Inveo Sp. z o.o.

IND Modbus

Instruction Manual

Overview

> **Warning**
>
> **This manual applies only to the firmware version v1.38 onwards. Inveo does not guarantee that the information contained in this document applies to previous firmware revisions.**



Purpose of the device

IND Modbus is an industrial tag reader, housed in a robust IP65 enclosure, offering protection from dust and water. The built-in RS485 module enables communication with a Master device, to which the codes read from RFID tags will be sent.

Additionally, thanks to the Modbus RTU protocol, the user has control over two relays that can be used to control various devices, and over two digital inputs that can be used to monitor the status of other devices. The device configuration is done via USB.

Changelog

1.1 11th of March 2026

- Firmware revision v1.38
  - Compatibility patch for AMD CPU's

1.0 **31st of March 2025**

- Firmware revision v1.36

Table of contents

- Overview
  - Purpose of the device
  - Changelog
    - 1.1 11th of March 2026
    - 1.0 31st of March 2025
- Device construction
  - Technical characteristics
  - Connectors description
  - Signalling
- IMPORTANT – Determining the device version
- v1 device configuration
  - Outputs
  - Inputs
  - LED and sound signal
  - Continuous Read Mode
  - RS485 communication settings
- v2 device configuration
  - Modbus
    - RS485 communication settings
    - Continuous reading
    - Outputs
  - LED's and sound
  - Main tab – Only in special versions
    - Keybord emulation mode
    - Virtual CDC serial port mode
    - Number sending format
- Modbus API
  - Modbus addressing
    - Holding Registers:
    - Single Coil:
  - Mifare tag structure - only Mif version
  - Memory buffer - Only Mif version
  - Data representation in Modbus addresses (Endianess)
- Application examples

- Card UID read operation
- Data read operation
- Data write operation
- Reading multiple tags
- Warranty and manufacturer's liability
  - Storage, operation and transport conditions
  - Disposal and decommissioning

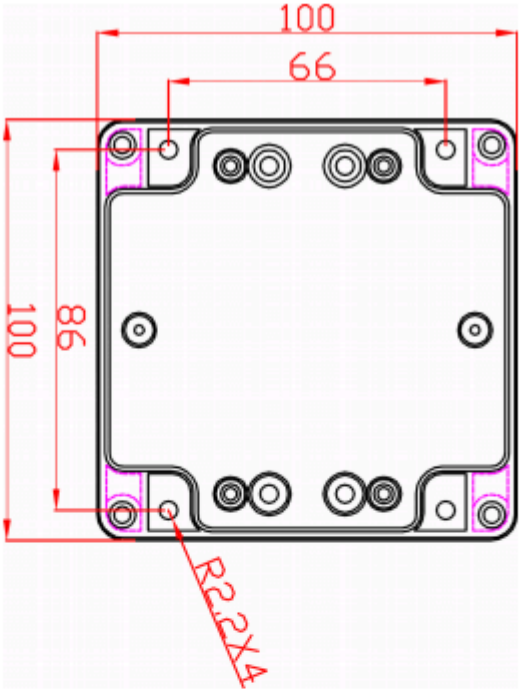
Device construction

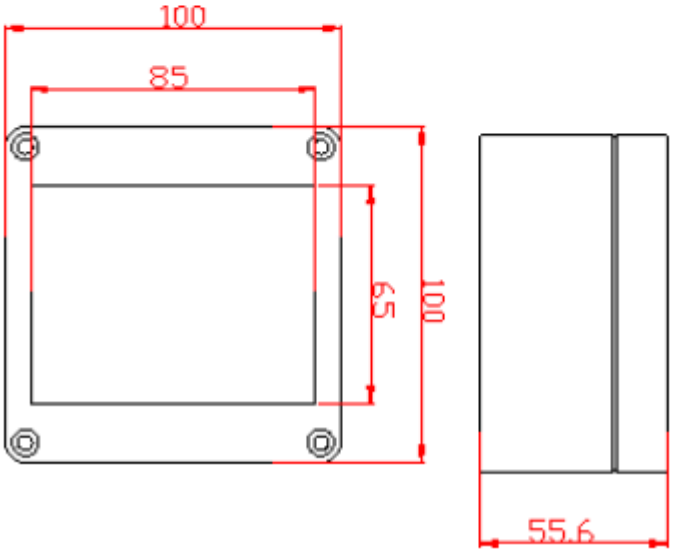
Technical characteristics

Parameter	Description
Power supply	10-24V DC
Power consumption	**Max 0,5W** (~40mA @ 12V)
Transponders	**The following IND Modbus versions are available:** **Mif:** Version supporting Mifare Classic®, Mifare Plus® (UID), Mifare DESFire® (UID) 13,56MHz transponders **Uni:** Unique 125kHz transponders **HT2:** HITAG 2 125kHz transponders
Inputs	- **Amount of inputs:** 2 - **Input type:** Normally Open (NO)
Outputs	- **Amount of outputs:** 2 - **Output type:** NO (relay), - **Maximum load:** 1A @ 30V DC
Komunikacja	- **RS485 port**, Modbus RTU - **USB type B port** for device configuration
Enclosure ingress protection	**IP65**

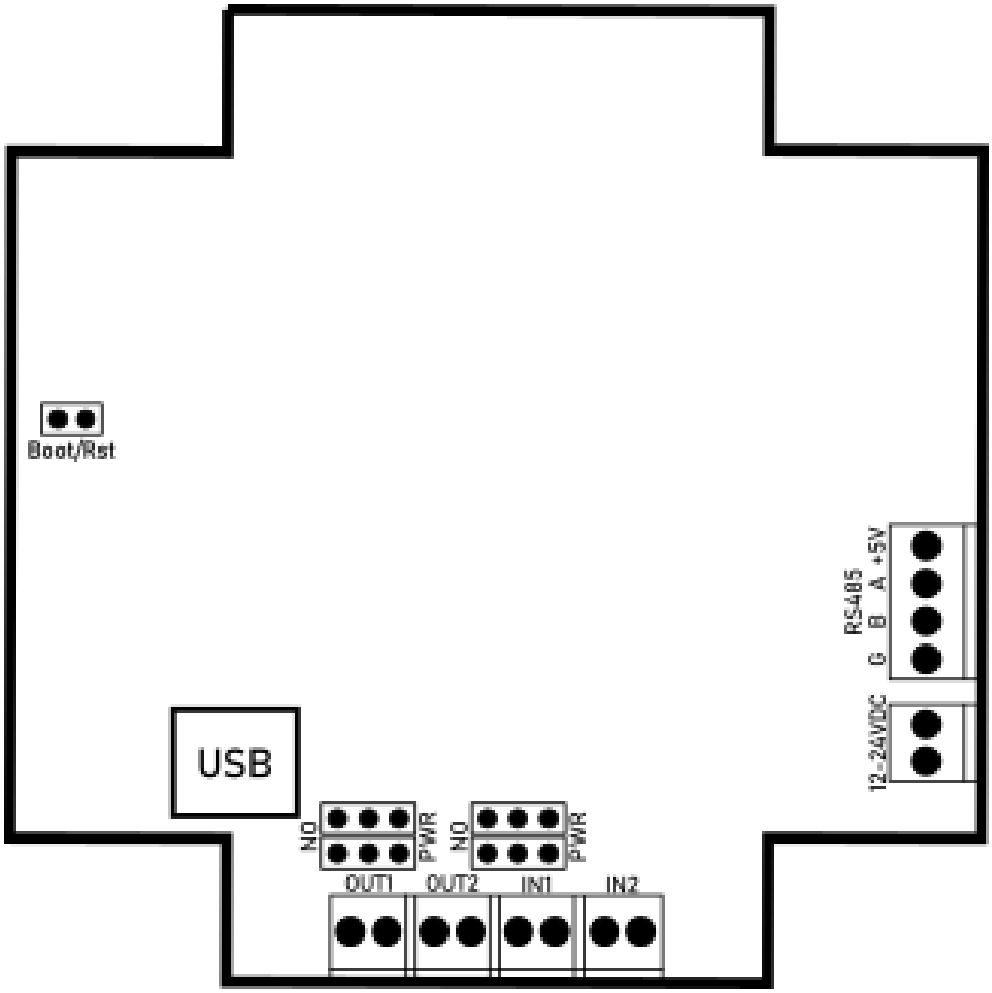
Because of different installation needs, the device isn't supplied with any mounting openings in the enclosure. Openings should be made by the customer anywhere in the rear section of the housing - this does not void warranty. Remember to use an appropriate cable gland in order to maintain the declared ingress protection.

Dimensions



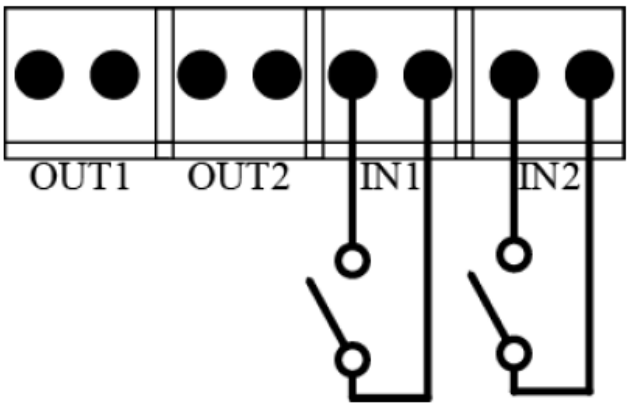


Connectors description



- **USB** – Used for device configuration,
- **OUT1, OUT2** – Relays 1 and 2 contacts,
- **IN1, IN2** – General purpose inputs 1 and 2,
- **12-24VDC** – Power input,
- **RS485** – Modbus connector,
- **Boot/Rst** – Shorting the pins before applying power will cause the module to enter the bootloader mode.

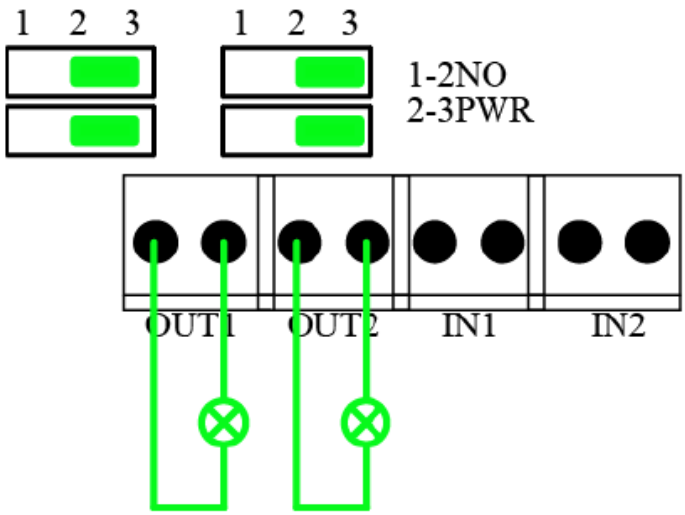
Connecting inputs



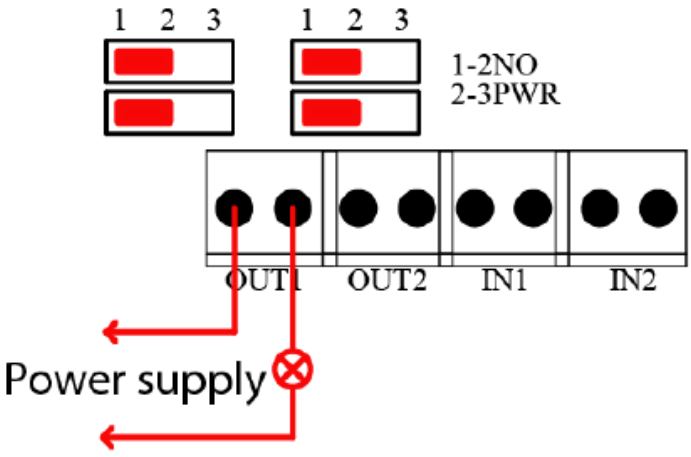
Outputs **OUT1** and **OUT2** can operate in two modes:

- **PWR mode** – Configuration jumpers set in position 2-3 (as shown in the image below). In this mode, an activated relay will provide the output with the power supply voltage eg. 12V DC. If the reader is powered with 24V DC, 24V DC will be provided.

