MBUS User Manual

Important note:
This User Manual contains patent pending technology. See page 6 for more information.

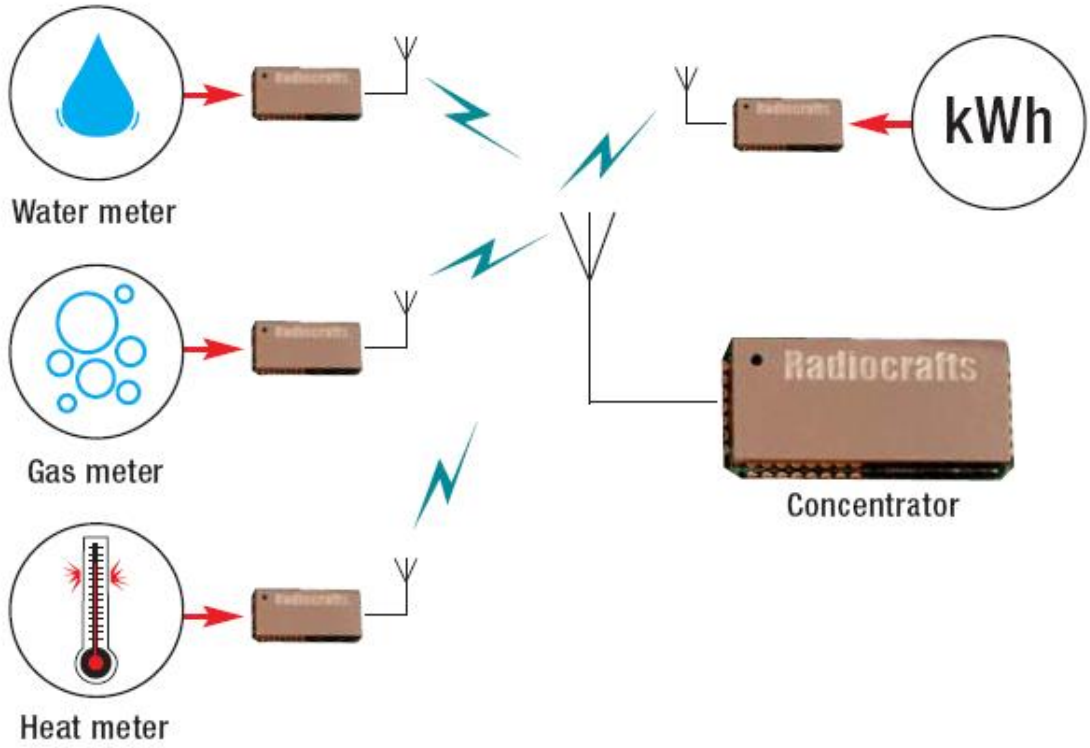


Table of contents

TABLE OF CONTENTS **2**
QUICK START **3**
MBUS FEATURE SETS **4**
OPTIONAL CUSTOM SPECIFIC VERSION **5**
IMPORTANT NOTE ON PATENT PENDING TECHNOLOGY **6**
INTRODUCTION: NETWORK TOPOLOGY **6**
WIRELESS M-BUS EMBEDDED PROTOCOL **7**
BASIC FUNCTIONALITY 7
UART INTERFACE FOR WIRELESS M-BUS PACKET HANDLING 8
FRAME FORMAT FOR TRANSMITTING DATA 9
FRAME FORMAT FOR RECEIVING DATA 9
UART INTERFACE FOR MODULE CONFIGURATION 10
UART TIMING INFORMATION 11
POWER MANAGEMENT 13
RSSI READING 13
TEMPERATURE READING 13
POWER SUPPLY VOLTAGE READING 13
MBUS1 DESCRIPTION **14**
MBUS1 TIMING TABLE 14
MBUS1 CONFIGURATION COMMANDS 15
MBUS1 CONFIGURATION MEMORY 17
MBUS2 DESCRIPTION **19**
MBUS2 AUTOMATIC ACKNOWLEDGE 19
MBUS2 AUTOMATIC ADDRESSING 19
MBUS2 AUTOMATIC SLEEP 19
MBUS2 INSTALLATION AND BINDING 20
MBUS2 ENCRYPTION 20
MBUS2 TIMING TABLE 23
MBUS2 CONFIGURATION COMMANDS 24
MBUS2 CONFIGURATION MEMORY 26
MBUS3 DESCRIPTION **29**
MBUS3 AUTO-MESSAGE GENERATOR 29
MBUS3 AUTOMATIC ADDRESSING 33
MBUS3 AUTOMATIC SLEEP 33
MBUS3 INSTALLATION AND BINDING 34
MBUS3 ENCRYPTION 34
MBUS3 REPEATER 35
MBUS3 TIMING TABLE 37
MBUS3 CONFIGURATION COMMANDS 38
MBUS3 CONFIGURATION MEMORY 40
MBUS4 DESCRIPTION **44**
MBUS4 MASTER REGISTERS 46
ITALIAN CIG EXTENSION OF WIRELESS M-BUS 47
CATEGORY 1 RECEIVER 48
MBUS4 CONFIGURATION COMMANDS AND CONFIGURATION MEMORY 49
APPENDIX 1: MBUS COMMAND LIST OVERVIEW **51**
APPENDIX 2: CONFIGURATION MEMORY FACTORY DEFAULT **52**
APPENDIX 3: ASCII TABLE **55**
DOCUMENT REVISION HISTORY **56**
DISCLAIMER **56**
TRADEMARKS **56**
LIFE SUPPORT POLICY **56**
CONTACT INFORMATION **57**

Quick Start

How do I transmit data?
Send your data to the RXD pin on the module. Use the UART format with settings (19200, 8, 1, N, no flow control). Up to 255 bytes are buffered in the module. The first byte of the message must contain the message length. The module will transmit the data when the whole packet is received.

How do I receive data?
Any received RF data packet with correct Wireless M-Bus format and check sums will be sent on the TXD pin. Optionally the meter address (first M-Bus block) is added to the data string. The RSSI value (received signal strength) can optionally be added to the message.

What about the antenna?
In most cases a simple quarter wavelength wire or a PCB track will do. Connect a piece of wire to the RF pin with length corresponding to the quarter of a wavelength. For space limited products, contact Radiocrafts and we will recommend the best antenna solution for your application.

How do I change the M-Bus mode, RF channel or any other parameter?
To change configurable parameters, send one byte to the module with the value 0x00 or assert the CONFIG-pin. This will take the module into configuration mode. Special commands are then used to access the configuration registers and test modes. Exit from configuration mode by sending the ‘X’ command. Parameters can be changed permanently and stored in non-volatile memory in the module.

MBUS feature sets

This User Manual describes the embedded protocol of the Wireless MBUS Modules from Radiocrafts. The MBUS firmware is available as different feature sets targeting specific applications. The hardware has the same size and pin-out for all frequency versions, and the different feature sets available are listed in the table below. The feature sets and the embedded functions are independent of the frequency, so this user manual is valid for all versions RC11xx-MBUSx. Detailed information on how to use the different feature sets is found in this User Manual. Additional information about the Wireless M-Bus packet structure for NTA 8130 compliance is described in Application Note 011 and is available on request.

Feature List	Feature set			
	MBUS1	MBUS2	MBUS3™	MBUS4™
General	Basic wireless M-bus functions	Added features for DSMR/NTA 8130 v. 3 compliance	Added features for OMS compliance	Wireless M-Bus mode N at 169 MHz
Network role	Master or Slave	Master or Slave	Master, Slave or Repeater	Master, Slave or Repeater
Modes	S1, S2, T1, T2, R2	T1, T2	C1, S1, S2, T1, T2	N1, N2, a-g
Encryption	No, must be handled externally	AES according to NTA 8130 (mode 4 and 5)	AES mode 4 and 5, and ELL encryption mode 1	AES mode 4 and 5, and ELL encryption mode 1
Installation mode	No, Must be handled externally	Yes, according to NTA 8130	Yes, according to OMS	Yes
Number of installed meters	None	Up to 8	Up to 64	256 internally, unlimited externally (>1000 meters per concentrator)
Filter function	No, receives any MBUS packet. Filtering must be handled externally	Master only receives messages from installed/registered meters (optional)	Master only receives messages from installed/registered meters (optional)	Master only receives messages from installed/registered meters (optional)
Automatic acknowledge in T2	No, must be handled externally	Yes, according to NTA8130 v. 3	Yes, according to OMS	Yes, for N2 mode
Automatic message acknowledge from Master	No	No	Yes, patent pending auto message generation. According to OMS, supporting two-way slaves; standard acknowledge or a predefined message from mailboxes or templates	Yes, patent pending auto message generation. Special support for handling of > 1000 meters (Slaves)

The command set used to configure the MBUS modules are different for each feature set and an overview is found in the appendixes.

Note that this user manual also is applicable for the RCxxxxTX-MBUS. This is a TX only hardware, and the RX features described in this user manual is not supported.

Optional custom specific version

As an option to the standard feature sets, a full wireless M-Bus application layer can be integrated in the module based on customer specification. In this case all the application layer protocol and timing will be handled internally by the module. The MPC1 (M-Bus Pulse Counter) is one such variant. See Data Sheet for details.

Important note on patent pending technology

Some of the technical solutions described in this User Manual are based on patent pending technology. In particular the methods used in the MBUS3 and MBUS4 to meet the T2 and N2 timing requirements for a master, using an address register, a flag register, an encryption key register combined with an auto-message generator for standard messages and its combination with a mailbox with pre-generated messages or templates, and a given message priority, depending on incoming messages, are subject to patenting.

Any infringements of patents and IP rights held by Radiocrafts will be prosecuted to the fullest extent.

Introduction: Network Topology

A Wireless M-Bus supported metering system normally consists of a number of heat-, gas-, water and/or electricity meters which reports their meteorological readings to a concentrator. The concentrator acts as the Master in the system while the meters are Slaves. In the standard the Master is referred to as “Other”.

The Radiocrafts Wireless M-Bus family of modules RC11xx-MBUSx can be configured to have a role as either Master or Slave. The Slave contains a unique address, and when sending a meter reading this address is added to the wireless message. The message from a Slave does not contain any Master address but the Master module within range will receive the message, and based on the Slave address (if the Slave is installed and the Master is configured for filtering), it will decode the message and send the data on its serial interface (TXD-pin).

In two-way communication modes, the battery operated meter (slave) will keep the receiver “on” for a short time. During this time slot the master can acknowledge the received message in order to open the communication channel (NTA 8130), or send a command (OMS) and thereby start a communication sequence. Note there is a difference in the addressing scheme between NTA 8130 and OMS: In NTA 8130 v.3 (MBUS2) the master returns an addressed acknowledgement to the Slave using the address field (Link Layer Address) originally received from the Slave. In OMS (MBUS3™) the master sends a command with it’s own address as Link Layer Address, and the slave’s address as Application Layer Address. Also prEN13757-4 (2013) use senders address as the Link Layer Address.

MBUS3™ (OMS) and MBUS4™ also allows for a one-way (unidirectional) repeater. The repeater will re-transmit all messages from slaves within range. Modules with MBUS3 and MBUS4 feature sets can be configured as a repeater.

MBUS3 has since its original release been extended to support the new C-mode (Compact mode), in addition to OMS functionality. A unique feature of MBUS3 is that T mode and C mode messages can be received at the same time.

Wireless M-Bus Embedded Protocol

Basic functionality

The module offers a buffered packet radio acting as a Wireless M-Bus modem. The module contains a fully embedded protocol supporting EN13757-4:2005 modes:

- Stationary mode S (S1, S1-m, S2)
- Frequent transmit mode T (T1 and T2)
- Frequent receive mode R2
- C1 mode according to EN13757-4:2013

The mode is configurable by the MBUS_MODE parameter.

The required M-Bus mode is configured by setting the module in configuration mode and entering appropriate UART commands. The following modes are supported:

S1/S2-mode:

Set MBUS_MODE = 0
Set PREAMBLE_LENGTH = 0 (for short preamble) or 1 (for long preamble)

The RF channel (channel 11) and data rate (32.768 kchip/s) are set internally in the module according to the S mode, and will override any settings in the RF_CHANNEL and RF_DATA_RATE configuration registers. This setting can also be used for T2 mode slave receive and master transmit.

T1-mode:

Set MBUS_MODE = 1

The RF channel (channel 12), data rate (100 kchip/s) and preamble length are set internally in the module according to the T mode, and will override any settings in the RF_CHANNEL, RF_DATARATE and PREAMBLE_LENGTH configuration registers. This setting can also be used for T2 mode slave transmit and master receive.

T2-mode:

Set MBUS_MODE = 2
Set NETWORK_ROLE = 0 or 1

The RF channel (channel 11 or 12), data rate (32.768 or 100 kchip/s) and preamble length are set internally in the module according to the T2 mode and the selected Network Role, either being a Slave (NETWORK_ROLE = 0) or a Master (NETWORK_ROLE = 1), and change according to receive/transmit. It will override any setting in the RF_CHANNEL configuration register.

R2-mode:

Set RF_CHANNEL = 1-10
Set MBUS_MODE = 4

The data rate (4.8 kchip/s) and preamble length are set internally in the module according to the R mode.

C1-mode:

Set MBUS_MODE = 10 or 11 (combined modes with T1 and T2 respectively)

The RF channel, data rate and preamble length are set internally in the module.

The module supports automatic generation of the Wireless M-Bus frame, i.e.;

- Preamble (header + synchronisation)
- Adding the first block (C-field and address/manufacturing ID)
- CRC
- Postamble

The RF signal is Manchester coded or "3 out of 6" coded for increased signal integrity.

The default M-Bus mode is entered and stored in the modules' non-volatile memory (MBUS_MODE). The M-Bus mode can also be changed using the 'G' command. Using the 'G' command, the value is not stored in non-volatile memory. To do a permanent change, use the 'M' command. The 'G' command should be used for frequent change of mode, to prevent excessive writing to the flash-based non-volatile memory.

The default C-field is entered and stored in the modules' non-volatile memory (CONTROL_FIELD). The C-field can also be changed using the 'F' command. Using the 'F' command, the value is not stored in non-volatile memory. To do permanent change, use the 'M' command.

The default Manufacturer ID and unique meter Address is entered and stored in the modules' non-volatile memory. The destination address (or module address) can also be changed using the 'T' command. Using the 'T' command, the address is not stored in non-volatile memory. To do a permanent change, use the 'M' command. MBUS2 sets the destination address automatically based on the last received message.

The module has an internal buffer and transmits application data as soon as the whole packet is received based on the packet length (first byte of the application frame). The module also has a timeout feature that will empty the input buffer in case of false data packets. The default timeout is 2 seconds.Max total payload is 246 bytes, or 255 including the header in the first block.

Sleep mode can be entered via an UART command and wake-up is triggered on UART traffic (one FFh byte). Sleep mode can also be entered automatically after a transmission (configurable by SLEEP_MODE).

The module acts as a buffered packet radio, hence all data to be sent is stored in the module before they are transmitted by the RF circuitry. Likewise, when data is received they are stored in the module before they are sent to the host. This allows the communication controller to add address information, CRC and encryption during transmission, and to do error check and decryption of the received data.

The Module has an UART interface that is used for both Wireless M-Bus packet data and module configuration.

UART Interface for Wireless M-Bus packet handling
The host use the UART Interface to send and receive Wireless MBUS data. The UART packet format can be changed in the configuration mode.

When the Module receives a Wireless M-Bus packet over RF it will send the packet over the UART interface on the TXD Line. When the host MCU wants to transmit a Wireless M-Bus packet over the RF, it must send the packet through the UART Interface on the RXD line.

