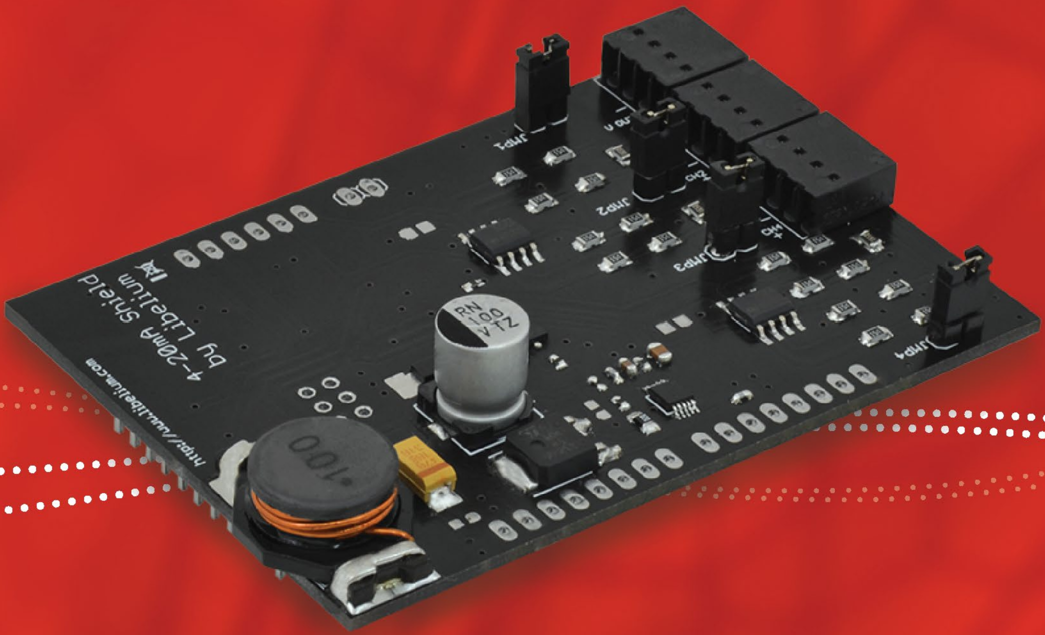
4-20 mA Current Loop

Sensor Board



Document version: v4.2 - 10/2015
© Libelium Comunicaciones Distribuidas S.L.

INDEX

1. Introduction

1.1. The standard

The 4-20 mA current loop is one of the most robust sensor signaling standard. Current loops are ideal for data transmission because of their inherent insensitivity to electrical noise. In a 4-20 mA current loop, all the signaling current flows through all components; the same current flows even if the wire terminations are less than perfect. All the components in the loop drop voltage due to the signaling current flowing through them. The signaling current is not affected by these voltage drops as long as the power supply voltage is greater than the sum of the voltage drops around the loop at the maximum signaling current of 20 mA.

Transmitting sensor information via a current loop is particularly useful when the information has to be sent to a remote location over long distances (500 meters, or more). The loop's operation is straightforward: a sensor's output voltage is first converted to a proportional current, with 4 mA normally representing the sensor's zero-level output, and 20 mA representing the sensor's full scale output. Then, a receiver at the remote end converts the 4-20 mA current back into a voltage which in turn can be further processed by a computer or display module.

This list includes some of the most common uses of the standard:

- Sensors and instruments
- Remote transducers
- Monitoring processes
- Data transmission in industrial ambients

