Wireless M-Bus Studio

User Manual

Document ID: 4100/6404/0053

IMST GmbH
Carl-Friedrich-Gauß-Str. 2-4
47475 KAMP-LINTFORT
GERMANY

Document Information

File name	WMBus_StudioManual.docx
Created	2011-05-02
Total pages	28

Revision History

Version	Note
0.1	Created, Initial Version
0.2	Draft Version Created For Review
0.5	Preliminary Version
0.6	Packet Monitor (Sniffer) Capabilities highlighted
1.0	Reviewed and released
1.1	AES Encryption added
1.2	C-Mode added

Aim of this Document

This document is intended to describe the Wireless M-Bus Studio, a Windows program which is to be used with the iM871A WiMOD module.

Table of Contents

1. Introduction

1.1 Overview

The Wireless M-Bus Studio is a Windows tool which allows to explore the capabilities of the iM871A, an ultra-low-power, high-performance, pre-certified Wireless M-Bus radio module which operates compliant to EN 13757 part 4. The GUI offers a comfortable way to configure and control the features of the embedded Wireless M-Bus Stack like:

- Wireless M-Bus Modes
- RF Message Header
- Radio Settings
- Automatic Power Saving
- Embedded Radio Link Test
- RF Message Generator
- Real Time Clock Support
- Packet Sniffer
- AES Encryption

1.2 Installation

The Wireless M-Bus Studio is delivered as part of the Wireless M-Bus Starter Kit which also contains the radio modules with pre-programmed firmware and a Demo Board which can be equipped with different radio modules. This Demo Board provides a USB Connector for communication and power supply purposes. An USB chip converts the serial interface signals from the iM871A into USB signals. For communication over this USB interface a Virtual COM port (VCP) driver must be installed on the Host PC.

1.2.1 USB Driver

The USB driver can be found in (<local folder>\drivers\CDM *version*.exe) on the Starter Kit CD/Memory Stick, but it is recommended to download the recent version from the USB drivers web site.

Run the CDM *version*.exe. A command box appears and the driver will be installed.

Plug in the USB cable to the USB port on the Demo Board. Plug in the other end of the USB cable into the USB port of the Host PC. Now the Windows OS will recognize the Demo Board as a USB serial converter.

To verify that the USB driver installation was successful, open the Windows Device Manager under "Start>Control>Panel>System>Hardware>Device Manager". A new USB – Serial Port (COMxx) entry under "Ports (COM & LPT)" should appear.

Figure 1-1: USB Driver Installation

1.2.2 Wireless M-Bus Studio

The Wireless M-Bus Studio is delivered as a self extracting setup.exe file. After execution the following files should be installed under Programs/WM-Bus Studio:

- WMBusStudio.exe - the executable application file
- WMBusStudio.ini - an internal configuration file
- WMBusHCI.dll - a Windows DLL including the host control interface protocol
- MessageConfig.cfg - includes some example RF messages

Note: It might be necessary to install the Microsoft Visual C++ 2008 Redistributable Package (x86) in case the Wireless M-Bus Studio doesn't start. Click the download button on the Microsoft web page. Double click the vcredist_x86.exe to install runtime components of Visual C++ libraries on a computer that does not have Visual C++ installed.

1.2.3 Hardware Setup

Please check the hardware configuration as described in [1].

1.2.4 Finish Installation

Mount an iM871A on the Demo Board and connect this by means of a USB cable to the PC. Start WMBusStudio.exe and continue with the following chapter.

2. Getting Started

The Wireless M-Bus Studio enables the user to evaluate the capabilities of the iM871A. Several features of the embedded radio firmware can be controlled from different pages which are described in more detail in the following chapters. At first a few general hints about this tool.

2.1 Navigation

As stated above the provided features of the embedded radio firmware are presented on several pages. The navigation from one page to another is implemented by means of two navigation bars. The vertical bar on the left side is used to change between the main sections: *Services*, *Configuration* and *Hardware Test*. Each section provides an individual horizontal bar on top which provides access to several subpages e.g. *Radio Link Test*, *Message Generator* and so on. Some pages provide timer controlled services. These services will automatically be stopped if another page is selected.