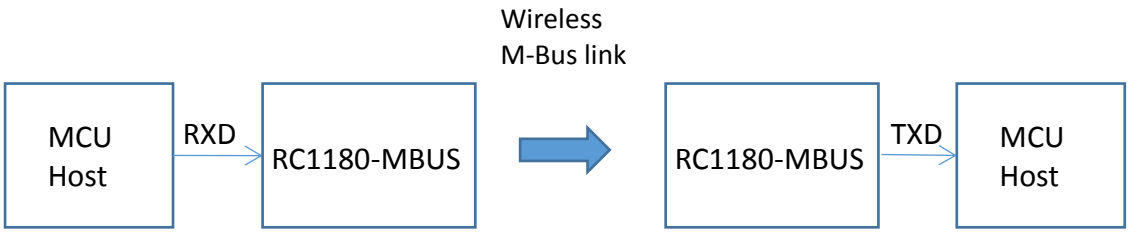


Figure 1: UART interface overview

Frame format for transmitting data
The data frame for the UART RXD pin (input for transmitting a Wireless M-Bus packet) is built like this:

Figure 2: UART interface packet transmission (RXD pin)

L is the length (not including the length byte itself), followed by the application data with the CI byte first. CI is the Control Information byte. The application data typically contains the application header, and data points with VIF and DIF codes. The application data can also be SML of DMLS.

An Extended Link Layer (ELL) can be added before the application data using this structure:
L + CI_{ELL} + ELL + CI_{APL} + APPL_DATA

The HEADER and C-field (and adjusted L value) is added to the Wireless M-Bus packet automatically by the module before transmitting over RF and both can be changed in configuration mode.

To transmit only a HEADER without Application data (CI+APPL_DATA) a L=0xFE can be sent to the module UART without additional bytes.

Frame format for receiving data
The data frame for the UART TDX pin (Output for received Wireless M-Bus packets) is built like this:

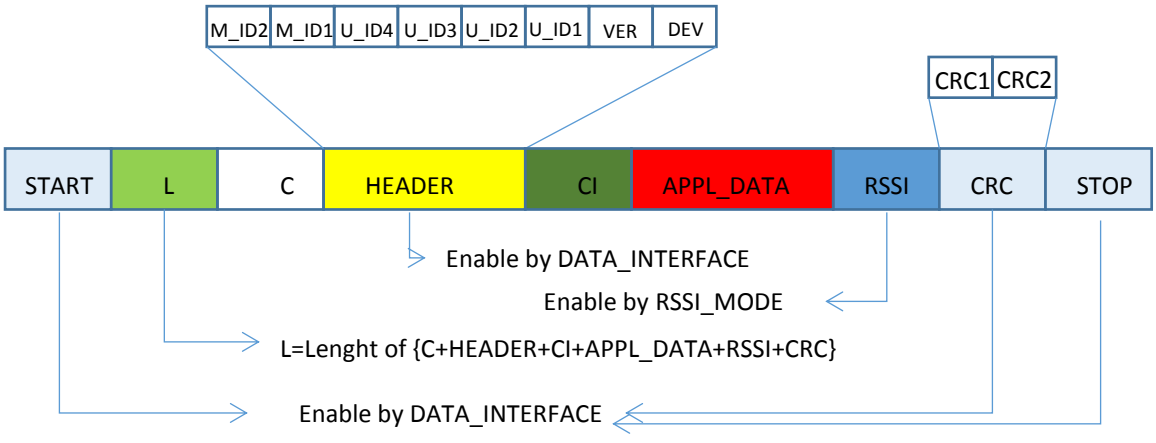


Figure 3: UART interface packet reception (TXD pin)

Data in blue and yellow are optional output parts of the UART message and can be enabled in configuration mode by the DATA_INTERFACE and RSSI_MODE configuration parameters.

L is the length byte and is always present. It does not include itself or the START/STOP bytes, but will include RSSI and CRC if enabled. CRC is calculated from length byte including RSSI (if enebled). Note that the length bytes itself must be reduced by 2 before calculating CRC on host. The length byte was shorter at calculation in module as the CRC was not added at time of calculation.

When setting DATA_INTERFACE = 1, the received HEADER will not be sent on the UART (typically used on a slave). However, to be able to notify the external application when an Acknowledgement is received ("empty" frame), a special string can be used. By setting DATA_INTERFACE = 3, the two byte string 00:E5h (i.e. L = 0) will be sent on the UART when an empty acknowledge frame is received.

Application data (CI + APPL_DATA) is always present (except when only a HEADER is received).

For host applications using a UART buffer the timing information used for parsing could be lost. In this case a start and stop byte can be used. Setting DATA_INTERFACE = 4 will add a START byte (68h) and a STOP byte (16h) to the message. This is only used for the module-to-host communication direction (TXD). Setting DATA_INTERFACE = 8 will add a two byte CRC checksum, and DATA_INTERFACE = 0Ch will add START/STOP bytes and CRC. The CRC is sent MSByte first.

The RSSI value is appended when RSSI_MODE = 1.

UART Interface for Module Configuration

The configuration of the module can be changed in-circuit from the host during operation, at the time of installation of the equipment, at the manufacturing test, or even as a stand-alone module. The configuration is changed by sending commands on the UART interface after the module is set in configuration mode. The configuration mode is entered by sending 00h to the module, or by asserting the CONFIG pin (set low).

In configuration mode the module will respond by sending a '>' prompt on the TXD pin. This indicates that the module is ready to receive commands. The CONFIG pin (if used) can then be de-asserted. Note that the CONFIG pin must be de-asserted *before* the Exit command ('X') is sent to the module in order to return to normal operation.

After a command is executed, the module responds with the '>' prompt character again, indicating it is ready for a new command. Do not send a new command before the '>' prompt is received. The time required to execute a command can vary depending on the command (see the Timing Information section). There is no '>' prompt after the 'X' exit command.

The parameters that are set by dedicated configuration commands ('C', 'P' etc) take immediate effect after returning to normal operation (IDLE), but will not be stored in non-volatile memory and will be lost in case the supply power is turned off or if the module is reset. These parameters are for example the radio channel and output power.

Permanent changes of parameters can be done by writing to the configuration memory using the memory command 'M'. These are for example *default* radio channel, *default* output power and M-Bus mode, see the Configuration Memory section for details.

The flow diagram below illustrates how to use the UART interface to enter configuration mode, change configuration parameters and return to IDLE mode.

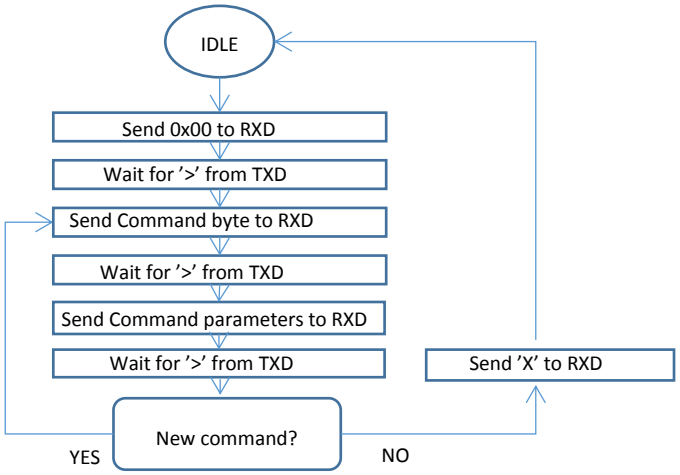


Figure 4: Configuration mode flow diagram

UART Timing Information

A UART byte consist of one start bit, 8 data bits, and one stop bit. In configuration mode a command to prompt reply will looks like this:

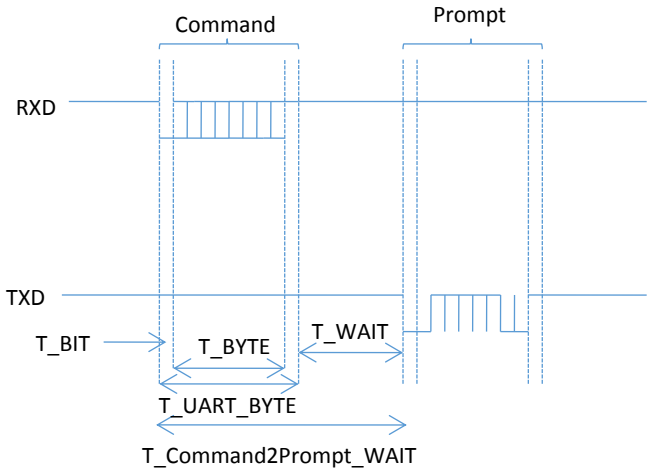


Figure 5: UART Command and prompt

The command-to-prompt wait time (T_Command2Prompt_WAIT) is different from command to command and values are shown in the timing table for each MBUS feature set.

The IDLE state is the normal state where the module both searches for preamble on the RF and wait for a character to be received on the UART. RXD is the state when receiving characters from the host filling up the internal buffer. TX state is when the data is transmitted on the air. RX state is when data is received from the air after preamble detection. TXD is the state where the received data is sent to the host on the UART.

CONFIG is the configuration mode, the state entered by sending 00h or asserting the CONFIG pin and is entered during parameter configuration, while MEMORY CONFIG is the sub-state entered by the 'M' command where the non-volatile configuration memory is being

programmed. Note the limitation on maximum number of write cycles using the 'M' command, see Electrical Specifications in the Data Sheet.

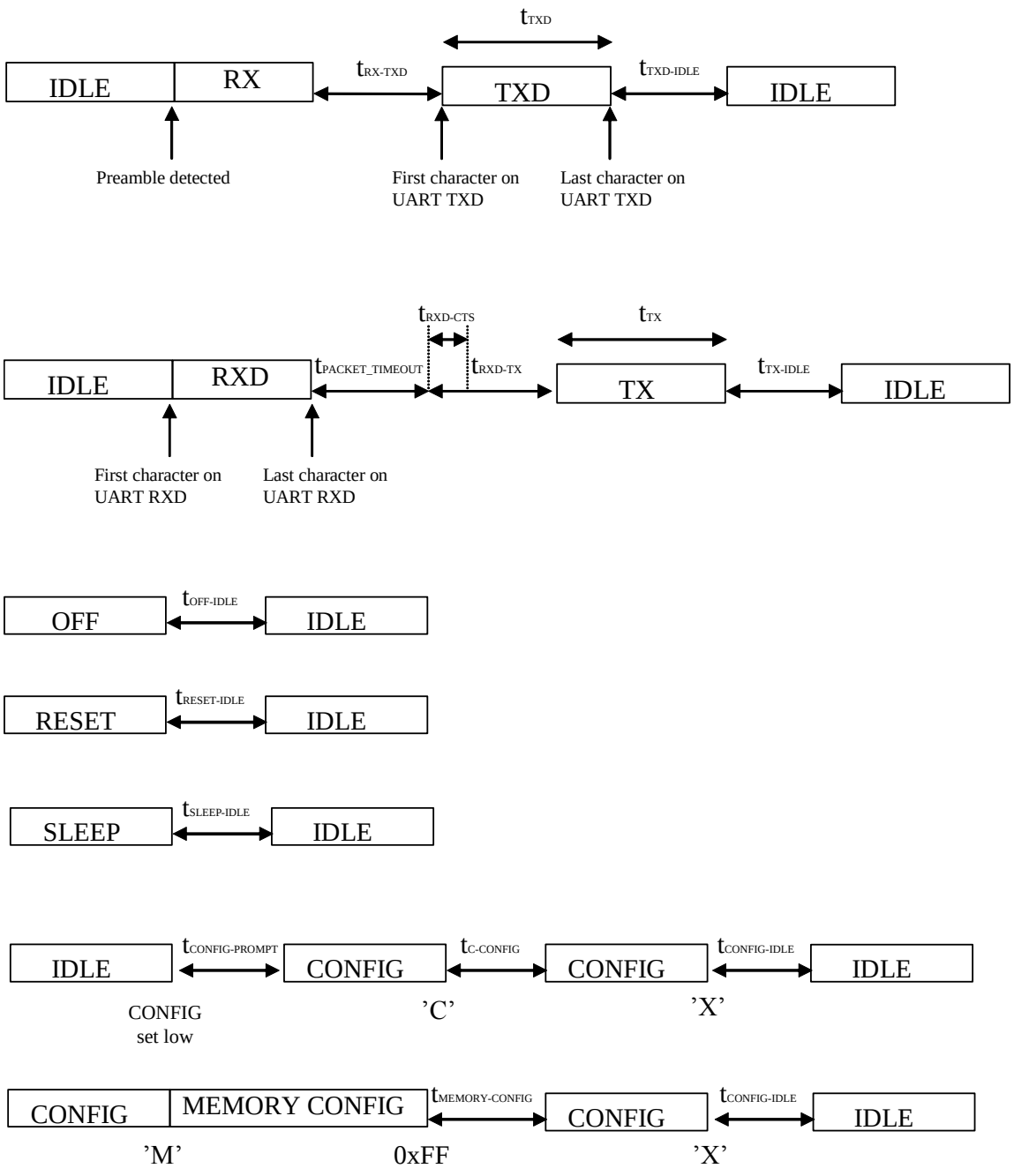


Figure 6: UART timing diagram

Timing values are shown in the timing table for each MBUS feature set.

Power Management
The module can be set in SLEEP mode in order to reduce the power consumption.

The low power SLEEP mode is manually entered by using the SLEEP command 'Z' after the module is set in configuration mode. It is also possible to configure the module to enter SLEEP automatically after a message has been transmitted (SLEEP_MODE=1). With this setup the module has to enter TX-mode (transmit a message) after power-on before entering SLEEP mode first time. In SLEEP mode the module will not receive or detect incoming data, neither from the host (UART port) nor from the air. The module is awakened from the SLEEP mode by sending the wake-up byte FFh on the UART RXD line (use a UART Baud rate > 4.8 kBd due to a maximum pulse length requirement). After the module has woken up (see Timing Information) it is ready to receive data on the UART or from the air. The SLEEP command can be used for both Master and Slave.

All configuration settings and RAM values are retained during SLEEP.

If the module is shut completely off (supply power turned off), all configuration settings in non-volatile memory is restored, but values in RAM are overwritten with default settings.

RSSI Reading
The module provide a digital Received Signal Strength Indicator (RSSI) through the 'S' command, or attached to the received messages. The RSSI value appended to a received message is the signal strength of that received packet. The RSSI value is an 8 bit character (one byte) indicating the current input signal strength or the signal strength of the received message. The signal strength can be used as an indication of fading margin, or as a carrier sense signal to avoid collisions.

The RSSI value increases with increased input signal strength in 0.5 dB steps. Input signal strength is given by (typ.):

P = - RSSI / 2 [dBm]

Temperature Reading
The module provides readings of a digital temperature monitoring sensor (TEMP) through the 'U' command. The module returns an 8 bit character (one byte) indicating the current temperature in degrees Celsius (°C) followed immediately by a second character which is the prompt ('>').

The TEMP value increases with increased temperature in 1 °C steps and has an accuracy of +/- 2 °C. The temperature is given by:

T = TEMP(dec) - 128 [°C] (example: TEMP=0x98 equals +24 °C)

Power Supply voltage Reading
The module provides readings of an internal power supply voltage monitoring sensor (VCC) through the 'V' command. The module returns an 8 bit character (one byte) indicating the current power supply voltage level followed immediately by a second character which is the prompt ('>'). The command can be useful for battery power monitoring.

The VCC value increases with increased supply voltage in 30 mV/step. The power supply voltage is given by:

V = VCC(dec)*0.030 [V] (example: VCC=0x68 equals 3.12 V)

MBUS1 Description

MBUS1 Timing table

The table below shows the timing information for the module when changing between different operating states. Timing symbols are according to figure 5 and 6.