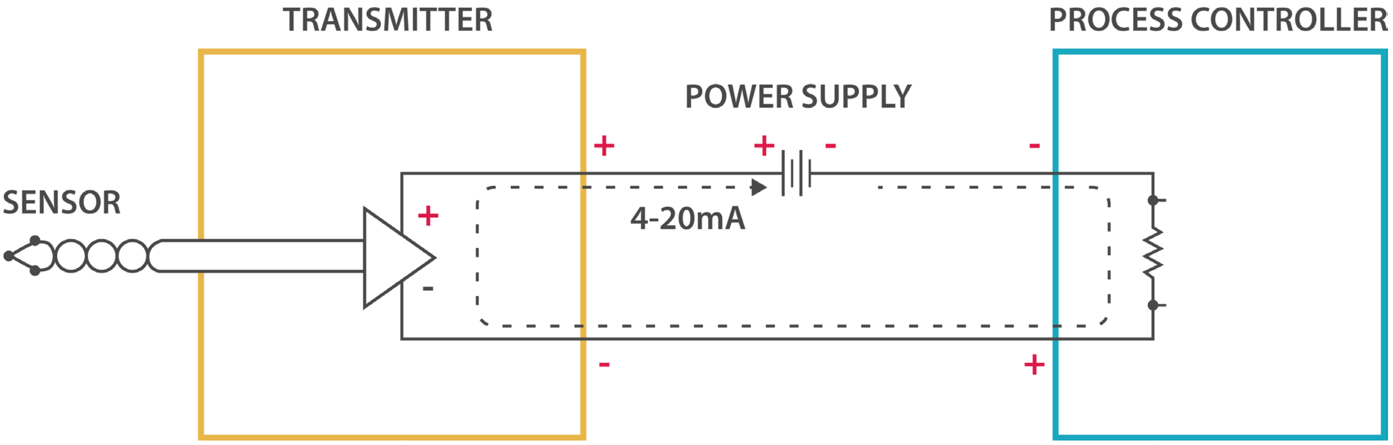


Figure : 4-20 mA standard typical connection diagram

1.2. Power Supply

The loop power-supply generally provides all operating power to the transmitter and receiver, and any other loop components that require a well-regulated DC voltage. In loop-powered applications, the power supply's internal elements also furnish a path for closing the series loop. +24 V is still the most widely used power supply voltage in 4-20 mA process monitoring applications. This is due to the fact that +24 V is also used to power many other instruments and electromechanical components commonly found in industrial environments. Lower supply voltages, such as +12 V, are also popular since they are used in computer based systems.

1.3. Transmitters categories

Depending on the source of current for the loop, devices may be classified as active (supplying power) or passive (relying on loop power).

1.3.1. Type 2 loop current

Type 2 transmitters are energized by the current loop, where the supply voltage is included in the receptor. The transmitter is floating and the ground is in the receptor.

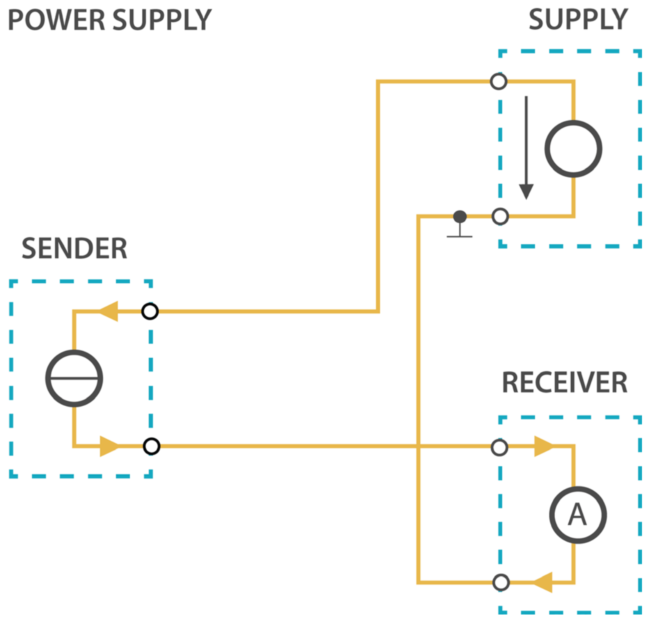


Figure : 4-20 mA type 2 connection

1.3.2. Type 3 loop current

Type 3 transmitters have 3 wires powered by the source voltage in them. In this case the transmitter is the power source for the current loop. The transmitter common is connected to the common of the receptor.

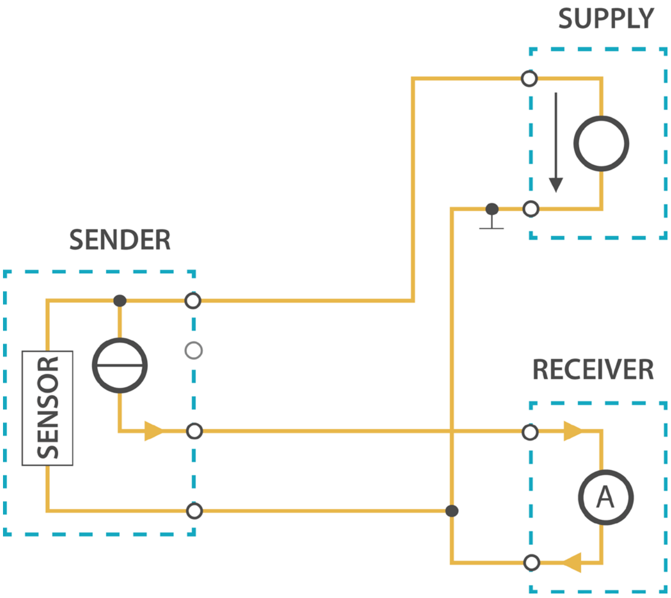


Figure : 4-20 mA type 3 connection

1.3.3. Type 4 loop current

Type 4 transmitters have 4 wires powered by the source voltage in them. The transmitter powers the current loop and the receptor acts a floating load.