> **Warning**
>
> **If the reader is powered with PoE 802.3af and does not have a supply connected to the power input, outputs OUT0 and OUT1 won't have any voltage supplied to them.**



- **Normally Open mode** – Jumpers set in position 1-2 (as shown in the image below). In this mode an external power source is necessary.



Signalling

The device is equipped with 3 LED indicators that inform the user about the module's state and with a sound generator that informs about tag application.
The LED can be disabled or set to respond to a tag.
The buzzer can be disabled or set to generate a signal after a tag is read.

Name	Description	Colour
POWER	Module power	Green
Status 1		Red
Status 2	Tag read signal	Green

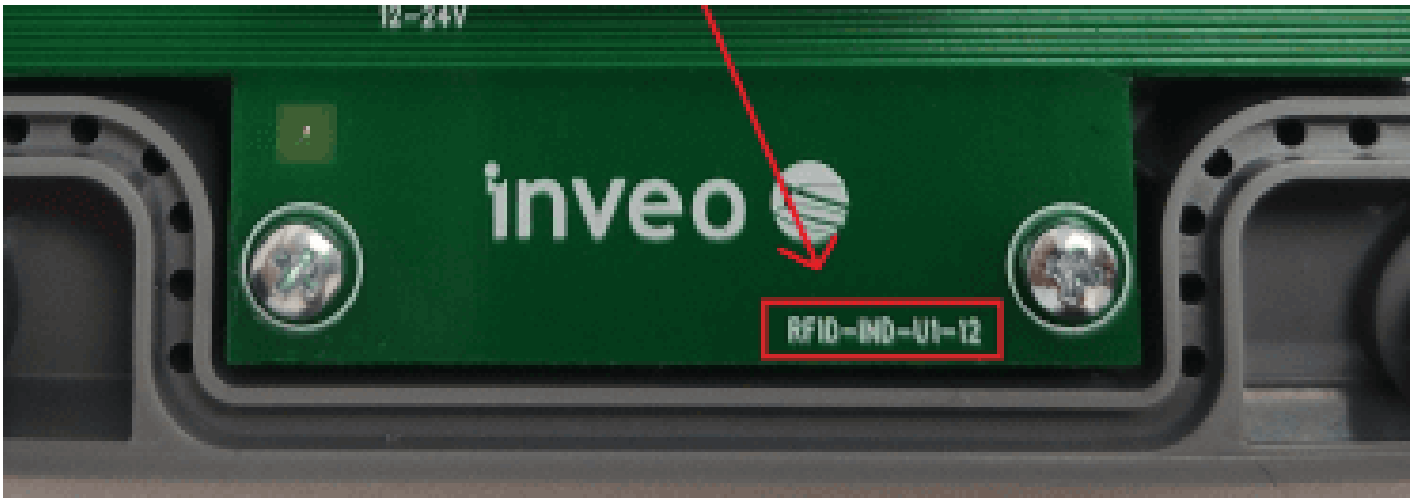
IMPORTANT – Determining the device version

Information

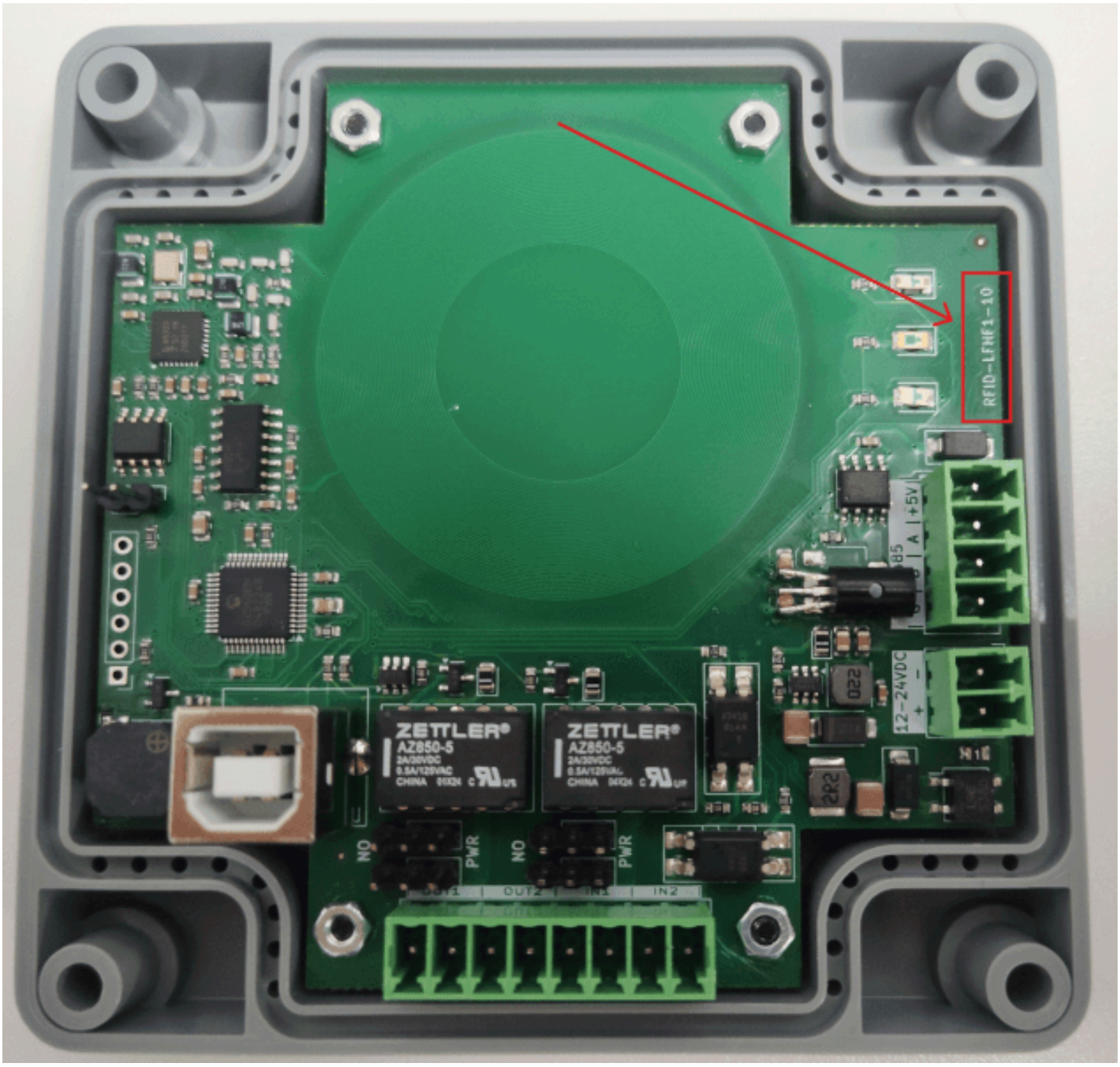
RFID IND Modbus readers with v2 hardware are supported by the Inveo RFID Config programme. V1 devices should be configured with the RFID U1 M1 Config software. To determine the hardware revision of your device:
- Open the enclosure,
- In the lower or right-hand side of the PCB, locate the marking:

RFID-IND-x1-xx indicates a v1 device
RFID-LFHF1-xx is a v2 reader.

Module version v1



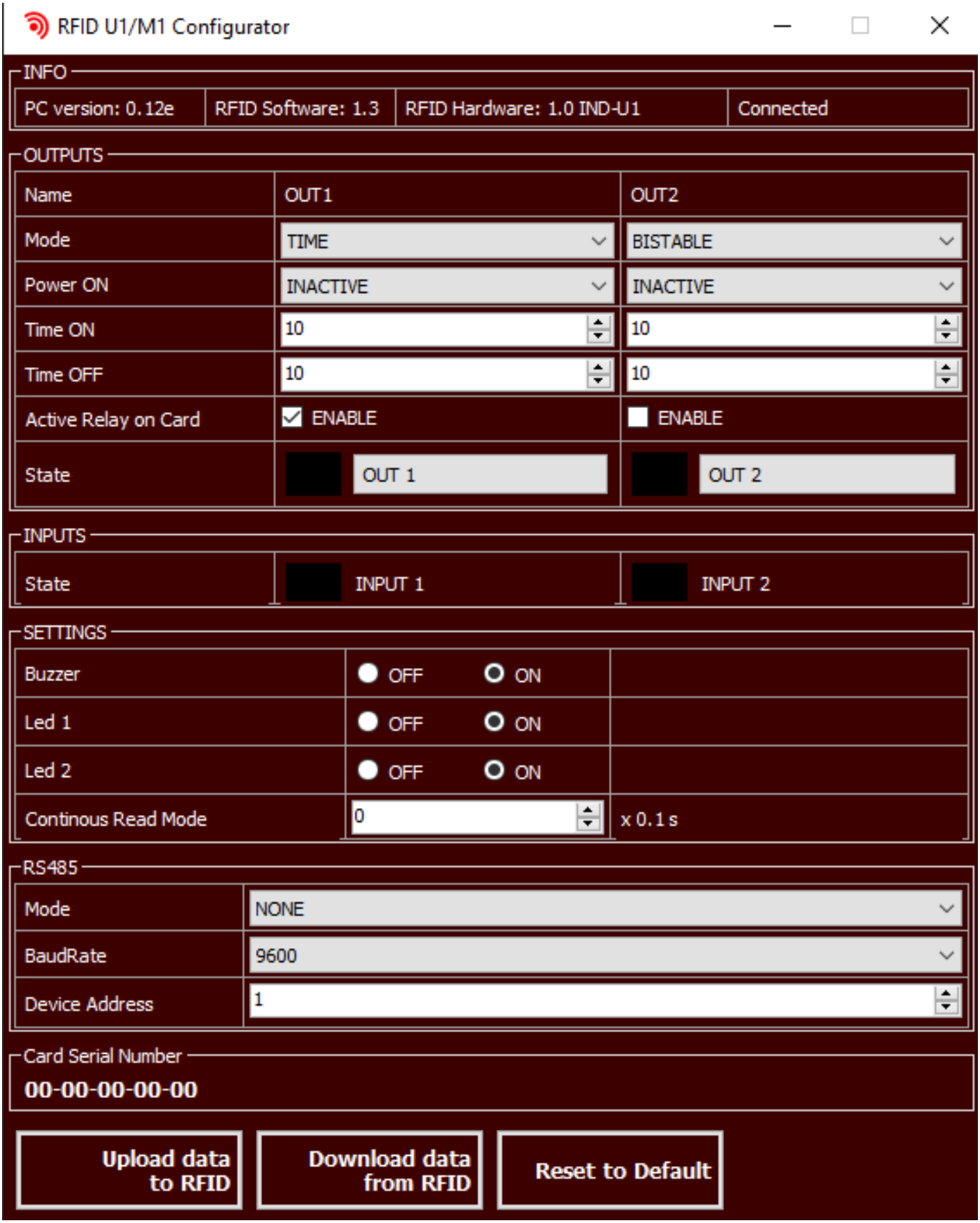
Module version v2



v1 device configuration

To configure the device, use the Inveo RFID M1 / U1 Configurator software, which allows you to define the basic functions of the device. The program can be downloaded from https://inveo.com.pl/software.