Symbol	Value	Description / Note
t_{RX-TXD}	180 us	Time from last byte is received from the air until first character is sent on the UART
t_{TXD}	Min 590 us	t_{TXD} = # bytes received x 590 us/char (10 bits at 19.2 kBd + 70 us delay per character)
$t_{TXD-IDLE}$	900 us	Time from last character is sent on the UART until module is in IDLE mode (ready for RXD and RX)
$T_{RXD-CTS}$	20 us	Time from last character is received by the UART (including any timeout) until CTS is activated
t_{RXD-TX}	960 us	Time from last character is received by the UART (including any timeout) until the module sends the first byte on the air.
$T_{TX-IDLE}$	960 us	Time from last character is sent on the air until module is in IDLE mode (ready for RXD and RX)
$t_{OFF-IDLE}$	3.2 ms	
$t_{RESET-IDLE}$	3.0 ms	
$t_{SLEEP-IDLE}$	1.3 ms	
$t_{CONFIG-PROMPT}$	60 us	Time from 00h / CONFIG pin is set low until prompt (‘>’)
$T_{C-CONFIG}$	1.1 ms	Delay after channel-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
$T_{G-CONFIG}$	1.1 ms	Delay after new M-Bus mode-byte is sent until prompt (‘>’). (For other volatile memory commands there is no delay but immediate prompt)
T_{WAIT}	1.55 ms (M command) 24 us (all other commands)	Delay from stop bit of the command byte to start bit of the prompt reply. See figure 5 for details.
$t_{MEMORY-CONFIG}$	31 ms	In this period the internal flash (non-volatile memory) is programmed. *Do not reset, turn the module off, or allow any power supply dips in this period as it may cause permanent error in the Flash configuration memory. After the last command parameter byte the host should wait for the ‘>’ prompt before any further action is done to ensure correct re-configuration.*
$T_{CONFIG-IDLE}$	1.1 ms	End of ‘X’ to IDLE
t_{TX}	3.6 ms	TX time for T1 mode when Length=1 on the UART. Preamble, sync, CRC and MBUS address field added internally.

MBUS1 Configuration Commands
A list of commands is shown in the table below. Commands must be sent as ASCII characters or their corresponding binary value. All arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Command	Argument in hex (decimal)	Note
Channel	'C' – 0x43	0x01-0x0C (1-10) Apply for R mode only	Data is stored in volatile memory only.
C-field	'F' – 0x46	0x00-0xFF (0-255)	Data is stored in volatile memory only.
M-Bus mode	'G' – 0x47	0x00-0x04 (0-4) 0: S 1: T1 2: T2 3: Reserved 4: R	Data is stored in volatile memory only.
Memory configuration	'M' – 0x4D	(Address, Data): see list of parameters below. 0xFF exits memory configuration.	Used to enter memory configuration menu. Parameters changed are stored in non-volatile memory.
Output power	'P' – 0x50	0x01-0x05 (1-5)	Data is stored in volatile memory only.
Quality Indicator	'Q' – 0x51	Returns one byte indicating the signal quality	Based on bit errors in preamble and synch word
Signal Strength (RSSI)	'S' – 0x53	Returns one byte indicating the signal strength of a detected signal or a valid packet.	If a valid packet has been received when in configuration mode, it will return the RSSI of the last received packet.
Destination / module address	'T' – 0x54	8 bytes; MAN_ID2 (Second manufacturer code), MAN_ID1 (First manufacturer code), ID4, ID3, ID2, ID1, VER (Version), DEV (Device Type),	Data is stored in volatile memory only.
Temperature monitoring	'U' – 0x55	Returns one byte indicating the temperature.	See page 12 for details
Battery monitoring	'V' – 0x56	Returns one byte indicating the power supply voltage.	See page 12 for details
Memory Read one byte	'Y' – 0x59	0x00 – 0x7F (The argument is the address in the configuration memory.)	Return one byte value from the configuration memory.
Exit command	'X' – 0x58	(none)	Exit to normal operation mode. All changes of parameters take effect.
Sleep mode	'Z' – 0x5A	(none)	Exit sleep mode by sending 0xFF on UART RXD pin

Test mode 0	'0' – 0x30	(none)	List all configuration memory parameters
Test mode 1	'1' – 0x31	(none)	TX carrier
Test mode 2	'2' – 0x32	(none)	TX modulated signal PN9 sequence
Test mode 3	'3' – 0x33	(none)	TX Off, RX mode

Note: ASCII characters are written as 'X', hexadecimal numbers are written like 0x00, and decimal numbers are written like 10 throughout the text. A table of ASCII characters and their respective hex and decimal values are found in the Appendix.

Any invalid command will be ignored and the '>' prompt will be re-sent.

If Test mode 1 or 2 is used, it is important to enter Test mode 3 before exiting the configuration mode ('X') in order to ensure proper operation in normal mode.

Example:
To select RF channel 3, send the follow sequence after asserting the CONFIG line and the '>' prompt is received:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'C'	0x43	'>'	
3	0x03	'>'	Wait for '>' prompt
[A new command could be issued here]			
'X'	0x58	(none)	Module returns to IDLE state

Note that the CONFIG line must be de-asserted after the first '>' prompt was received, but before the 'X' command.

MBUS1 Configuration Memory

The table below shows the complete list of configurable parameters stored in non-volatile memory. These values can be changed using the 'M' command. All addresses and arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for R mode only	0x00	1-10	0x01 (1)	See data sheet for channel frequencies. Only used for R mode.
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels.
MBUS_MODE	M-Bus mode	0x03	0-4 0: S 1: T1 2: T2 3: Reserved 4: R	0x01 (1)	Use 'G' command to change value in volatile memory only
SLEEP_MODE	Sleep mode	0x04	0: Disable Sleep 1: Enable Sleep	0x00 (0)	When enabled the module enter Sleep mode after transmission
RSSI_MODE	Append RSSI to received data	0x05	0:Disabled 1: Enabled	0x00 (0)	When enabled the RSSI value is appended to the received data
Radio packet configuration					
PREAMBLE_ LENGTH	Short of long preamble in S mode only	0x0A	0x00 (0): Short 0x01 (1): Long	0x00 (0)	Preamble (header) length, apply for S mode only
Medium access, addressing and network management					
NETWORK_ROLE		0x12	0x00 (0): Slave/Meter 0x01 (1): Master/Concentrator	0x00 (1)	
M_ID1	Manufacturer ID, first byte	0x19	0x00-0xFF (0-255)	0x0C (12)	
M_ID2	Manufacturer ID, second byte	0x1A	0x00-0xFF (0-255)	0xAE (174)	
U_ID1	Unique ID, first byte	0x1B	0x00-0xFF (0-255)	0x12 (18)	
U_ID2	Unique ID, second byte	0x1C	0x00-0xFF (0-255)	0x34 (52)	
U_ID3	Unique ID, third byte	0x1D	0x00-0xFF (0-255)	0x56 (86)	
U_ID4	Unique ID, forth byte	0x1E	0x00-0xFF (0-255)	0x78 (120)	
VER	Version	0x1F	0x00-0xFF (0-255)	0x01 (1)	
DEV	Device	0x20	0x00-0xFF (0-255)	0x07 (7)	
Data and configuration interface, UART Serial Port					
UART_BAUD_RATE	Baud rate	0x30	0x00: Not used 0x01: 2400 0x02: 4800 0x03: 9600 0x04. 14400 0x05: 19200 0x06: 28800 0x07: 38400 0x08: 57600 0x09: 76800 0x0A: 115200 0x0B: 230400	0x05 (5)	BE CAREFUL IF CHANGING AS HOST MAY LOOSE CONTACT WITH MODULE! Does not take effect until module is re-booted / reset.

UART_FLOW_CTRL	UART flow control	0x35	0: None 1:CTS only 3:CTS/RTS 4:RXTX(RS485)	0x00 (0)	
DATA_INTERFACE	Data interface	0x36	0x00: MBUS packet with ID and address 0x01: Application data only 0x02: Reserved 0x03: Application data only with ack (00:3Eh) 0x04: Add start/stop byte 0x08: Add CRC 0x0C: Add start/stop byte and CRC	0x00 (0)	Sets receiver data format. First byte is always packet length (except when using start byte)
CONTROL_FIELD	C-field	0x3B	0x00-0xFF (0-255)	0x44 (68)	Use 'F' command to change value in volatile memory only
PART_NUMBER		0x3C-0x48		RCxxxx-MBUS1	
HW_REV_NO		0x4A-0x4D		x.yz	x, y and z; Any number 0d-9d
FW_REV_NO		0x4F-0x52		x.yz	x, y and z; Any number 0d-9d
Exit from memory configuration		0xFF	No argument should be sent		To exit from command mode the 'X' command must be sent after '>' is received.

To make permanent changes to default values and other parameters, the Memory Configuration command 'M' is used. This command should be followed by pairs of byte being the memory address and the new value to be stored at that address. In order to exit the Memory Configuration mode, the 'address' 0xFF must be sent, but without any data argument. Then wait for the '>' prompt while the internal memory is re-programmed (see Timing Information for typical delay). To completely exit from command mode, the normal exit command 'X' must be sent.

Example:
To change the MAN_ID (at address 0x19 and 0x1A) and set it to (100,200) (0x64, 0xC8), send the following sequence:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'M'	0x4D	'>'	Module ready to receive address
0x19	0x19	(none)	
100	0x64	(none)	
0x1A	0x1A	(none)	
200	0xC8	(none)	
[new address could be sent here]			
[new value could be sent here]			
0xFF	0xFF	'>'	Wait for '>' prompt
'X'	0x58	(none)	Module returns to IDLE state

Test mode 0 ('0' command) can be used to list all parameters stored in non-volatile memory. This command can be used to verify and check the module configuration.

MBUS2 Description

MBUS2 Automatic Acknowledge

The Master must reply with an acknowledge message within 3 ms after a received Access Demand, if further communication shall take place. To meet this timing requirement, the module has built-in automatic acknowledge support. Use the Acknowledge flag (set using the A –command) to indicate which slave shall be acknowledged at the next access. The flag is automatically cleared but can also be cleared manually with the A-command.

MBUS2 Automatic Addressing

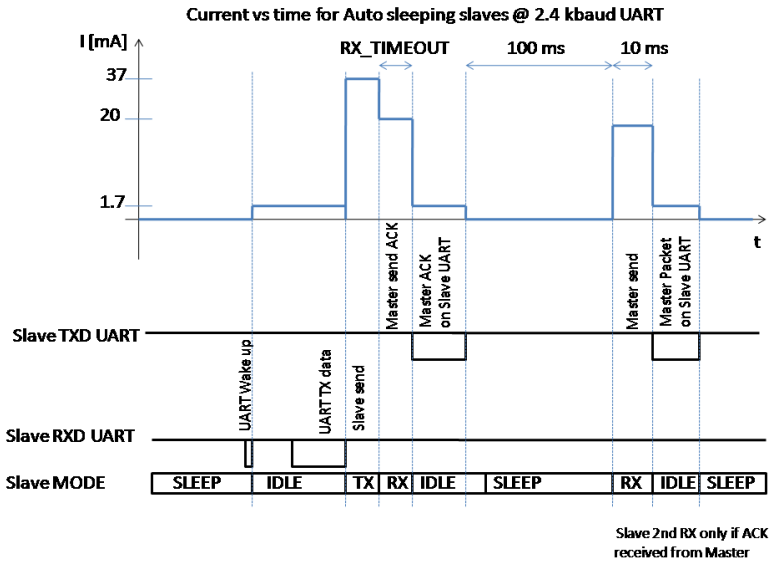
When the Master receives a message from a slave, this slave HEADER will be used as the HEADER for the next transmission from the master. If, for example during installation, messages are received from several meters, the last message received will be the default address. This feature can be overridden by using the 'T' command.

MBUS2 Automatic Sleep

The Slave has special support for automatic sleep after data transmission. If automatic SLEEP is enabled (SLEEP_MODE = 1), the module will automatically go to sleep in a configurable time after data transmission, if acknowledge is not received. The receiver timeout after slave TX is configured by RX_TIMEOUT. If acknowledge is received after the slave transmission, the slave goes to sleep for 100 ms before it goes into RX for 10 ms waiting for a new Master message. This reduces the burden of the slave host controller when meeting NTA 8130 (v.3.0) data exchange timing.

A '0xFF' byte will wake up a sleeping Slave into a low power idle mode (IDLE). The Slave will only accept UART input (RXD UART) in this IDLE mode to save current. The current saving depends on UART rate and gives less current consumption for low UART rates (but data transfer takes longer time). The figure below illustrates current vs time for a Slave configured for auto sleep at 2.4 kBaud UART rate when receiving acknowledge from Master after the first transmission.

An additional '0xFF' byte in IDLE mode will force the Slave into active RX mode to enable RF reception before transmitting.



MBUS2 Installation and Binding
The module (Master) can be set in Installation Mode using the “I” command. When the module is in Installation Mode it will accept all Access Demand Install messages (C-field is 46h in T1 and 06h in T2).

Slaves can be bound to a Master by registering their addresses in the Address Register. This is done by using the “B” (Bind) command followed by a register number (1-8) and an 8 bytes slave address. Thus, a maximum of 8 meters can be bound to one Master for the MBUS2 feature set (meeting NTA 8130 request for minimum 4 meters).

Note; the host must know which registers are used and which are free at any time.

MBUS2 Encryption
The module supports AES-128 encryption. When a slave is registered into the masters address register, the master host should request a new encryption key from the utility data base. The new key is specific for each slave and related to the slave equipment ID or unique address. The new key should be provided in two versions; plain and encrypted using the slave’s default key. That is, the utility needs to keep a register with default keys linked to each meter.

The master host should send the new encrypted key to the slave. And the slave host should configure this new key into the module using the ‘K’ command, followed by 16 bytes (the encrypted key). The slave module will automatically de-crypt the new key using its default key.

The master host should then send the new (plain) key to the master module using the ‘K’ command, followed by the register location number, and the 16 bytes (the new key). The register location number must correspond to the address register location for that slave.

The ENCRYPT_FLAG and DECRYPT_FLAG parameters are used to enable / disable the encryption when transmitting and receiving messages. The 8 bit values are interpreted as bit maps corresponding to the 8 address registers, LSB being register 1.

For a message to be encrypted, the encryption flag for the particular slave must be set, and the CI-field and Signature field sent to the module must be according to the standard for encryption to take place. The module will do byte stuffing if required to get a full 16 byte encryption block. Only CI-fields 0x5A, 0x5B and 0x72 allows encryption. The Signature field must be encryption mode 0x04 or 0x05 according to NTA 8130.

When using Signature field 0x05, the application must add the two encryption verification bytes (0x2F) after the header. The Initialization Vector for the encryption is extracted from the long header (for CI-fields 0x5B and 0x72). For the short header (CI-field 0x5A) the Initialization Vector is partly from the MAC header (destination address) and the short application header.

The Access Counter byte in the application header is used by the encryption, and the host application must increment the counter in order to avoid repetitive messages.

For a message to be decrypted the decryption flag for the particular slave must be set, and the CI-field and Signature field must be according to the standard for encryption to take place.

In the slave, only LSB is used as a flag for encryption/decryption.

The ‘D’ and ‘E’ commands are used to set decryption and encryption flags without storing in non-volatile memory. The value following the D and E commands is interpreted as bit maps corresponding to the 8 address registers.

To test the encryption feature you need to have a valid key set for the master and the slave. In addition you need to send a valid UART frame into the module in order for the internal encryption and decryption feature to be activated on this message. The RCTools PC software from Radiocrafts (MBUS_CCT and MBUS_DEMO) can be used to configure key sets and send and receive encrypted messages.

Example of a key set:
Master Key: 0x00 0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88 0x99 0xAA 0xBB 0xCC 0xDD 0xEE 0xFF
Slave Key: 0x0A 0x90 0xE5 0xB7 0x4D 0x28 0x07 0xA6 0x51 0xF6 0x9A 0xC0 0x89 0x6A 0x09 0xF6
Use factory default for Init vector and Default key in the configuration memory.

