User Manual

MC100 Terminal
MC100 Gateway
MC100 GPIO
MC100 SensT2
MC100 WMBUS

Version v1.0.1

MC Technologies GmbH, Kabelkamp 2, 30179 Hannover, Germany
Tel: +49511 67 69 99-0, Fax: +49511 67 69 99-150, Website & Shop: www.mc-technologies.net, Email: info@mc-technologies.net

This documentation applies to MC 100 SW version 1.0.4.

1 General

These instructions enable the safe and efficient handling of the product. The instructions are an integral part of the product and must be kept accessible at all times to installation, commissioning and operating personnel.

1.1 Warranty provisions

Unauthorized use, non-observance of this documentation, the use of insufficiently qualified personnel and unauthorized modifications exclude the manufacturer's liability for resulting damages. The manufacturer's warranty is void. The provisions of our terms and conditions of delivery and purchase (AGB) apply. These can be found on our website (https://www.mc-technologies.net/en/terms-and-conditions.php)

2 Safety

The Safety section provides an overview of the safety instructions to be observed when operating the product.
The product has been constructed in compliance with the currently valid rules of technology and is safe to operate. It has been tested and has left the factory in perfect condition in terms of safety. In order to maintain this condition over the operating time, the information in the applicable publications and certificates must be observed and followed.
The basic safety instructions must be strictly maintained when operating the product. In addition to the basic safety instructions, the individual sections of the documentation contain descriptions of procedures and operating instructions with specific safety information.
Furthermore, the local guidelines for the prevention of accidents and general safety regulations for the area of application of the device should be applied.
Only the consideration of all safety instructions enables the optimum protection of personnel and the environment against hazards as well as the safe and trouble-free operation of the product.

2.1 Technical limiting values

The product is exclusively intended for use within the technical limits specified in the data sheets.
The following limit values must be observed:

- The surrounding temperature limits must not be exceeded or dropped below.
- The supply voltage range must not be violated under or above.
- The maximum air humidity must not be exceeded and condensation must be avoided.
- The maximum switching voltage and the maximum switching current load must not be exceeded.
- The maximum input voltage and the maximum input current must not be exceeded.

2.2 Obligations of the operator

The operator must always observe the national regulations applicable in his country regarding the operation, functional testing, repair and maintenance of electronic devices.

2.3 Qualification of personnel

Installation, commissioning and maintenance of the product may only be carried out by trained specialist personnel authorised by the system operator. The qualified personnel must have read and understood this documentation and follow the instructions.
The electrical installation and commissioning of the product may only be carried out by a person who, due to his or her specialist training, knowledge and experience as well as knowledge of the relevant standards and regulations, is able to carry out work on electrical systems and independently detect and avoid possible hazards.

2.4 Guidelines for transport and storage

The following instructions must be observed:

- Do not expose the product to moisture or other potentially harmful environmental conditions (radiation, gases, etc.) during transport and storage. Pack the product properly.
- Pack the product so that it is protected from shocks during transport and storage, e.g. by using air-cushioned packaging.
- Before installing the product, check it for possible damage that may have been caused by improper transport or storage. Damage in transit must be noted on the shipping documents. All claims for damages must be made immediately and before installation to the carrier / company responsible for storage.

2.5 Labels on the product

The type plate of the product is located as a sticker on one surface of the product. It may contain the following markings, among others, which are explained in more detail here.

	**Dispose of old equipment in an environmentally responsible manner** This symbol indicates that old devices must be disposed of separately from residual waste at suitable collection points. See also the disposal section in this manual.
	**CE-label** By affixing the CE marking, the manufacturer confirms that the product complies with the product-specific applicable European regulations.

2.6 Environmental protection

Dispose the product and its packaging in accordance with the relevant environmental protection regulations. Refer to the disposal section of this manual for instructions on how to dispose of the product. Separate the packaging components made of cardboard and paper as well as plastic and recycle them using the appropriate collection systems.

