

TapNPass, TnP-FSR103

Quick Start Guide

Portable Serial to Bluetooth Low energy (BLE)
adapter for deported HMI apps on mobile devices

August - 2019

Package Contents

TapNPass Fix NFC-BLE to Serial fieldbus adapter includes:

- 1 x **TnP-FSR103**
- 1 x **RS-232 cord**
(RJ45-DB9)
- 1 x **RS-485 cord**
(RJ45-DB9, blue marker)
- 1 x **Modbus adapter**
(RJ45-RJ45)
- 1 x **Terminal Block Plug**
(2-wire)



Use an off-the-shelf **USB (A) cord** when connecting to a target's USB connector. This cord is not provided with TapNPass.

Note: For Modbus connections, plug the provided Modbus adapter into the TapNPass RJ45 port and use an off-the-shelf network cord to connect to the target. The network cord is not provided with TapNPass.

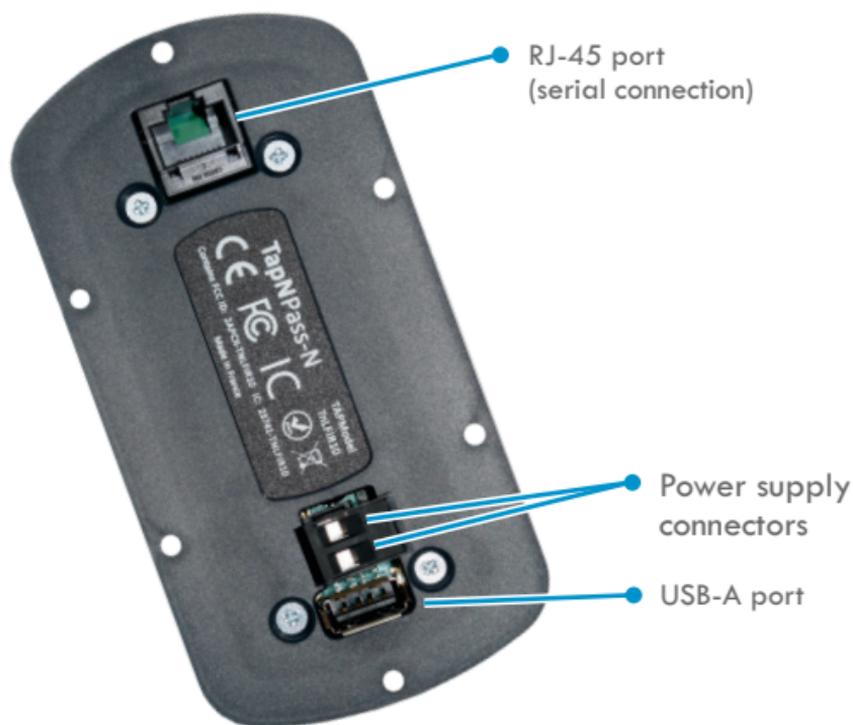
LED descriptions

	LED	Condition	Description
	Power	Flashing Green Off	Power on, BLE emitting. Power off.
	Communication	Flashing Orange Flashing Blue Off	Receiving data from target. Sending data to target. No communication.

Note:

A placement for a battery LED () may be visible on the front of **TapNPass Fix** products. This LED is only implemented for the **Nomad** versions to indicate the charge state of the battery. It is not used on the **Fix** products.

Hardware Description



Power Consumption Characteristics

TapNPass is a low power-consumption device (typically 200 mW). However, overall consumption depends on the type of connection to the target system. For example, a USB device connected on the USB (A) port may consume up to 2.5W (5V - 500mA). The following are examples of measured consumption with a 5V source:

- RS-232: 35mA
- RS-485: 45mA
- USB: 110mA (with a FTDI or CP210x connected cable)

Power Supply

Connect an external power supply on the provided connector. The supported voltage range is 10V to 30V.

It is recommended to use a power source of 5W (12V/400mA or 24V/200mA) and connection wires of 2.5mm².

