Inveo Sp. z o.o.

# USB Pocket

Instruction Manual

# Overview

**Warning**

**This manual applies only to the firmware version v1.32 onwards. Inveo does not guarantee that the information contained in this document applies to previous firmware revisions.**




## Purpose of the device

RFID USB Pocket is used to read Unique EM4100 and EM4102 transponders.

The device reads the UID code and can execute advanced tag conversion operations depending on the configuration. The module allows a prefix and postfix in the form of words or special characters (eg. Enter) to be defined.

## Changelog

**1.0** 31st of March 2025

- Firmware revision v1.32

Table of contents


- Overview
  - Purpose of the device
  - Changelog
    - 1.0 31st of March 2025
- Device construction
  - Technical characteristics
  - General characteristics
- Configuring the device using the "RFID Config" programme
  - Basic settings
  - LED's and sound
  - Number reading formats
  - Advanced conversion functions
- Warranty and manufacturer's liability
  - Storage, operation and transport conditions
  - Disposal and decommissioning

# Device construction

## Technical characteristics

| Parameter | Description |
|---|---|
| **Power supply** | **USB 5V DC** (device is powered from the PC's USB port) |
| **Transponders** | **Tag standard:** Unique EM4100, EM 4102<br>**Carrier frequency:** 125kHz<br>**Read distance:** up to 5cm |
| **Communication** | USB port:<br>- keyboard emulation<br>- virtual COM port |
| **Dimensions** | 62,5mm (width) x 15,5mm (height) x 25mm (depth) |

## General characteristics

The reader is equipped with a USB port supporting CDC (Virtual Com Port) and HID (keboard)

The device has two LED indicators and a buzzer.

The LED's can be disabled or set to react to:

- Tag read,
- Tag application,
- Power.

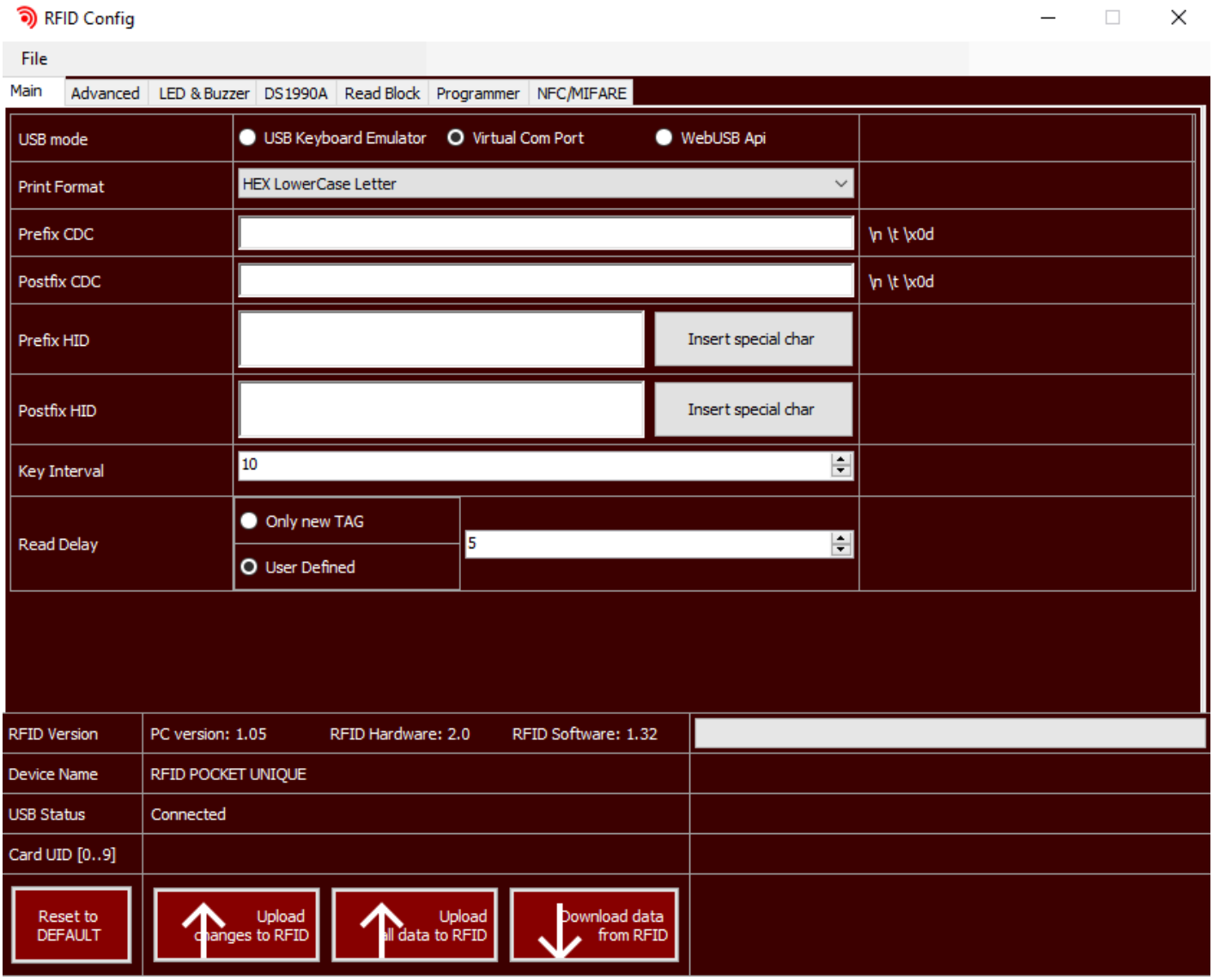
The sound signal can be disabled or set to generate a sound after reading an RFID tag.