2.7 Basic safety requirement

	**Electrostatic discharges can damage the product!** **Damage the product.** Observe the general precautions for handling electrostatically sensitive components.
	**Moisture and liquids from the surrounding area can get inside the product!** **Risk of fire and damage to the product.** The product must not be used in wet or humid environments or in the immediate vicinity of water. Install the product in a dry place protected from splashing water. Turn off the power before performing any work on a device that has been exposed to moisture.
	**Short circuits and damage due to improper repairs and modifications and opening of maintenance areas!** **Risk of fire and damage to the product.** It is not permitted to open the product for repair work or modifications beyond the removal and insertion of the plug-in cards provided for this purpose.
	**Overvoltage and voltage peaks from the power supply system!** **Fire hazard and damage to the device due to overvoltage.** Install a suitable surge protection.
	**Distance between antennas and persons!** **Too small distance between mobile phone antennas and people can affect their health.** Please note that the mobile phone antenna must be at least 20 cm away from persons during operation.

3 Use of open-source software

3.1 General

This product includes, among other things, so-called open-source software, which was produced by third parties and published for free use by everyone. The open-source software is subject to special open-source software licenses and the copyright of third parties. Every customer can use the open-source software according to the license terms of the respective manufacturer. The rights of the customer to use the open-source software beyond the purpose of our product are regulated in detail by the respective open source software license. The customer may freely use the open-source software, as provided for in the respective valid license, beyond the purpose of the open-source software in our product. In the case that there is a contradiction between our license terms for our product and the respective open-source software license, the respective open-source software license takes priority over our license terms as far as the respective open-source software is affected.
The use of the open-source software used is free of charge. We do not charge any user fees or comparable fees for the use of the open-source software contained in our product. The use of the open-source software by the customer in our product is not part of the profit we make with the contractual remuneration.

The provided list shows all open source software programs that are included in our product. The most important open-source software licenses are listed in the licenses section at the end of this publication.
As far as programs contained in our product are under the GNU General Public License (GPL), GNU Lesser General Public License (LGPL), the Clarified Artistic License or any other open-source software license, which stipulates that the source code must be made available, we will send the source code at any time on request. Our offer to send the source code on demand automatically ends 3 years after delivery of our product to the customer. Requests should be sent to the following address, if possible stating the serial number of our product:

MC Technologies GmbH
Kabelkamp 2
30179 Hannover

3.2 Liability regulations

We do not assume any warranty or liability if the open-source software programs contained in our product are used by the customer in a way that no longer corresponds to the purpose of the contract on which the purchase of our product is based. This applies in particular to any use of the open source software programs outside of our product. For the use of the open source software beyond the purpose of the contract, the warranty and liability provisions provided for in the applicable open source software license for the corresponding open source software, as listed below, shall apply. In particular, we shall not be liable if the open source software in our product or the entire software configuration in our product is changed. The warranty given in the contract on which the purchase of our product is based shall only apply to the unchanged open source software and the unchanged software configuration in our product.

3.3 Used open-source software

Please contact our support department (support@mc-technologies.net) for a list of open source software used in this product. Alternatively, a list of the open source software used can be found in the MC100 web interface under System -> Software -> Installed.

4 Device variants

This manual describes different variants of the MC100 terminal / gateway from MC Technologies GmbH. The devices are collectively referred to as MC100 in the manual. The MC100 are:

- 4G LTE MC100 Terminal
- 4G LTE MC100 Gateway
- 4G LTE MC100 GPIO Terminal
- 4G LTE MC100 GPIO Gateway
- 4G LTE MC100 SensT2 Terminal
- 4G LTE MC100 SensT2 Gateway
- 4G LTE MC100 WMBUS Gateway

If the MC100 differ, this is mentioned separately in the corresponding sections.

5 Contents

The package includes the accessories listed below. Please check that all accessories listed are included in your box. If a part is missing or damaged, please contact MC Technologies directly.