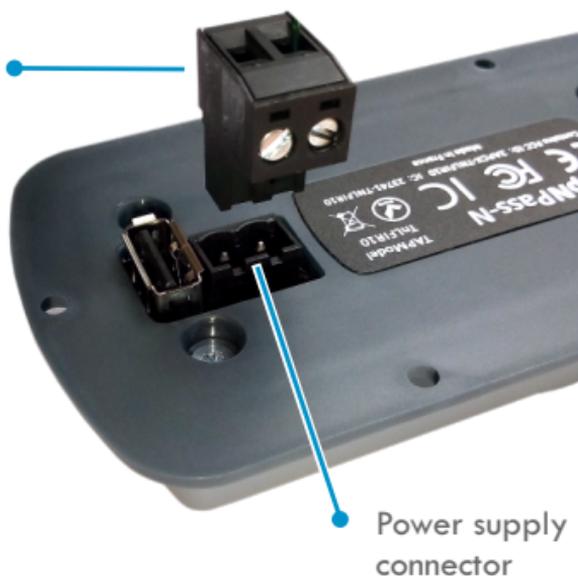
To connect the power supply:

- Insert the provided terminal block plug (Molex 039530-0002) into the power supply connector on the back of the TapNPass.
- Insert the two wires from the power source

Note: The polarity does not matter. The voltage input port is equipped with a bridge rectifier.

- Tighten the 2 screws to secure the wires in place

Terminal block plug



Power supply
connector

I. Connect to a Target

Connect TapNPass to the target system. Available connections are:

Connection Type	Cord	TapNPass Connector	Target Connector
Serial RS-232	RS-232 cord ¹	RJ45	DB9
Serial RS485	RS-485 cord ¹	RJ45	DB9
Serial USB (host)	USB (A) ²	USB (A)	USB
Serial w/Modbus-RTU	Modbus adapter ¹ + network cable ²	RJ45	RJ45

Notes:

1. Provided with TapNPass.
2. Not provided. Use any off-the-shelf cord of this type.

2. Connect to your Mobile

TapNPass connects to Android and iOS mobiles via Bluetooth Low Energy (BLE) and Near Field Communication (NFC).

It is pre-configured for serial or Modbus communication with target systems using the **Tap Toolbox** app.

Note: TapNPass can also be reconfigured. See section 4.

Get Tap Toolbox

Go to **Google Play Store** or to the **App Store**. Download and install IoTize **Tap Toolbox**.



Note: For Android mobiles (with NFC) and iOS (iPhone XR/XS). Activate NFC and a data connexion, then hold the mobile to the NFC indicator on TapNPass. **Tap Toolbox** will be proposed automatically for installation.

Connect to mobile to TapNPass

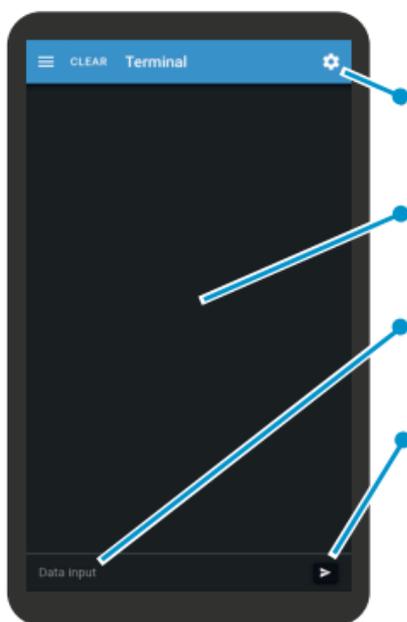
1. Activate the mobile's Bluetooth and data connections.
2. Press TapNPass's Power button to start its Bluetooth.
3. Launch **Tap Toolbox** and select **Scan** to find your TapNPass.
4. Select your TapNPass from the list of Bluetooth devices.

After you are connected, **Tap Toolbox** provides the utilities:

- **Terminal:** sends/receives characters via RS-232, RS-485 or USB interfaces.
- **Modbus:** drive one or more Modbus (slave) devices. Only Modbus-RTU is supported.

3a. Terminal utility

Selecting **Terminal** opens this utility for sending text commands and visualizing responses.



Settings: set input format and visualization parameters

Display: visualization area for commands and responses.

Data input: click to access keyboard and type text commands.

Send button

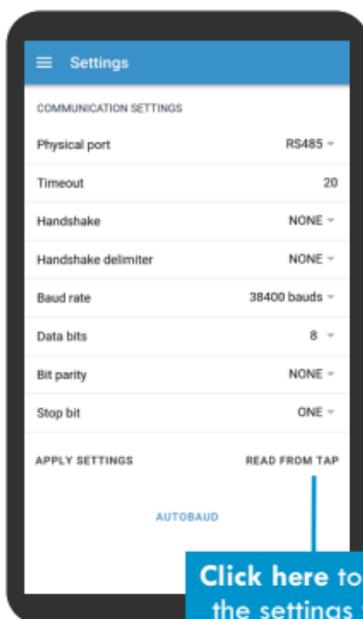
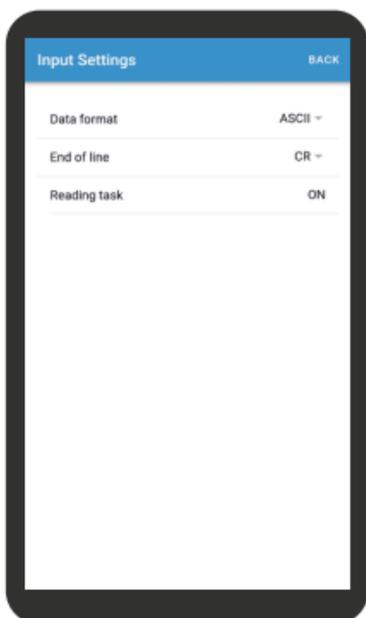
UART settings

Click the menu icon on the left, and select **Settings**.