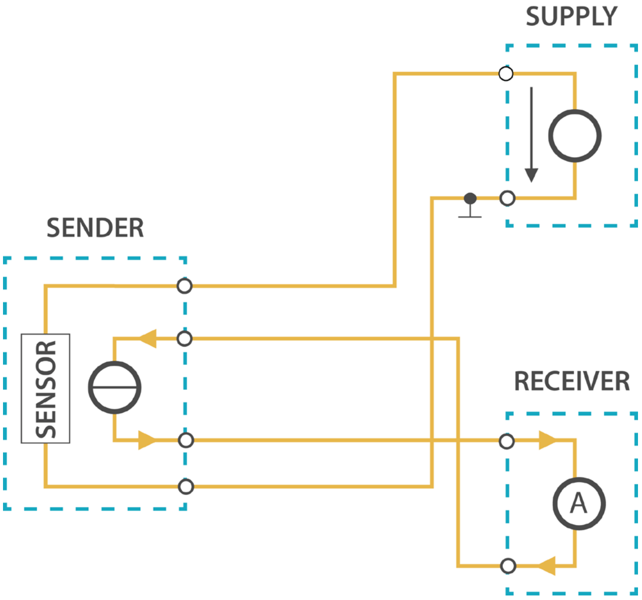


Figure : 4-20 mA type 4 connection

2. Hardware

2.1. Electrical characteristics

- **Board power voltages:** 4.2 V
- **Output voltage max:** +12 V
- **Output current max:** 180 mA
- **Converting resistance tolerance:** 0.1%
- **Temperature Range:** [0 ºC, 85 ºC]
- **Dimensions:** 73.5 x 51 x 1.3 mm