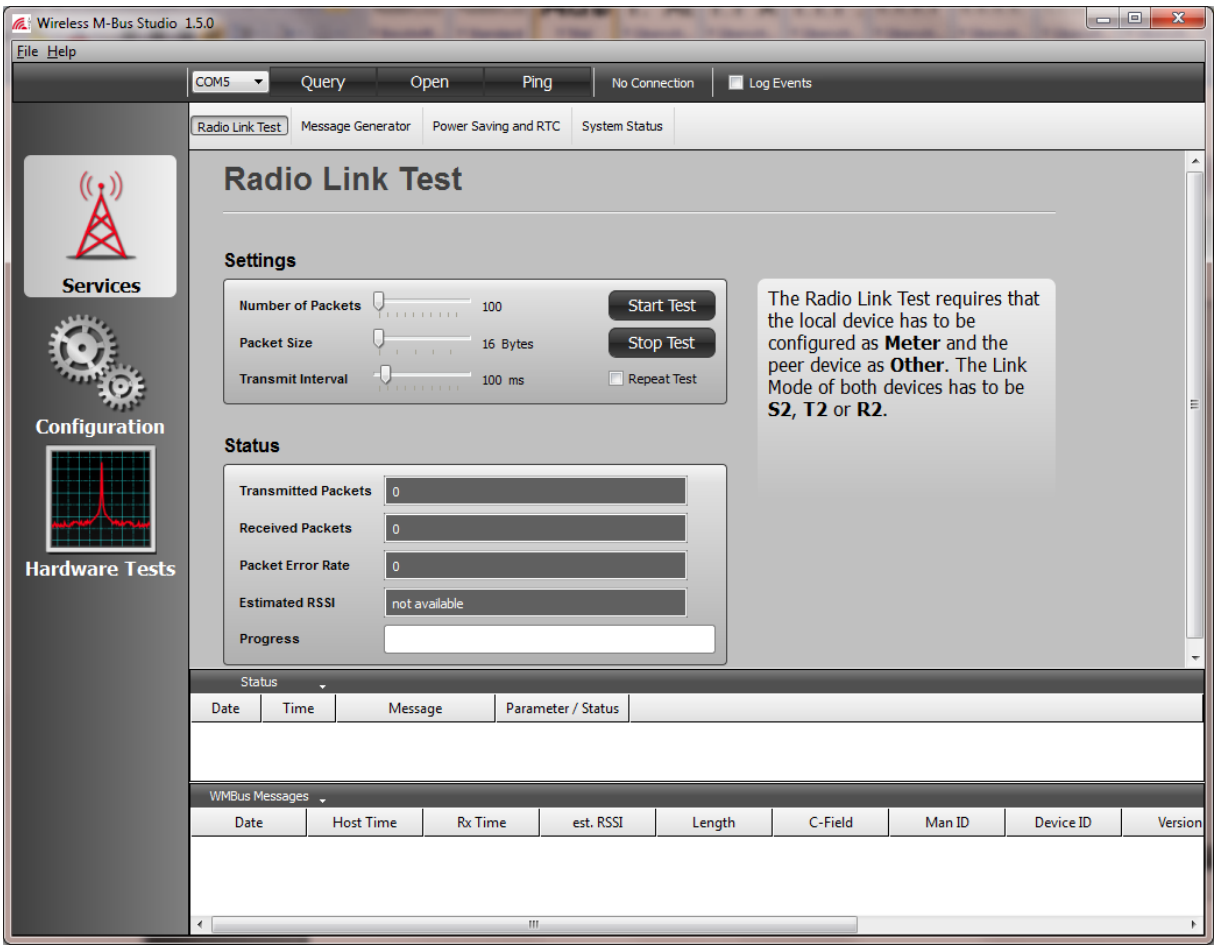


Figure 2-1: Navigation Bars

2.2 Connection to iM871A

After connecting an iM871A via USB cable it is necessary to open a Serial COM port. The Com Port can be selected from a drop down list on top of the window. The COM port list can be updated by pressing the *Query* button. The connection can be checked by pressing the *Ping* button. On success some information about the local connected device is displayed within the top toolbar.

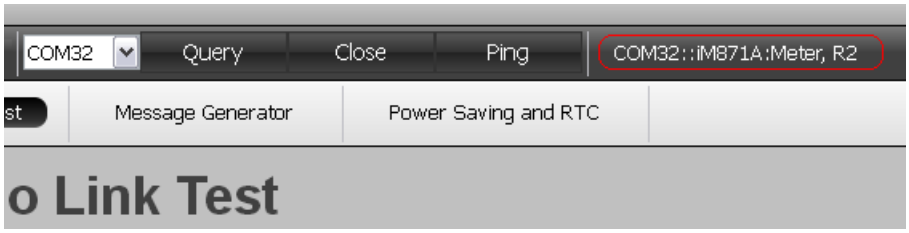


Figure 2-2: Device Information of connected device

2.3 Status List Boxes

The bottom area of the main window contains two list boxes, one for the commands which are sent to the connected radio device and a second to display the received WM-Bus RF messages according to the Wireless M-Bus Message Format. An estimated received signal strength indicator (RSSI) and an *RxTime* stamp indicating the time of reception (generated within the firmware) can be optionally displayed (see chapter Device Configuration).

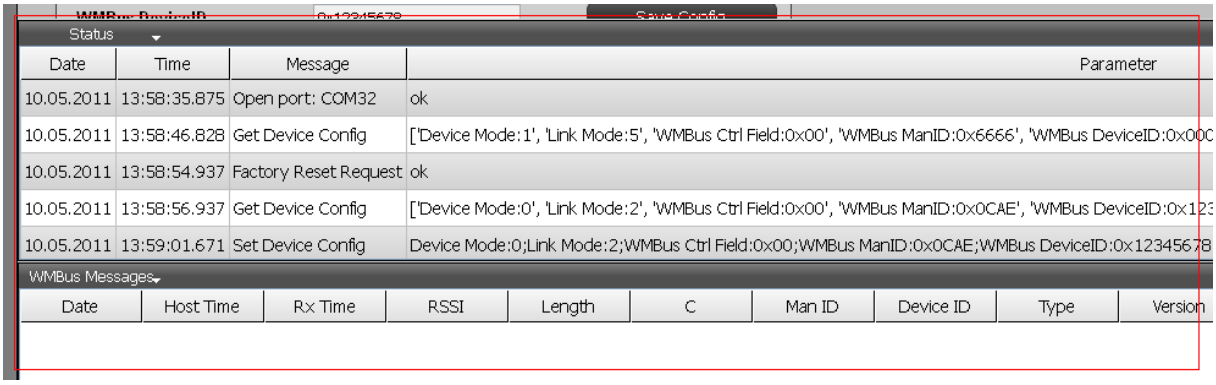


Figure 2-3: Status List Boxes

2.4 Logging

This tool supports automatic logging of all the information of the Status List Boxes into a file.

Figure 2-4: Logging

The logging will be activated by selecting the checkmark ***Log Events*** in the upper control bar. Unselecting this feature will close the log file which can then be opened with any kind of text editor. The file format is plain ASCII text.

2.5 Hardware Notes

The embedded Firmware can be configured to use LED 1 – 3 of the Demo Board (see chapter Device Configuration).

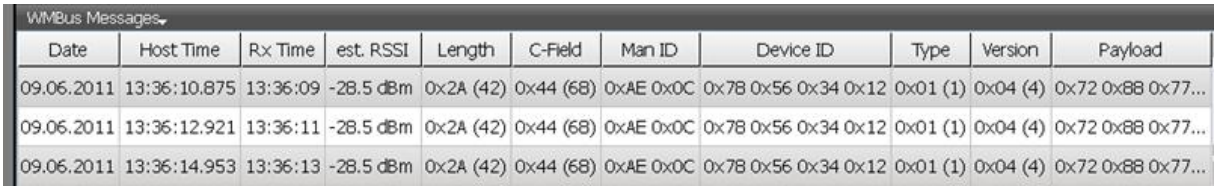
3. Feature Set

The feature set of the embedded radio firmware is divided into the following main sections: *Services*, *Configuration* and *Hardware Test*.