Example of UART RXD frames that enable encryption:

Test packet A: No filling byte
Slave TX message: C=6, L=2D, CI=72,
Data=78563412AE07010701002004 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFF

Slave TX message: C=6, L=25, CI=5A,
Data=02002004 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFF

Slave TX message: C=6, L=2D, CI=5B,
Data=78563412AE07010703002004 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFF

Test packet B: Adding filling bytes
Slave TX message: C=6, L=1E, CI=72,
Data=78563412AE07010704001104 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCC

Slave TX message: C=6, L=17, CI=5A,
Data=05001204 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDD

Slave TX message: C=6, L=2E, CI=5B,
Data=78563412AE07010706002104 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAA

Test packet C: Adding filling bytes and un-encrypted bytes at the end of the packet.
Slave TX message: C=6, L=22, CI=72,
Data=78563412AE07010707001104 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCAAAAAAAA

Slave TX message: C=6, L=1B, CI=5A,
Data=08001204 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDAAAAAAAA

Slave TX message: C=6, L=2B, CI=5B,
Data=78563412AE07010709001A04 2F2FAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAABBCCDDEEFFAAAAAAAA

Green: Number of encrypted bytes including the two 0x2F bytes in the start of the encryption part
Read: Second block Header (un-encrypted)
Blue: Signature field. To enable encryption this field must be 04 (NTA3130 encryption) or 05 (NTA8130 encryption harmonized to OMS).
Purple: Encrypted part of UART message. Internal encryption and not encrypted on UART
Black: Un-encrypted part of message

Slave step-by-step guide to send an encrypted message to a Master:
1. Enter configuration mode
2. Send the slave key to the module using the ‘K’ command and ‘Slave Key’ as command parameter.
3. Enable Encryption using the ‘E’ command and ‘0x01’ as command parameter
4. Leave configuration mode using the ‘X’ command

5. Send one of the valid test packets above.

This packet will now be encrypted over the RF link.

Master step-by-step guide to receive an encrypted message from a Slave:
1. Enter configuration mode
2. Install the Slave to this Master using the ‘B’ and ‘Address Register’ as command parameter followed by the slave address. Address register is 1-8.
2. Send the Master key to module using the ‘K’ command and ‘Address Register’ + ‘Master Key’ as command parameters.
3. Enable Decryption using the ‘D’ command and ‘Address Register’ as command parameter.
4. Leave configuration mode using the ‘X’ command

An encrypted packet from the installed slave will now be decrypted before it is provided on the UART TXD.

Note that you also have the option to use the M command to permanently set the Encryption/Decryption flag in the configuration non-volatile memory, instead of using the ‘D’ and ‘E’ commands. Encryption / decryption will only take place when the signature field indicate mode 0x04 or 0x05. If encryption mode 0x00 is used, the message will not be encrypted / decrypted even if the flags are set.

MBUS2 Timing table
The table below shows the timing information for the module when changing between different operating states. Timing symbol is according to figure 5 and 6.

Symbol	Value	Description / Note
t_{RX-TXD}	180 us	Time from last byte is received from the air until first character is sent on the UART
t_{TXD}	Min 590 us	t_{TXD} = # bytes received x 590 us/char (10 bits at 19.2 kBd + 70 us delay per character)
$t_{TXD-IDLE}$	900 us	Time from last character is sent on the UART until module is in IDLE mode (ready for RXD and RX)
$T_{RXD-CTS}$	20 us	Time from last character is received by the UART (including any timeout) until CTS is activated
t_{RXD-TX}	960 us	Time from last character is received by the UART (including any timeout) until the module sends the first byte on the air.
$T_{TX-IDLE}$	960 us	Time from last character is sent on the air until module is in IDLE mode (ready for RXD and RX)
$t_{OFF-IDLE}$	3.2 ms	
$t_{RESET-IDLE}$	3.0 ms	
$t_{SLEEP-IDLE}$	1.3 ms	
$t_{CONFIG-PROMPT}$	60 us	Time from 00h / CONFIG pin is set low until prompt (“>”)
$T_{G-CONFIG}$	1.1 ms	Delay after channel-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
$T_{G-CONFIG}$	1.1 ms	Delay after new M-Bus mode-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
T_{WAIT}	1.55 ms (B, K and M command) 24 us (all other commands)	Delay from stop bit of the command byte to start bit of the prompt reply. See figure 5 for details.
$T_{MEMORY-CONFIG}$	31 ms	In this period the internal flash (non-volatile memory) is programmed. *Do not reset, turn the module off, or allow any power supply dips in this period as it may cause permanent error in the Flash configuration memory. After the last command parameter byte the host should wait for the ‘>’ prompt before any further action is done to ensure correct re-configuration.*
$T_{CONFIG-IDLE}$	1.1 ms	End of ‘X’ to IDLE
t_{TX}	3.6 ms	TX time for T1 mode when Length=1 on the UART. Preamble, sync, CRC and MBUS address field added internally. Depends on M-Bus mode (T, S, R) and L

MBUS2 Configuration Commands

A list of commands is shown in the table below. Commands must be sent as ASCII characters or their corresponding binary value. All arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Command	Argument in hex (decimal)	Note
Acknowledge	'A' – 0x41	One byte bitmap, address register 1 is LSB.	Sets bitmap for acknowledge from Master. 0x00 will clear all flags.
Bind	'B' – 0x42	Register number (1-8) followed by 8 byte address (same order as for Destination/module address)	Used to bind slaves to master. Data stored in non-volatile memory.
Channel	'C' – 0x43	0x01-0x0C (1-10) Apply for R mode only	Data is stored in volatile memory only.
Decrypt	'D' – 0x44	One byte bitmap, address register 1 is LSB.	Sets bitmap for decryption of data. 0x00 will clear all flags.
Encrypt	'E' – 0x45	One byte bitmap, address register 1 is LSB.	Sets bitmap for encryption of data. 0x00 will clear all flags.
C-field	'F' – 0x46	0x00-0xFF (0-255)	Data is stored in volatile memory only.
M-Bus mode	'G' – 0x47	0x00-0x04 (0-4) 0: S 1: T1 2: T2 3: Reserved 4: R	Data is stored in volatile memory only. S and R mode not supported in NTA8130
Install	'I' – 0x49	0: Normal operation 1: Install mode 2: Accept all messages	In install mode messages with C-field = 06h and 46h are accepted. Use in Master only.
Key register	'K' – 0x4B	Slave: 16 byte key. Master: Register number (1-8) followed by 16 byte key	Used to set encryption key. Data stored in non-volatile memory.
Memory configuration	'M' – 0x4D	(Address, Data): see list of parameters below. 0xFF exits memory configuration.	Used to enter memory configuration menu. Parameters changed are stored in non-volatile memory.
Output power	'P' – 0x50	0x01-0x05 (1-5)	Data is stored in volatile memory only.
Quality Indicator	'Q' – 0x51	Returns one byte indicating the signal quality of the last received packet	Based on bit errors preamble and synch word
Signal Strength (RSSI)	'S' – 0x53	Returns one byte indicating the signal strength of a detected signal or a valid packet.	If a valid packet has been received when in configuration mode, it will return the RSSI of the last received packet.
Destination / module address	'T' – 0x54	8 bytes; M_ID2, M_ID1,	Data is stored in volatile memory only.

		U_ID4, U_ID3, U_ID2, U_ID1, VER (Version), DEV (Device Type),	
Temperature monitoring	'U' – 0x55	Returns one byte indicating the temperature.	See page 12 for details
Battery monitoring	'V' – 0x56	Returns one byte indicating the power supply voltage.	See page 12 for details
Memory Read one byte	'Y' – 0x59	0x00 – 0xFF (The argument is the address in the configuration memory.)	Return one byte value from the configuration memory.
Exit command	'X' – 0x58	(none)	Exit to normal operation mode. All changes of parameters take effect.
Sleep mode	'Z' – 0x5A	(none)	Exit sleep mode by sending 0xFF on UART RXD pin
Test mode 0	'0' – 0x30	(none)	List all configuration memory parameters
Test mode 1	'1' – 0x31	(none)	TX carrier
Test mode 2	'2' – 0x32	(none)	TX modulated signal PN9 sequence
Test mode 3	'3' – 0x33	(none)	TX Off, RX mode

Note: ASCII characters are written as 'X', hexadecimal numbers are written like 0x00, and decimal numbers are written like 10 throughout the text. A table of ASCII characters and their respective hex and decimal values are found in the Appendix.

Any invalid command will be ignored and the '>' prompt will be re-sent.

If Test mode 1 or 2 is used, it is important to enter Test mode 3 before exiting the configuration mode ('X') in order to ensure proper operation in normal mode.

Example:
To select RF channel 3, send the follow sequence after asserting the CONFIG line and the '>' prompt is received:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'C'	0x43	'>'	
3	0x03	'>'	Wait for '>' prompt
[A new command could be issued here]			
'X'	0x58	(none)	Module returns to IDLE state

Note that the CONFIG line must be de-asserted after the first '>' prompt was received, but before the 'X' command.

MBUS2 Configuration Memory

The table below shows the complete list of configurable parameters stored in non-volatile memory. These values can be changed using the 'M' command. All addresses and arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for R mode only	0x00	1-10	0x01 (1)	See data sheet for channel frequencies. Only used for R mode.
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels.
MBUS_MODE	M-Bus mode	0x03	0-4 0: S 1: T1 2: T2 3: Reserved 4: R	0x01 (1)	Use 'G' command to change value in volatile memory only S and R mode not supported in NTA8130
SLEEP_MODE	Sleep mode	0x04	0: Disable Sleep 1: Enable Sleep 3: Enable with sleep timeout	0x00 (0)	When enabled the module enter Sleep mode after transmission. Delay set by RX_TIMEOUT. If enabled with sleep timeout, the module goes directly to Sleep after a Reset, and to Sleep after TIMEOUT when wakeup from Sleep or exit Config mode.
RSSI_MODE	Append RSSI to received data	0x05	0:Disabled 1: Enabled	0x00 (0)	When enabled the RSSI value is appended to the received data
TIMEOUT	Time before modem clear buffer without transmitting if Buffer size < Length byte (first byte).	0x10	<1-254> 0x01 (1): 32 ms 0x02 (2): 48 ms 0x03 (3): 64 ms 0x7C (124): 2 s 0xF9 (249): 4 s	0x7C	IF SLEEP_MODE=3 the same timeout is used to auto sleep slaves. Modem transmit without timeout when Buffer size = length byte.
Radio packet configuration					
PREAMBLE_ LENGTH	Short of long preamble in S mode only	0x0A	0x00 (0): Short 0x01 (1): Long	0x00 (0)	Preamble (header) length, apply for S mode only
Medium access, addressing and network management					
NETWORK_ROLE		0x12	0x00 (0): Slave/Meter 0x01 (1): Master/Concentrator	0x00 (0)	
M_ID1	Manufacturer ID, first byte	0x19	0x00-0xFF (0-255)	0x0C (12)	
M_ID2	Manufacturer ID, second byte	0x1A	0x00-0xFF (0-255)	0xAE (174)	
U_ID1	Unique ID, first byte	0x1B	0x00-0xFF (0-255)	0x12 (18)	
U_ID2	Unique ID, second byte	0x1C	0x00-0xFF (0-255)	0x34 (52)	
U_ID3	Unique ID, third byte	0x1D	0x00-0xFF (0-255)	0x56 (86)	

U_ID4	Unique ID, forth byte	0x1E	0x00-0xFF (0-255)	0x78 (120)	
VER	Version	0x1F	0x00-0xFF (0-255)	0x01 (1)	
DEV	Device	0x20	0x00-0xFF (0-255)	0x07 (7)	
Data and configuration interface, UART Serial Port					
UART_BAUD_RATE	Baud rate	0x30	0x00: Not used 0x01: 2400 0x02: 4800 0x03: 9600 0x04. 14400 0x05: 19200 0x06: 28800 0x07: 38400 0x08: 57600 0x09: 76800 0x0A: 115200 0x0B: 230400	0x05 (5)	BE CAREFUL IF CHANGING AS HOST MAY LOOSE CONTACT WITH MODULE! Does not take effect until module is re-booted / reset.
UART_FLOW_CTRL	UART flow control	0x35	0: None 1:CTS only 3:CTS/RTS 4:RXTX(RS485)	0x00 (0)	
DATA_INTERFACE	Data interface	0x36	0x00: MBUS packet with ID and address 0x01: Application data only 0x02: Reserved 0x03: Application data only with ack (00:3Eh) 0x04: Add start/stop byte 0x08: Add CRC 0x0C: Add start/stop byte and CRC	0x00 (0)	Sets receiver data format. First byte is always packet length (except when using start byte)
FREQ_CAL		0x39		Different for each module.	Found in factory and used by the module to minimise the total frequency tolerance. For firmware upgrade, read back the value and write it back after the upgrade.
LED_CONTROL		0x3A	0: Disabled 1: RX/TX indicator 2: UART/RF IDLE indicator	0x00 (0)	Use to enable LED0/LED1 for RX/TX packet indication or UART/RF IDLE mode indicator.
CONTROL_FIELD	C-field	0x3B	0x00-0xFF (0-255)	0x06 (6)	Use ‘F’ command to change value in volatile memory only
RX_TIMEOUT		0x3C	0x00-0xFF (0-255)	0x0B (11)	Delay before Sleep mode, n x 0.6 ms
INSTALL_MODE		0x3D	0: Normal mode (accept installed MBUS meters only) 1: Install mode 2: Filter off (accept all MBUS types)	2	
ENCRYPT_FLAG		0x3E		0	Bit mask for encryption, enabled when set
DECRYPT_FLAG		0x3F		0	Bit mask for decryption, enabled when set
DEFAULT_KEY		0x40-0x4F		All 0xFF (255)	

INIT_VECTOR		0x50-0x5F		All 0x00 (0)	
PART_NUMBER		0x61-0x6C		RCxxxx-MBUS2	
HW_REV_NO		0x6E-0x71		x.yz	x, y and z; Any number 0d-9d
FW_REV_NO		0x73-0x76		x.yz	x, y and z; Any number 0d-9d
ADDRESS_ID1		0x80-0x87		All 0x00	Address for installed meters.
ADDRESS_ID2		0x88-0x8F		All 0x00	Address for installed meters.
ADDRESS_ID3		0x90-0x97		All 0x00	Address for installed meters.
ADDRESS_ID4		0x98-0x9F		All 0x00	Address for installed meters.
ADDRESS_ID5		0xA0-0xA7		All 0x00	Address for installed meters.
ADDRESS_ID6		0xA8-0xAF		All 0x00	Address for installed meters.
ADDRESS_ID7		0xB0-0xB7		All 0x00	Address for installed meters.
ADDRESS_ID8		0xB8-0xBF		All 0x00	Address for installed meters.
SERIAL_NUMBER		0x78-0x7F		All 0x00	8 bytes reserved for serial number for traceability. Is programmed by Radiocrafts during test.
Exit from memory configuration		0xFF	No argument should be sent		To exit from command mode the 'X' command must be sent after '>' is received.

To make permanent changes to default values and other parameters, the Memory Configuration command 'M' is used. This command should be followed by pairs of byte being the memory address and the new value to be stored at that address. In order to exit the Memory Configuration mode, the 'address' 0xFF must be sent, but without any data argument. Then wait for the '>' prompt while the internal memory is re-programmed (See Timing Information for typical delay). To completely exit from command mode, the normal exit command 'X' must be sent.