After installing the RFID M1 / U1 Configurator configuration program and starting it, connect the USB cable to the computer and the module (in this case, the external power supply of the module is not required – the device is powered via the USB port).

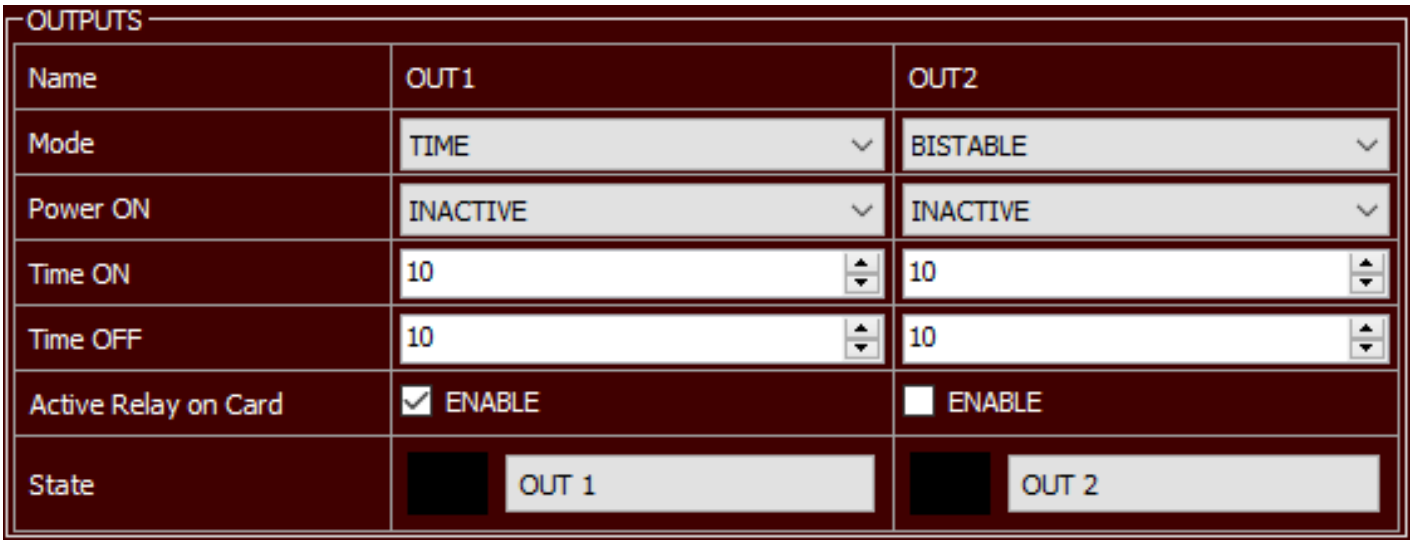


The first line of the programme window displays information about the version of the configuration program – **PC version**, reader software version – **RFID Software** and reader hardware version – **RFID Hardware**.

It also informs the user whether that the reader is connected or disconnected to/from the PC.

Outputs

The reader is equipped with 2 relay outputs. Each output can be programmed individually.



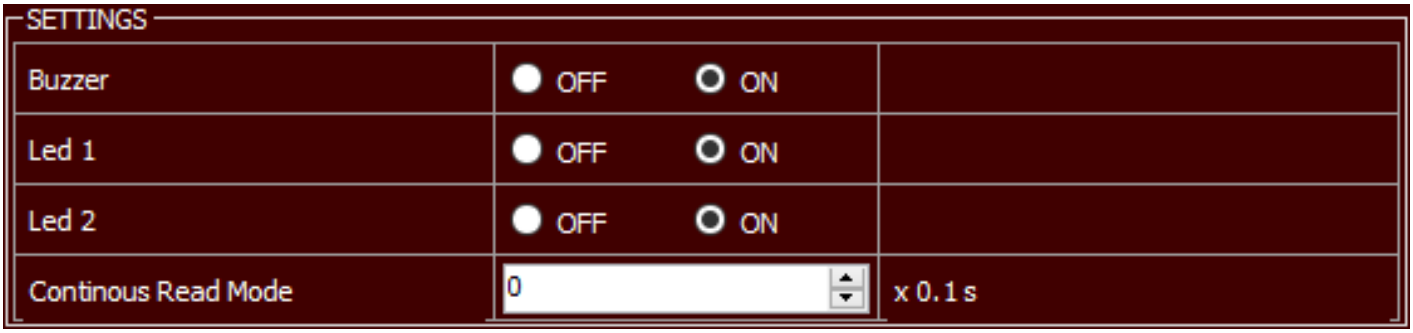
- **Mode** – Output mode. The following options are available:
  - **Disable** – Output control is disabled,
  - **Bistable** – The relay has only one determined status (active or inactive),
  - **Astable** – The relay is activated and deactivated with defined time periods (**Time On** and **Time Off**),
  - **Time** – Output single trigger mode, the output is activated for the defined time (**Time on**) after the specified time passes (**Time off**).
- **Time On** – Time defining how long the relay will be active (Used in **Toggle** and **One-pulse** modes),
- **Time Off** – Time defining how long the relay will be inactive (Used in **Toggle** and **One-pulse** modes),
- **Active Relay on Card** – Activates the relay after a card is applied and read by the reader,
- **State** – Displays the current relay state (yellow - activated, black - deactivated) and gives the user the option to activate it manually.

Inputs

State INPUT 1 and **State INPUT 2** display the current input state. When the rectangle is yellow, the input is active. A black rectangle indicates that the input is inactive.

LED and sound signal

The software allows the user to customize visual and sound signaling.



For the buzzer and LED's, two options can be selected:

- **OFF** – Buzzer/LED off,
- **ON** – Sound a signal/glow when a tag is read.

Name	Description
LED 1	By default, the diode informs about a successful card read. Led colour - red
LED 2	By default, the diode informs about a successful card read. LED colour - green

Note

Regardless of the selected setting, it is always possible to control the signaling via Modbus RTU.

Continuous Read Mode

Using Continuous Read Mode makes reading the same tag possible only after the defined time passes. Other tags are read instantly. This prevents the situation where the same card is accidentally read multiple times. The delay time is specified in 0,1s units.

Tip

It should be considered that using this option WILL NOT set the new tag read flag. The device will operate autonomically and allow the same tag to be read after the specified time passes.

RS485 communication settings



- **Mode** – 9th transmission bit settings:
  - **None**,
  - **Even**,
  - **Odd**.
- **BaudRate** – Transmission speed selection field (1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200),
- **Device address**.

Card Serial Number – The field displays the last RFID transponder code read and the type of card.

> **Information**
>
> **Upload the configuration to the RFID reader after making changes.**

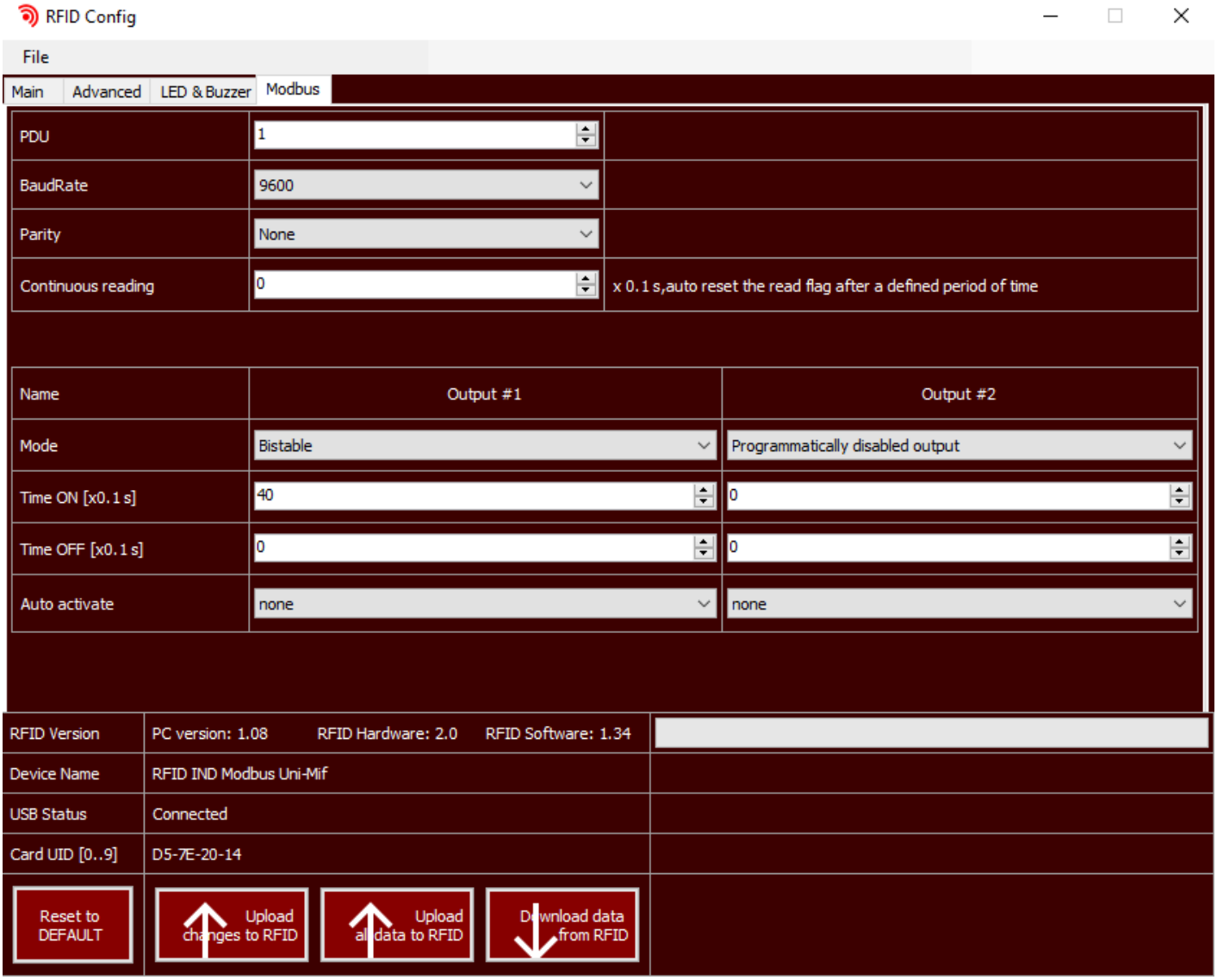
v2 device configuration

RFID IND Modbus readers with v2 hardware are supported by the Inveo RFID Config programme. V1 devices should be configured with the RFID U1 M1 Config software. To determine the hardware revision of your device, read the **Determining the module version** chapter.

