IMST GmbH

Carl-Friedrich-Gauß-Str. 2-4, D-47475 Kamp-Lintfort

Wireless M-Bus Gateway Firmware

Feature Overview

Version 1.3

Document State

wip

Date

27.03.2024

Document ID

4000/40140/0179

© 2024 IMST GmbH - All rights reserved

History

Version	Date	Chapter	Comment
1.0	10.09.2021	All	Initial Version
1.1	16.11.2021	All	Updates with respect to firmware version V0.21
1.2	25.07.2023	All	Updates for iM881A-XL firmware version V1.1, BC50
1.3	27.03.2024	All	Updates with respect to new hardware iM891A-XL, iU891A-XL (USB-Stick), new firmware eatures and other deliverables

Aim of this document

This document provides a feature overview about the WM-Bus Gateway Firmware and other related deliverables by IMST.

Content

- 1 - Overview
- 2 - General Firmware Features
- 3 - Wireless M-Bus Feature
- 4 - WM-Bus Gateway Studio (PC - Tool)

1 - Overview

Th following figure depicts the main components IMST delivers in combination with the WM-Bus Gateway Firmware which enables reception and transmission of Wireless M-Bus messages according to EN13757-2/-4/-5-/-6..

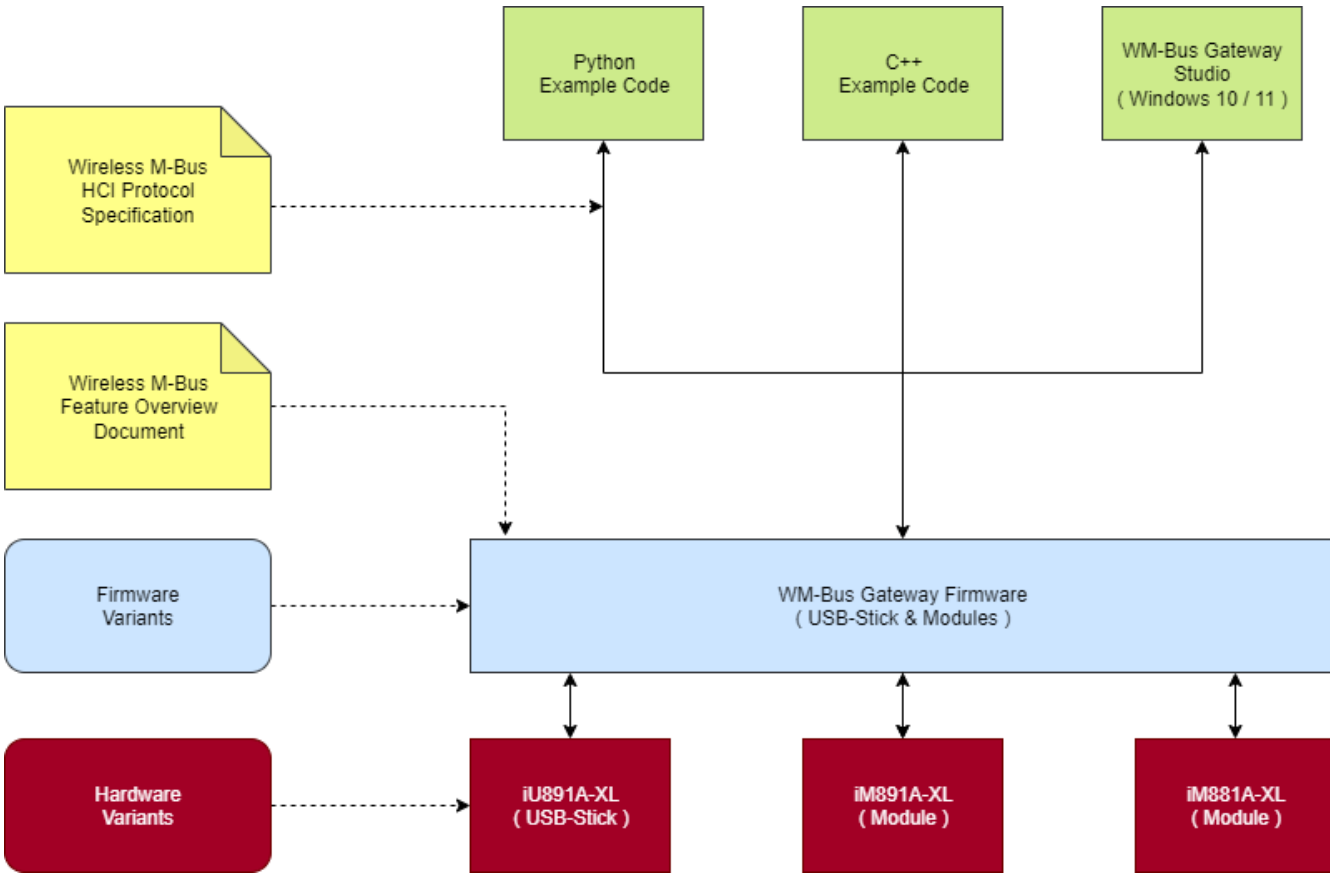

Figure 1 : Main deliverables around WM-Bus Gateway Firmware.