- MC 100 Terminal / Gateway
- Quick Installation Guide
- Safety Instructions
- Screw terminals

Optional accessories are not included. Among others, the following parts are available at MC Technologies:

- Mobile phone antennas
- Antenna extensions and accessories
- DIN Rail
- Ethernet cable
- GSM (LTE)-Antenna

6 Technical specifications

The following specifications apply to all versions of the MC100. If these variants differ from one another, the different values are indicated separately.

6.1 Technical Data

6.1.1 Physical properties

Physical characteristics	Value
Supply Voltage	8 V ... 30 V DC
Dimensions (W x H x D)	120 x 75 x 35mm
Operation temperature	-20°C to +70°C
Housing	Plastic
Mounting	Wall mounting (DIN rail adapter optional)

The extended temperature range allows temporary operation at increased temperatures. This may result in functional restrictions (especially during data transmission). This serves to protect the internal electronics.

6.1.2 Technological mobile network features

Technological feature	Description
Supporting networks	LTE Cat1 (4G): Band 1,3,5,7,8 and 20 HSDPA/HSUPA/HSPA+ (3G): Band 1,5,8, all bands with diversity GSM/GPRS/EDGE (2G): Band 3,8
Transmission rates LTE	Up to 50 Mbps uplink and 10 Mbps downlink
Transmission rates HSPA+	Up to 5.76 Mbps uplink and 21.6 Mbps downlink
Transmission rates EDGE	Up to 236.8 kbps uplink and 236.8 kbps downlink
Antenna connections	SMA (female)

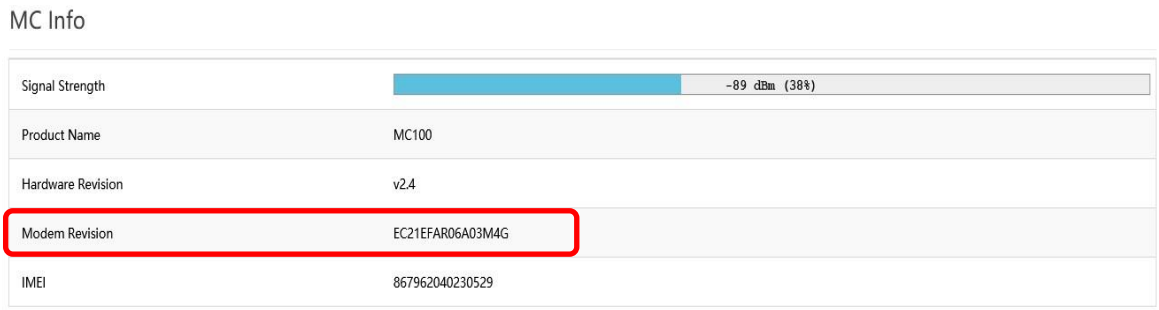
LED 4 shows the statues of the network connection:

LED 4:	**Long-time Off: Searching for connection.** **Long-time ON: Connected.** **Flashing quickly: Packet Data transfer.**

7 Check Modem Version

To check the firmware version of your MC100:

1. Login to Luci web interface using your device IP address (default: 192.168.2.1)
2. On the main page, scroll down to MC info Tab



3. Under Modem Revision, you can read the firmware version of your MC100 Model Chip.
    a. As of release 1.0.4 the latest firmware for the model chip is AR06A03.

8 Ports and display

The following figures show a maximum equipped version of the MC100. Depending on the variant, your MC100 may not have all connections, display or control elements.

Port	Description
POW	Power supply
GPIO	I/O-Interfaces
RS485	Serial RS485 interface (no galvanic isolation)
CAN	CAN-Port

Port	Description
RS232	Serial RS232-Interface
USB	USB-Port (USB 2.0 Host)
ETH	Ethernet-Port (RJ45, 10/100 BT)

Port	Description
A1	LTE Antenna - SMA socket (main connection)
A2	LTE Diversity Antenna - SMA socket (MIMO)
GPS	GPS antenna – SMA socket

LED	Description
LED 1	Green, Power supply
LED 2	Green, Configurable.
LED 3	Red, Configurable.
LED 4	Long-time Off: Searching for connection. Long-time ON: Connected. Flashing quickly: Packet Data transfer.
LED 5	Green, Reserved for GPS.