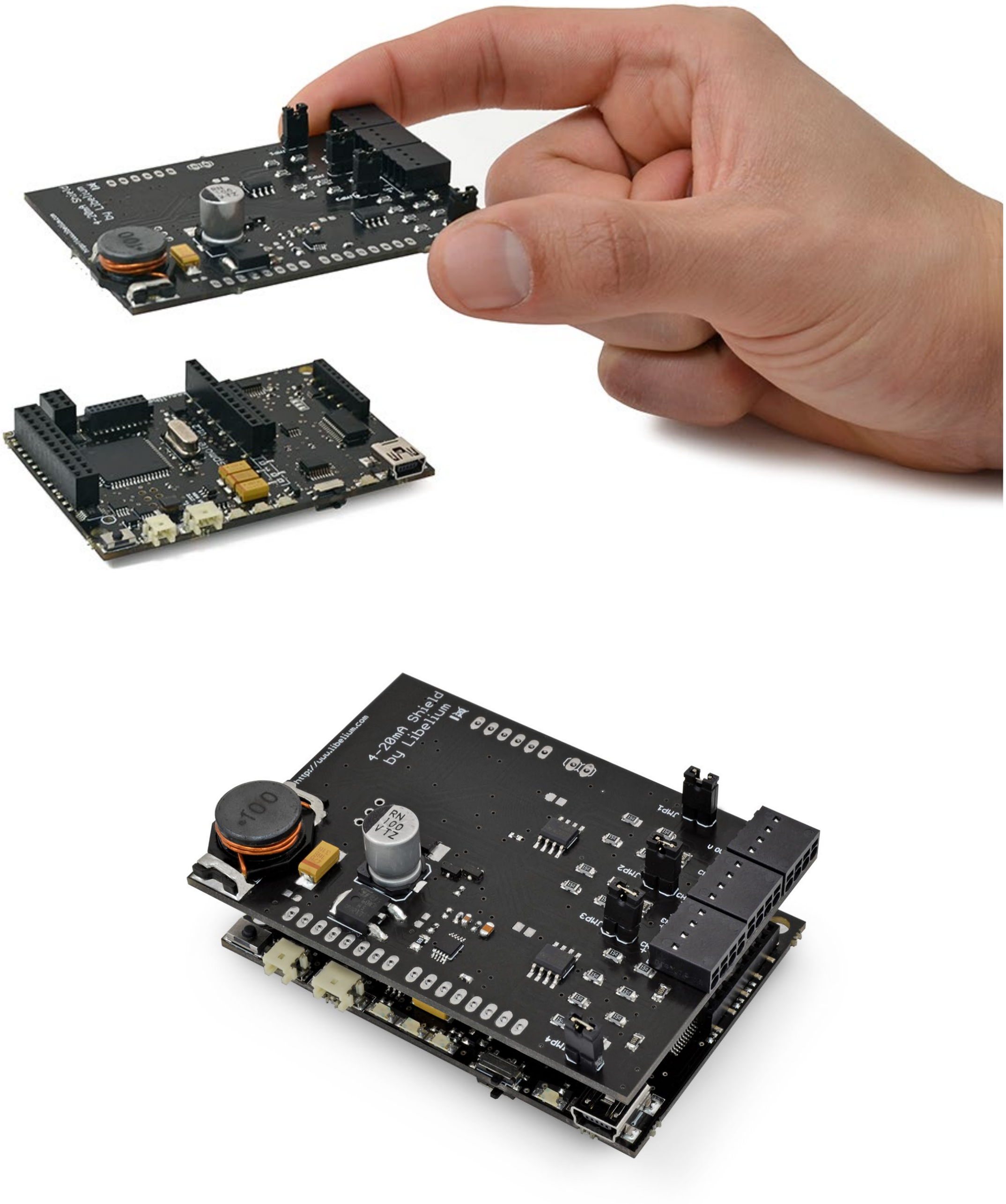


Figure : 4-20 mA Board connected over Waspmote

2.2. Connection diagram

The 4-20 mA Board for Waspmote has 4 input channels which allow the user to convert signals that come from 3rd party sensors or devices with this industrial standard. The 4-20 mA Board works with type 2, 3 and 4 current transmitters. Each channel can be configured for working with these types of connections. If the jumper is moved away, the receptor (the 4-20 Board) will be configured as float (type 4 connection).