These settings are common to **Terminal** and **Modbus** utilities.

Input Settings

Click  in the **Terminal** display. These settings are specific to the **Terminal** utility.



Click here to read the settings from TapNPass.

3b. Modbus utility

Selecting **Modbus** opens this utility for reading and writing Modbus (slave) registers.

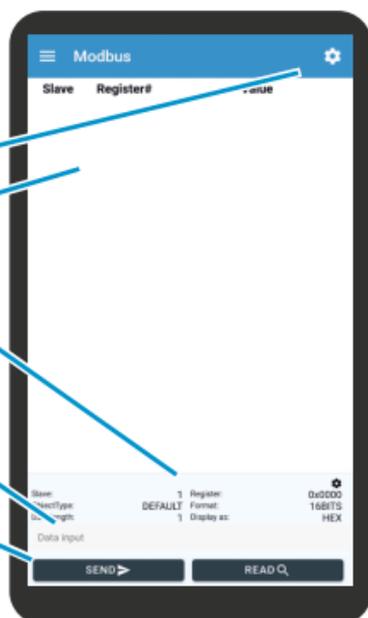
Settings: Set parameters that are specific to the Modbus communication with the target.

Registers: List of registers to monitor.

Register definition: Displays values of parameters of the selected register.

Input field: Enter the value to be written when using the write register command.

Command buttons: Execute the read or write commands.



Define a register to monitor

Click in the **Registers** area, then enter:

- **Slave id** (8-bit value)
- **Register address** (16-bit value)
- **Type** (default is 'holding register')
- **Format:** registers are 16-bit. They are truncated for 8-bit values. 32-bit values use 2 consecutive registers.
- **Length:** the number of consecutive registers to be accessed
- **Display mode** (HEX or DEC)

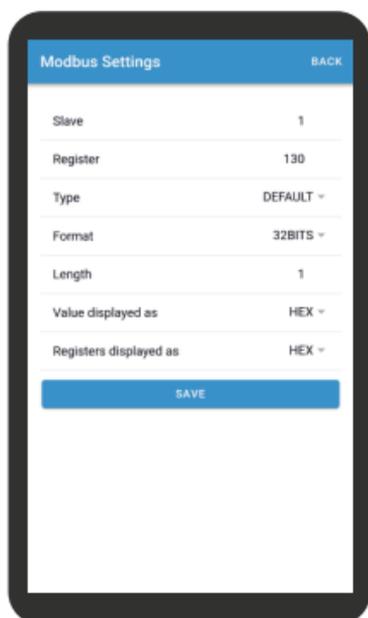
Click on **Save** and then **Read**.

The register is listed in the **Register** display and the values are in the **Register definition** field.

Save a register for later use

- Click on the register in the **Registers** area
- Swipe to the left
- Click on the **Save** button

This saves the register and related settings to a list of registers.



4. Advanced Configuration

Your TapNPass can be configured to:

- Require login for access to features or data
- Open an HMI app that you created for your systems or your user

The **IoTize Studio** configuration environment, detailed documentation, application notes and app development resources are available online at:

<http://docs.iotize.com/GettingStarted/TapNPass/>

5. Troubleshooting

App won't display any register values

If Tap Toolbox Modbus utility does not display any values from your Modbus slave, verify that:

- Connection cord is plugged into TapNPass and the target.
- Connection cord pin out is correct for your target. The problem may be caused by inverted D0 and D1 signals.

To comply with the Modbus specification, TapNPass uses an RJ45-RJ45 cord with a specific pin specification. Try this cord first.

If the provided cord does not work, you can also try an Ethernet cord with the standard pin out.

If neither cord works, you will have to verify the pin connections on the target and may have to adapt a cord to your target's pin out.

- The UART settings are appropriate for the target.
- The slave address is correct for the target.
- You have access to the Modbus features in the current TapNPass configuration. For a new TapNPass, the initial configuration allows access without login for Modbus and Terminal features. If the configuration has been changed, you will have to obtain the login from the configuration project or the user who configure the TapNPass.