Example:
To change the MAN_ID (at address 0x19 and 0x1A) and set it to (100,200) (0x64,0xC8), send the following sequence:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'M'	0x4D	'>'	Module ready to receive address
0x19	0x19	(none)	
100	0x64	(none)	
0x1A	0x1A	(none)	
200	0xC8	(none)	
[new address could be sent here]			
[new value could be sent here]			
0xFF	0xFF	'>'	Wait for '>' prompt
'X'	0x58	(none)	Module returns to IDLE state

Test mode 0 ('0' command) can be used to list all parameters stored in non-volatile memory. This command can be used to verify and check the module configuration.

MBUS3 Description

The OMS specification differs from NTA 8130 in the way the communication session between a slave (meter) and master (MUC) is done. More recently, also the EN13757-4 (2013) has been updated according to OMS, and even the latest NTA 8130 v 4.0 is now aligned with these. For battery operated devices the slave is always initiating the communication, and the master must then transmit (if need) within a short time window (2-3 ms in the T mode). After one such “ping-pong” sequence, the slave will have a pause (enter sleep mode) for 2-5 seconds, before it again do a new transmission allowing the master to do another transmission. This means that all messages to be sent from the master must be ready and transmitted within a very short time. The MBUS3 Auto-message generator and Mailbox features make this task easy and doable (patent pending implementation).

The installation scenarios in OMS are supposed to support many meters per MUC, hence the address and registers has been extended to 64 positions in MBUS3.

MBUS3 also support reception of mode C. Mode C can be received at the same time as mode T. The module also automatically receives and distinguish between Frame Format A and B. The current implementation supports C1 mode (unidirectional) as specified in the preliminary EN 13757-4 (2013). The timing of C2 mode is currently not supported. C mode meters can be installed in the Master module, optionally with encryption keys, mixed with T mode meters.

The module supports the new Extended Link Layer (ELL), its new AES-128-CTR mode of encryption, and the new compact frame formats, all according to EN13757-4 (2013). The new ELL can be used for all modes.

MBUS3 Auto-message Generator
In T mode the Master must reply with a new message within 3 ms after a received Access Demand, if further communication shall take place. To meet this timing requirement, the module has built-in an Auto-message Generator and a Mailbox.

Auto-message Flags (set using the A –command) are used to indicate which slave shall be replied at the next access, what type of message, and if it is sent from the mailbox or is a standard message. The module supports two schemes of auto message replies; single level or two level.

In the single level scheme (MAILBOX.TLAM = 0) the message is sent from the Master without regard to the previous message or Access Number of the incoming message. The flag is automatically cleared when a reply is sent (when enabled by MAILBOX.ACMB and ACSM), but can also be cleared manually with the A-command.

A Frequent Access Cycle (FAC) between a Slave and a Master consist of several Master messages depending on the slave message and its Access Number in the case of lost messages. A two levels Auto-message handler can be enabled (MAILBOX.TLAM = 1) to streamline the communication with the meter in this case. Using two level Auto-messages the Master can be configured to check replies from the Slave and respond differently from the Mailbox or standard message depending on the Slave packet Access Number.

For each address register there is a corresponding flag register (default two bytes, TLAM=0):
SSSSSSSS, E:D:FCB:R:MB
Where
SSSSSSSS selects the standard message, see below
E is the Encryption flag bit
D is the Decryption flag bit

FCB is the Frame Count Bit (to be used in the next transmission)
R is one bit reserved for future use
MB is the Mailbox selected by 4 bits (0000 to 1111), where 0000 means none

If two level Auto-message handler is used (TLAM=1 in MAILBOX) an additional byte is added to this flag register and gives:

SSSSSSSS, E:D:FCB:R:MB, ANV:SMV:MBV:R:MB2

Where

ANV enables Access Number Valid
SMV enables Standard Message Valid
MBV enables Mailbox Valid
R is one bit reserved for future use
MB2 is the Mailbox level 2 selected by 4 bits (0000 to 1111)

Only the lsb nibble (MB2) should normally be altered by the A-command. ANV, SMV and MBV are used internally by the module to keep track of the message sequence. These flags should be read and 'or-ed' when writing a new MB2 setting.

The ANV (Access Number Valid) flag is defined to tell if the Access Number should be checked by the Master before reply. (Strictly, the master transmission is not a "reply", as the Master is the primary station in unbalanced communication. But we use the term "reply" because the Slave defines the timing by its reception window). For the first Message in the Frequent Access Cycle (FAC) the ANV flag is not set, as the first reply is in response to the first initiating access (e.g. SND-NR) from the Slave.

The SMV and MBV flags store information about the last sent message by the Master, if it was from a Mailbox or if it was a Standard message, and type of standard message. Normally these flags shall not be altered by the user. If MBV was not set it means that a Standard Message was sent. SMV=1 means the message sent was a response to RSP-UD and msb of 0x38 in SSSSSSSS will be cleared. SMV=0 means the message sent was a response to RSP-UD or ACK and msb of 0x07 in SSSSSSSS will be cleared.

The flags are read by using the O command (letter "O"). If two level Auto-message handler is used (TLAM=1 in MAILBOX), the 3 flag registers are returned, and an additional byte containing the last Access Number.

The flags are set by using the A command: First send the register number, followed by two bytes if TLAM=0 and three bytes if TLAM=1. The flags are stored in volatile memory and will be lost during power off.

The E and D flags can be set by default for all register positions using the ENCRYPT and DECRYPT configuration parameters.

The MAILBOX configuration parameter is used to set certain features of the Auto-message Generator:

RR:TLAM:ACSM:ACMB:AMMB:DFC:A

Where

RR are two bits reserved for future use
TLAM enables the two level auto message handler
ACSM is the Auto Clear Standard Message bit
ACMB is the Auto Clear Mailbox bit
AMMB is the Accept all Messages (C-fields) for Mailbox transmission bit
DFC is the check Data Flow Control bit in the C-field
A is the check Accessibility bit

If the ACSM or ACMB bit is set, the auto-message flag will be automatically cleared after a transmission. If the DFC bit is set, the DFC bit of the incoming message (in the C-field) will be checked before an auto-message is sent (otherwise it will be ignored, and the message is sent regardless of the incoming message). If the A bit is set, the Accessibility of the incoming message (in the Transport Layer or Application Header Configuration Word) will be checked before an auto-message is sent, sending only to meters that signals they are accessible (otherwise it will be ignored, and the message sent regardless of the incoming message).

If TLAM is set the Master checks the Slave reply Access Number and use two level auto-message handling. When the two level message handler is used the auto clearing flags should not be set (ACSM=0, ACMB=0).The auto-message flags/mailbox will be cleared as a part of the two level auto-message handler after a correct response is received from the Slave.

The Auto-message Generator supports three types of messages:

- Standard messages with fixed frame format
- Message from a mailbox
- A template messages from a mailbox

There are several standard messages that can be sent, depending on the incoming message from the slave. The master module will automatically recognize the type of message, and reply accordingly. The table below shows which messages can be sent for each incoming message, and the corresponding flags (SSSSSSSS) for the master reply. Standard messages are shown in green cells, messages from the mailbox in red cells.

<table>
<tr><th colspan="2" rowspan="2"></th><th colspan="7">Master reply</th></tr>
<tr><th>Mail-box</th><th>CNF-IR

(0x06)</th><th>ACK

(0x00)</th><th>REQ-UD1
(0x5A/
7A*)</th><th>REQ-UD2

(0x5B/7B)</th><th>SND-NKE
(0x40)</th><th>SND-UD

(0x53/73)</th></tr>
<tr><td rowspan="5">Slave mess-age</td><td>SND-IR
(0x46)</td><td></td><td>10000000</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>ACC-DMD
(0x48)</td><td></td><td></td><td>01000000</td><td></td><td></td><td></td><td></td></tr>
<tr><td>SND-NR
(0x44)</td><td></td><td></td><td></td><td colspan="2">00000000: No reply
00100000: REQ-UD1
00010000: REQ-UD2
00001000: Reserved</td><td></td><td>←</td></tr>
<tr><td>RSP-UD
(0x08/28
0x18/38**)</td><td></td><td></td><td></td><td colspan="4" rowspan="2">00000000: No reply
00000100: REQ-UD1
00000010: REQ-UD2
00000001: SND-NKE</td></tr>
<tr><td>ACK
(0x00/20
0x10/30**)</td><td></td><td></td><td></td></tr>
</table>

*Note: The module will automatically set the Frame Count Bit in the C-field depending on the last transmitted FCB bit (as stored in the Flag Register). The FCB bit is automatically alternated.

**Note: A reply is only sent if the DFC bit in the MAILBOX configuration parameter is cleared

The Status byte will automatically be set to the incoming packet RSSI value, and the Access Number will be set automatically, depending on the incoming message. Before a transmission, the Access Number can be set to a new value using the N-command. A reply to SND-IR and ACC-DMD use the incoming Access Number for the reply.

If several flags are set, the most significant bit flag has the highest priority. The Mailbox has priority above the standard messages. Mailbox messages will only be sent in reply to certain incoming message types as shown in the table above (in red). However, the module can also be configured (the AMMB parameter) to use the mailbox for any incoming message. The MAILBOX configuration parameter is used to set auto-clearing of flags for standard messages and the mailbox.

There are up to 15 mailboxes that can be used to store “pre-cooked” messages. Each mailbox is 64 bytes, except mailbox number 15 which is 128 bytes. Mailboxes can be combined to support messages up to 255 bytes (less the header). When using more than 64 bytes for one message, the following mailbox cannot be used. That is, if four mailboxes of 255 bytes are to be used, they should be address as number 1, 5, 9 and 13.

IMPORTANT NOTE: The binding (B-command) and Encryption Key entry (K-command) will erase the Mailbox (due to memory constraints in the module).

Mailboxes can be written and read using the W – write command, and the R – read command. The format for writing to the mailbox is:

Mailbox number, C-field, Length, CI-field, followed by the rest of the message

The Mailbox number should be 1-15. There is no restriction on the C-field value, but do note that OMS specify only a set of allowable C-fields. The Length byte shall be the number of bytes following, not including the Length byte itself. Note; if the message (including C and Length) is more than 64 bytes, the following mailbox cannot be used.

Long mailbox messages must be pre-encrypted in order to meet the time constraints in T mode. This is done by using the E-command. If the message holds a valid address, and the Configuration Word is set for encryption, and the Encryption flag is disabled, the message will be encrypted and re-stored in the Mailbox. Note, if the Encryption flag is not disabled, the message will be encrypted (again!) when transmitted.

If the message in the mailbox has an Application Layer Address = 0, Access Number = 0 and Status = 0, the message is called a *template*. When using a template, the blank fields will be added on-the-fly when responding to the slave. This makes it possible to point at the same Template message for many meters.

The Access Number (AN) will automatically be increased for every new message transmission if set to zero in the mailbox.

Normally a Standard Message is sent as an answer to SND-IR and ACC-DMD using the same AN, but even if an answer is sent from the Mailbox, the AN must be the same as the incoming message. For all other cases the Master will set a new AN. Using the single level Auto-message handler (TLAM = 0) the Access Number is always incremented with no regard to the incoming message.

Using the two level Auto-message handler (TLAM=1) the Master AN will not be incremented if the message is repeated (because the last Master reply was not received by the Slave, the Master AN will be used again). For a new message the Master AN is incremented automatically.

The RSSI of the last received packet will automatically be added in the Status byte by the module if set to zero.

Note; if the Template message requires encryption (as determined by the signature, and the Encryption enable flag is set), the time to encrypt the message on-the-fly might violate the 2-3

ms response time in T-mode. Templates with more than one block to be encrypted can only be used for S mode (up to 50 ms response time). In this case the pre-encrypted mailbox message must be used.

MBUS3 Automatic Addressing

When the Master receives a message from a slave, this slave's address will be used as the Application Layer Address for the next transmission from the master when using the auto-message feature.

When messages are sent from the UART buffer, the Application Layer Address used is determined by the host. Note; the Link Layer Address (in the MAC header) is always the masters own address as stored in the configuration memory. The Link Layer Address stored in configuration memory can be overridden by using the 'T' command (volatile memory), or permanently changed using the 'M' command (non-volatile memory).

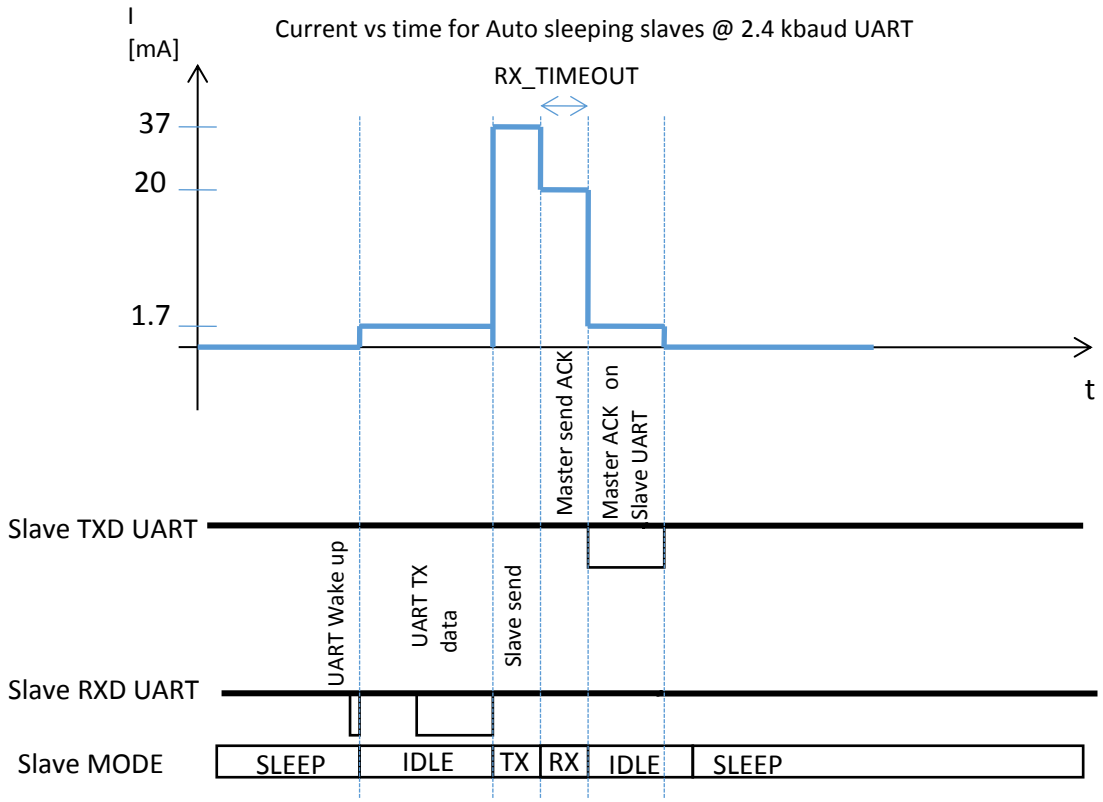
MBUS3 Automatic Sleep

The Slave has special support for automatic sleep after data transmission. If automatic SLEEP after TX is enabled (SLEEP_MODE = 1), the module will automatically go to sleep in a configurable time after data transmission, if a message is not received. The receiver timeout is configured by RX_TIMEOUT. The automatic sleep is done after any message transmitted. The slave host application is responsible for waking up the module for a re-transmission in 2-5 seconds if a communication sequence was started, but no reply received from the master.

If automatic SLEEP after RX (and TX) is enabled (SLEEP_MODE = 3), the module will automatically go to sleep after data reception and the message is sent to the host over the UART. This feature should normally only be used for a Slave. In T-mode the RX_TIMEOUT should be set to 6. In S-mode the RX_TIMEOUT should be set to 86.

A '0xFF' byte will wake up a sleeping Slave into a low power idle mode (IDLE). The Slave will only accept UART input (RXD UART) in this IDLE mode to save current. The current saving depends on UART rate and gives less current consumption for low UART rates (but data transfer takes longer time). The figure below illustrates current vs time for a Slave configured for auto sleep at 2.4 kBaud UART rate when receiving acknowledge from Master after the first transmission.