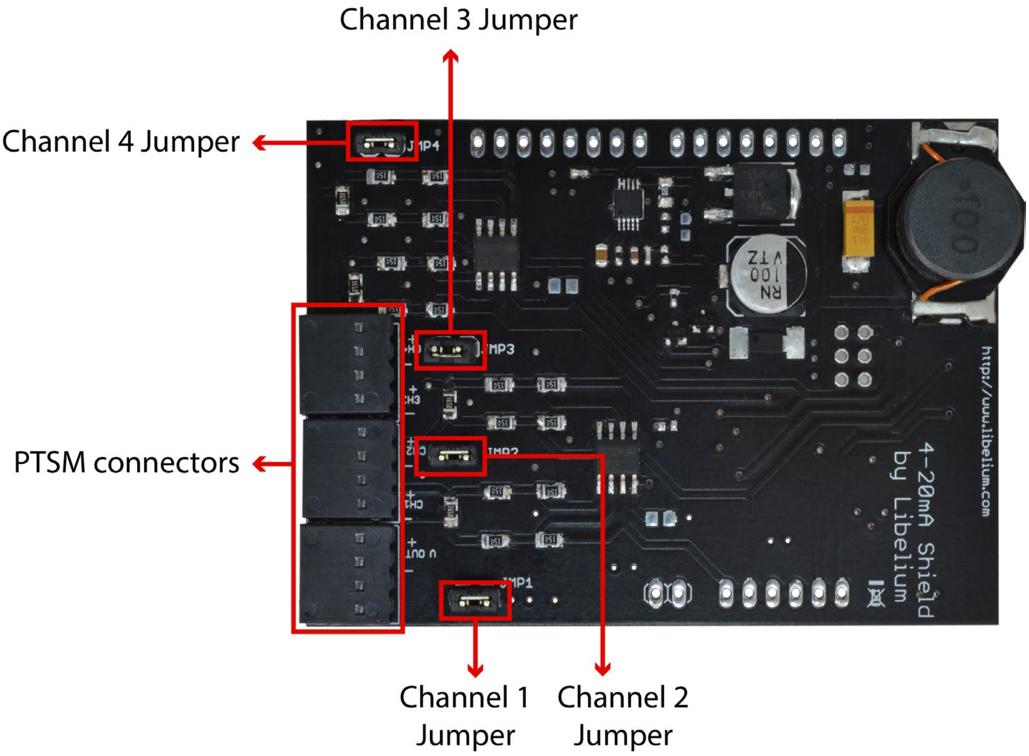


Figure : 4-20 mA Board diagram of connections

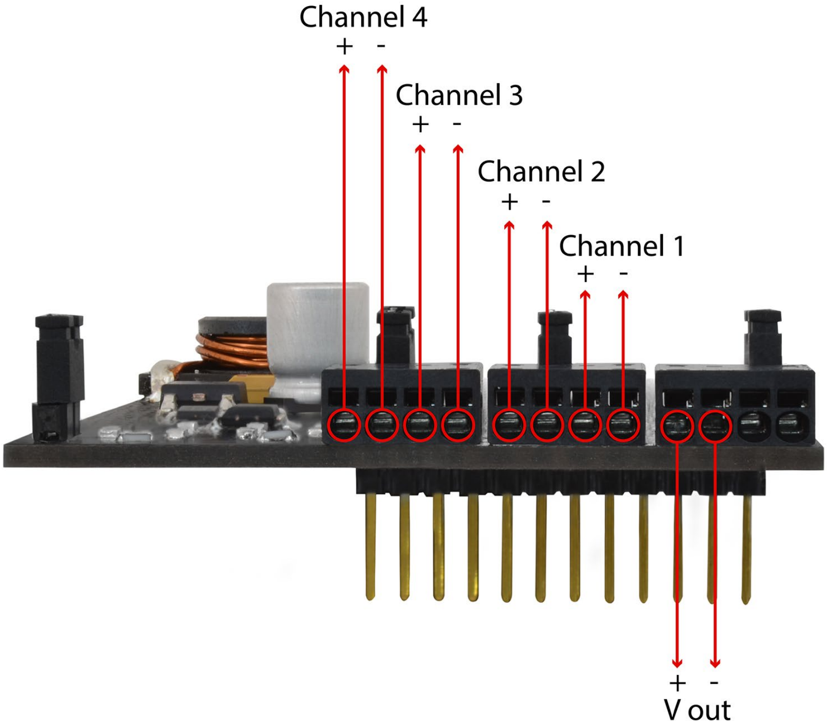


Figure : Channels from 1 to 4 and power supply connections

2.3. Consumption

The consumption of the 4-20 mA Board depends on several factors. When the battery is fully charged, the consumption of the board is lower due to the DC-DC converter. This DC-DC converter rises the voltage from the battery voltage level to 12 V, so if the battery level decreases, the energy needed to maintain the 12 V level is higher. The nominal consumption of the board is 8 mA without sensor. Also, if the 4-20 mA Board is used to power other sensors, the consumption will increase.

2.4. Connectors

For connecting the 4-20 mA sensors, the 4-20 mA Board includes PTSM connectors. These connectors (PTSM from Phoenix Contact) allow to assemble the wires of the probe simply by pressing them into it. To remove the wire, press the slot above the input pin and pull off the wire softly.

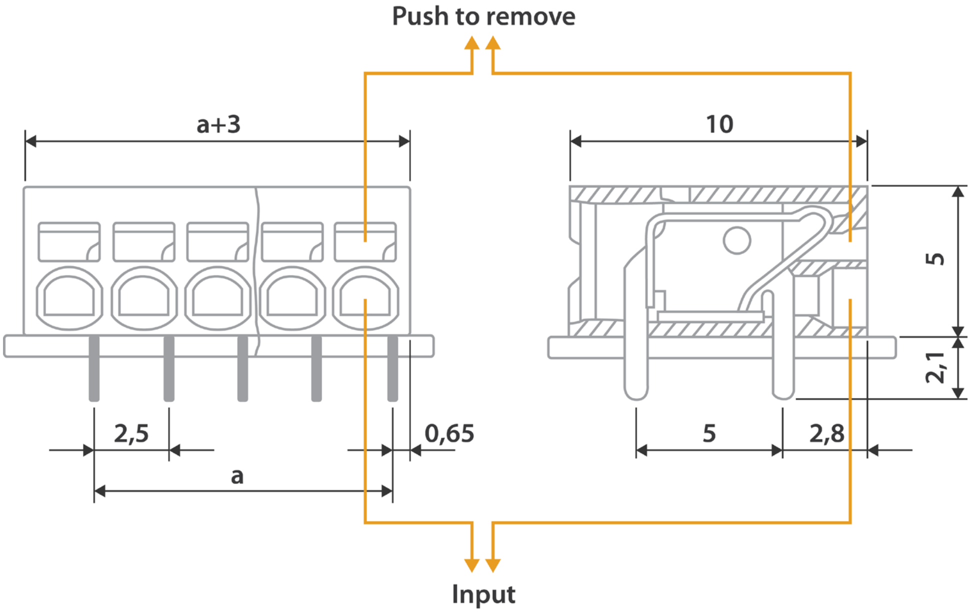


Figure : Diagram of a socket extracted from the Phoenix Contact data sheet

2.5. Powering sensors from the 4-20 mA Board

The 4-20 mA Board includes a 12 V output that can be used as power supply for sensors. The majority of 4-20 mA sensor works in the 9-24 V range. Before connecting a sensor to the 4-20 mA Board, you must be sure that 12 V fits the sensor's range of the power supply. If you are going to use all channels, the consumption of the board will increase. It is important to know that the 4-20 mA Board is not designed to power industrial devices such as PLC's and thermostats: the maximum current output is about 180 mA.

3. Applications

The 4-20 mA Sensor Board allows the user to interface the Waspmote ecosystem with 4-20 mA systems. Waspmote allows to perform two main applications:

1º- Add wireless connectivity to 4-20 mA devices

Waspmote can be configured to read the information coming from 4-20 mA sensors and send it wirelessly using any of the wireless modules available in the platform to a base station or to another node connected to another bus. The available wireless technologies are: WiFi, 3G, GPRS, 802.15.4, ZigBee, Sigfox, LoRa, Bluetooth, Bluetooth Low Energy, RF-868MHz, RF-900MHz.