After installing and running the configuration programme, connect an USB cable between the PC and the module (external power is not required in this case - the device is powered via USB).

Modbus

The **Modbus** tab enables the user to configure the RS485 communication and assign parameters of the module's outputs.



RS485 communication settings

PDU	1	
BaudRate	9600	
Parity	None	

- **PDU** – Device's modbus address,
- **BaudRate** – Transmission speed selection field (1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200),
- **Parity** – 9th transmission bit settings:
  - **None**,
  - **Even**,
  - **Odd**.

Continuous reading

The device allows definition of a tag read delay.

Continuous reading	0	x 0.1 s,auto reset the read flag after a defined period of time

Using Continuous Read Mode makes reading the same tag possible only after the defined time passes. Other tags are read instantly. This prevents the situation where the same card is accidentally read multiple times. The delay time is specified in 0,1s units.

> **Tip**
>
> **It should be considered that using this option WILL NOT set the new tag read flag. The device will operate autonomically and allow the same tag to be read after the specified time passes.**

Outputs

The reader is equipped with 2 relay outputs. Each output can be programmed individually.

Name	Output #1	Output #2
Mode	Bistable	Programmatically disabled output
Time ON [x0.1 s]	40	0
Time OFF [x0.1 s]	0	0
Auto activate	none	none

- **Mode** – Output mode. The following options are available:
  - **Disable** – Output control is disabled,
  - **Bistable**,

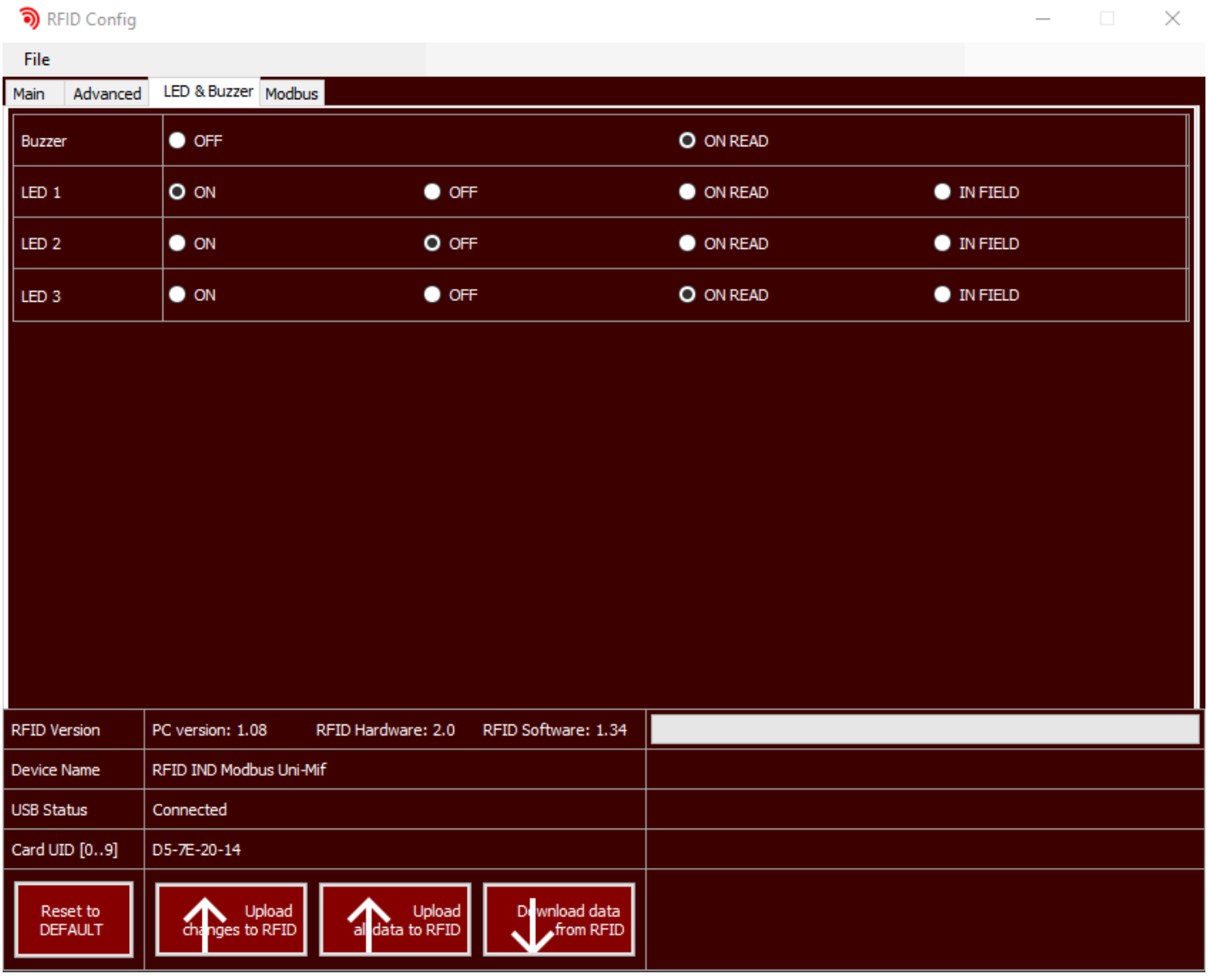
- **Toggle** – The relay is activated and deactivated with defined time periods (**Time On and** Time Off**)
- **One-pulse** – Output single trigger mode, the output is activated for the defined time (**Time on**) after the specified time passes (**Time off**),

- **Time On** – Time defining how long the relay will be active (Used in **Toggle** and **One-pulse** modes),
- **Time Off** – Time defining how long the relay will be inactive (Used in **Toggle** and **One-pulse** modes),
- **Auto activate** – Select action that will trigger the output:
  - **on read** – Output will be triggered after a card is applied to the reader,
  - **on card** – Output will be triggered if the card remains in the read field.

> **Information**
>
> **Upload the configuration to the RFID reader after making changes.**

LED's and sound

The **Inveo RFID Config** programme enables simple configuration of visual and sound signalling. All you have to do is to go to the **LED & Buzzer** tab and select the appropriate settings.



For the buzzer, two options can be selected:

- **OFF** – Buzzer off,
- **ON READ** – Sound a signal when a tag is read.

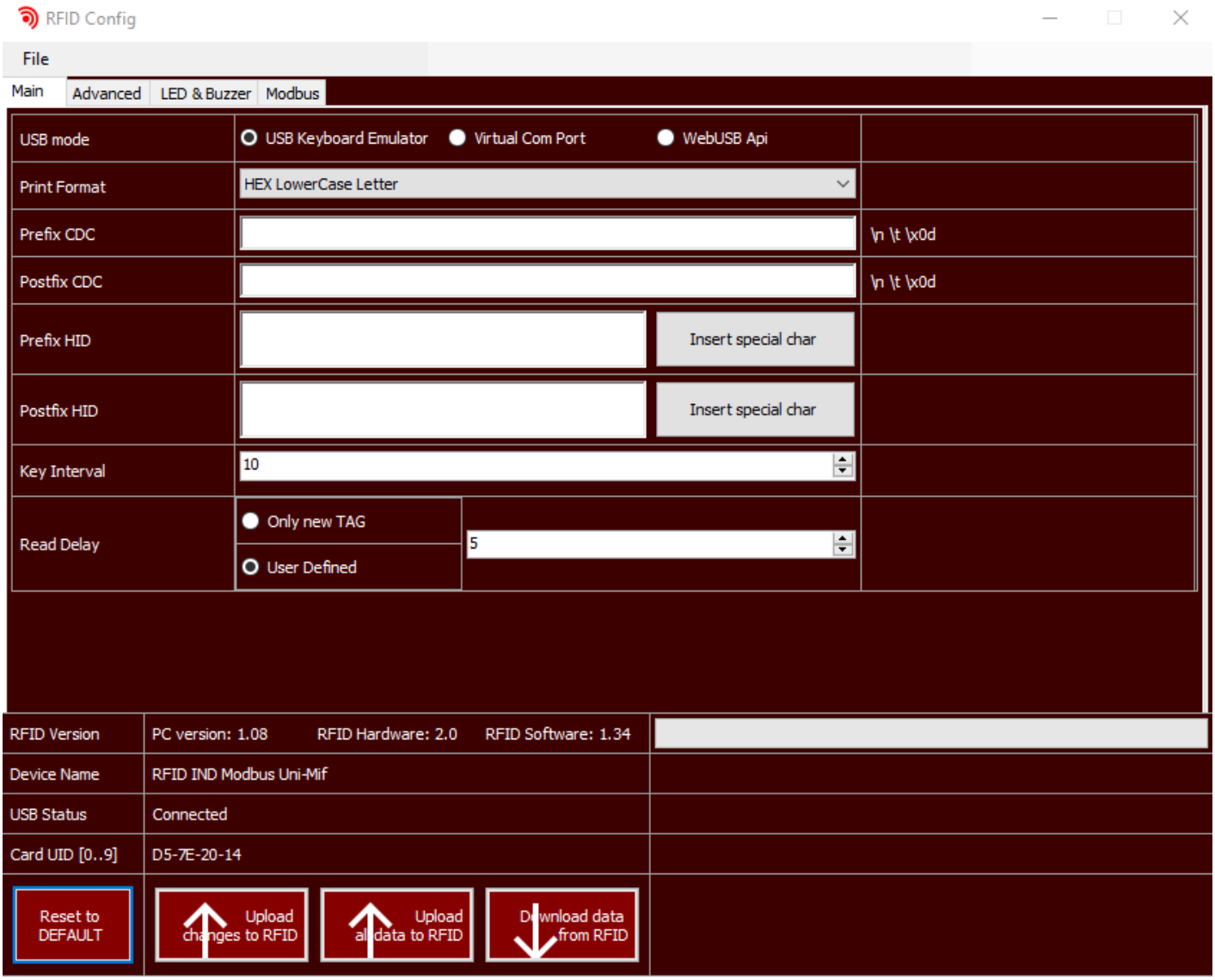
Name	Description
LED 1	By default, the diode informs about the fact that power is connected to the module. LED colour - green.
LED 2	By default, the diode is not used. Led colour - red.
LED 3	By default, the diode informs about a successful card read. LED colour - green.