An additional '0xFF' byte in IDLE mode will force the Slave into active RX mode to enable RF reception before transmitting.



MBUS3 Installation and Binding
The module (Master) can be set in Installation Mode using the "I" command. When the module is in Installation Mode it will accept all Send Installation Request (SND-IR) messages (C-field is 46h).

Slaves can be bound to a Master by registering their addresses in the Address Register. This is done by using the "B" (Bind) command followed by a register number (1-64) and an 8 bytes slave address. The Slave address format is the same as used in the Application Layer Address. Thus, a maximum of 64 meters can be bound to one Master (MUC).

Note; the host must know which registers are used and which are free at any time.

MBUS3 Encryption
The module supports AES-128 encryption. When a slave is registered into the masters address register, the master host should request the encryption key from the utility data base. The new key is specific for each slave and related to the slave equipment ID or unique address.

Currently, OMS does not specify how to distribute new keys over the air to the slave.

The master host should send the new (plain) key to the master module using the 'K' command, followed by the register location number, and the 16 bytes (the new key). The register location number must correspond to the address register location for that slave.

The ENCRYPT_FLAG and DECRYPT_FLAG parameters are used to set the flags to enable / disable the encryption when transmitting and receiving messages.

Wireless M-Bus offers two options for encryption of data; either on the Link Layer using the ELL (Extended Link Layer), or on the Application Layer using the TPL (Transport Layer / short or long application header). The module will automatically detect which option is to be used, and perform the encryption and decryption. The ELL uses the counter mode (AES-128-CTR) and does not need any padding of data. The TPL encryption uses block mode (AES-128-CBC) and need padding to 16 byte blocks.

For a message to be encrypted, the encryption flag for the particular slave must be set, and the CI-field and Configuration Word (TPL) or Communication Control field (ELL) sent to the module must be according to the standard for encryption to take place. The message to be encrypted must have padding bytes (“2F”) if required to get a full 16 byte encryption block. Only TPL CI-fields using short of long header allows encryption. The Configuration Word must set encryption mode 0x04 or 0x05 according to OMS.

When using Signature field 0x05, the application must add the two encryption verification bytes (0x2F) after the header. The Initialization Vector for the encryption is extracted from the long header (for CI-fields 0x5B, 0x60, 0x64, 0x6C, 0x6D, 0x72, 0x7C, 0x7E, 0x80 and 0x8B). For the short header (CI-field fields 0x5A, 0x61, 0x65, 0x7A, 0x7D, 0x7F and 0x8A) the Initialization Vector is partly from the MAC header (destination address) and the short application header.

For ELL encryption the module will automatically add or check the Payload CRC if the ENCRYPT_FLAG or DECRYPT_FLAG parameters are set with &0x02.

The Access Counter byte in the application header is used by the encryption, and the counter must be incremented in order to avoid repetitive messages. When using the mailbox template the counter is automatically incremented by the module (see description of two level buffering).

For a message to be decrypted the decryption flag for the particular slave must be set, and the CI-field and Configuration Word (TPL) or Communication Control field (ELL) must be according to the standard for encryption to take place.

In the slave, register 1 is used to hold the flags for encryption/decryption.

The set flags command ‘A’ is used to set decryption and encryption flags without storing in non-volatile memory.

The ‘E’ command can be used to encrypt a message in the Mailbox, as described above. Note that you also have the option to use the M command toset all the Encryption/Decryption flag in the configuration non-volatile memory, instead of using the ‘A’ command. Encryption / decryption will only take place when the signature field indicate mode 0x04 or 0x05. If encryption mode 0x00 is used, the message will not be encrypted / decrypted even if the flags are set.

MBUS3 Repeater

The MBUS3 feature set contains complete autonomous unidirectional unregistered repeater functionality. The module can be configured as a Repeater by setting NETWORK_ROLE = 2. The MBUS_MODE must be set to 0 (S-mode) or 1 (T1 mode).

The Repeater operates as a stand-alone unit after power on, no installation is required. The Repeater should be mains powered as its receiver is working continuously. The Repeater can handle up to 15 messages simultaneously, each with a random delay before the transmission.

The Repeater will repeat all SND-NR and SND-IR messages within reach after a random delay of 5-25 seconds, with the Hop-Count bit set to 1. If it is a SND-IR message, the repeater will in addition generate a SND-NKE message after 2-5 seconds to indicate it is within range to support an installation tool. The SND-NKE message contains the RSSI level for the received SND-IR message.

The Repeater supports repetitions using the ELL or the TPL.

The Repeater generates a SND-NR “management message” every 240 minutes. This message indicates that the Repeater is alive and working.

The Repeater can be set in Installation mode by activating (setting low) the Install button input (pin 25, Demo Board S5) for < 1 second. The LED driver output (pin 29, Demo Board D1, red LED) will blink rapidly when in Installation mode. In Installation mode the module transmits SND-IR every 30-60 seconds. After 90 transmissions it returns to normal mode.

The installation mode can also be turned off by activating the Install button input again. The LED output will then stop toggling.

All pending messages are cleared when entering Installation mode, but the module also works as a normal repeater when in Installation mode.

MBUS3 Timing table

The table below shows the timing information for the module when changing between different operating states. Timing symbol is according to figure 5 and 6.

Symbol	Value	Description / Note
t_{RX-TXD}	180 us	Time from last byte is received from the air until first character is sent on the UART
t_{TXD}	Min 590 us	t_{TXD} = # bytes received x 590 us/char (10 bits at 19.2 kBd + 70 us delay per character)
$t_{TXD-IDLE}$	900 us	Time from last character is sent on the UART until module is in IDLE mode (ready for RXD and RX)
$T_{RXD-CTS}$	20 us	Time from last character is received by the UART (including any timeout) until CTS is activated
t_{RXD-TX}	960 us	Time from last character is received by the UART (including any timeout) until the module sends the first byte on the air.
$T_{TX-IDLE}$	960 us	Time from last character is sent on the air until module is in IDLE mode (ready for RXD and RX)
$t_{OFF-IDLE}$	6.5 ms	
$t_{RESET-IDLE}$	6.5 ms	
$t_{SLEEP-IDLE}$	1.3 ms	For UART data rates up to 4.8 kBd the sequence 0xFF : 0x00 can be sent without delay (from Sleep to Config). For higher UART baud rates, add 2 ms delay before setting Config mode
$t_{CONFIG-PROMPT}$	60 us	Time from 00h / CONFIG pin is set low until prompt (“>”)
$T_{G-CONFIG}$	1.1 ms	Delay after channel-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
$T_{G-CONFIG}$	1.1 ms	Delay after new M-Bus mode-byte is sent until prompt (“>”).(For other volatile memory commands there is no delay but immediate prompt)
T_{WAIT}	1.55 ms (B, K and M command) 24 us (all other commands)	Delay from stop bit of the command byte to start bit of the prompt reply. See figure 5 for details.
$T_{MEMORY-CONFIG}$	31 ms	In this period the internal flash (non-volatile memory) is programmed. *Do not reset, turn the module off, or allow any power supply dips in this period as it may cause permanent error in the Flash configuration memory. After the last command parameter byte the host should wait for the ‘>’ prompt before any further action is done to ensure correct re-configuration.*
$T_{CONFIG-IDLE}$	1.1 ms	End of ‘X’ to IDLE
t_{TX}	3.6 ms	TX time for T1 mode when Length=1 on the UART. Preamble, sync, CRC and MBUS address field added internally. Depends on M-Bus mode (T, S, R) and L

MBUS3 Configuration Commands

A list of commands is shown in the table below. Commands must be sent as ASCII characters or their corresponding binary value. All arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Command	Argument in hex (decimal)	Note
Auto-message flags	'A' – 0x41	Register number (1-64) followed by 2 bytes	Sets auto-message flags for the Master and encryption flags.
Bind	'B' – 0x42	Register number (1-64) followed by 8 byte address (NB: Use application layer format, NOT the same order as for Destination/module address in link layer)	Used to bind slaves to master. Data stored in non-volatile memory.
Channel	'C' – 0x43	0x01-0x0C (1-10) Apply for R mode only	Data is stored in volatile memory only.
Encrypt	'E' – 0x45	Mailbox number (1-15)	
C-field	'F' – 0x46	0x00-0xFF (0-255)	Data is stored in volatile memory only.
M-Bus mode	'G' – 0x47	0x00-0x04 (0-4) 0: S2 1: T1 2: T2 3: S1 4: R 10: T1+C 11: T2+C	Data is stored in volatile memory only. R and C-mode not supported in OMS
Install	'I' – 0x49	0: Normal operation 1: Install mode 2: Accept all messages	In install mode messages with C-field = 06h and 46h are accepted. Use in Master only.
Key register	'K' – 0x4B	Register number (1-64) followed by 16 byte key. Slave must use register 1.	Used to set encryption key. Data stored in non-volatile memory.
List binding	'L' – 0x4C	Register number (1-64)	Module responds with the address stored in the register (8 bytes)
Memory configuration	'M' – 0x4D	(Address, Data): see list of parameters below. 0xFF exits memory configuration.	Used to enter memory configuration menu. Parameters changed are stored in non-volatile memory.
Access Number	'N' – 0x4E	0x00 – 0xFF (0-255)	Set new Access Number
Read AutO-message flags	'O' – 0x4F	Register number (1-64)	Module responds with the auto-message flags (2 bytes)
Output power	'P' – 0x50	0x01-0x05 (1-5)	Data is stored in volatile memory only.
Quality Indicator	'Q' – 0x51	Returns one byte indicating the signal quality of the last received packet	Based on bit errors preamble and synch word

Read mailbox	'R'– 0x52	Mailbox number (1-15). Will send the selected mailbox content to the UART.	
Signal Strength (RSSI)	'S' – 0x53	Returns one byte indicating the signal strength of a detected signal or a valid packet.	If a valid packet has been received when in configuration mode, it will return the RSSI of the last received packet.
Destination / module address	'T' – 0x54	8 bytes; M_ID2, M_ID1, U_ID4, U_ID3, U_ID2, U_ID1, VER (Version), DEV (Device Type),	Data is stored in volatile memory only.
Temperature monitoring	'U' – 0x55	Returns one byte indicating the temperature.	See page 13 for details
Voltage monitoring	'V' – 0x56	Returns one byte indicating the power supply (battery) voltage.	See page 13for details
Write to mailbox	'W' – 0x57	Writes data to selected mailbox (1-15)	Sequence: W > (prompt from module) # (mailbox number) C L CI Data
Memory Read one byte	'Y' – 0x59	0x00 – 0xFF (The argument is the address in the configuration memory.)	Return one byte value from the configuration memory.
Sleep mode	'Z' – 0x5A	(none)	Exit sleep mode by sending 0xFF on UART RXD pin
Test mode 0	'0' – 0x30	(none)	List all configuration memory parameters
Test mode 1	'1' – 0x31	(none)	TX carrier[1]
Test mode 2	'2' – 0x32	(none)	TX modulated signal PN9 sequence
Test mode 3	'3' – 0x33	(none)	TX off, RX mode
Test mode 4	'4' – 0x34	(none)	IDLE (TX off, RX off)
	'@RC'	Reset to factory settings	CONFIG pin must be asserted, or CONFIG_INTERFACE=1
	'@RR'	Reset radio	CONFIG pin must be asserted, or CONFIG_INTERFACE=1

[1] TX carrier: One has to send a user packet in normal mode prior to "Test mode 1" in order to run a frequency calibration.

Note: ASCII characters are written as 'X', hexadecimal numbers are written like 0x00, and decimal numbers are written like 10 throughout the text. A table of ASCII characters and their respective hex and decimal values are found in the Appendix.

Any invalid command will be ignored and the '>' prompt will be re-sent.

If Test mode 1 or 2 is used, it is important to enter Test mode 3 before exiting the configuration mode ('X') in order to ensure proper operation in normal mode.

Example:
To select RF channel 3, send the follow sequence after asserting the CONFIG line and the '>' prompt is received:

Command	Hex	Response	Comment/Note
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'C'	0x43	'>'	
3	0x03	'>'	Wait for '>' prompt
[A new command could be issued here]			
'X'	0x58	(none)	Module returns to IDLE state

Note that the CONFIG line must be de-asserted after the first '>' prompt was received, but before the 'X' command.

MBUS3 Configuration Memory
The table below shows the complete list of configurable parameters stored in non-volatile memory. These values can be changed using the 'M' command. All addresses and arguments must be sent as binary values to the module (not as ASCII representation for hex or decimal).

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for R mode only	0x00	1-10	0x01 (1)	See data sheet for channel frequencies. Only used for R mode.
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels.
MBUS_MODE	M-Bus mode	0x03	0-4 0: S2 1: T1 2: T2 3: S1 4: R 10: T1+C mode 11: T2+C mode	0x01 (1)	Use 'G' command to change value in volatile memory only R mode not supported in OMS Mode 10 and 11 allows reception of both modes
SLEEP_MODE	Sleep mode	0x04	0: Disable Sleep 1: Enable Sleep after TX 2: Reserved 3: Enable Sleep after TX and RX 5: As 1 with sleep timeout 7: As 3 with sleep timeout	0x00 (0)	When enabled the module enter Sleep mode after transmission (or reception). Delay set by RX_TIMEOUT If enabled with sleep timeout, the module goes directly to Sleep after a Reset, and to Sleep after TIMEOUT when wakeup

					from Sleep or exit Config mode.
RSSI_MODE	Append RSSI to received data	0x05	0:Disabled 1: Enabled	0x00 (0)	When enabled the RSSI value is appended to the received data
Radio packet configuration					
PREAMBLE_ LENGTH	Short of long preamble in S mode only	0x0A	0x00 (0): Short 0x01 (1): Long	0x00 (0)	Preamble (header) length, apply for S mode only
TIMEOUT	Time before modem clear buffer without transmitting if Buffer size < Length byte (first byte).	0x10	<1-254> 0x01 (1): 32 ms 0x02 (2): 48 ms 0x03 (3): 64 ms 0x7C (124): 2 s 0xF9 (249): 4 s	0x7C	IF SLEEP_MODE=3 the same timeout is used to auto sleep slaves. Modem transmit without timeout when Buffer size = length byte.
Medium access, addressing and network management					
NETWORK_ROLE		0x12	0x00 (0): Slave/Meter 0x01 (1): Master/Concentrator 0x02 (2): Repeater	0x01 (1)	
MAILBOX	Auto-message and mailbox configuration	0x16	0x10: Clear standard message flag 0x08: Clear mailbox flag 0x04: Use mailbox only for special messages* 0x02: Check DFC bit before transmission 0x01: Check Accessibility before transmission		*Special messages are: SND-NR, RSP-UD and ACK
M_ID1	Manufacturer ID, first byte	0x19	0x00-0xFF (0-255)	0x0C (12)	
M_ID2	Manufacturer ID, second byte	0x1A	0x00-0xFF (0-255)	0xAE (174)	
U_ID1	Unique ID, first byte	0x1B	0x00-0xFF (0-255)	0x12 (18)	
U_ID2	Unique ID, second byte	0x1C	0x00-0xFF (0-255)	0x34 (52)	
U_ID3	Unique ID, third byte	0x1D	0x00-0xFF (0-255)	0x56 (86)	
U_ID4	Unique ID, forth byte	0x1E	0x00-0xFF (0-255)	0x78 (120)	
VER	Version	0x1F	0x00-0xFF (0-255)	0x01 (1)	
DEV	Device	0x20	0x00-0xFF (0-255)	0x07 (7)	
Data and configuration interface, UART Serial Port					
UART_BAUD_RATE	Baud rate	0x30	0x00: Not used 0x01: 2400 0x02: 4800 0x03: 9600 0x04. 14400 0x05: 19200 0x06: 28800 0x07: 38400 0x08: 57600 0x09: 76800 0x0A: 115200	0x05 (5)	BE CAREFUL IF CHANGING AS HOST MAY LOOSE CONTACT WITH MODULE! Does not take effect until module is re-booted / reset.