As shown above the firmware is available for three different hardware variants:

- iM881A-XL (radio module for system integration)
- iM891A-XL (radio module for system integration)
- iU891A-XL (ready to use USB stick)

In addition to this document, the Wirless M-Bus HCI-Protocol Specification explains the serial messages, which are used to control and configure the hardware in more detail.

Furthermore IMST delivers API Example Code in C/C++ and Python, including the most important message encodes and decoders of the HCI (Host Controller Interface) protocol.

Finally, the so-called WM-Bus Gateway Studio allows customers to explore the capabilities of the hardware and embedded WM-Bus Gateway firmware.

Last but not least, the Studio supports an integrated firmware update service for the different hardware variants.

Back to Top

2 - General Firmware Features

The firmware provides the following general features:

- **Configurable Default Settings**
  The so-called Default Configuration includes a set of changeable parameters that will be loaded from the non-volatile memory during firmware start into the WM-Bus Stack and immediatley become active.

- **Configurable Active Settings**
  During runtime the WM-Bus Stack can be re-configured with different parameters without storing them into the nonvolatile memory.

- **Scan - Mode**
  The Scan-Mode allows listening on several WM-Bus Link Mode dependent channels one after another with configurable timeout. This feature is helpful for scanning the environment for known WM-Bus devices, aiding in the configuration of built-in packet filters and encryption keys.

- **WM-Bus Packet Reception, Frame-Format Decoding & Decryption**
  This main firmware feature enables the reception of wireless M-Bus packets. The set of implemented Link Modes, Frame Decoders, and Decryption Modes is outlined in chapter 3. Our goal is to fulfill all required modes listed in the Open Metering Specification, Volume 2, Primary Communication, Issue 4.1.2 / 2016 - 12 - 16

- **WM-Bus Packet Filters & Encryption / Decryption Keys**
  The firmware offers a configurable list of WM-Bus devices. Each list entry includes an 8-byte Device Address (Manufacturer ID, Device ID, Version, Type) for identification purposes, along with a 128-bit AES Key used for packet encryption and decryption. This list can be stored in non-volatile memory upon request and will be automatically loaded into RAM during startup. The capacity of the list depends on both hardware and firmware, as outlined in the ***Wireless M-Bus Host Controller Interface Specification*** document .

- **Packet Filtering**
  The configurable WM-Bus Device List can also serve for packet filtering, ensuring that only received packets with 'known' Device Addresses are forwarded to the host controller through the serial interface.

- **Firmware and Hardware Identification**
  This service allows the identification of the given hardware and firmware variant to support future firmware updates.

- **Firmware Updates**
  IMST's radio modules and USB sticks are based on ST microcontrollers, which offer a built-in bootloader with a corresponding protocol.

Important Note

- **Software and Hardware Reset**
  In case of unexpected behavior or malfunction of the radio module or USB stick, at first a software reset by means of a HCI Message must be intitiated by the host. If this is not successful, then a reset by means of the **hardware reset signal** should be performed afterward. In case of an USB-Stick the control of the reset pin must be implemented via USB. IMST provides C/C++ example code for this.

- **Firmware Update Support**
  Additionally, it is highly recommended that the host has the capability to conduct a firmware update utilizing the integrated bootloader of both the radio-module and USB stick. To facilitate this, the necessary bootloading protocol (from ST) and control over hardware reset and boot signals must be integrated on the host controller side. For USB sticks, the hardware signals can be managed via USB. IMST offers C/C++ example code for the entire firmware update process upon request..

Back to Top

3 - Wireless M-Bus Feature

The following sub-chapters outline the wireless M-Bus specific features in more details.