The user can configure the LED's according to his needs. Each LED can be configured in 4 ways:

- **ON** – The LED glows constantly when power is connected,
- **OFF** – LED disabled,
- **ON READ** – The LED signals a successful tag read,
- **IN FIELD** – The LED signals that a tag has been applied and glows when it is in range of the module.

Main tab – Only in special versions

> **Warning**
>
> **The Main tab is relevant only to special RFID IND Modbus versions.**


RFID Config
File
Main
Advanced
LED & Buzzer
Modbus
USB mode
USB Keyboard Emulator
Virtual Com Port
WebUSB Api
Print Format
HEX LowerCase Letter
Prefix CDC
\n \t \x0d
Postfix CDC
\n \t \x0d
Prefix HID
Insert special char
Postfix HID
Insert special char
Key Interval
10
Read Delay
Only new TAG
User Defined
5
RFID Version
PC version: 1.08
RFID Hardware: 2.0
RFID Software: 1.34
Device Name
RFID IND Modbus Uni-Mif
USB Status
Connected
Card UID [0..9]
D5-7E-20-14
Reset to DEFAULT
Upload changes to RFID
Upload all data to RFID
Download data from RFID

In the programme's bottom secion, the current USB port (to which the device is connected) status and basic reader information should be displayed:

RFID Version	PC version: 1.08	RFID Hardware: 2.0	RFID Software: 1.34
Device Name	RFID IND Modbus Uni-Mif		
USB Status	Connected		

- **PC version** – PC application version,
- **RFID Hardware** – RFID reader hardware version,
- **RFID Software** – RFID reader software version.

If the reader was detected by the programme, configuration can begin.

The USB mode mode of operation should be selected first. Available options:

- **USB Keyboard emulator** – Standard keyboard emulation (HID),
- **Virtual Com Port** – Virtual COM port communication (CDC),
- **WebUSB Api** – Communication via a web page.

Keybord emulation mode

In keyboard emulation mode, the reader behaves like a standard USB keyboard. After an RFID tag is applied, its number is stored in the space selected by the cursor in an appropriate format.

Virtual CDC serial port mode

If the **Virtual Com Port mode** is selected, the reader sends the applied tag's UID number to a virtual serial port. In this mode both the sound and LED indicators can be controlled.

Sending the byte to the virtual com port causes a reaction:

Bit	Function	dec	hex	Description
0		1	1	LED 1
1		2	2	LED 2
2		4	4	LED 3
3		8	8	Continuous tone

4		16	10	Ascending tone
5		32	20	Descending tone
6	*	64	40	**0** – Standard operating mode, the code is sent after a card is applied **1** – The code is sent by request only
7	**	128	80	**1** – Card code send request

Tip

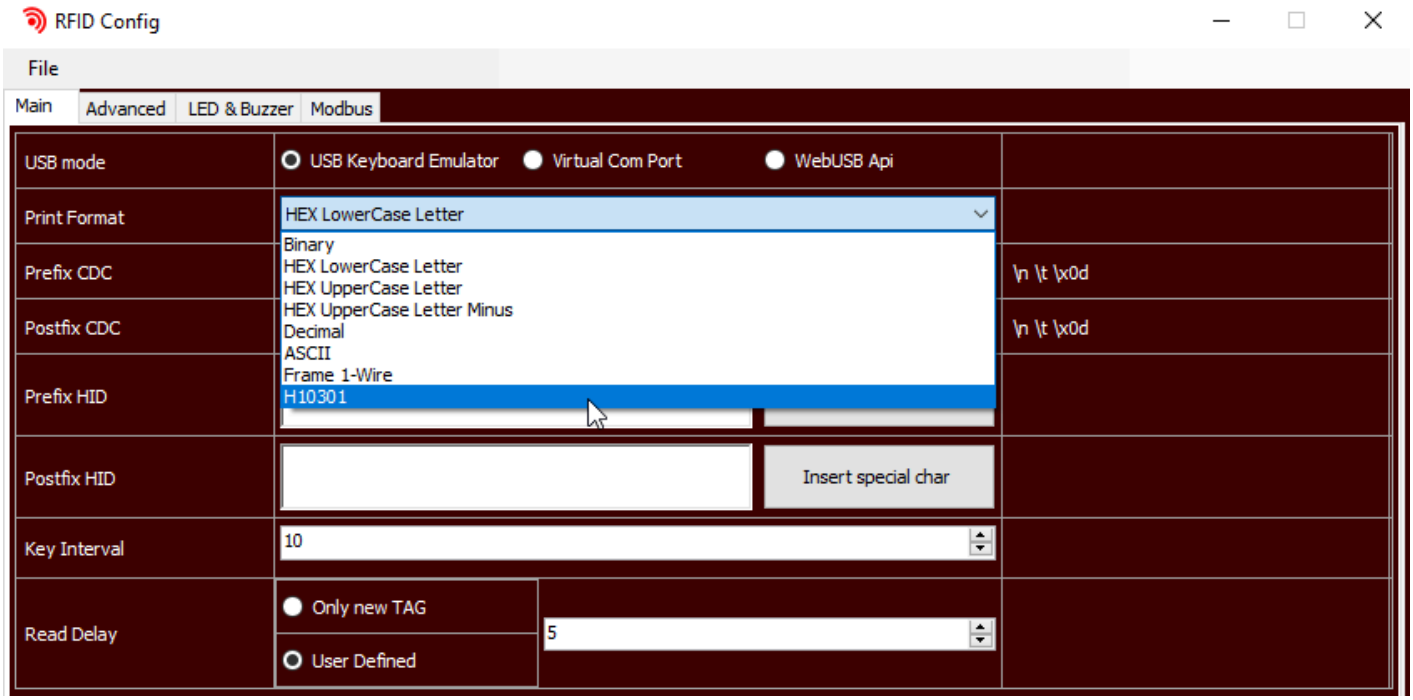
If a request (7th bit) is sent but there is no card in the read field, the device will not return a value.

Examples

- To enable **LED1** and **LED 3**, value of 5 (0x05) should be sent,
- To enable the sound signal (continuous tone, bit 3), value of 8 (0x08) should be sent,
- To enable the sound signal (momentary tone, bit 5), **LED1**, and **LED3**, value of 37 (0x25) should be sent,
- To send a card read request, value of 192 (0xC0 - sum of bit 6 and 7) should be sent.

Number sending format

The next step is to set the format for sending the read card number.



The following output data formats are available:

- **Binary** – The card code bytes are sent in the binary format (applies to the Virtual Com Port mode),
- **HEX LowerCase Letter** – The tag code is sent in the hexadecimal format with lower case letters,
- **HEX UpperCase Letter** – The tag code is sent in the hexadecimal format with upper case letters,
- **HEX UpperCase Letter Minus** – The tag code is sent in the hexadecimal format with upper case letters, dashes separating the code bytes:

Data format	Example
HEX LowerCase Letter	`aabbccdd`
HEX UpperCase Letter	`AABBCCDD`
HEX UpperCase Letter Minus	`AA-BB-CC-DD`

- **Decimal** – The tag is converted to decimal format:

Example

Tag read: `A6-0A-9D-95`
Value displayed: `2785713557`

- **ASCII** – Decoding of card data to the ASCII format (applies to Mifare tags). The string must end with a "0".
- **Frame 1-Wire** – DS2401 emulation.
- **H10301**

Prefix CDC – Is a string of numbers and letters sent before a tag is read in the virtual serial port mode.

Postfix CDC – Is a string of numbers and letters sent after a tag is read in the virtual serial port mode.

Prefix HID – Is a string of numbers and letters or special characters (Alt, Enter, key combination) sent before a tag is read in the keyboard emulation mode.

Postfix HID – Is a string of numbers and letters or special characters (Alt, Enter, key combination) sent after a tag is read in the keyboard emulation mode.

Key Interval – the delay between sent characters. This setting is used on slower devices such as tablets. If the device to which the reader is connected isn't fast enough to input the UID or characters defined in the Prefix/Postfix fields correctly (eg. doesn't input all characters), define a longer time delay between characters (Key Interval). The default value is 10.

Read Delay - Only new TAG – readout of the same tag is possible only after the defined time has passed, other tags are read immediately. This setting prevents the card from being read multiple times. The delay time is defined in x * 0,1s.

Read Delay - User Defined – User defined tag readout delay time. The delay time is defined in x * 0,1s.

To define a special character to be sent as a HID prefix or postfix, click **Insert special char** next to the appropriate position.

Using the virtual keyboard window, any key and character combination can be created.



Example

The reader should display the following text string when a card is applied:
User logged in: (UID NUMBER HERE). and input enter key.

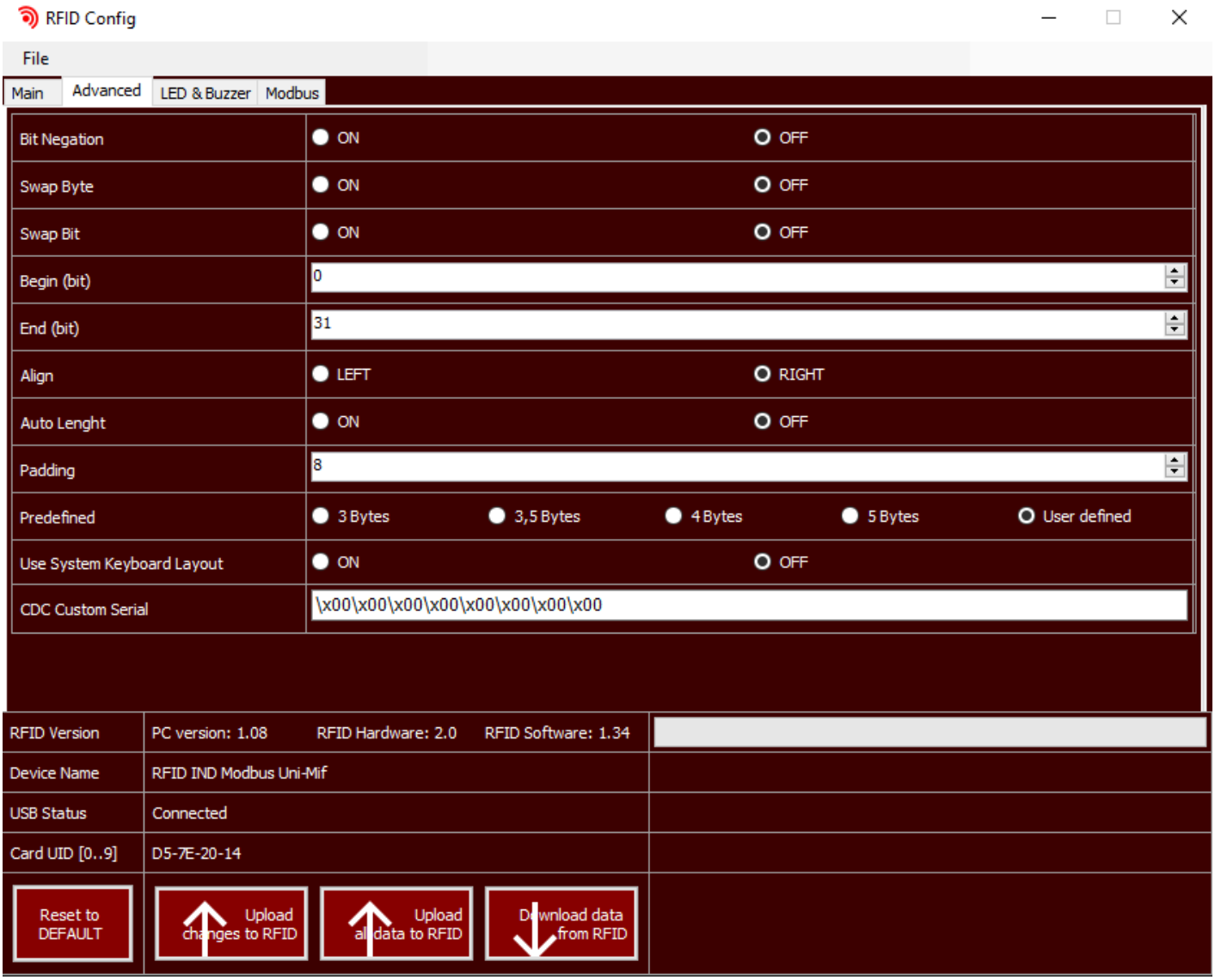
First, the **Prefix HID** field should be filled with: "User logged in:"