			0x0B: 230400		
UART_FLOW_CTRL	UART flow control	0x35	0: None 1:CTS only 3:CTS/RTS 4:RXTX(RS485)	0x00 (0)	
DATA_INTERFACE	Data interface	0x36	0x00: MBUS packet with ID and address 0x01: Application data only 0x02: Reserved 0x03: Application data only with ack (00:3Eh) 0x04: Add start/stop byte 0x08: Add CRC 0x0C: Add start/stop byte and CRC	0x00 (0)	Sets receiver data format. First byte is always packet length (except when using start byte)
CONFIG_ INTERFACE	Configuration interface	0x37	0x00: Default 0x01: CONFIG pin not required for @ commands		As a special protection the @ commands require the CONFIG pin to be asserted, but this can be disabled
FREQ_CAL		0x39		Different for each module.	Found in factory and used by the module to minimise the total frequency tolerance. For firmware upgrade, read back the value and write it back after the upgrade.
LED_CONTROL		0x3A	0: Disabled 1: RX/TX indicator 2: UART/RF IDLE indicator	0x00 (0)	Use to enable LED0/LED1 for RX/TX packet indication or UART/RF IDLE mode indicator.
CONTROL_FIELD	C-field	0x3B	0x00-0xFF (0-255)	0x06 (6)	Use ‘F’ command to change value in volatile memory only
RX_TIMEOUT		0x3C	0x00-0xFF (0-255) 0x08 (8) = 4.8 ms 0x56 (86) = 50 ms	0x08 (8)	Delay before Sleep mode, n x 0.6 ms Minimum 3 ms (6) in T mode, minimum 50 ms (86) in S mode.
INSTALL_MODE		0x3D	0: Normal mode (accept installed MBUS meters only) 1: Install mode 2: Filter off (accept all MBUS types)	2	
ENCRYPT_FLAG		0x3E	0: Disabled 1: Enabled 3: Enabled and add Payload CRC	0	Default setting for encryption, enabled when set
DECRYPT_FLAG		0x3F	0: Disabled 1: Enabled 3: Enabled and check Payload CRC	0	Default setting for decryption, enabled when set
DEFAULT_KEY		0x40-0x4F		All 0xFF (255)	
INIT_VECTOR		0x50-0x5F		All 0x00 (0)	

PART_NUMBER		0x61-0x6C		RCxxxx-MBUS3	
HW_REV_NO		0x6E-0x71		x.yz	x, y and z; Any number 0d-9d
FW_REV_NO		0x73-0x76		x.yz	x, y and z; Any number 0d-9d
SERIAL_NUMBER		0x78-0x7F		All 0x00	8 bytes reserved for serial number for traceability. Is programmed by Radiocrafts during test.
Exit from memory configuration		0xFF	No argument should be sent		To exit from command mode the 'X' command must be sent after '>' is received.

To make permanent changes to default values and other parameters, the Memory Configuration command 'M' is used. This command should be followed by pairs of byte being the memory address and the new value to be stored at that address. In order to exit the Memory Configuration mode, the 'address' 0xFF must be sent, but without any data argument. Then wait for the '>' prompt while the internal memory is re-programmed (See Timing Information for typical delay). To completely exit from command mode, the normal exit command 'X' must be sent.

Example:
To change the MAN_ID (at address 0x19 and 0x1A) and set it to (100,200) (0x64,0xC8), send the following sequence:

Command	**Hex**	**Response**	**Comment/Note**
Enter	0x00	'>'	Or assert CONFIG pin De-assert CONFIG after '>' prompt
'M'	0x4D	'>'	Module ready to receive address
0x19	0x19	(none)	
100	0x64	(none)	
0x1A	0x1A	(none)	
200	0xC8	(none)	
[new address could be sent here]			
[new value could be sent here]			
0xFF	0xFF	'>'	Wait for '>' prompt
'X'	0x58	(none)	Module returns to IDLE state

Test mode 0 ('0' command) can be used to list all parameters stored in non-volatile memory. This command can be used to verify and check the module configuration.

MBUS4 Description

The MBUS4 supports the mode N (169 MHz) variant of the Wireless M-Bus standard as specified in the preliminary EN 13757-4 (2013). The implementation supports both N1 (unidirectional communication) and N2 (bi-directional communication). That is, two-way communications with transmission and reception by both Slave and Master is implemented, including the two-way timing for sleeping Slaves

The MBUS4 functionality is similar to the MBUS3 functionality, with the addition of the Master supporting 256 meters (Slaves) internally and > 1000 Slaves registered externally (in the host).

As for MBUS3, the MBUS4 Master can be configured to receive all messages, or only installed meters. Messages to/from installed meters can be encrypted/decrypted.

The MBUS4 receives and decodes both Frame Format A and B automatically in real-time. The Frame Format used for transmission of messages is set by the “PREAMBLE_LENGTH” parameter, as described below.

The number of channels in mode N were extended in 2018 revision of EN13757-4. This revision also added a new data rate at 6.4 kbps. The channel and data rate settings to support this is shown below.

The module can be configured by these parameters:

RF_CHANNEL (0x00)

Channel number	Frequency [MHz]	Subband (EN13757:2018)	Supported data rates [kbps]
1	169.40625	A	2.4/4.8/6.4
2	169.41875	A	2.4/4.8/6.4
3	169.43125	A	2.4/4.8/6.4
4	169.44375	A	2.4/4.8/6.4
5	169.45625	A	2.4/4.8/6.4
6	169.46875	A	2.4/4.8/6.4
7	169.41250	*Legacy EN13757:2013*	2.4/4.8
8	169.43750	*Legacy EN13757:2013*	2.4/4.8
9	169.46250	*Legacy EN13757:2013*	2.4/4.8
10	169.43750	A	19.2
11	169.48125	B	2.4/4.8
12	169.49375	C	2.4/4.8/6.4
13	169.50625	C	2.4/4.8/6.4
14	169.51875	C	2.4/4.8/6.4
15	169.53125	C	2.4/4.8/6.4
16	169.54375	C	2.4/4.8/6.4
17	169.55625	C	2.4/4.8/6.4
18	169.56875	C	2.4/4.8/6.4
19	169.58125	C	2.4/4.8/6.4
20	169.59375	D	2.4/4.8/6.4
21	169.60625	D	2.4/4.8/6.4
22	169.61875	D	2.4/4.8/6.4
23	169.63125	D	2.4/4.8/6.4
24	169.64375	D	2.4/4.8/6.4
25	169.65625	D	2.4/4.8/6.4
26	169.66875	D	2.4/4.8/6.4
27	169.68125	D	2.4/4.8/6.4
28	169.69375	D	2.4/4.8/6.4
29	169.70625	D	2.4/4.8/6.4
30	169.71875	D	2.4/4.8/6.4
31	169.73125	D	2.4/4.8/6.4

32	169.74375	D	2.4/4.8/6.4
33	169.75625	D	2.4/4.8/6.4
34	169.76875	D	2.4/4.8/6.4
35	169.78125	D	2.4/4.8/6.4
36	169.79375	D	2.4/4.8/6.4
37	169.80625	D	2.4/4.8/6.4
38	169.62500	D	19.2
39	169.67500	D	19.2
40	169.72500	D	19.2
41	169.77500	D	19.2

RF_POWER (0x01)

5: 24/27 dBm (RC1701/RC1701HP/RC1701VHP), default
4: 20/24 dBm (RC1701/RC1701HP)
3: 17/20 dBm (RC1701/RC1701HP)
2: 14/17 dBm (RC1701/RC1701HP)
1: 10/14 dBm (RC1701/RC1701HP)

RF_DATA_RATE (0x02)

1: 2.4 kbps
2: 4.8 kbps
4: 19.2 kbps (4GFSK)
5: 6.4 kbps (GFSK)

MBUS_MODE (0x03)

16: N2 mode, default
17: N1 mode (Slave will not receive data)

PREAMBLE_LENGTH (0x0A)

0: Frame format A, default
1: Reserved
2: Frame format B, used for transmission

The MBUS4 support both Application Layer (i.e. Transport Layer) encryption and the new Link Layer Encryption, using the Extended Link Layer, as specified in EN13757-4 (2013). The module accepts all CI-fields and will automatically use the correct encryption scheme. Encryption / decryption is enabled in the Flag Register. The default value of the Flag Register encryption / decryption flags are set using the ENCRYPT_FLAG / DECRYPT_FLAG configuration parameters as for MBUS3. When ELL is used and encryption is enabled, the module also automatically adds the correct PayLoad CRC. The PayLoad CRC can therefore be set to 0x00 by the host in this case.

Test commands 1-4 (TX CW, TX PN), RX and IDLE) are as for MBUS3, but do note that test mode 4 should be used between 1 and 2.

The ‘M’ command is normally used to change Configuration parameters, as for MBUS3. To restore default factor settings, the ‘@RC’ command sequence can be used. To prevent unintentional restoring of the configuration memory, the CONFIG pin must be activated (low) when the ‘RC’ is sent to the module, for the command to take effect. Do note that this command also will overwrite the serial number of the module.

In normal data mode, the first byte sent to the module is interpreted as the length field, except:

0x00 set the module into configuration mode
0x01 – 0xF6 legal length values
0xF7 – 0xF9 illegal length values, module will return to Idle

0xFA	override LBT
0xFB	no response message to be sent, module will go to Idle mode
0xFC	“key challenge” transfer from host to module
0xFD	set module in Idle mode, enable RF receiver (use after sleep)
0xFE	module will send “empty” message (link layer only, no application layer)
0xFF	set module in Idle mode, disable RF receiver (UART only)

The maximum length of the message sent to the module is 0xF6, the minimum length is 0x01. The maximum length corresponds to 255 bytes including the link layer as transmitted on the air.

The High Power (HP) module has an internal temperature compensation to regulate the output power over temperature. Normally the temperature compensation is done before every transmission (output power level 5 only), but can be disabled in time critical applications by setting HP_TEMPCOMP_ENABLE = 0.

The instantaneous RSSI can be read using the ‘S’ command. This reading has a delay due to the settling time of the receiver. For continuous monitoring RSSI, a faster way is to use

‘3’ to set RX mode
‘s’ (small caps) to read the RSSI multiple times
‘4’ to exit RX mode

MBUS4 Master registers

The N2 timing includes a response time of 100ms (fast) or 1100/2100ms (slow) after the Slave message is received. This timing is handled by the Master module. Also, a sleeping Slave will wake up and listen after this response time.

Due to the long range of the N mode, a large number of meters are expected to be handle by one single master. Hence the MBUS4 Master is designed to support more than 1000 meters. Several mechanisms are used to achieve this (patent pending):

- 128 slaves can be registered in non-volatile memory Address and Key Registers, with corresponding Flag Register supporting auto-message generation
- 4 slaves can be registered in volatile memory Address and Key Registers (no write cycle limitations), with corresponding Flag register supporting auto-message generation
- 122 slaves can be registered in non-volatile memory Address and Key Registers, but without corresponding Flag register.
- An “infinite” number of slaves can be supported through a special protocol between the module and the host. In this case, the host must store the Addresses and Keys.

The auto-message generator can be used for the 128 slaves in non-volatile memory, and the 4 slaves in volatile memory, by using the Flag Register. The 4 slaves in volatile memory can be registered by “b” and “k” command, and can be listed using the “l” command. The Flag Register can be access by “a” and “o” command (they are also mapped as Flag Registers 129-132 using the “A” and “O” command. Note, non-volatile registers 129-132 should not be used.

The 122 slaves (registers 133 to 254) in non-volatile memory without corresponding Flag Register are using ENCRYPT_FLAG and DECRYPT_FLAG in configuration memory for enabling encryption and decryption, respectively. Incoming messages from these Slaves can be decrypted using the registered key if the decrypt flag is set. But note that the auto-message generation of standard or mailbox messages are not possible for these slaves. However using the special protocol to the host, it is still possible to respond to these slaves with the correct response time as handled by the module.

An “infinite” number of slaves can be supported in the host, only limited by the memory and processing power of the host controller, interacting with the module over a special protocol. The master module is still handling the response timing and message encryption. The same protocol can be used for the 122 slaves without flag registers.

When a slave message is received, the master module will search for the slave address in the Address register. If the slave was registered, the message will be decrypted before sent to the host (if decryption was enabled in the individual Flag register or by the DECRYPT_FLAG). If the slave address was not found, the (still encrypted) message is sent to the host. The host may now send a Key (“Key challenge” using 0xFC) to the module, and the module will decrypt the message using this Key, and send the decrypted message to the host. Further, the host may now send a new message to the module. This message will be encrypted using the previously transferred Key challenge, or the Key already registered in the module (as for the 122 slaves). If a Key challenge was not sent, a Key may be sent to the module after the message was transferred. The fast/slow timing of the response is handled by the module. If no response is to be sent, the host may terminate the response cycle by sending the 0xFB command.

The “Key challenge”, or the Key following the message, is sent to the module using the 0xFC command (instead of length byte) followed by the 16 Key bytes.

Using this special protocol, the timing is important. When the Slave expect a fast response, they Key challenge, new message (and following key), must be transferred to the module within 90 ms. If a slow response is used, within 1090/2090 ms. It is recommended to use a UART Baud rate of minimum 115 kBd to meet these timing requirements.

If a message is sent to the module during the response time cycle that does not match the last incoming message address, the response time cycle is terminated and the message is sent as a normal message without further time delay.

Italian CIG extension of Wireless M-Bus

The Italian “CIG Interchangeability Task Force” has published UNI/TS 11291-11-4, Gas measurement systems – Hourly based gas metering systems, Part 11-4, Communication profile PM1. This is a companion standard, on top of EN 13757-4 mode N and EN 13757-3 but using DLMS / COSEM as application layer. It contains some additional requirements to EN13757-4, such as Listen Before Talk (LBT) and output power control in fine steps. This is supported by MBUS4, and can be enabled through configuration memory parameters.

To enable the output power control in fine steps, set PA_TABLE_EXTENDED > 0. This is the default value, and can be set from 1 to 19 (or 20 in case of VHP variant). It is possible to change the output power by using the ‘P’ command. The output power changes approximately 3 dB for each step. See data sheet for details. If PA_TABLE_EXTENDED = 0, the module is backward compatible using the 5 steps in RF_POWER.

The LBT feature is enabled by setting LBT_ENABLE = 0x01. The following parameters configure the LBT: LBT_RSSI_THRESHOLD, LBT_MAX_ATTEMPT, LBT_BO_PERIOD, LBT_BO_FLAT, LBT_MAX_DELAY. These parameters should be set according to CIG recommendations.

The host controller may send 0xFA to override LBT for the following message. It applies only for one message and is automatically cleared after this one transmission.

If the LBT algorithm fails, a command/result message is sent from the module on UART. The result message is 0x01 for failure. After successful TX, the 0x00 is sent. This result message is sent always when LBT is enabled, even if it is overridden.

Note, CIG only use Frame Format B, so PREAMBLE_LENGTH must be set to 0x02.

Category 1 receiver
In some cases it is advisable to trade off some sensitivity for better selectivity and blocking properties. In noisy environments the radio communication range is not limited by thermal noise, but interference from other strong transmitters. In this case it is possible to configure the module for Category 1 receiver settings giving increased blocking properties as specified in EN 300220. To enable this feature, set CAT1_ENABLE = 0x01.

Antenna tuning feature
The test mode ‘7’ and ‘8’ enable frequency sweep and pulsed transmissions and can be used to tune and optimize an antenna. See application note on the detailed procedure.

MBUS4 Configuration commands and Configuration Memory
MBUS4 has the same configuration commands and register numbers as MBUS3 with the addition of command 7 and 8 as shown below.