Port	Description
SIM	Slot for SIM card

Port	Description
Isolated Inputs	8 digital inputs
Isolated Outputs	8 digital outputs (solid state relay)
0 – 10V	2 analog inputs 0 to 10V
0 – 20mA	2 analog inputs 0 to 20mA
PWM	1 PWM (Pulse With Modulation) outputs

Port	Description
Isolated RS485	1 serial RS485 interface (with galvanic isolation)
Isolated Inputs	8 digital inputs
Isolated Outputs	2 digital outputs (solid state relay)
PT100/PT1000	4 PT100/PT1000 inputs
4 – 20mA Outputs	2 analog 4 - 20mA outputs
0 – 10V	2 analog inputs 0 to 10V
4 – 20mA LS	2 analog inputs 4 to 20mA Loop Current
1-Wire	1 one-wire or single-wire bus input

9 Installation

The MC100 Terminal/Gateway provides easy access to the Internet via the cellular network. For its installation, a SIM card from your mobile network operator that has been activated for parcel services (GPRS, UMTS, HSPA or LTE) is required.

Please follow the instructions step by step and configure the settings for the SIM card. An incorrect PIN can lead to the SIM card being blocked.

9.1 Antenna

Mount the supplied antenna on connector "A1" of the MC100. Check whether there is sufficient local network coverage from your mobile phone provider.
A2 is for MIMO usage, also see 9.1 for further information

9.2 Inserting the SIM card

Insert your SIM card as shown in the figure below until it locks into place. Place your finger on the SIM card latch (on the left of the SIM card) and gently slide the latch (Check from the device) to the right until it clicks into place.

9.3 Power supply

The MC100 is operated with a supply voltage of 10 - 30V DC. Please connect an external DC power supply with an output voltage of 10 - 30V DC to the power supply socket "POW" (e.g. 12V > 1.2A or 24V > 600mA).

Please connect the power supply unit to the power supply or switch on the power supply. A green "Power" LED will light up (see transparent bottom shell on the side of the antenna connectors).
After a short initialization phase (approx. 40 seconds) the device is ready for operation.

9.4 Configuration via web interface

The MC100 can be configured to set mobile phone parameters (SIM pin, APN, etc.) via an integrated web interface. To do this, connect the Ethernet interface of your PC and the Ethernet interfaces of the MC100 (ETH) with a network cable.

- The PC must obtain its IP address automatically.
- Open a browser on the PC. Firefox browser is preferred. Using Chrome browser Luci interface load slowly and often react only after repeated "refresh".
- The MC100 can be reached via web interface using the IP address **192.168.2.1**.
- For authentication, a user name and password must be entered. The default user name here is "root". During the first installation you can click directly on the login button without entering a password.

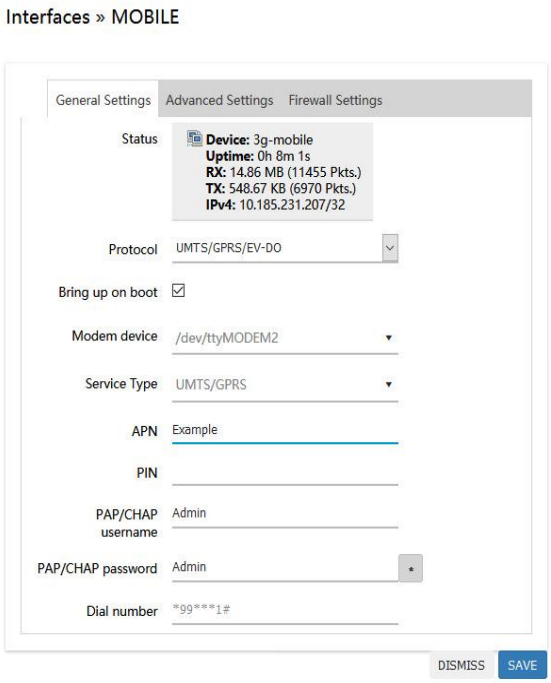
For security reasons, click on "Go to Password Configuration" in the yellow window and change/set the "Password" here.