3.1 Services

3.1.1 Packet Monitor (Sniffer)

The firmware stack of the iM871A supports to monitor WM-Bus Messages according to EN 13757-4. The received radio link messages will be sent to the Wireless M-Bus Studio and displayed in WM-Bus Message Status List at the bottom of the main window. In conjunction with the *Logging* feature it is possible to store these messages into a file.


WMBus Messages

Date	Host Time	Rx Time	est. RSSI	Length	C-Field	Man ID	Device ID	Type	Version	Payload
09.06.2011	13:36:10.875	13:36:09	-28.5 dBm	0x2A (42)	0x44 (68)	0xAE 0x0C	0x78 0x56 0x34 0x12	0x01 (1)	0x04 (4)	0x72 0x88 0x77...
09.06.2011	13:36:12.921	13:36:11	-28.5 dBm	0x2A (42)	0x44 (68)	0xAE 0x0C	0x78 0x56 0x34 0x12	0x01 (1)	0x04 (4)	0x72 0x88 0x77...
09.06.2011	13:36:14.953	13:36:13	-28.5 dBm	0x2A (42)	0x44 (68)	0xAE 0x0C	0x78 0x56 0x34 0x12	0x01 (1)	0x04 (4)	0x72 0x88 0x77...

Figure 3-1: Packet Monitor (Sniffer)

The output format of the firmware stack can be configured. Every message can be extended by an estimated RSSI value and a Real Time Clock timestamp (see *Device Configuration*).

Due to the fact that the EN 13757 part 4 defines different radio settings for Meters and Other devices the following configurations for packet monitoring are recommended:

Monitoring of devices in Meter Mode

Meter Configuration (Device Mode = Meter)	Monitor (Sniffer) Configuration (Device Mode = Other)
S1, S1-m, S2	S2
T1, T2	T2
R2 (Radio Channel 1..x)	R2 (Radio Channel 1..x)
C1, C2	C2

Monitoring of devices in Other Mode

Other Configuration (Device Mode = Other)	Monitor (Sniffer) Configuration (Device Mode = Meter and RxWindow = 0 !!!)
S2	S2
T2	T2
R2 (Radio Channel 1..x)	R2 (Radio Channel 1..x)
C2	C2

3.1.2 Radio Link Test

This application is used to verify the radio link quality between two iM871A modules.

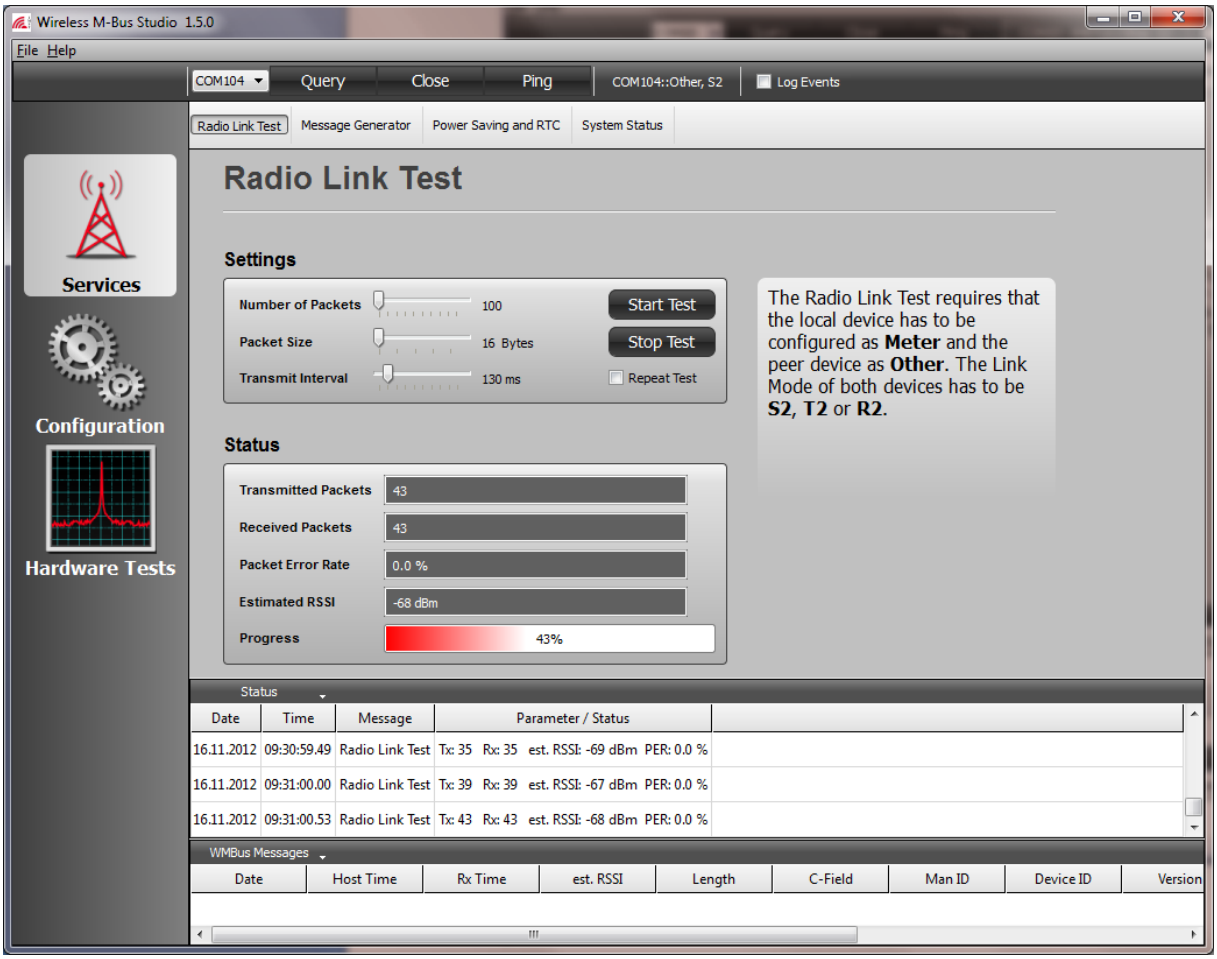


Figure 3-2: Radio Link Test

A configurable number of test packets is sent from the local connected iM871A to a peer device which might be powered by battery. The peer device increments an internal *RxCounter* and returns this number back to the sender every time it receives a test packet. The local device sends a status message to the PC every 500ms. This message includes the *TxCounter* of the local device and the *RxCounter* and an estimated *RSSI* (Received Signal Strength Indicator) of the last received packet. The displayed packet error rate is derived by means of the following formula:

$$PER[\%] = (1 - RxCounter/TxCounter) \cdot 100 \qquad \textit{(Equation 3-1)}$$

Configuration

The following parameters can be configured:

- *Number of packets*: number of RF test messages to transmit per test run
- *Packet Size*: number of bytes in one test packet
- *Transmit Interval*: time in milliseconds between the transmission of two packets
- The *Repeat Test* checkbox determines if the test should be repeated automatically or not

> **Note:** The firmware timer supports a time resolution of 10ms !

Prepare Test

- Check the device configuration (Device Mode, Link Mode) for both iM871A modules (local and peer device). The *Device Mode* of the peer device must be set to *Other*. The Device Mode of the local device must be *Meter*. Furthermore the *Link Mode* of both devices must be the **same** and can be configured to *S2*, *T2*, *R2*, *C2*[1].
- Change the configuration parameter on both devices if desired.
- Ensure that no other interfering devices are in range !

Start Test

- Press *Start* button.
  The transmission of packets and its timing is now controlled by the firmware itself and not by the GUI.

Stop Test

- Press *Stop* button (or select another page).
  A stop command will be sent to the local device to finish the test procedure.

[1] C-Mode features two different Telegram Formats, make sure that both Radio Link Test devices use the same Telegram Format

3.1.3 Message Generator

The Message Generator can be used to send WM-Bus messages from one iM871A to another.

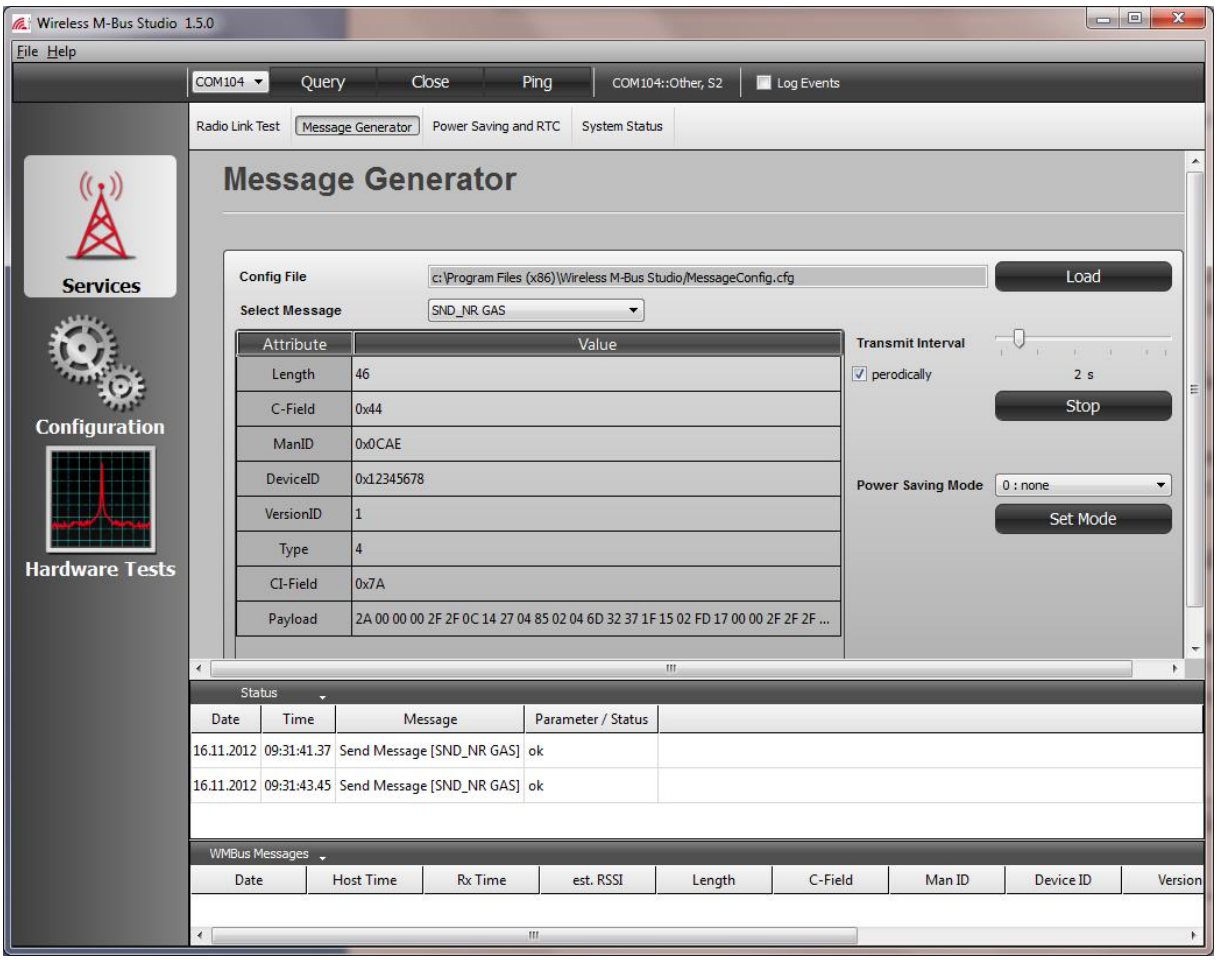


Figure 3-3: Message Generator

The tool allows to select several pre-defined WM-Bus messages which are loaded from a configuration file. The transmission can be done once or in a periodically way. The *Transmit Interval* can be changed if desired.