- 3.1 - Wireless M-Bus Features - Supported Link Modes
- 3.2 - Wireless M-Bus Features - Frame Format A & B
- 3.3 - Wireless M-Bus Features - Encoding and Decoding of several Message Layers
- 3.4 - Wireless M-Bus Features - Packet Decryption and Encryption
- 3.5 - Wireless M-Bus Features - Supported CI-Values

Back to Top

3.1 - Wireless M-Bus Features - Supported Link Modes

The following table outlines the supported Wireless M-Bus Link Modes:

	Frequency	Coding	Chiprate	Bitrate	Frame Format
S - Mode	868.30 MHz	Manchester	32786 cps	16384 bps	A
T - Mode (Rx) (Meter to Other)	868.95 MHz	3-Out-Of-6	10000 cps	66666 bps	A
T - Mode (Tx) (Other to Meter)	868.30 MHz	Manchester	32768 cps	16394 bps	A
C - Mode (Rx) (Meter to Other)	868.95 MHz	NRZ	100000 cps	100000 bps	A, B
C - Mode (Tx) (Other to Meter)	869.525 MHz	NRZ	50000 cps	50000 bps	A, B
C / T-Mode (Rx) (Meter to Other)	868.95 MHz	NRZ 3-Out.Of-6	10000 cps 10000 cps	10000 bps 66666 bps	A, B A

Table 3 - 1 : Link Modes (Details)

Back to Top

3.2 - Wireless M-Bus Features - Frame Format A & B

The firmware is able to support encoding, decoding, CRC generation and validation for both required frame formats A + B.

Frame Format A (S -, T -, C - Mode)

	First Block							Second Block			...	Optional Blocks (s)	
Field	L	C	Man ID	Device ID	Version	Type	CRC	CI	Data	CRC	...	Data	CRC
Octets	1	1	2	4	1	1	2	1	15 or ((L - 9) mod 16) - 1	2		16 or (L - 9) mod 16	2

Table 3 - 2.1 : Frame Format A (Details)

Frame Format B (C - Mode)

	First Block						Second Block			Optional Block	
Field	L	C	Man ID	Device ID	Version	Type	CI	Data	CRC	Data	CRC
Octets	1	1	2	4	1	1	1	115 or (L - 12)	2	L - 129	2

Table 3 - 2.2 : Frame Format B (Details)

Back to Top

3.3 - Wireless M-Bus Features - Encoding and Decoding of several Message Layers

A wireless M-Bus message consists of several (optional) layers. The following table outlines the supported and not supported parts:

Abbreviation	Layer Name	Supported	Releated Part of Standard	OSI Model Layer
PHY	Physical Layer (see 1.1)	Yes	EN13757-2/-4/-5-/-6	Physical
DLL	Data Link Layer	Yes	EN13757-2/-4/-5	Data Link
ELL	Extended Link Layer	Yes	EN13757-4	
NWL	Network Layer	No	EN13757-5	Network
AFL	Authentication and Fragmentation Sublayer	Yes, for single fragment	EN13757-3	Presentation Session Transport
TPL	Transport Layer	Yes	EN13757-3/-4	
APL	Application Layer	No	EN13757-1/-3/-5	Application

Table 3 - 3 : Supported Wireless M-Bus Layers

Back to Top

3.4 - Wireless M-Bus Features - Packet Decryption and Encryption

The following table outlines supported and not supported decryption modes:

Mode	Encryption Type	Authentication	Supported
0	None	None	Yes
2	DES CBC	None	No
3	DES CBC	None	No
5	AES-128 CBC	None, MIC	Yes
7	AES-128 CBC, dynamic key	CMAC	Yes
8	AES-128 CTR	CMAC	No
9	AES-128 GCM	GCM/GMAC	No
10	AES-128 CCM	CCM	No
128	ELL AES-128 CTR	None, CRC	Yes
129	Custom Modes	None	e.g. AT-WMBUS-NA-1

Table 3 - 4 : Supported Decryption & Encryption Modes

Back to Top

3.5 - Wireless M-Bus Features - Supported CI-Values

The following list outlines the supported CI-Values. Theses values are used to identify the correct TPL-header-type for packet encryption / decryption.