9.5 Setting up the SIM and establishing a connection

Click on "Network" in the left menu and then on "Interfaces".
Click on the "EDIT" button at the "MOBILE" interface.

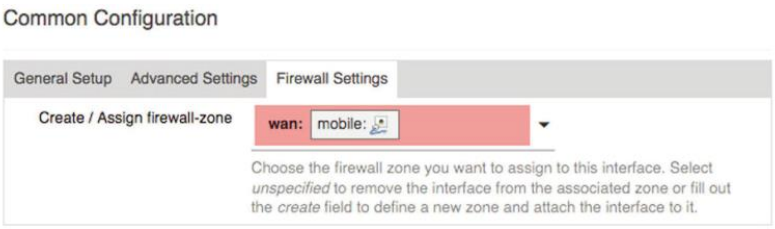
Setting APN:

1. Change Protocol to: ”UMTS/GPRS/EV-DO.” (Supports LTE-connection)
2. Click on “SWITCH PROTOCOL.”
3. Change Modem device to “/dev/ttyUSB2”
4. Enter the access data, APN, user and password for data transmission in your mobile network.
5. Enter the PIN number of your SIM card. Leave the entry field empty for a SIM card without PIN.



You can obtain the data from your mobile phone provider. Many providers do not require additional entries for user and password.

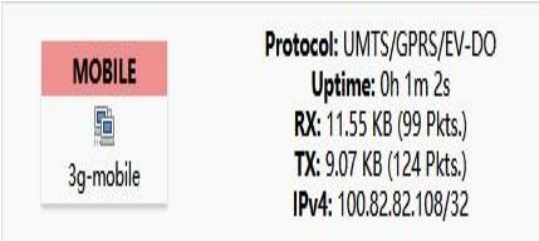
On the tab "Firewall Settings" select the setting "**wan**: mobile" under "Create / Assign firewall-zone.



Confirm your entries by clicking on the "SAVE & APPLY" button.

Please disconnect the MC100 briefly from the power supply.

After a startup phase of several minutes, log in to the web interface again. Navigate to the menu item "Network" and "Interfaces". In the **MOBILE** area, the transmission values in MB for RX and TX should now appear.



The quality of the connection can be in Luci under Status -> Overview -> MCInfo

RSRP		
> -70 dBm to -79 dBm	Very good signal	Telecommunication and Internet operate without any disturbance
-80 dBm to -89 dBm	Good, solid signal	Telecommunication and Internet operate with slight disturbance
-90 dBm to -100 dBm	Medium signal	Telecommunication and Internet operate with slight disturbance
-101 dBm to -110 dBm	Poor signal	Telecommunications and Internet operate with significant disruption
< -110 dBm	Slow or no signal	No connection or disconnected

9.6 Online-Test

Navigate to the menu item "Network" and "Diagnostics". Click on the "PING" button under "Network Utilities". The selected server on the Internet should answer with PING times.

9.7 Internet access

After successfully logging into the mobile network, you can now connect to the Internet using your browser. To do this, open the browser on your PC and enter a URL known to you in the address field (e.g. www.mc-technologies.net).

9.8 Access via SSH

To access the Linux system of the MC100, you must connect to the same IP address (default: 192.168.2.1) as the web interface.
The following access data are required:

User: `root`
Password: In delivery state no password is necessary here

To secure your MC100 change the password by entering "passwd" under SSH.