For devices which are configured to operate in *Meter* mode the feature *Auto Low Power Mode* can be enabled.

Auto Low Power Mode means, that the device enters the configured *Power Saving Mode* automatically when a WM-Bus message was sent. The next command on the serial interface will wake the device.

The message transmission is controlled by the GUI itself and will be stopped if another page is selected.

WM-Bus Messages which are received from another device will be displayed according to the Wireless M-Bus Message format in the lower status list box. In addition to the message an estimated RSSI value and a timestamp can be displayed optionally (see chapter Device Configuration).

Note: The time stamp is generated by means of a *Real Time Clock* which must be enabled !

LED Usage

The embedded Firmware can be configured to use *LED 1 – 3* of the Demo Board as follows:

- *LED 1 (Alive Indicator)* indicates if the module is active (on) or sleeping (off)
- *LED 2 (Tx Indicator)* toggles every time a message was sent
- *LED 3 (Rx Indicator)* toggles every time a message was received

3.1.4 Power Saving and Real Time Clock

This page exposes the Power Saving and Real Time Clock feature of the iM871A firmware.

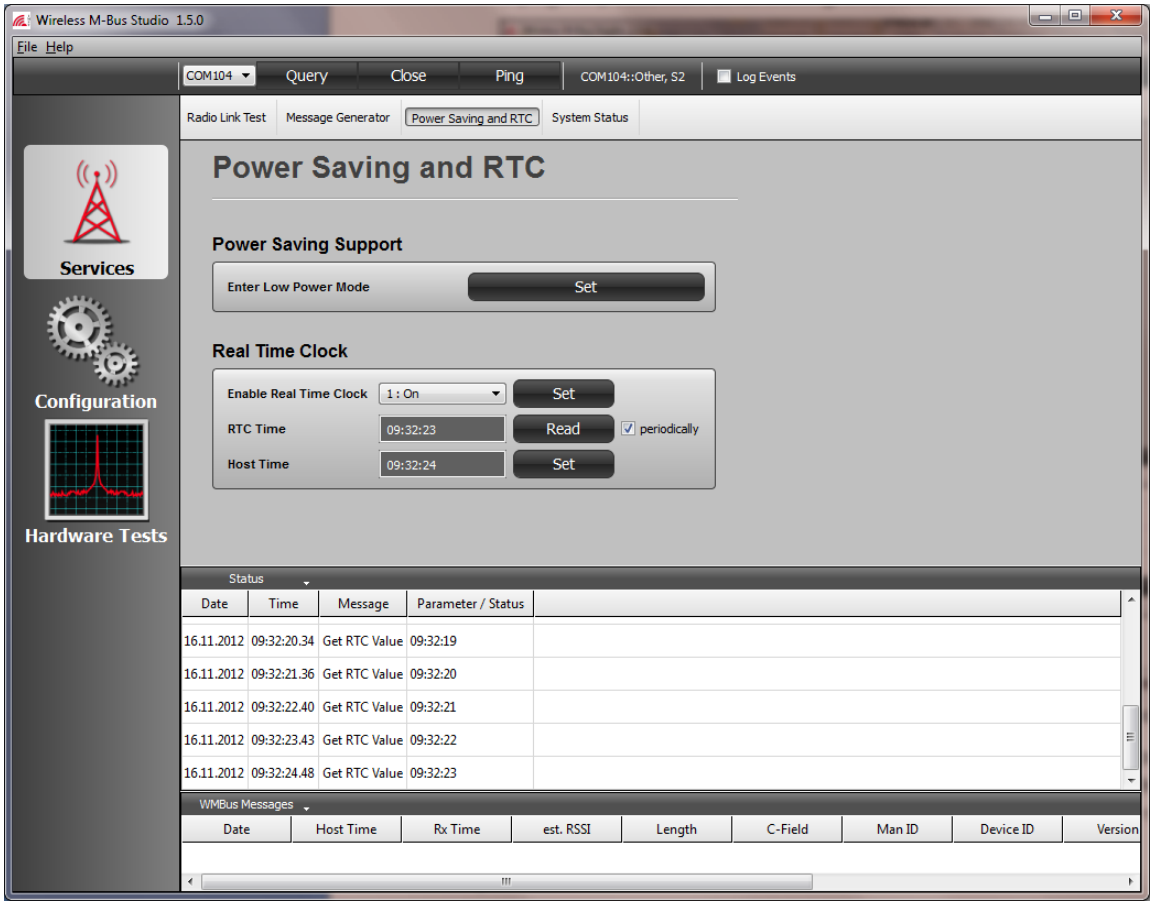


Figure 3-4: Power Saving and Real Time Clock

The firmware provides a *Low Power Mode* which can be entered automatically (see Device Configuration) or manually by sending a command via serial interface. In this mode the radio part is in shutdown state and the CPU is in sleep mode. The module can be wake up again by sending a new command.

Real Time Clock Support

The iM871A provides an embedded RTC which can be used for timer controlled operations e.g. automatic transmission[1] of WM-Bus messages at specific times or with a configurable interval. The RTC is currently used (when enabled) to generate time stamps for received RF messages. The GUI allows to enable/disable the RTC and provides an easy mean to read the RTC and to synchronize it to the PC time.

Note: The lowest current consumption during Low Power Mode can only be achieved with **disabled** RTC !

[1] Automatic message transmission is not part of the firmware yet.

3.1.5 System Status

This page displays the system status of the iM871A.

Figure 3-5: System Status

The *System Status* consists of the following information elements:

- *Flash Status* - indicates if the internal non-volatile memory which is used for device configuration, is ok or not. The stored information is protected by a checksum. During system start-up this checksum is tested. If the checksum is valid the configuration is loaded from the Flash into the RAM and propagated to the related firmware and hardware components. In case of a wrong checksum a default configuration is used.

