

Graphical User Interface GUI

For EM3028TP14 Extreme Low Power RTC Using USB -- I2C Bus Dongle

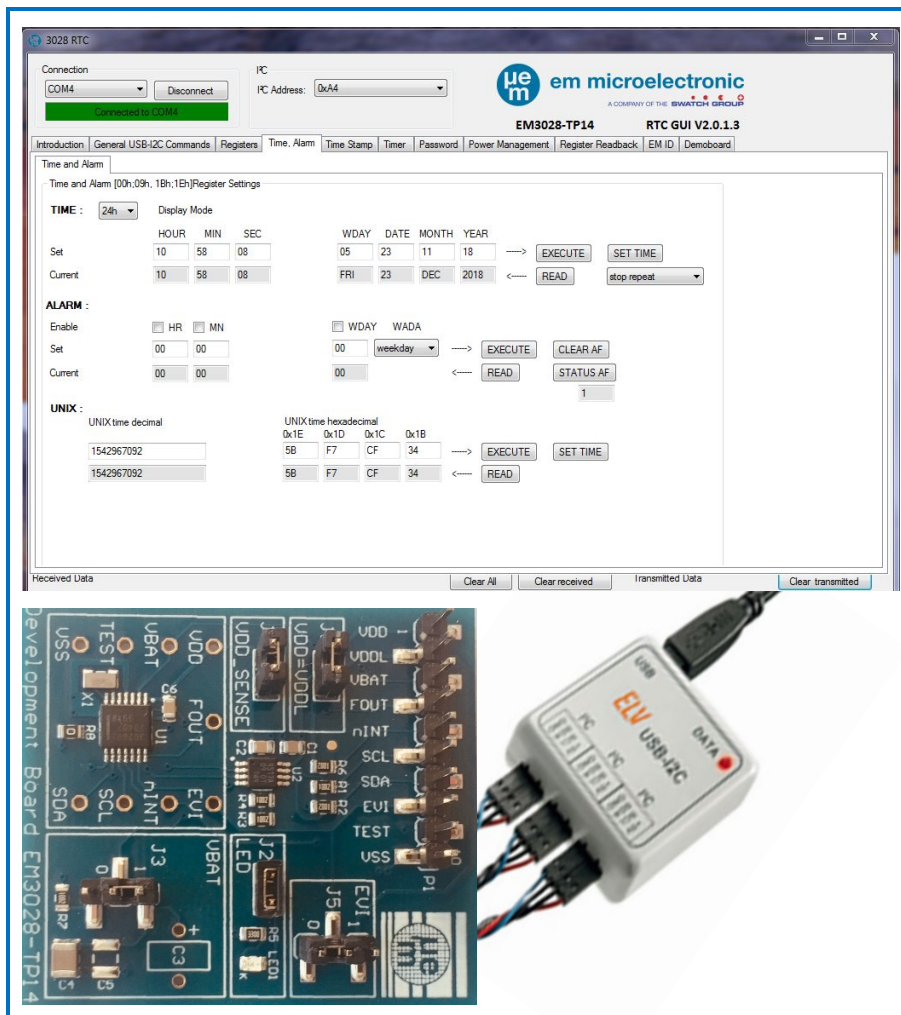


TABLE OF CONTENTS

1. Summary	4
2. Key Functions	4
2.1. Hardware: USB-I ² C Interface Dongle	4
2.2. Software	4
3. Bridge Circuit USB to I ² C-Bus.....	6
3.1. Circuit Diagram of Dongle.....	6
4. Installation.....	8
4.1. Software Driver	8
4.2. Hardware	8
4.3. Software GUI	8
5. Operation	10
5.1. Procedure	10
5.2. General USB I ² C-Bus Command.....	12

1. SUMMARY

The growing number of features of Real-time Clocks can optimally be evaluated through an intuitive Graphical User Interface (GUI). To simplify the hardware setup EM offers a kit with the USB port to I²C-Bus dongle and the RTC development board. The GUI it is the straight forward approach for communicating with the RTC and monitoring all functions including register and flag settings. The string of applied I²C-Bus commands is reflected in real time as reference for the firmware development. A dedicated mask for communication with other I²C-Bus circuits is provided in addition.