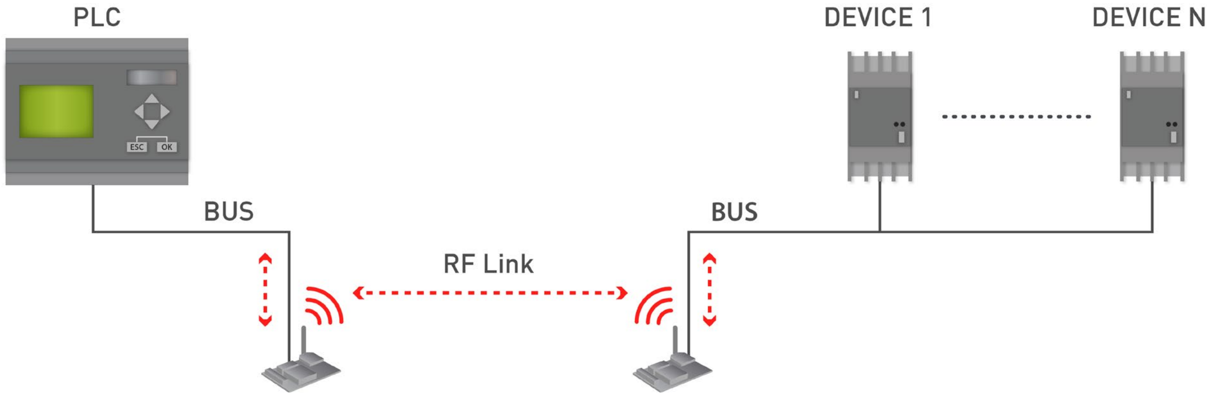


Figure : 4-20 mA wireless application

2º- Connect to the Cloud 4-20 mA devices

Waspmote can be configured to read the information coming from the 4-20 mA sensors and send it wirelessly directly to the Cloud using WiFi, 3G and GPRS radio interfaces.

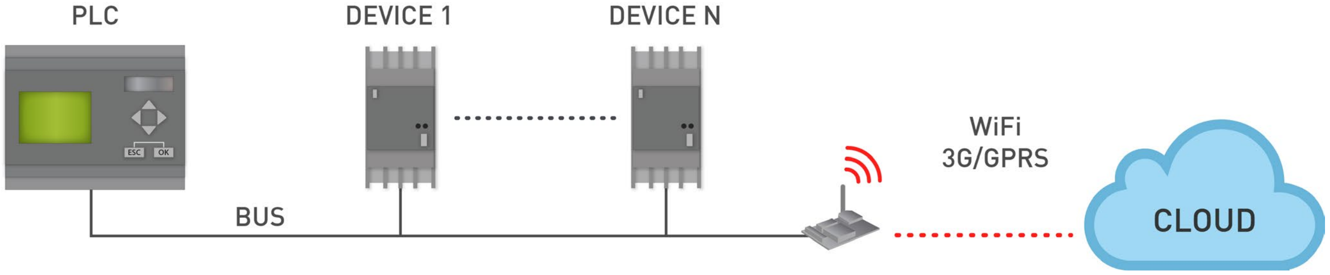


Figure : Cloud connection

4. Libelium's API

It is mandatory to include the 4-20 mA library when using this board. The following line must be introduced at the beginning of the code:

```
#include <currentLoop.h>
```

Waspmote's API 4-20 mA files:

- currentLoop.cpp
- currentLoop.h

API's functions

- Public functions:

`uint8_t ON(uint8_t powerSupply)`	Powers ON the 4-20 mA supply (5 V or 12 V)
`uint8_t OFF(uint8_t powerSupply)`	Powers OFF the 4-20 mA supply (5 V or 12 V)
`int readChannel(uint8_t channel)`	Gets the sensor value in integer format
`float readVoltage(uint8_t channel)`	Gets the sensor value as a voltage
`readCurrent(uint8_t channel, float offSet)`	Gets the sensor value as a current in mA and introduce an offSet correction factor
`float readCurrent(uint8_t channel)`	Gets the sensor value as a current in mA
`uint8_t isConnected(uint8_t channel)`	Checks if the current loop is well connected

Figure : Table of public functions

5. Library functions

5.1. Library constructor

To start using Waspmote 4-20 mA library, an object from class 'currentLoop' must be created. This object, called 'currentLoopBoard', is created inside Waspmote 4-20 mA library and it is public to all libraries. It is used through this guide to show how Waspmote 4-20 mA library works. When creating this constructor, all the variables are defined with an initial value by default.

5.2. Switching the board ON

The 4-20 mA Board includes a 12 V output that can be used to supply sensors, and can be controlled from the library functions, by a digital pin of Waspmote. The electronic measurement circuits use the 5 V power (so it is always mandatory to switch this option on), and is necessary to switch ON this power supply before getting data from the sensors.

On the other hand, it is only necessary to switch the 12 V on when we want to power 3rd party sensors.

Example of use:

```
{
 // Sets the 5 V switch ON
 currentLoopBoard.ON(SUPPLY5V);
 delay(100);

 // Sets the 12 V switch ON
 currentLoopBoard.ON(SUPPLY12V);
 delay(100);
}
```

See an example of use here:
http://www.libelium.com/development/waspmote/examples/4-20ma-01-current-loop-basic-example

5.3. Switching the board OFF

This function can be used to switch OFF the power supplies of the 4-20 mA Board. The 12 V and 5 V power supply must be switched off separately as shown in the next example.