The *System Status* contains the following items which are reset during system start-up.

- *System Ticks* - indicates the number of counted system ticks

> **Note:** The firmware supports a tick resolution of 10ms ! The *System Tick* is not updated when the CPU is in the *Low Power Mode*.

- *Transmitted Messages* - indicates the number of transmitted RF messages
- *Received Messages* - indicates the number of received RF messages with CRC ok
- *RxCRC Errors* - indicates the number of received messages with CRC error
- *RxPhy Errors* - indicates the number of errors from the physical RF layer e.g. decoding errors

3.2 Configuration

3.2.1 Device Configuration

The iM871A firmware provides several device parameters which can be read and changed on this page.

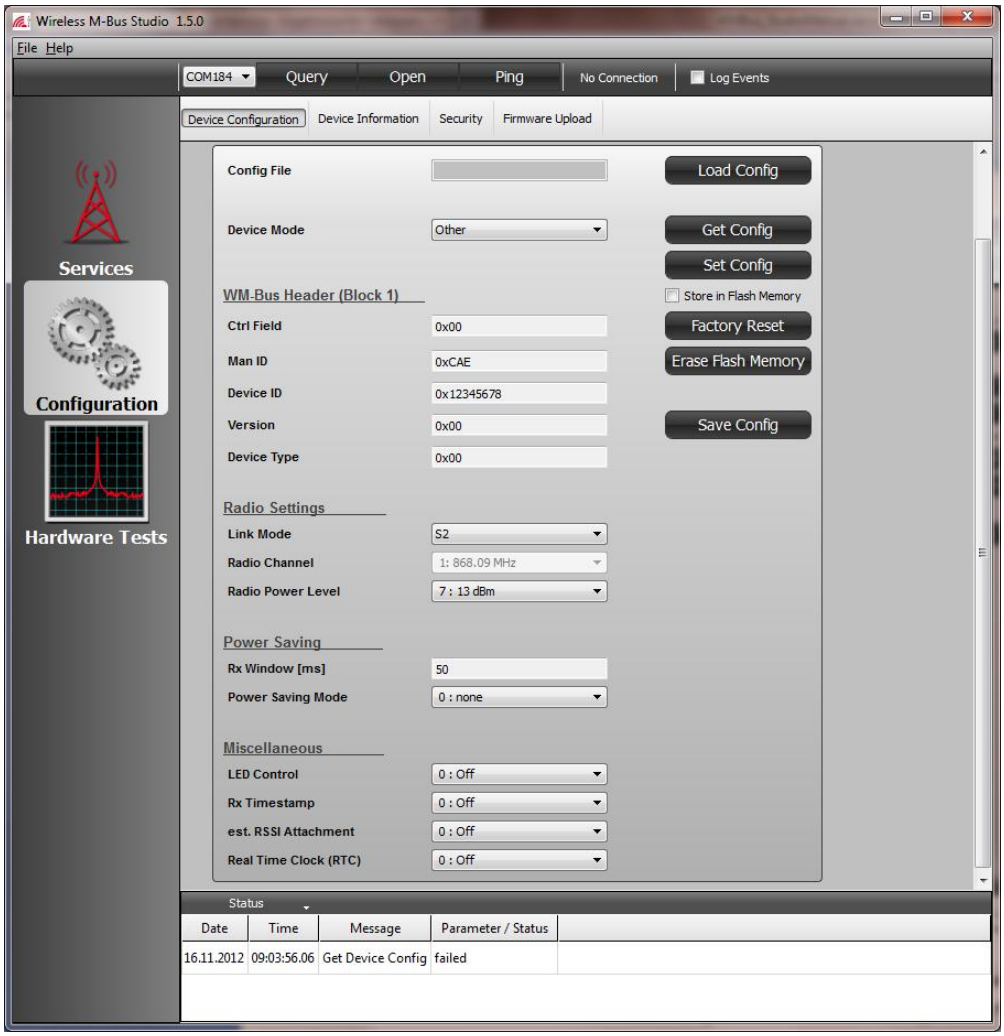


Figure 3-6: Device Configuration

The following parameters can be configured:

- *Device Mode* - defines if the device operates in *Meter Mode* or *Other Mode* according to EN 13757-4
- *WM-Bus Header (Block 1)* – includes all fields of Block 1 for message exchange according to EN 13757-4
- *Radio Settings* – includes the standard radio parameters:
  - *Link Mode* – modes according to EN 13757-4
  - *Radio Channel* – selectable physical channels for R2-mode
  - *RF Power Level* – RF output power level

- *Power Saving* – includes parameters for devices which operate in *Meter Mode*:
  - *Power Saving Mode* – determines the Power Saving Mode which is entered *automatically* after sending/receiving an RF message.
  - *Rx Window* – configurable time the receiver is waiting for a response message from a peer device after sending a message. This window is only applied in one of the bidirectional *Link Modes S2, T2, R2 or C2.* Note: a value of zero (0) enables the receiver permanently (useful for Packet Sniffer).

- *Miscellaneous* – includes the following configuration options:
  - *LED Control* – determines if the firmware should control the Demo Board LEDs via port pin to indicate several internal events.
  - *RxTimestamp* – determines if an RTC based time stamp should be attached for every received RF message when send over serial interface to a host controller.
  - *Est. RSSI* – determines if an estimated received signal strength indicator should be attached for every received RF message.
  - *Real Time Clock* – determines if the RTC should be enabled or not.

Load/Save Configuration

The GUI offers the possibility to load/save a complete set of configuration parameters from/to file.

Get/Set Configuration

These buttons allow to read/modify the configuration of a connected iM871A.

Store in Flash Memory

The device parameters can be changed temporarily (in RAM) or persistently (in non-volatile flash memory).

Factory Reset

This feature can be used to restore the default device configuration which is also used in case of corrupt non-volatile memory content.

Erase Flash Memory

This function allows to erase the non-volatile configuration data memory.

3.2.2 Device Information

This page displays some general firmware information elements.