For the correct development board setup, please refer to the specific documentation

- EM3028-TP14_DevelopmentBoard.pdf

Features:

- Ready to run the dongle establishing bidirectional USB to I²C-Bus translation.
- Three parallel wired I²C-Bus ports include also +5 V for supplying power to the development board.
- USB and customized cables for the I²C-Bus are included

2. KEY FUNCTIONS

2.1. HARDWARE: USB-I²C INTERFACE DONGLE

The dongle is a ready to run module. It creates a virtual COM-port via the USB connection. It provides three I²C-Bus connections with optional output of 5 V for powering up the application (max 450 mA).

Power consumption:	Module / total: <50 mA / max 500 mA
I ² C-Bus clock frequency:	245 Hz – 400 kHz
USB driver for Windows:	Windows XP, Windows 7, Windows 8
Size:	50 mm × 40 mm × 15 mm

2.2. SOFTWARE

The software control via a GUI allows a fast setup for communicating with the Real-Time-Clock module. Aside from the detailed GUI pages for the target development board, it allows communicating with any I²C-Bus device by entering directly the data hex codes.

Example: s A4 2A p

Nomenclature: All terms are in hexadecimal format!

s	stands for the I ² C START condition
A4	for the slave address
00	for the first data byte to be sent
p	for the I ² C STOP condition

3. BRIDGE CIRCUIT USB TO I²C-BUS

3.1. CIRCUIT DIAGRAM OF DONGLE

The block diagram illustrates both connections: to the PC and to the application under evaluation.

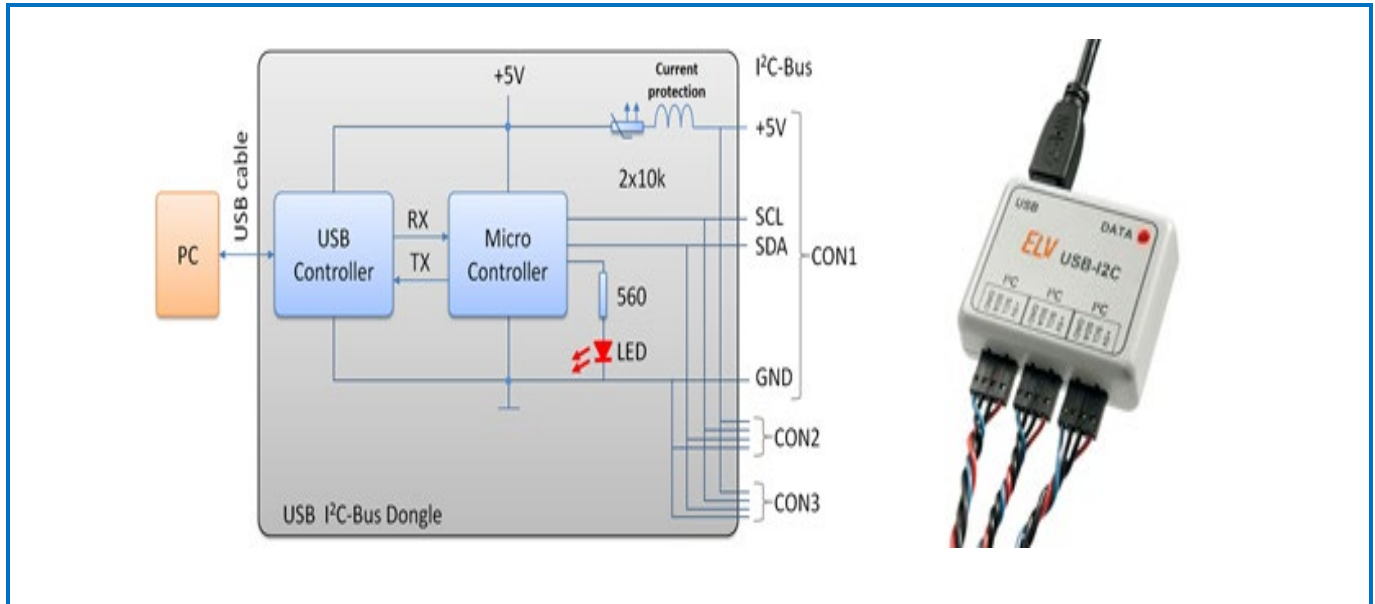


Figure 3-1 Dongle Block Diagram

The three connectors CON1, CON2, and CON3 are wired in parallel and carry the four required signals; their order is imprinted on the package:

+5 V: Optional 5 V supply can be used. Total consumption of all 3 outputs must be kept below 450 mA.