Example of use:

```
{
 // Sets the 5 V switch OFF
 currentLoopBoard.OFF(SUPPLY5V);
 delay(100);

 // Sets the 12 V switch OFF
 currentLoopBoard.OFF(SUPPLY12V);
 delay(100);
}
```

5.4. Reading data

The 4-20 mA library includes the necessary functions to read data in several formats. The 4-20 mA standard sends the sensor information as a current, and this information can be transformed in voltage with a simple conversion function.

Example of use:

```
{
  // Get the sensor value in integer format (0-1023)
  int value = currentLoopBoard.readChannel(CHANNEL1);
  USB.print("Int value read from channel 1: ");
  USB.println(value);

  // Get the sensor value as a voltage
  float voltage = currentLoopBoard.readVoltage(CHANNEL1);
  USB.print("Voltage value read from channel 1: ");
  USB.print(voltage);
  USB.println("V");

  // Get the sensor value as a current in mA
  float current = currentLoopBoard.readCurrent(CHANNEL1);
  USB.print("Current value read from channel 1: ");
  USB.print(current);
  USB.println("mA");
}
```

Sometimes it is necessary to introduce a correction factor to correct offset deviations in the measurement process. The next function can be used for this:

```
{
  // Get the sensor value as a current in mA
  float current = currentLoopBoard.readCurrent(CHANNEL1, OFFSET);
  USB.print("Current value read from channel 1: ");
  USB.print(current);
  USB.println("mA");
}
```

See an example of use here:

http://www.libelium.com/development/waspmote/examples/4-20ma-01-current-loop-basic-example

5.5. Current loop state

One of the most important features of the 4-20 mA standard is the possibility of detecting a broken wire or failed instrument. The 4-20 mA library includes a function to detect the current state of the line.

Example of use:

```
{
  if (currentLoopBoard.isConnected(CHANNEL1))
  {
        // Get the sensor value as a current in mA
        current = currentLoopBoard.readCurrent(CHANNEL1);
        USB.print("Current value read from channel 1: ");
        USB.print(current, 3);
        USB.println("mA");
  }
  else
  {
        USB.println("The sensor is not connected...");
  }
}
```

See an example of use here:

http://www.libelium.com/development/waspmote/examples/4-20ma-02-current-loop-connection-state

6. Real application example

As a real application we are going to connect a generic 4-20 mA sensor to the 4-20 Sensor Board and power the sensor with the Board's 12 V output. This sensor is a type 2 transmitter. Type 2 transmitters are energized by the current loop, where the supply voltage is included in the receptor. The transmitter voltage is floating and the ground is in the receptor, so it is necessary to connect the corresponding jumper.