In the **Postfix HID** field, "." should be inserted and an Enter key pressed on the virtual keyboard.

To confirm the settings, it's required to press the **Upload data to RFID** button.

Once this is done, every tag application will cause this string to display:
"User logged in: `UIDnumber`."

Under the **Advanced** tab, advanced code readout options are located.



The application allows modification of the read code in a wide range.
The following options are available:

Bit Negation – If selected, the read values are negated:

	HEX	BIN
Read tag	04-00-1B-B7-BC	00000100-00000000-00011011-10110111-10111100
Modified tag	FB-FF-E4-48-43	11111011-11111111-11100100-01001000-01000011

Swap Byte – Function enabling byte swap:

	HEX	BIN
Read tag	04-00-1B-B7-BC	00000100-00000000-00011011-10110111-10111100
Modified tag	BC-B7-1B-00-04	10111100-10110111-00011011-00000000-00000100

Swap Bit – Function causing the bit string to change:
Replacement of the first bit with the last one, the second bit with the second last one etc.

Before modification HEX	Before modification BIN	After modification BIN	After modification HEX
04BC	0000 0100 1011 1100	0011 1101 0010 0000	3D20

	HEX	BIN
Read tag	04-00-1B-B7-BC	00000100-00000000-00011011-10110111-10111100
Modified tag	3D-ED-D8-00-20	00111101-11101101-11011000-00000000-00100000

Begin (bit) – Number of the bit the reader begins the UID readout with.

End (bit) – Number of the bit the reader ends the readout operation with.

Input of a wrong value to the **Begin / End (bit)** field will result in a change to the entire tag value, because the bits of each byte will be shifted:

The UID code is in a 0-39 range. The user filled the fields as follows:
Begin (bit) = 1
End (bit) = 40
correct TAG: 04-00-1B-B7-BC → TAG read: 08-00-37-6F-78,

Begin	End	TAG (in bytes)	TAG (in bits)
0	39	04-00-1B-B7-BC	**0100000000000001101110110111101111 00**
1	40	08-00-37-6F-78	**1000000000000011011101101111011110 00**

The entire tag (in bits) was shifted, that's why the value in bytes is completely different.

Align – Alignment of the read tag,

Example 1

Bytes 3, 5 are to be read off the tag: 01-0F-A0-D2-61
Begin (bit): 12
End (bit): 39
Auto Length: ON

Operation	Value
Complete tag code (bytes)	01-0F-A0-D2-61
Complete tag code (bits)	00000001 00001111 10100000 11010010 01100001
Byte to be aligned	XF-A0-D2-61
Align RIGHT (B)	FA-0D-26-10
Align RIGHT (b)	11111010 00001101 00100110 00010000
Align LEFT (B)	0F-A0-D2-61
Align LEFT (b)	00001111 10100000 11010010 01100001

Example 2

Bytes 1, 5 are to be read off the tag: 05-00-EE-9C-86
Begin (bit): 28
End (bit): 39
Auto Length: ON

Operation	Value
Complete tag code (bytes)	05-00-EE-9C-86
Complete tag code (bits)	00000101 00000000 11101110 10011100 10000110
Byte to be aligned	XC-86
Align RIGHT (B)	C8-60
Align RIGHT (b)	11001000 01100000
Align LEFT (B)	0C-86
Align LEFT (b)	00001100 10000110

Auto Length – Automatic adjustment of the read code length.

Padding – Manual setting of the code value length. The value of 1 equals to one character.

Example

UNIQUE standard, each UID code has 5 bytes (for HEX format).
Input 0 / 39 to the Begin / End (bit) fields respectively.
In the Padding field input the value of 10:
From Begin (bit)=0 to End (bit) = 39 → 40 bits = 5 bytes (for the HEX format it will be 10 characters → Padding = 10).

Predefined – Predefined values for **Begin (bit)** and **End (bit)**.

Use System Keyboard Layout – The system keyboard layout is taken into consideration when letters and numbers are inputted.

Example

Polish keyboard layout:



Hungarian keyboard layout:



CDC Custom Serial – Optional number that can be read in the virtual COM port mode.

Modbus API

The device is designed to be controlled via the Modbus serial communication protocol.

Modbus API allows the user to:

- Read and write any block of the Mifare tag (only Mif version),
- Set the security (authorization) to write and read data to each Mifare block (only Mif version),
- Control signaling (LED, Buzzer),
- Check the process result.

Modbus addressing

Modbus is available via RS485 (Modbus RTU).
The device supports the following Modbus RTU functions:

- 0x01 Read Coils
- 0x03 Read Holding Register
- 0x05 Write Single Coil
- 0x06 Write Single Register
- 0x0F Write Multiple Coils
- 0x10 Write Multiple Registers

> **Tip**
>
> **In order for the Modbus protocol to operate correctly, the RFID Config programme must be closed!**

Holding Registers:

Address	R/W	Description
1000	R/W	**IsNewTag** – a new tag has been read: 1 - Tag recognized 0 - no tag The read flag must be reset (zeroized) before reading the next tag. The Holding Registers address corresponds to the Single Coil Register 1016 function
1001	R	**UID Length** – For Unique: typically 5, Mifare: 4, 7 or 10
1002 - 1011	R	**UID** bytes

1012	R	**Card Type** – only Mif version
1017	R	**MODEL ID** – device ID
1018	R	**Software Version**
1019	R	**Hardware Version**
1020	R/W	**Mode OUT1** – Output No. 1 mode: 0 – Disable 1 – Bistable 2 – Toggle – astable mode 3 – One-pulse – single trigger mode
1021	R/W	**Time On OUT1** – time, defining how long the output will be activated for (1-65535) (*0,1s) eg. 120 means 12 seconds
1022	R/W	**Time Off OUT1** – time, defining how long the output will be deactivated for (1-65535) (*0,1s)
1023	R/W	**Mode OUT2** – OUT2 mode Similarly to Mode OUT1
1024	R/W	**Time On OUT2** Similarly to TimeOn OUT1
1025	R/W	**Time Off OUT2** Similarly to TimeOff OUT1

Only Mif version

Address	R/W	Description
2000	W	**WriteEnable** – enable TAG writing
2001	W	**ReadEnable** – enable TAG reading
2002	R/W	**TagType** – 1k Mifare (0) or 4k Mifare (1)
2003	R/W	**MemoryMode** – Linear mode (0) or Full mode (1) memory buffer mode
2004	R/W	**DataMode** – data presentation mode in the buffer (0-2)
2005	R/W	**Recognized card signalling mode** Format (binary): xxxx xxxx xxxx BB12, where: BB – 0 no sound signal, 1 accept signal, 2 reject signal

		1 – led 1 2 – led 2
2006	R/W	**Read operation signalling mode:** Format (binary): xxxx xxxx EE34 BB12, where: EE – 0 no sound signal, 1 accept signal, 2 reject signal for wrong operation 1 – led 1 for wrong operation 2 – led 2 for wrong operation BB – 0 no sound signal, 1 accept signal, 2 reject signal for correct operation 1 – led 1 for correct operation 2 – led 2 for correct operation
2007	R/W	**Write operation signalling mode** As shown above.
2008	R/W	**HaltTag** – Writing 1 will cease reading the current tag. The reader enters the mode of awaiting a new tag. IsNewId and HaltTag are zeroized.
2009	R/W	**WakeAll** – Activation of all tags – Writing 1 momentarily cuts the power to the RFID antenna. After that, all previously stored tags are available again. IsNewId and WakeAll are zeroized.
2010	R/W	**ReadResultGlobal** – Last readout result: 0 no read error, 1 – error. This is a logic sum of read errors from all blocks (ReadResultCode)
2011	R/W	**WriteResultGlobal** – Last write result: 0 no write error, 1 – error. This is a logic sum of write errors from all blocks (WriteResultCode)
2020-2035	R/W	**RunReadFlag** – selection of the block to be read. One bit controls one block. Eg. 0x0031 means readout of block 0 from sector 0 and block 0+1 from sector 1
2040-2055	R/W	**RunWriteFlag** – selection of the block to be written. One bit controls one block.
2100-2355	R/W	**ReadAuthorization** – set authorization to any block during the read operation. Format (hex): xAxK, where: A – block authorization type (0 for key A, 1 for B), K – number of the stored key 0-7. Eg. 0001 indicates type A authorization and second key
2400-2655	R/W	**WriteAuthorization** – Write operation authorization. As shown above.
2700-2955	R/W	**ReadResultCode** – result code of all block read operations (0 – no error)
3000-3255	R/W	**WriteResultCode** – result code of all block write operations (0 – no error)

4000-8095	R/W	**ReadBufferMemory**
10000-14095	R/W	**WriteBufferMemory**

Single Coil:

Address	R/W	Description
1000	R/W	**ON 1** – relay 1 control (on/off)
1001	R/W	**ON 2** – relay 2 control (on/off)
1002	R	**COIL STATE 1** – relay 1 coil state
1003	R	**COIL STATE 2** – relay 2 coil state
1010	R	**INPUT 1** – input 1 state
1011	R	**INPUT 2** – input 2 state
1012	R/W	**LED1** – LED 1 (green, power supply) control
1013	R/W	**LED2** – LED 2 (red, fault) control
1014	W	**BUZZ ACCEPT** – enable acceptation signal
1015	W	**BUZZ REJECT** – enable rejection signal
1016	R/W	**IsNewTag** – a new tag has been read: 1 - Tag recognized 0 - no tag The read flag must be reset (zeroized) before reading the next tag. Same as the Holding Coil Register 1016 address.
1017	R/W	**ResetFlag** – read flag reset 1 – default reader state after it has been turned on. Can be zeroized and set by the user for diagnostic purposes.