SCL: Serial Clock line *

SDA: Serial Data line *

GND: Ground

*) : The pull-up resistors are located on the development board

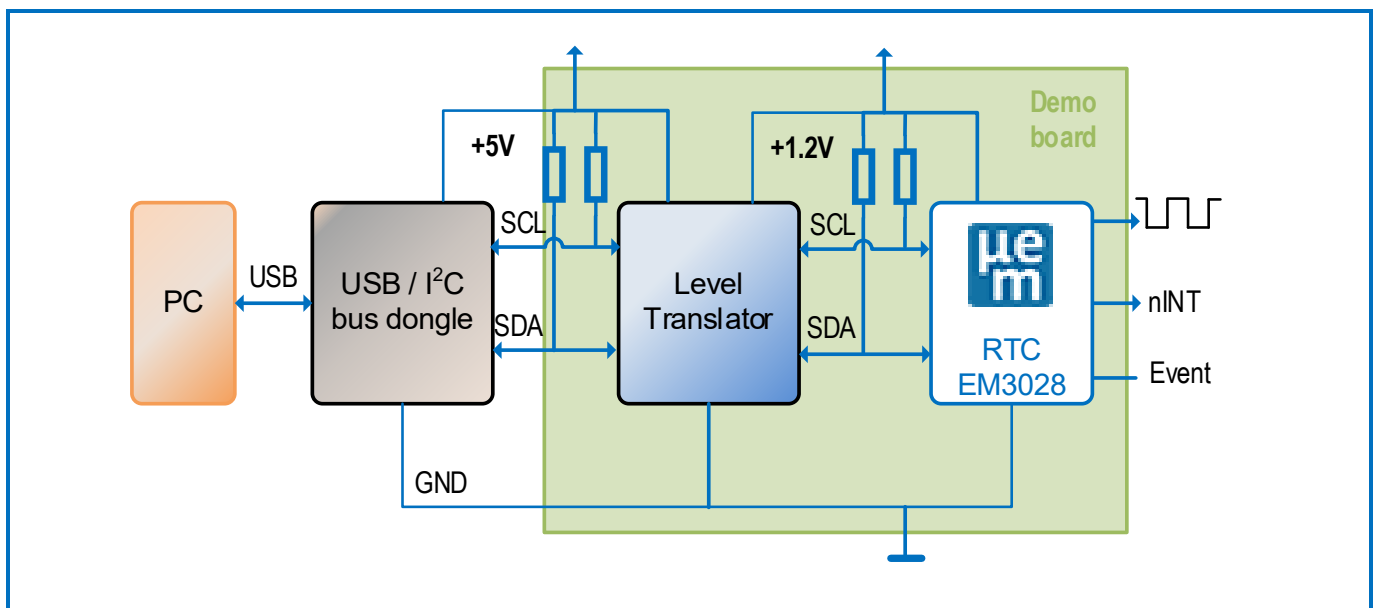


Figure 3-2 Interface Low Voltage RTC

4. INSTALLATION

Use the following sequence:

4.1. SOFTWARE DRIVER

Install the USB software driver provided before connecting the interface module!
Unpack the file: i.e **cp210x_vcp_win_xp_s2k3_vista_7_8_v6_6_1.zip**
Install the driver in administrator mode, by double click.

4.2. HARDWARE

Connect the USB cable between the Dongle and your PC as well the as the I²C-Bus cable to your target RTC board. In case an external supply voltage is used, make sure it is now turned on.
The red LED will light up indicating the dongle successfully started up. The system is now ready for starting the GUI software.

4.3. SOFTWARE GUI

Installation of the GUI

- Run the exe file: USB-I2C-**EM3028TP14**.exe (program name **EM3028TP14** is RTC specific)
- The start window will pop op (Figure 4-1)
- Choose the appropriate com-port from the drop down list (usually the last one on the list)
- Press Connect: Status changes to *Connected to COMn* and turns green