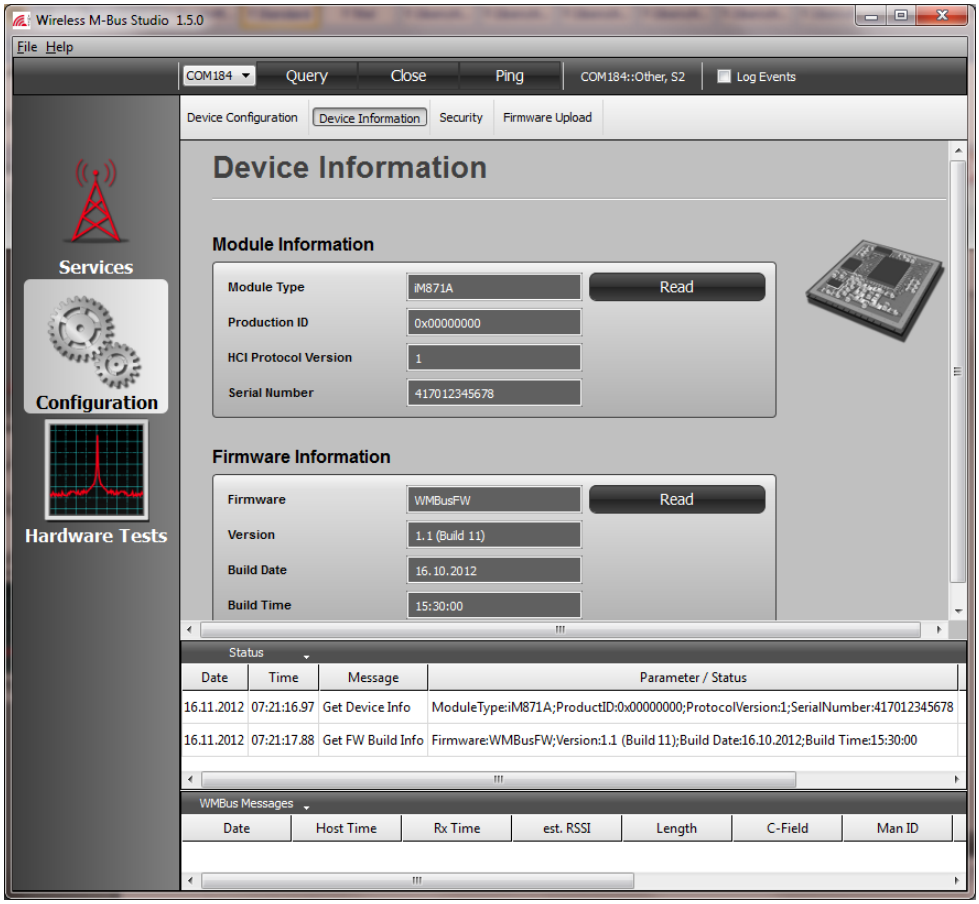


Figure 3-7: Device Information

The following information elements can be read:

- *Module Type* - 8-Bit number which identifies the connected radio module type e.g. iM871A
- *Production ID* - a unique 32-Bit production number which identifies the radio module
- *HCI Protocol Version* - 8-Bit number which identifies the supported Host Controller Interface (HCI) Protocol version
- *Firmware* - a character string which identifies the firmware
- *Version* - a version number and build counter which identify the firmware version
- *Build Date* - identifies the firmware build date
- *Build Time* - identifies the firmware build time

3.2.3 Security

The iM871A supports automatic AES-128 encryption and decryption of Wireless M-Bus radio messages.

Figure 3-8: Security

AES Encryption

This section allows the configuration and activation of the automatic encryption algorithm. The encryption is used whenever a radio message is transmitted. The AES key can be set in by means of four 32-Bit input fields.

Configure Key

This function can be used to set a new AES encryption key.

Enable/Disable Key

This function activates / deactivates the automatic encryption algorithm.

Store in Flash

The encryption parameters can be changed temporarily (in RAM) or persistently (in non-volatile flash memory).

AES Decryption

This section allows the configuration of AES decryption keys. The automatic decryption algorithm is used whenever a radio message is received. The firmware provides decryption keys for multiple transmitting devices which are identified by means of their WM-Bus Header information (see also *Device Configuration*). The configuration data is only stored in RAM in will not resist a power-cycle.

Slot

AES decryption keys and corresponding WM-Bus Header data are stored within so called memory slots. The number of available memory slots depends on the firmware version.

Man ID / Device ID / Version / Device Type

These input fields are used to set the WM-Bus Header information elements of the transmitting device. During message reception these fields are used to find the corresponding AES decryption key in the memory slots of receiver's configuration table.

AES Key

These four 32-Bit input fields are used to configure the decryption key.

Set Key

This function can be used to write a new AES decryption key into a memory slot.

3.3 Hardware Tests

3.3.1 Radio Tests

This page provides some test functions which allow to verify the iM871A radio performance.