The Coil 1016 register is queried the most often. If its value changes to 1, it means that a new tag has been read by the device. The Holding Registers 1002-1011 contain the tag ID.

After the ID has been read, the read flag should be reset (Coil 1016 or Holding Register 1000) by sending '0'. Only then the reading of next tag will be possible.

The output parameters can be modified using the Modbus protocol. The settings are not stored in the non-volatile memory of the module. After the device is restarted, the settings previously stored in the EEPROM using the configuration programme will be restored.

Mifare tag structure - only Mif version

The 1k Mifare tag structure is presented below. (source: NXP Semiconductor MF1S503x pdf):

		Byte Number within a Block															
Sector	Block	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Description
15	3	Key A						Access Bits				Key B						Sector Trailer 15
	2																	Data
	1																	Data
	0																	Data
14	3	Key A						Access Bits				Key B						Sector Trailer 14
	2																	Data
	1																	Data
	0																	Data
:	:																	
:	:																	
:	:																	
1	3	Key A						Access Bits				Key B						Sector Trailer 1
	2																	Data
	1																	Data
	0																	Data
0	3	Key A						Access Bits				Key B						Sector Trailer 0
	2																	Data
	1																	Data
	0	Manufacturer Data																Manufacturer Block

The Mifare tag is equipped with a 1kB EEPROM, separated into 16 blocks.

Manufacturer Data

The first block of the tag contains the manufacturer data, that is the UID number (often described as Card ID or Serial Number). It is assumed that this block is read-only, but cards that allow writing data to the first block do exist.

Data blocks

Each sector contains 3 data blocks (except for sector 0, containing only 2 blocks). Each block can store up to 16 bytes of data. Each data block can be configured using the access byte as a:

- Read/write block,
- Value block.

Sector Tralier

The Sector Trailer block is located at the end of each sector. It contains the access keys and data block configuration bits:

- Key A (required key),
- Key B (optional key).

Reading and writing

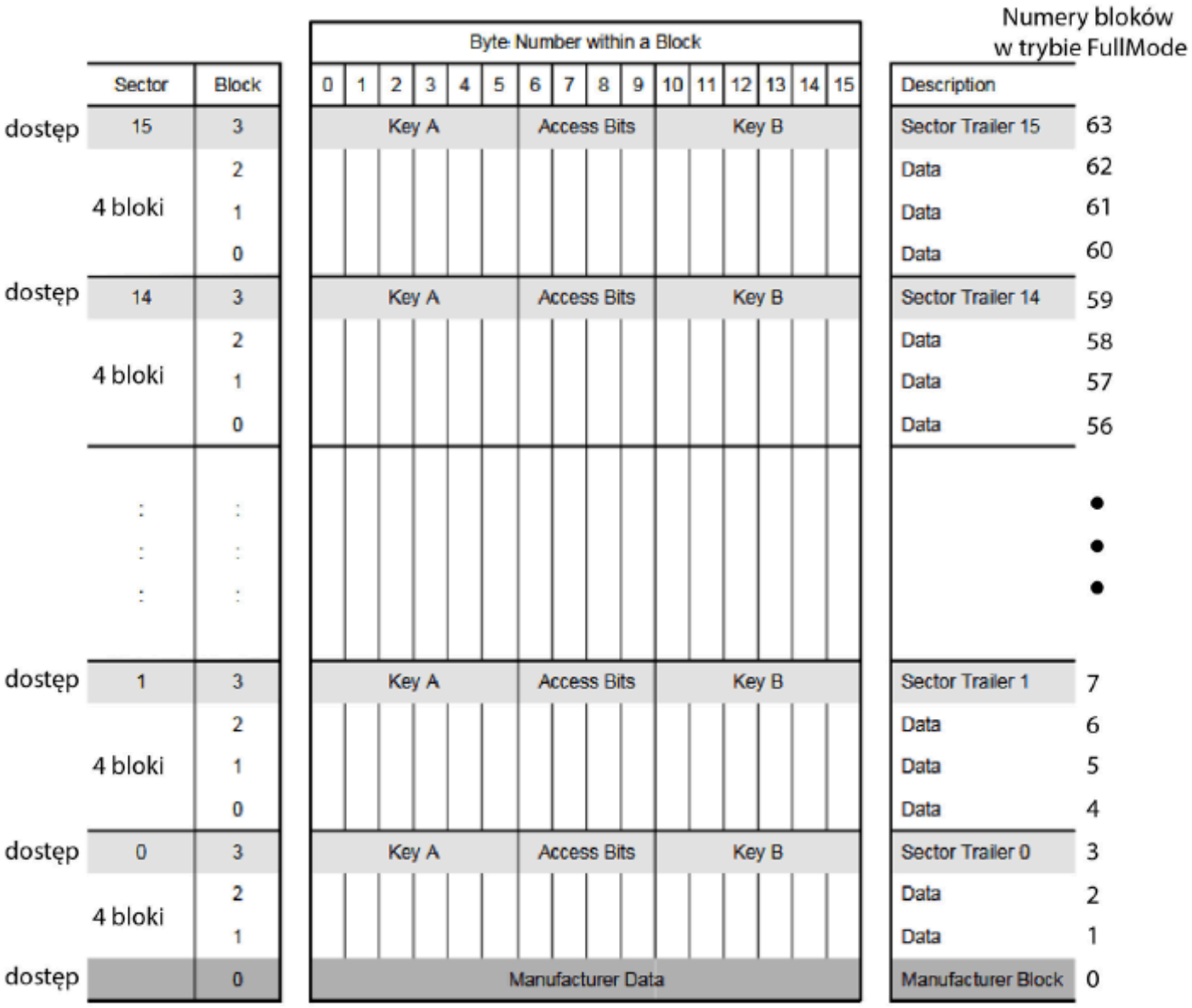
Precisely 16 bytes can be read or written to the tag at once. If the user wants to change 2 bytes of the tag without deleting the other bytes, it's required to read 16 bytes, change the 2 required and write all 16 bytes.

Memory buffer - Only Mif version

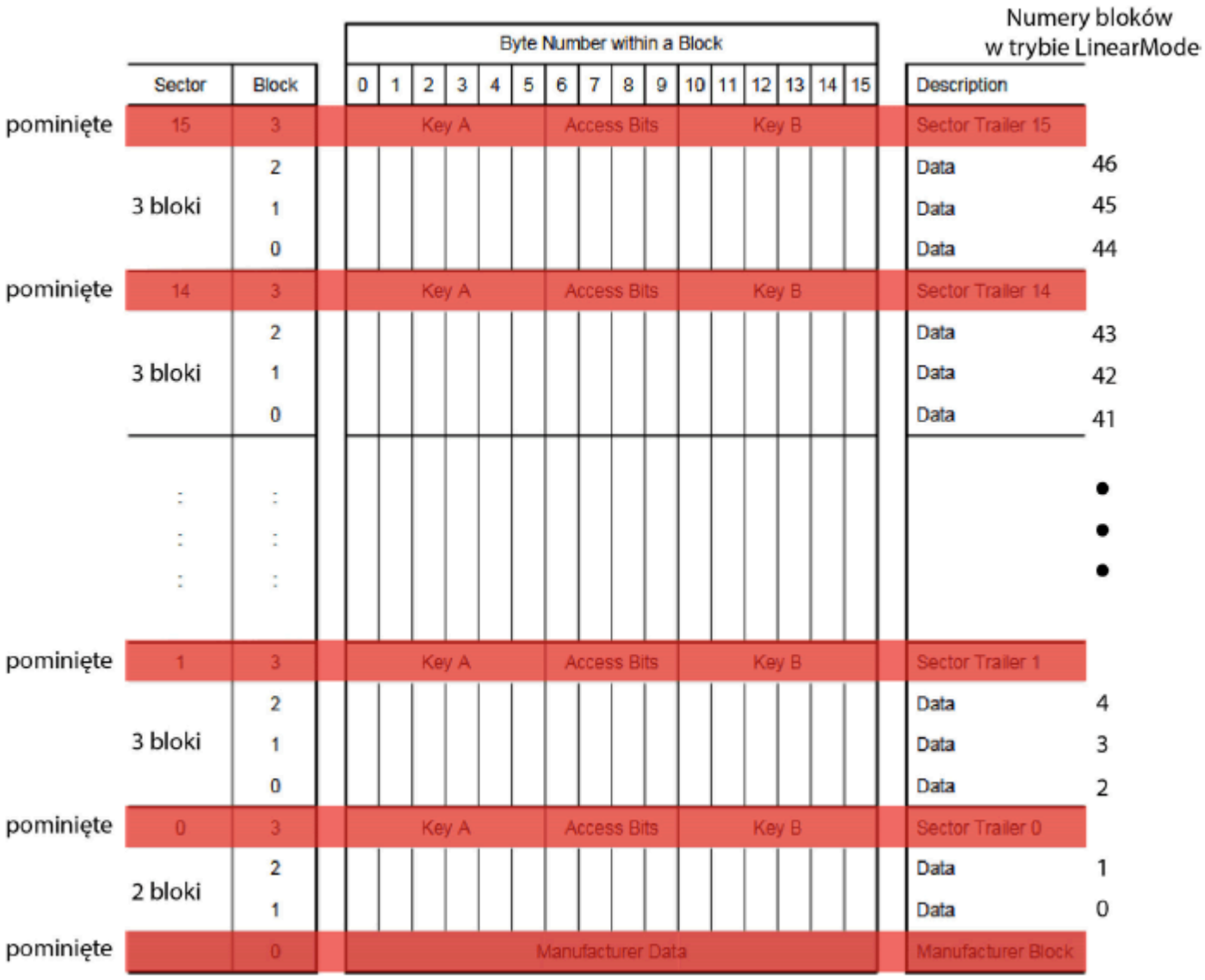
The reader is equipped with a built-in tag data memory. Two buffers (4kB each) are used: one for the read and one for the write operation. Access to the memory can be gained with Modbus Holding Registers.

The reader can operate in two memory addressing modes (**MemoryMode** register):

- **Full mode** – Read and write memory is addressed exacly like the tag memory structure. To read the second byte of the first data block, read 18 and the Modbus Register buffer shift (1*16 + 2 = 18). Block 0 from the 15th sector begins with 15 (sector number)*4 (block in the sector)*(byte in the block) + Modbus Register buffer shift. Make sure that no unwanted data is stored to the Sector Trailer, as this can block sector access.



- **Linear Mode** – The reader calculates the addresses, skipping the manufacturer and all Sector Trailer blocks. Thanks to that, the user has 752 free bytes from the 1024 available. This mode guarantees that no unwanted data will be written to the Sector Trailer blocks, as the application will not have access to them.



Tip

The RunReadFlag and RunWriteFlag command contents depend on the defined memory addressing modes.