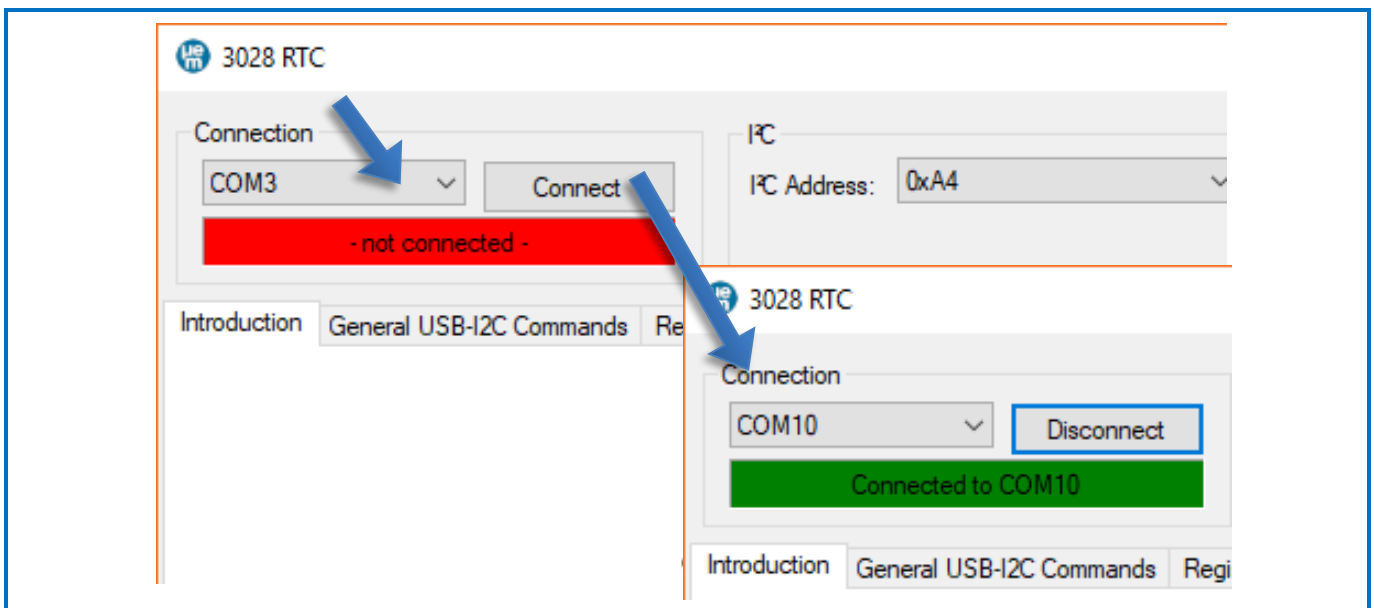


Figure 4-1 Start Window of the GUI

5. OPERATION

The Graphical User Interface is organized by a number of drop-down menus. Specific windows visualize the actual status and allow controlling the specific Micro Crystal RTC-module. In a separate menu (General USB-I2C commands) the GUI can be used as a universal I²C-Bus interface for controlling any peripheral circuit.

5.1. PROCEDURE

The drop-down windows follow a general structure (Figure 5-1)

1. Input fields to set e.g. the current time, to start transmission to the RTC press EXECUTE.
2. Output fields are updated after the READ button is pressed
3. Interactive knobs allow activating different functions or selecting specific parameters
4. Monitoring bus transmission: After every executed bus transmission the bus protocol is listed in the field *Transmitted Data*. Any data received by the controller is listed in the field *Received Data*. This is ideal for verifying the code needed for proper execution