# Configuring the device using the "RFID Config" programme

## Basic settings

To configure the device, download the free **RFID Config** programme from the **INVEO** website.

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




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

## LED's and sound

The **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.

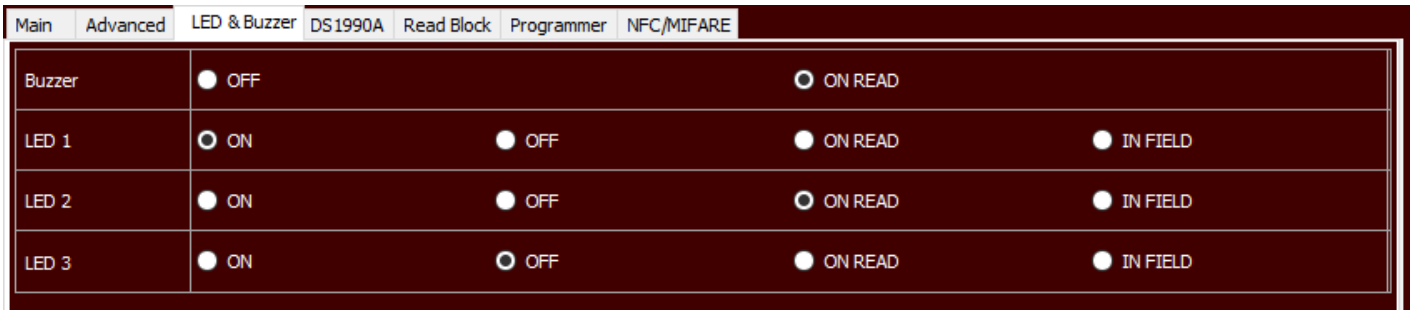
For the buzzer, two options can be selected:

- **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 informs about a successful card read. LED colour - red. |
| **LED 3** | Not used |

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.