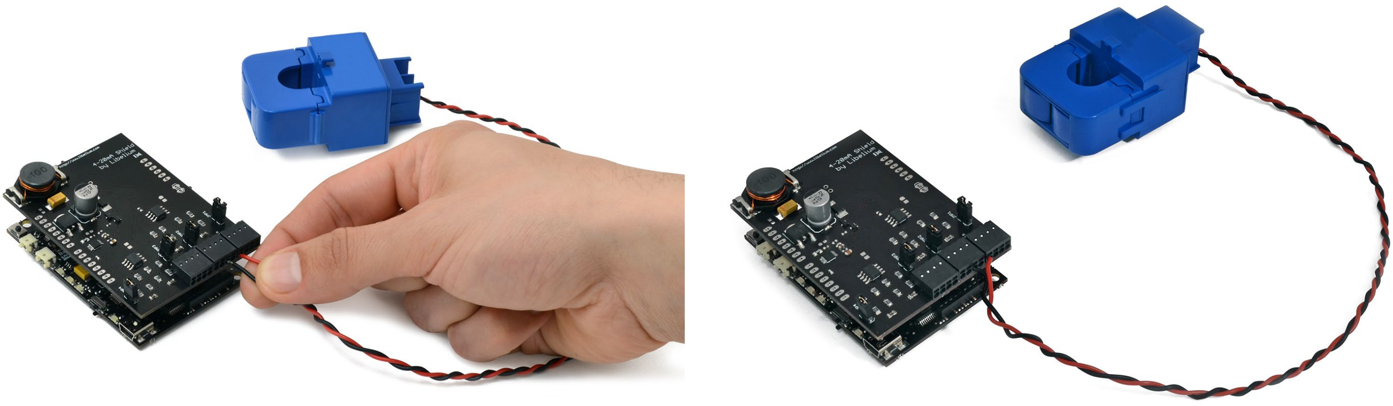


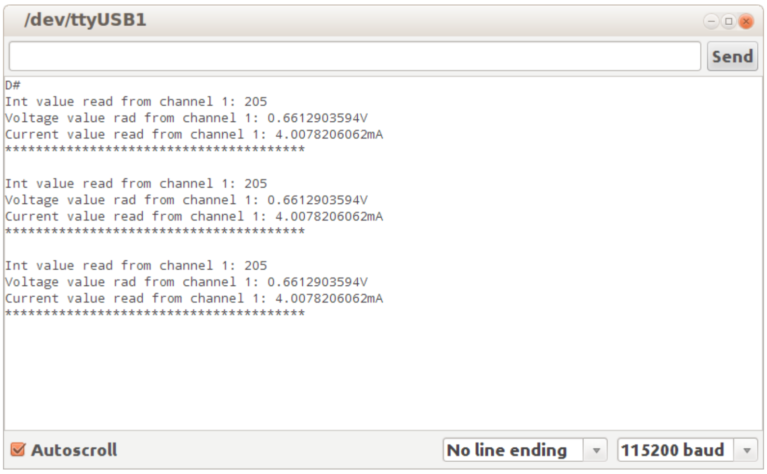
Figure : 4-20 mA sensor connected to the 4-20 mA Board

After connecting the sensor, you only have to upload one of the example codes included in the Waspmote IDE, for instance the example number one:

http://www.libelium.com/development/waspmote/examples/4-20ma-01-current-loop-basic-example

In the serial monitor you should see that the sensor is transmitting 4 mA. The 4 mA "bottom of span" signal allows the receiver to **detect a broken wire or failed instrument**. Remember that in normal operation, a 4-20 mA sensor delivers between 4 and 20 mA. Any current above or below means malfunction.

The constant-current feature of a current loop cancels out voltage drop errors due to long wiring runs (of course this would also be true if you selected different current values for zero and span [e.g., 5-30 mA]).



```
D#
Int value read from channel 1: 205
Voltage value rad from channel 1: 0.6612903594V
Current value read from channel 1: 4.0078206062mA
***************************************

Int value read from channel 1: 205
Voltage value rad from channel 1: 0.6612903594V
Current value read from channel 1: 4.0078206062mA
***************************************

Int value read from channel 1: 205
Voltage value rad from channel 1: 0.6612903594V
Current value read from channel 1: 4.0078206062mA
***************************************
```

Figure : Data reception from the 4-20 mA sensor

7. Code examples and extended information

For more information about the Waspmote hardware platform go to:
http://www.libelium.com/waspmote
http://www.libelium.com/development/waspmote

In the Waspmote Development section you can find complete examples:
http://www.libelium.com/development/waspmote/examples

Example:

```
/*
 *  ------ [4-20mA_01] Current Loop Basic Example --------
 *
 *  Explanation: This sketch shows how to use the most important
 *  features of the 4-20 mA Current Loop Board in Waspmote. This
 *  standard is used to transmit information of sensor over long
 *  distances. Waspmote uses analog inputs for reading the sensor
 *  values.
 *
 *  Copyright (C) 2014 Libelium Comunicaciones Distribuidas S.L.
 *  http://www.libelium.com
 *
 *  This program is free software: you can redistribute it and/or modify
 *  it under the terms of the GNU General Public License as published by
 *  the Free Software Foundation, either version 2 of the License, or
 *  (at your option) any later version.
 *
 *  This program is distributed in the hope that it will be useful,
 *  but WITHOUT ANY WARRANTY; without even the implied warranty of
 *  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 *  GNU General Public License for more details.
 *
 *  You should have received a copy of the GNU General Public License
 *  along with this program. If not, see <http://www.gnu.org/licenses/>.
 *
 *  Version:          0.1
 *  Design:           David Gascon
 *  Implementation:   Ahmad Saad
 */

//Include this library for using current loop functions
#include <currentLoop.h>

//Instantiate currentLoop object in channel 1
currentLoop sensor(CHANNEL1);

void setup()
{
  // Power on the USB for viewing data in the serial monitor
  USB.ON();
  delay(100);

  // Sets the 5 V switch ON
  PWR.setSensorPower(SENS_5V, SENS_ON);
  delay(100);
}

void loop()
```

```
{
  // Get the sensor value in int format (0-1023)
  int value = sensor.readChannel();
  USB.print(“Int value read from channel 1: “);
  USB.println(value);

  // Get the sensor value as a voltage
  float voltage = sensor.readVoltage();
  USB.print(“Voltage value rad from channel 1: “);
  USB.print(voltage);
  USB.println(“V”);

  // Get the sensor value as a curren in mA
  float current = sensor.readCurrent();
  USB.print(“Current value read from channel 1: “);
  USB.print(current);
  USB.println(“mA”);

  USB.println(“***************************************”);
  USB.print(“\n”);

  //Delay after reading
  delay(2500);
}
```

8. API changelog

Keep track of the software changes on this link:

www.libelium.com/development/waspmote/documentation/changelog/#4_20_mA

9. Documentation changelog

From v4.1 to v4.2

- References to the new Sigfox module

From v4.0 to v4.1

- References to the new LoRa module
- Created new chapter "API changelog"