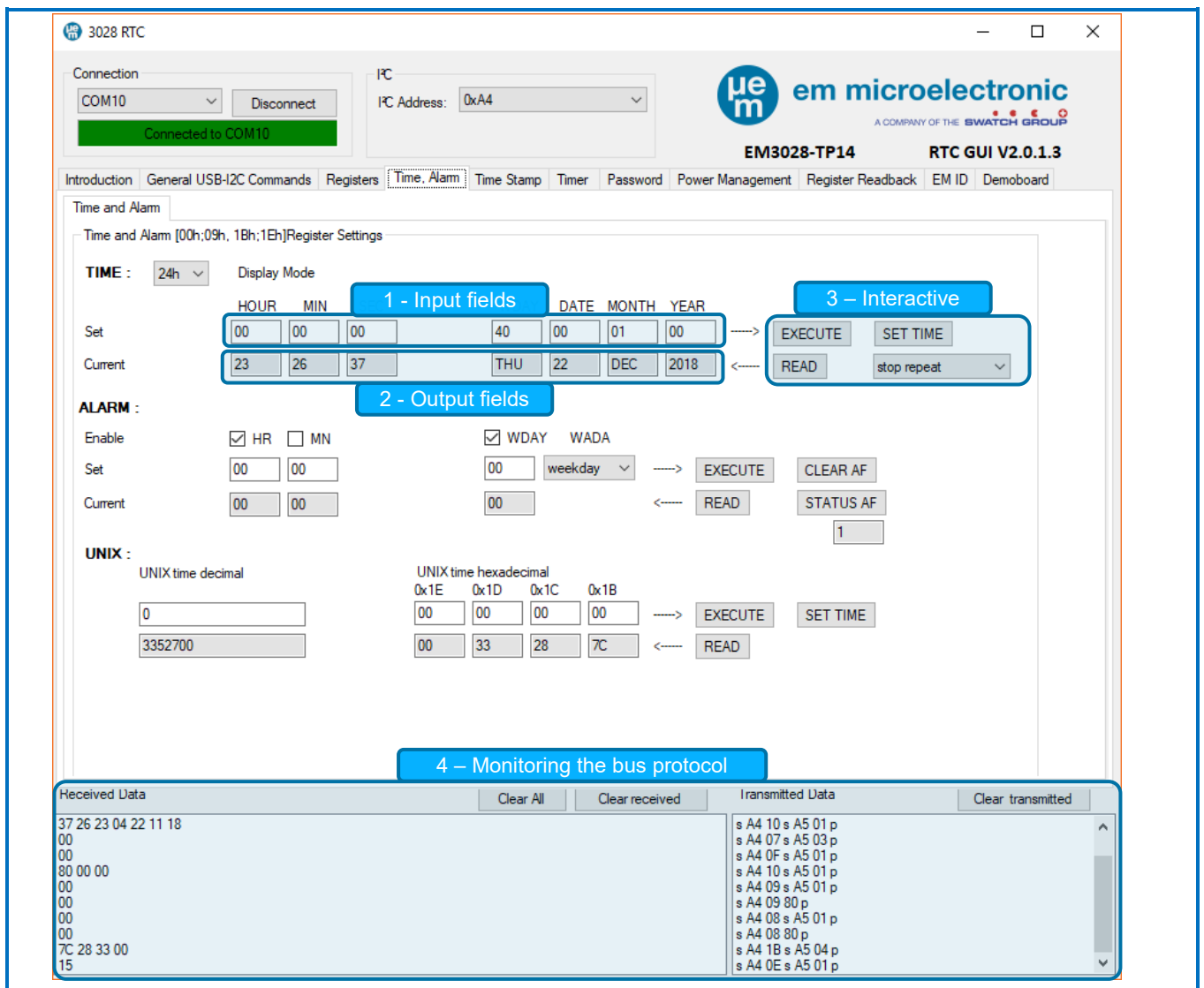


Figure 5-1 Structure of the GUI

5.2. GENERAL USB I²C-BUS COMMAND

The Universal I²C interface allows controlling any circuit by entering directly the I²C-Bus instructions.