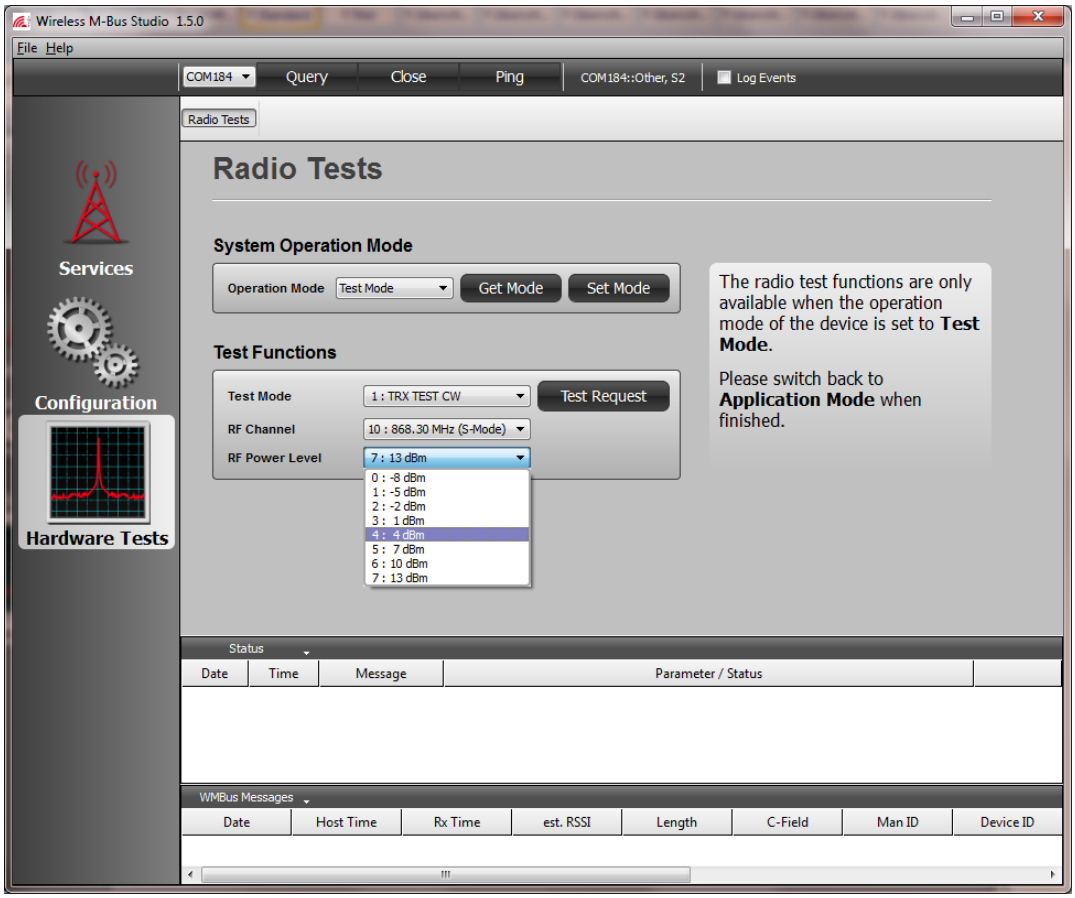


Fig. 3-1: Radio Tests

System Operation Modes

The firmware can operate in different kind of so called *System Operation Modes*. These modes enable the firmware to align its behaviour according to a given use case e.g. *Application Mode* or *Test Mode*. The default and standard mode is called *Application Mode* and designed for application purposes like *Radio Link Test* or *Message Exchange*. For test purposes the so called *Test Mode* must be set.

Note: The new requested mode will be stored into the non-volatile flash memory since a mode change requires a firmware reset. During next firmware start-up the operation mode is read from the flash memory and the related firmware and hardware parts get initialized.

Radio Test Functions

The firmware provides a *Continuous Wave Test (CW)* for different *Radio Channels* and different *RF Power Levels*.

Note: Don't forget to switch back to *Application Mode* when tests are finished !

4. Hardware Configuration

The configuration of the WiMOD Demo Board can be found in [1].

5. Appendix

5.1 List of Abbreviations

DLL	Dynamic Link Library
FW	Firmware
GUI	Graphical User Interface
HCI	Host Controller Interface
HW	Hardware
RAM	Random Access Memory
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
RTC	Real Time Clock
SW	Software
UART	Universal Asynchronous Receiver/Transmitter
WM-Bus	Wireless M-Bus

5.2 List of Figures

5.3 References

[1] iM871A_StarterKit_QuickStartGuide.pdf

6. Regulatory Compliance Information

The use of radio frequencies is limited by national regulations. The radio module has been designed to comply with the European Union's R&TTE (Radio & Telecommunications Terminal Equipment) directive 1999/5/EC and can be used free of charge within the European Union. Nevertheless, restrictions in terms of maximum allowed RF power or duty cycle may apply.

The radio module has been designed to be embedded into other products (referred as "final products"). According to the R&TTE directive, the declaration of compliance with essential requirements of the R&TTE directive is within the responsibility of the manufacturer of the final product. A declaration of conformity for the radio module is available from IMST GmbH on request.

The applicable regulation requirements are subject to change. IMST GmbH does not take any responsibility for the correctness and accuracy of the aforementioned information. National laws and regulations, as well as their interpretation can vary with the country. In case of uncertainty, it is recommended to contact either IMST's accredited Test Center or to consult the local authorities of the relevant countries.

7. Important Notice

7.1 Disclaimer

IMST GmbH points out that all information in this document is given on an "as is" basis. No guarantee, neither explicit nor implicit is given for the correctness at the time of publication. IMST GmbH reserves all rights to make corrections, modifications, enhancements, and other changes to its products and services at any time and to discontinue any product or service without prior notice. It is recommended for customers to refer to the latest relevant information before placing orders and to verify that such information is current and complete. All products are sold and delivered subject to "General Terms and Conditions" of IMST GmbH, supplied at the time of order acknowledgment.

IMST GmbH assumes no liability for the use of its products and does not grant any licenses for its patent rights or for any other of its intellectual property rights or third-party rights. It is the customer's duty to bear responsibility for compliance of systems or units in which products from IMST GmbH are integrated with applicable legal regulations. Customers should provide adequate design and operating safeguards to minimize the risks associated with customer products and applications. The products are not approved for use in life supporting systems or other systems whose malfunction could result in personal injury to the user. Customers using the products within such applications do so at their own risk.

Any reproduction of information in datasheets of IMST GmbH is permissible only if reproduction is without alteration and is accompanied by all given associated warranties, conditions, limitations, and notices. Any resale of IMST GmbH products or services with statements different from or beyond the parameters stated by IMST GmbH for that product/solution or service is not allowed and voids all express and any implied warranties. The limitations on liability in favor of IMST GmbH shall also affect its employees, executive personnel and bodies in the same way. IMST GmbH is not responsible or liable for any such wrong statements.

7.2 Contact Information

IMST GmbH

Carl-Friedrich-Gauss-Str. 2-4
47475 Kamp-Lintfort
Germany

T +49 2842 981 0
F +49 2842 981 299
E wimod@imst.de
I www.wireless-solutions.de