CI-Field	Function / Layer	Up- or Down-link	TPL header - Type	Protocol / Service
53_h	Application Reset or Select	Down	Long	Application Reset or Select
$5B_h$	Command	Down	Long	M-Bus
60_h	Command	Down	Long	DLMS
$6C_h$	Time Sync	Down	Long	Generic
$6D_h$	Time Sync	Down	Long	Generic
$6E_h$	Application Error	Up	Short	Gereric
$6F_h$	Application Error	Up	Long	Generic
72_h	Response	Up	Long	M-Bus
74_h	Alarm	Up	Short	Generic
75_h	Alarm	Up	Long	Generic
78_h	Response	Up	None	M-Bus
$7A_h$	Response	Up	Short	M-Bus
$7C_h$	Response	Up	Long	DLMS
$7D_h$	Response	Up	Short	DLMS
80_h	Pure Transport Layer	Down	Long	None
$8A_h$	Pure Transport Layer	Up	Short	None
$8B_h$	Pure Transport Layer	Up	Long	None
$8C_h$	Extended Link Layer	Up / Down	Short	Lower Layer Service (2 Byte)
$8D_h$	Extended Link Layer	Up / Down	Long	Lower Layer Service (8 Byte)
$8E_h$	Extended Link Layer	Up / Down	Long	Lower Layer Service (10 Byte)
$8F_h$	Extended Link Layer	Up / Down	Long	Lower Layer Service (16 Byte)
$C3_h$	Command	Down	Long	Security Information Transport
$C4_h$	Response	Up	Short	Security Information Transport
$C5_h$	Response	Up	Long	Security Information Transport

Table 3 - 5 : Supported CI-Values

Back to Top

4 - WM-Bus Gateway Studio (PC - Tool)

The WM-Bus Gateway Studio can be used to configure and test the WM-Bus Gateway firmware. Furthermore it provides a firmware update service for the different hardware variants.

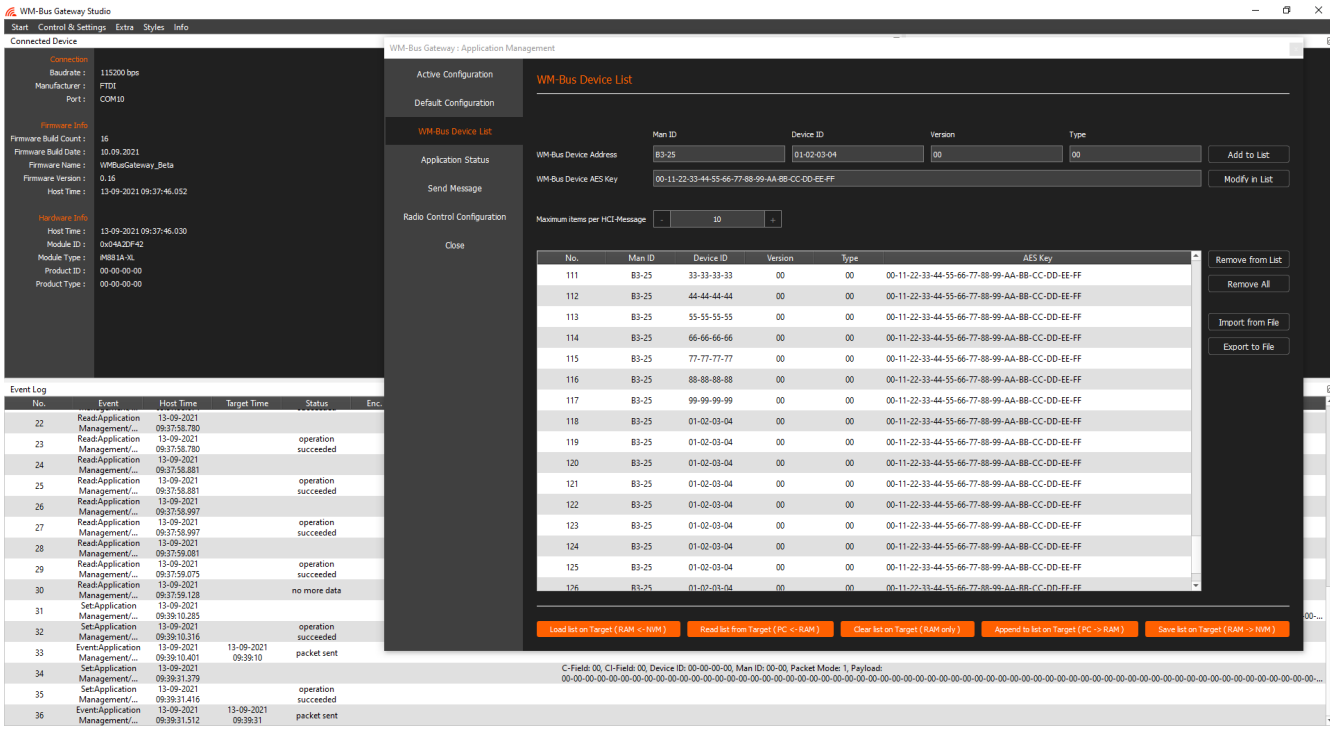

Figure 2 - 1 : WM-Bus Gateway Studio

This application is availlable vor Windows 10, 64 Bit.

Back to Top