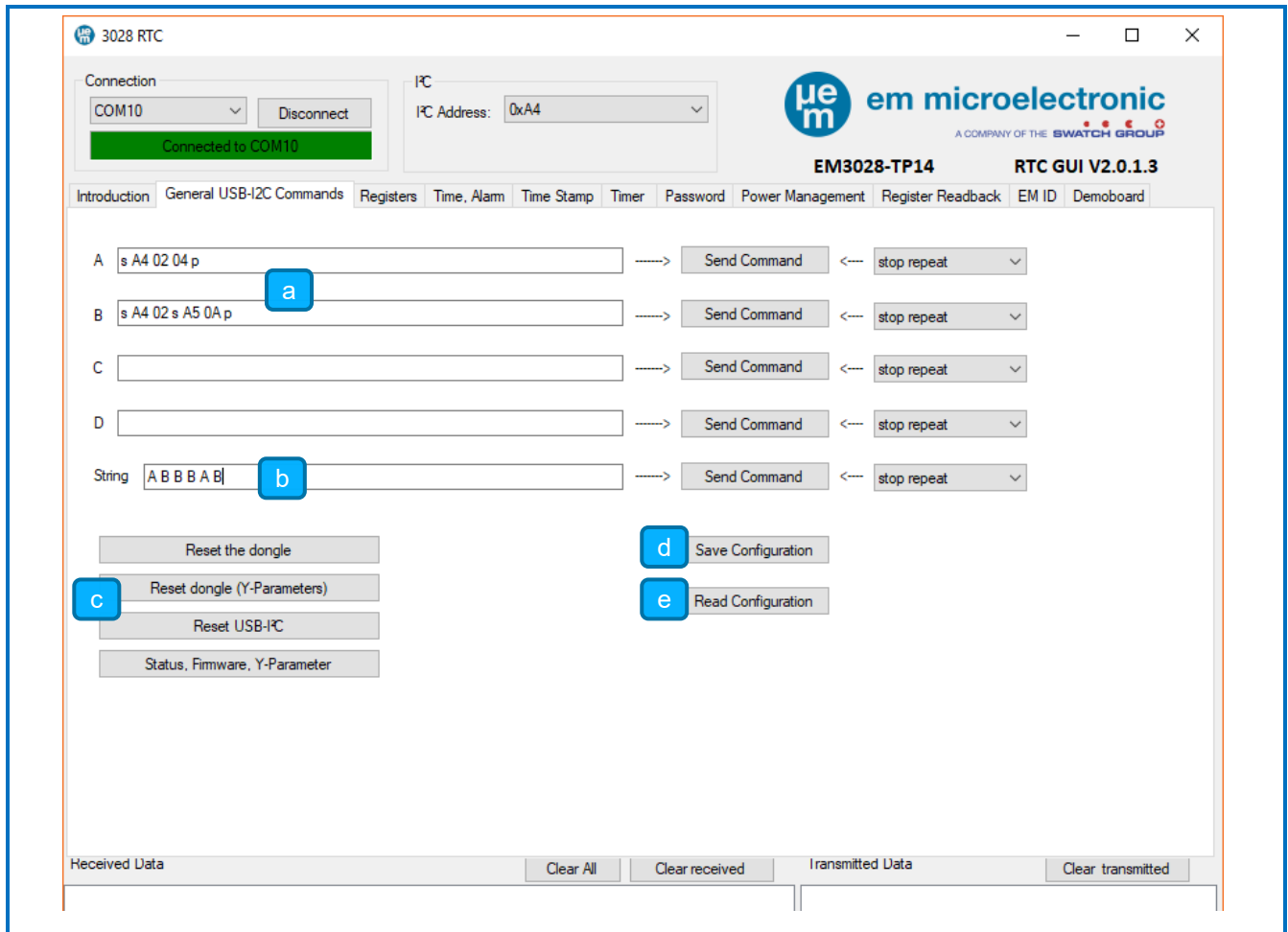


Figure 5-2 General USB-I²C Bus Interface

- There are 4 command lines A to D for preparing communication. The data format is straight forward
- I²C-Bus protocol:
e.g.: for sending data: s A4 02 04 p press *Send Command*
e.g.: for reading data: s A4 02 s 65 0A p: press *Send Command* setting address pointer, then reading 10 bytes (0Ah)
(s = START, A4 = slave address, 02 04 = data, p = STOP)
The command lines can be fired off once per click or repeatedly every 100ms, 1s, 2s, 10s.
- The string allows to sequentially firing the above command lines: e.g. A B B D C. When using fast repetition rate it is of good practice to increase the I²C-Bus speed. (Use the T400000 instruction for setting the bus speed to 400kHz, details are shown below under 'Additional instructions')
- Reset functions can be executed
- Saving the actual configuration on the PC, including all settings on the RTC. Assign a specific name and select a directory.
- Reloading of stored settings to continue former tasks.

Additional instructions:

- ^ Space is ignored, letters can be in capital or small.
Sequences like SA400P or S A4 00 p are interpreted identically
- S I²C-Bus Start condition, also for repetitive start condition
- P I²C-Bus-Bus STOP condition
- L n Inserts a delay of n ms: 1 to 65535ms (n= 0001...FFFF)
- T m Setting the I²C-Bus frequency: minimum 226Hz m= 000226, maximum 409.6kHz m= 409600
- ? System status and settings are read from the dongle

EM Microelectronic-Marin SA ("EM") makes no warranties for the use of EM products, other than those expressly contained in EM's applicable General Terms of Sale, located at <http://www.emmicroelectronic.com>. EM assumes no responsibility for any errors which may have crept into this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein.

No licenses to patents or other intellectual property rights of EM are granted in connection with the sale of EM products, neither expressly nor implicitly.

In respect of the intended use of EM products by customer, customer is solely responsible for observing existing patents and other intellectual property rights of third parties and for obtaining, as the case may be, the necessary licenses.

Important note: The use of EM products as components in medical devices and/or medical applications, including but not limited to, safety and life supporting systems, where malfunction of such EM products might result in damage to and/or injury or death of persons is expressly prohibited, as EM products are neither destined nor qualified for use as components in such medical devices and/or medical applications. The prohibited use of EM products in such medical devices and/or medical applications is exclusively at the risk of the customer.