Parameter	Command	Argument in hex (decimal)	Note
Test mode 7	'7' – 0x37	(none)	Antenna tuning function. Scan frequency range +/-2 MHz around the centre frequency. 5 seconds between transmissions and 200 kHz separation between test frequencies.
Test mode 8	'8' – 0x38	(none)	Pulsed transmission. 100ms packet for every 5 seconds. Used to verify antenna.

The following table shows *only* the *differences* in module configuration between MBUS4 and MBUS3.

Parameter	Description	Address hex	Argument dec	Factory setting hex (dec)	Comment
Radio configuration					
RF_CHANNEL	Default RF channel for N mode	0x00	1-41	0x03 (3)	See data sheet for channel frequencies
RF_POWER	Default RF output power	0x01	1-5	0x05 (5)	See data sheet for output power levels
DATA_RATE	Data rate	0x02	1: 2.4 kbps 2: 4.8 kbps 3: NA 4: 19.2 kbps 5: 6.4 kbps	0x01 (1)	
MBUS_MODE	M-Bus mode	0x03	16: N2 mode 17: N1 mode	0x10 (16)	Use 'G' command to change value in volatile memory only
PA_TABLE_EXTENDED		0x06	0: Disabled 1-20: Default power step	0	Disabled. 1 – 20 is on. See data sheet for output power levels
CAT1_ENABLE		0x07	0: Disabled 1: Enabled	0	
Radio packet configuration					
PREAMBLE_ LENGTH	Frame Format	0x0A	0x00 (0): FFA 0x01 (1): N.A. 0x02 (2): FFB	0x00 (0)	Transmit Frame format A or B
LBT_ENABLE		0x18		0	
HP_TEMPCOMP_ ENABLE	Power amplifier temperature compensation	0x29	0x00 (0): Off 0x01 (1): On	0x00 (0)	
LBT_RSSI_THRESHOLD		0x2A		80	80 (50-110) [-dBm]
LBT_MAX_ATTEMPT		0x2B		5	5 (3-8)
LBT_BO_PERIOD		0x2C		4	4 (=40 ms) In 10 ms step
LBT_BO_FLAT		0x2D		3	3 (1-8)
LBT_MAX_DELAY		0x2E		75	750 (=75) 250 to 1000 ms, in 10 ms step

Appendix 1: MBUS Command list overview

Command list	Feature set		
	MBUS1	MBUS2	MBUS3/MBUS4
'A' – 0x41	N.A	Acknowledge	Auto-Message Flag
'B' – 0x42	N.A	Bind	Bind
'C' – 0x43	Channel	Channel	Channel
'D' – 0x44	N.A	Decrypt RF message before send to UART	N.A (controlled in flag register)
'E' – 0x45	N.A	Encrypt UART message before RF transmit	Encrypt mailbox message
'F' – 0x46	C-field	C-field	C-field
'G' – 0x47	M-Bus mode	M-Bus mode	M-Bus mode
'I' – 0x49	N.A	Install	Install
'K' – 0x4B	N.A	Key register	Key register
'L' - +x4C	N.A	N.A	List Binding
'M' – 0x4D	Memory configuration	Memory configuration	Memory configuration
'N' – 0x4E	N.A	N.A	Access number
'O' – 0x4F	N.A	N.A	Read Auto message flag register
'P' – 0x50	Output power	Output power	Output power
'Q' – 0x51	Quality Indicator	Quality Indicator	Quality Indicator
'R – 0x52'	N.A	N.A	Read Mailbox
'S' – 0x53	Signal Strength (RSSI)	Signal Strength (RSSI)	Signal Strength (RSSI)
'T' – 0x54	Destination address	Destination address	Destination address
'U' – 0x55	Temperature monitoring	Temperature monitoring	Temperature monitoring
'V' – 0x56	Battery monitoring	Battery monitoring	Battery monitoring
'W' – 0x57	N.A	N.A	Write Mailbox
'X' – 0x58	Exit command	Exit command	Exit command
'Y' – 0x59	Memory Read one byte	Memory Read one byte	Memory Read one byte
'Z' – 0x5A	Sleep mode	Sleep mode	Sleep mode
'0' – 0x30	Test mode 0	Test mode 0	Test mode 0
'1' – 0x31	Test mode 1	Test mode 1	Test mode 1
'2' – 0x32	Test mode 2	Test mode 2	Test mode 2
'3' – 0x33	Test mode 3	Test mode 3	Test mode 3
'4' – 0x34	Test mode 4	Test mode 4	Test mode 4
'7' – 0x37	N.A	N.A	Test mode 7 – Antenna tuning feature (MBUS4 only)
'8' – 0x38	N.A	N.A	Test mode 8 (MBUS4 only)

Commands in grey are stored in non-volatile memory (Flash). The rest of the commands are stored in volatile memory (RAM) and is lost after a power off or a reset.

Appendix 2: Configuration Memory Factory Default

Address	MBUS1 factory default Values (FW 1.26)							
0x00-0x07	0x01	0x05	0x03	0x01	0x00	0x00	0x64	0x00
0x08-0x0F	0x05	0x3C	0x00	0xD3	0x91	0xDA	0x80	0x80
0x10-0x17	0x7C	0x00	0x00	0x01	0x00	0x00	0x00	0x00
0x18-0x1F	0x00	0x0C	0xAE	0x12	0x34	0x56	0x78	0x01
0x20-0x27	0x07	0x01	0x01	0x00	0x00	0x00	0x00	0x04
0x28-0x2F	0xFF	0x08	0x00	0x00	0x00	0x00	0x00	0x00
0x30-0x37	0x05	0x08	0x00	0x01	0x05	0x00	0x00	0x01
0x38-0x3F	0x2B	0x00	0x01	0x44	0x00	0x52	0x43	0x31
0x40-0x47	0x31	0x38	0x30	0x2D	0x4D	0x42	0x55	0x53
0x48-0x4F	0x31	0x2C	0x32	0x2E	0x30	0x30	0x2C	0x31
0x50-0x57	0x2E	0x32	0x36	0x20	0x20	0x00	0xFF	0xFF
0x58-0x5F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x60-0x67	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x68-0x6F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x70-0x77	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x78-0x7F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
Address	**MBUS2 factory default Values (FW 2.21)**							
0x00-0x07	0x01	0x05	0x03	0x02	0x00	0x00	0x64	0x00
0x08-0x0F	0x05	0x3C	0x00	0xD3	0x91	0xDA	0x80	0x80
0x10-0x17	0x7C	0x00	0x01	0x01	0x00	0x00	0x00	0x00
0x18-0x1F	0x00	0x0C	0xAE	0x12	0x34	0x56	0x78	0x01
0x20-0x27	0x07	0x01	0x01	0x00	0x00	0x00	0x00	0x04
0x28-0x2F	0xFF	0x08	0x00	0x00	0x00	0x0E	0x10	0x11
0x30-0x37	0x05	0x08	0x00	0x01	0x05	0x00	0x00	0x01
0x38-0x3F	0x2B	0x00	0x00	0x06	0x0B	0x02	0x00	0x00
0x40-0x47	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x48-0x4F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x50-0x57	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x58-0x5F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x60-0x67	0x00	0x52	0x43	0x31	0x31	0x38	0x30	0x2D
0x68-0x6F	0x4D	0x42	0x55	0x53	0x32	0x2C	0x32	0x2E
0x70-0x77	0x30	0x30	0x2C	0x32	0x2E	0x32	0x31	0x00
0x78-0x7F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x80-0x87	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x88-0x8F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x90-0x97	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x98-0x9F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xA0-0xA7	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xA8-0xAF	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xB0-0xB7	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xB8-0xBF	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xC0-0xC7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xC8-0xCF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD0-0xD7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD8-0xDF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE0-0xE7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE8-0xEF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF0-0xF7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF8-0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

Address	MBUS3 factory default Values (FW 3.14)							
0x00-0x07	0x01	0x05	0x03	0x01	0x00	0x00	0x09	0x00
0x08-0x0F	0x05	0x3C	0x00	0xD3	0x91	0xDA	0x80	0x80
0x10-0x17	0x7C	0x00	0x01	0x01	0x00	0x00	0x17	0x00
0x18-0x1F	0x00	0x48	0x24	0x12	0x34	0x56	0x78	0x01
0x20-0x27	0x07	0x01	0x01	0x00	0x00	0x00	0x00	0x04
0x28-0x2F	0xFF	0x08	0x00	0x00	0x00	0x00	0x00	0x00
0x30-0x37	0x05	0x08	0x00	0x01	0x05	0x00	0x00	0x01
0x38-0x3F	0x2B	0x00	0x00	0x06	0x08	0x02	0x00	0x00
0x40-0x47	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x48-0x4F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x50-0x57	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x58-0x5F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x60-0x67	0x00	0x52	0x43	0x31	0x31	0x38	0x30	0x2D
0x68-0x6F	0x4D	0x42	0x55	0x53	0x33	0x2C	0x32	0x2E
0x70-0x77	0x30	0x30	0x2C	0x33	0x2E	0x31	0x34	0x00
0x78-0x7F	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0x80-0x87	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x88-0x8F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x90-0x97	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0x98-0x9F	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xA0-0xA7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xA8-0xAF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xB0-0xB7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xB8-0xBF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xC0-0xC7	0xE0	0xEA	0xFA	0x07	0x84	0x84	0xE0	0xEA
0xC8-0xCF	0xFA	0x47	0x84	0xC4	0xFF	0xFF	0xFF	0xFF
0xD0-0xD7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xD8-0xDF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE0-0xE7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xE8-0xEF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF0-0xF7	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF
0xF8-0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF

Grey: Reserved (do not change).
Blue: Reserved for Module part number and version information (do not change). Typical values shown, the actual content of memory location 60d-83d depends on part number and version number
Red: Module specific values (Calibration and unique serial number). Read back before Firmware upgrade take place

MBUS4 Factory Reset values (FW 1.01)

```
03 05 01 10|00 00 09 00  ________
05 3C 00 D3|91 DA 80 80  _<_Ó‘Ú□□
7C 00 01 01|00 00 17 00  |_______
00 48 24 12|34 56 78 01  _H$_4Vx_
07 01 01 00|00 00 00 04  ________
FF 00 00 00|00 00 00 00  ÿ_______
05 08 00 01|05 00 00 00  ________
00 00 00 44|00 02 00 00  ___D____
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
00 00 00 00|00 00 00 00  ________
00 00 00 00|00 00 00 00  ________
93 93 93 93|73 03 5E 03  ““““s_^_
0D 07 06 07|10 10 10 10  ________
0E 0E 0E 0E|10 10 10 10  ________
01 00 00 00|09 9A 8C 9A  ____šŒš
3A B0 A0 B0|30 2F 2E 2C  :° °0/.,
00 52 43 31|37 30 31 48  _RC1701H
50 2D 4D 42|55 53 34 2C  P-MBUS4,
31 2E 30 30|2C 31 2E 30  1.00,1.0
30 20 20 20|20 20 20 20  0
20 00 00 00|00 00 00 00   _______
00 FF FF FF|FF FF FF FF  _ÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
FF FF FF FF|FF FF FF FF  ÿÿÿÿÿÿÿÿ
```

Appendix 3: ASCII Table

HEX	DEC	CHR	CTRL
0	0	NUL	^@
1	1	SOH	^A
2	2	STX	^B
3	3	ETX	^C
4	4	EOT	^D
5	5	ENQ	^E
6	6	ACK	^F
7	7	BEL	^G
8	8	BS	^H
9	9	HT	^I
0A	10	LF	^J
0B	11	VT	^K
0C	12	FF	^L
0D	13	CR	^M
0E	14	SO	^N
0F	15	SI	^O
10	16	DLE	^P
11	17	DC1	^Q
12	18	DC2	^R
13	19	DC3	^S
14	20	DC4	^T
15	21	NAK	^U
16	22	SYN	^V
17	23	ETB	^W
18	24	CAN	^X
19	25	EM	^Y
1A	26	SUB	^Z
1B	27	ESC	
1C	28	FS	
1D	29	GS	
1E	30	RS	
1F	31	US	
20	32	SP	
21	33	!	
22	34	"	
23	35	#	
24	36	$	
25	37	%	
26	38	&	
27	39	'	
28	40	(	
29	41	)	
2A	42	*	
2B	43	+	
2C	44	,	
2D	45	–	
2E	46	.	
2F	47	/	
30	48	0	
31	49	1	
32	50	2	
33	51	3	
34	52	4	
35	53	5	
36	54	6	
37	55	7	
38	56	8	
39	57	9	
3A	58	:	
3B	59	;	
3C	60	<	
3D	61	=	
3E	62	>	
3F	63	?	

HEX	DEC	CHR
40	64	@
41	65	A
42	66	B
43	67	C
44	68	D
45	69	E
46	70	F
47	71	G
48	72	H
49	73	I
4A	74	J
4B	75	K
4C	76	L
4D	77	M
4E	78	N
4F	79	O
50	80	P
51	81	Q
52	82	R
53	83	S
54	84	T
55	85	U
56	86	V
57	87	W
58	88	X
59	89	Y
5A	90	Z
5B	91	[
5C	92	\
5D	93	]
5E	94	^
5F	95	_
60	96	`
61	97	a
62	98	b
63	99	c
64	100	d
65	101	e
66	102	f
67	103	g
68	104	h
69	105	i
6A	106	j
6B	107	k
6C	108	l
6D	109	m
6E	110	n
6F	111	o
70	112	p
71	113	q
72	114	r
73	115	s
74	116	t
75	117	u
76	118	v
77	119	w
78	120	x
79	121	y
7A	122	z
7B	123	{
7C	124	|
7D	125	}
7E	126	~
7F	127	DEL

Document Revision History

Document Revision	Changes
1.0	First release
1.10	Detailed UART interface and Encryption example included. Minor changes and corrections
1.11	Timing and figure 3 corrections. Auto sleep included. Minor corrections in text.
1.20	Added MBUS3
1.21	Clarified order of address bytes for binding.
1.22	MBUS3 updates before official release.
1.30	MBUS2 new features for FW 2.20: - Auto sleep functionality update for slaves. - 8 Byte serial number reservations in configuration memory. - 1 byte frequency tolerance calibration value in configuration memory.
1.31	- Changed terminology from RC1180-MBUSx to MBUSx - Included quick start chapter (and removed it from data sheet 2.20) - RSSI reading included (and removed it from data sheet rev 2.20) - Optional custom specific version chapter included for MBUS2 - SLEEP_MODE, TIMEOUT and LED CONTROL update for MBUS2
1.40	MBUS3 new features for FW 3.09: - 8 Byte serial number reservations in configuration memory. - 1 byte frequency tolerance calibration value in configuration memory. - -SLEEP_MODE, TIMEOUT and LED CONTROL update
1.50	- Correction of SERIAL_NUMBER location - Correction PART_NUMBER, HW_REV_NO and FW_REV_NO location - info about only HEADER packets sending (L=0xFE) included - Auto sleep info for MBUS3 Included. - U, V and Y command info included - Appendix 1&2 update - Two level auto-message handler info included (New Feature FW 3.11)
1.60	- MBUS4 Description added - MBUS3 Description for C1 mode added
1.71	Additional information and corrections for MBUS3 and MBUS4 versions
1.72	Updated RF_CHANNEL frequencies
1.73	-Updated MBUS4 description with CIG features; extended power table, LBT and Category 1 receiver - Minor corrections
1.75	Added description of new features in FW 1.05 (MBUS4) More channels(11-41), new data rate(5) and new test modes ('7' and '8') added for MBUS4

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. –MBUS3™ and -MBUS4™ are trademarks of Radiocrafts AS.. 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

Address:
Radiocrafts AS
Sandakerveien 64
NO-0484 OSLO
NORWAY

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