## Number reading formats

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:"

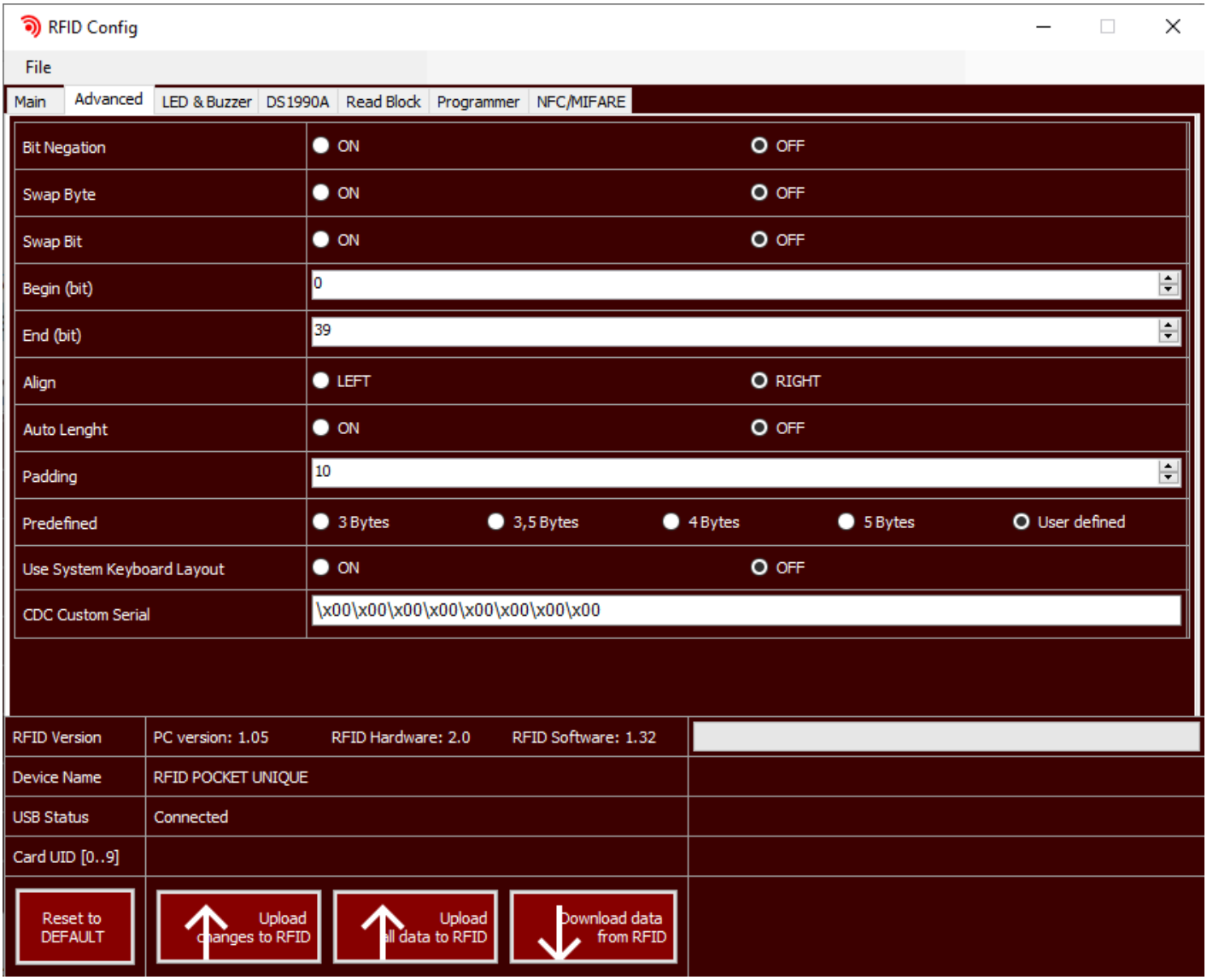
In the **Postfix HID** field, "." should be inserted and an Enter key pressed on the virtual keyboard.

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`."

## Advanced conversion functions

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.

## 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.