Setting **Linear mode** skips the manufacturer and all Sector Trailer blocks. In the RunReadFlag and RunWriteFlag commands, the firs bit indicates the second block of the first sector, in **Full mode** first block of the first sector.

The memory addressing mode is set by sending the appropriate value: **Linear mode** (0) or **Full mode** (1) to the Holding Register 2003.

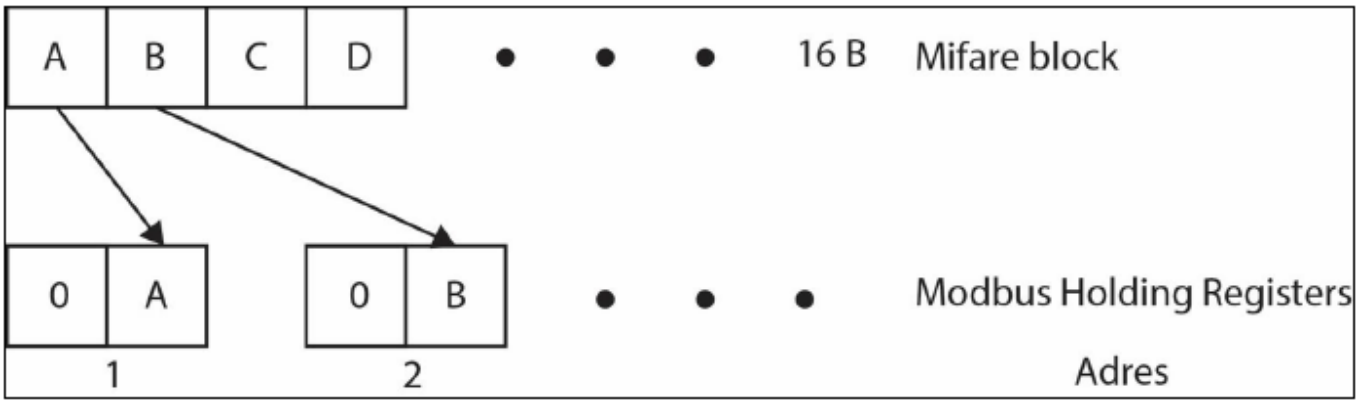
Data representation in Modbus addresses (Endianess)

3 modes are available (**DataMode** register):

- **default** – Each Modbus Holding Register contains one byte of data.

Example

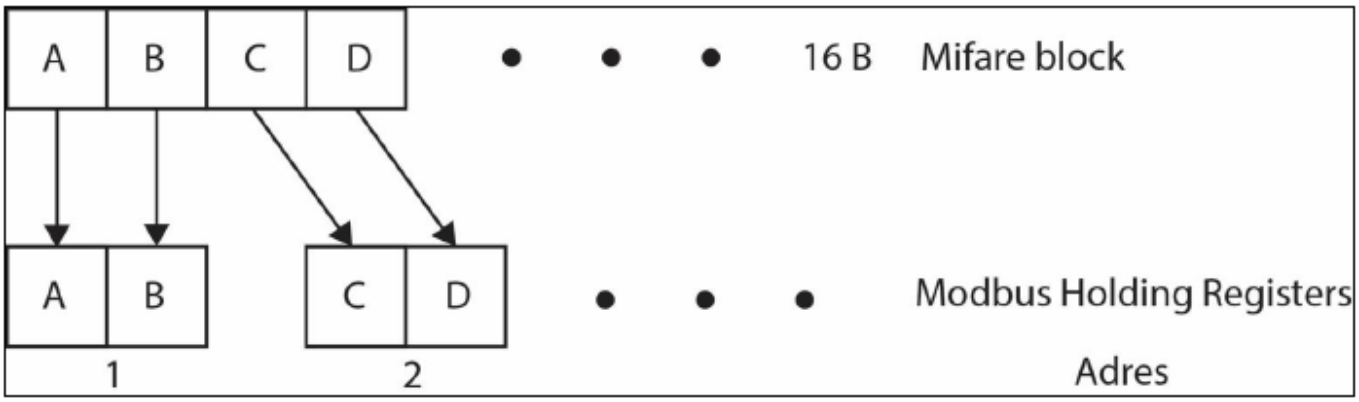
If the tag block 0 has the first bytes: MSB:0x55 and LSB:0xAA, then Modbus Reg 0 (+ Buffer Memory offset) contains „0xAA”, and Modbus Reg 1 contains „0x55”.



- **Endian 1** – Each Modbus Holding Register contains two bytes of data.

Example

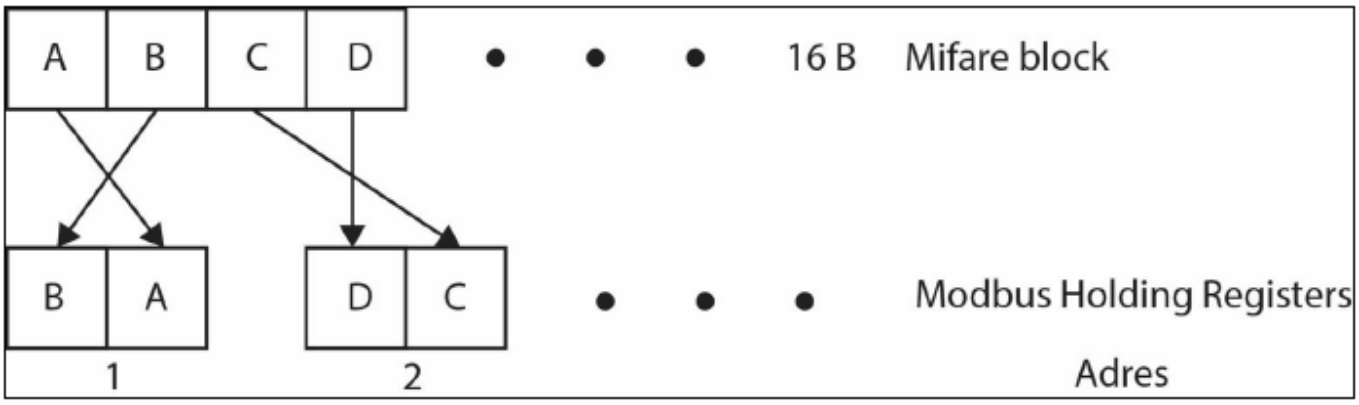
If the tag block 0 has the first bytes: MSB:0x55 and LSB:0xAA, then Modbus Reg 0 (+ Buffer Memory offset) contains „0xAA”.



- **Endian 2** – Each Modbus Holding Register contains two bytes of data.

Example

If the tag block 0 has the first bytes: MSB:0x55 and LSB:0xAA, then Modbus Reg 0 (+ Buffer Memory offset) contains „0xAA55”.



The **Endian 1 i 2** settings can reduce the amount of data transfered between the RFID reader and the main controller (PLC or other).

The data representation is set by sending an appropriate value to the Holding Register 2004.

Application examples

Card UID read operation

1. Apply a tag to the reader.
2. Wait for a 1 to appear in the **IsNewTag** register (1000 Holding Register/1016 Coil).
3. The **UID** will appear in the Holding Registers 1002-1005.
4. Reset the **IsNewTag** flag (1000 Holding Register/1016 Coil), to enable reading of next tag.

Data read operation

1. Select the tag signalling (this step can be omitted).
2. Select **MemoryMode** (2003 Holding Register) and **DataMode** (2004 Holding Register).
3. If required, set **ReadAuthorization** to the register without a default key.
4. Select the block that will be read **RunReadFlag** (2020-2035 Holding Registers).
5. Set the **ReadEnable** flag (2001 Holding Register) to 1.
6. Wait for a 1 to appear in the **IsNewTag** register (1000 Holding Register/1016 Coil).

After a tag is read (**IsNewTag** = 1):

1. Read the selected memory area from the **ReadBuffer** (registers 4000-).
2. Initiate the response signal if not selected automatically.

3. Check the **ReadResultGlobal** register (2010 Holding Register) for errors.
4. Reset the **IsNewTag** flag to unlock the reader.
5. Return to step 3.

Data write operation

1. Select the tag signalling (this step can be omitted).
2. Select **MemoryMode** (2003 Holding Register) and **DataMode** (2004 Holding Register).
3. If required, set **WriteAuthorization** to the register without a default key.
4. Write the selected memory area using **WriteBuffer** (10000-). **RunWriteFlags** assigned to the write block will be set automatically while data is written.
5. Optionally, set or reset the **RunWriteFlag** for the written block.
6. Set the **WriteEnable** flag (2000 Holding Register) to 1.
7. Wait for a 1 to appear in the **IsNewTag** register.

After a tag is read (**IsNewTag** = 1):

1. Initiate the response signal if not selected automatically.
2. Check the **WriteResultGlobal** register (2011 Holding Register) for errors.
3. Reset the **IsNewTag** flag to unlock the reader.
4. Return to step 3.

Reading multiple tags

1. Wait for a 1 to appear in the **IsNewTag** register.
2. Execute the operations needed (read, write etc.)
3. After all tag operations are complete, recall **HaltTag**, to block the specified tag and operate on another one.

OR

1. Recall **WakeAll**, to wake all tags and start the operations on all of them from the beginning.

Warranty and manufacturer's liability

> **Warning**
>
> **The manufacturer provides a two-year warranty for the device and post-warranty service for a period of 10 years from the date of introduction of the device to the market. The warranty covers all material and production defects.**

The manufacturer undertakes to respect the warranty agreement, if the following conditions are met:

- All repairs, changes, expansions and device calibrations are carried out by the manufacturer or an authorized service center,
- The power supply system meets the applicable standards,
- The device is operated in accordance with the suggestions presented in this manual,
- The device is operated in accordance with its intended purpose.

The manufacturer assumes no responsibility for consequences resulting from improper installation, improper use of the device, failure to comply with the instruction manual, and repairs made by unauthorized personnel.

> **Warning**
>
> **The device contains no user serviceable parts inside.**

Storage, operation and transport conditions

The device should be stored in enclosed rooms, where the atmosphere is free from vapours and corrosive substances:

- Environment temperature from -30°C to +60°C (-22°F - 140°F),
- Humidity from 25% to 90% (condensation unacceptable),
- Atmospheric pressure from 700 to 1060 hPa.

The device is intended to operate in the following conditions:

- Environment temperature from -10°C do +55°C (14°F - 131°F),
- Humidity from 30% to 75%,
- Atmospheric pressure from 700 to 1060 hPa.

Recommended transport conditions:

- Environment temperature from -40°C do +85°C (-40°F - 185°F),
- Humidity from 5% to 95%,
- Atmospheric pressure from 700 to 1060 hPa.

Installation and device operation:

- The module should be operated in accordance with recommendations provided later in this manual.

Disposal and decommissioning

In an event the device needs to be decommissioned (eg. after its intended life period is surpassed), it is recommended to contact the manufacturer or his representative, who are responsible to respond appropriately, i.e., to collect the device from the user. The user can alternatively contact companies specializing in electronic device or computer equipment disposal and/or decommissioning. Under no condition should the device be placed with other waste.