9.9 Command Line Interface

Open the Command Prompt to login your device:

1. Type: ssh root@192.168.2.1
2. Type the password.

Command	Use to
mcinfo info	Print general information about the modem.
mcinfo mobile	Print information about mobil communication status.
mcinfo gpio	Print information about external GPIO module pins.

```
root@MC100:~# mcinfo
Usage: mcinfo [options] [command[ command]]

Options:
  -h            Show this help message and exit.
  -v            Print verbose debug information to error
  -V            Show version information and exit.
  -d DEVICE     Set the tty device (default: /dev/ttyUSB3)
  -c COMMAND    Send COMMAND to modem.
  -t TIME       define the timeout in deciseconds
                    default: 1

Command:
info   Print general information about the modem.
mobile Print information about mobil communication status
gpio   Print information about external GPIO module pins
```

9.10 Change IP-address

Change using Command Prompt

Open the Command Prompt to login your device:

1. Type: ssh root@192.168.2.1
2. Type the password.
3. Open the configuration file using command: vim /etc/config/network

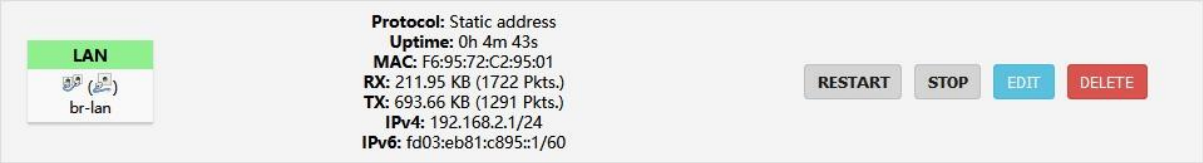
```
config interface 'lan'
        option type 'bridge'
        option ifname 'eth0'
        option proto 'static'
        option netmask '255.255.255.0'
        option ip6assign '60'
        option ipaddr '192.168.2.1'
```

4. Press “i” to enter the input mode, and change the line "option ipaddr" in the interface "lan" to the corresponding address.
5. If the MC100 is to go online via a router, the line option gateway 'aaa.bbb.ccc.ddd' can be added (with the IP address of the router).
6. Then press Escape to save the file, and then type :wq and press Enter.
7. Finally, type service network reload and press Enter.

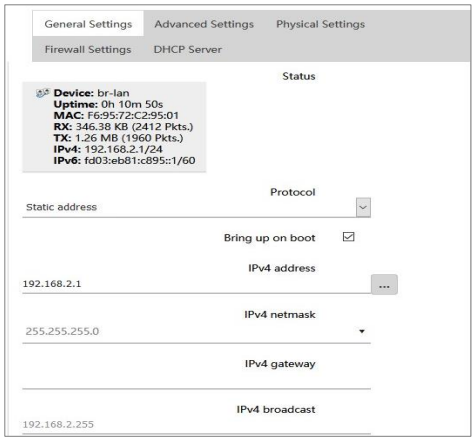
Restart the network services with the new settings. After a few seconds the MC100 has the new IP address. If necessary, the network connection at the host PC must be briefly disconnected and reconnected so that the PC can obtain a new IP.

Change using LuCi Web Interface

1. Using LuCi Web interface, login to your MC100.
2. Go to Network →Interfaces.



3. Under LAN Click on “EDIT”.
4. Under “General Settings”, change “IPv4 address”.



5. Click on “SAVE”, and then click on “SAVE & APPLY”

9.11 Node-RED Web Interface

Node-RED is a programming tool that allows hardware devices, APIs and online services to be wired together. It is builds on Node.js, fully utilizing the advantages of its event-driven, non-blocking model.
It offers a browser-based editor that enables to easily wire flows by using the wide range of nodes in the palette.

If Node-RED is installed, you can reach the Node-RED web interface with the same IP address as the standard web interface and the specification of the port (e.g. 1880)
Example with default address: http://192.168.2.1:1880
(Note: Using https:// will not work.)

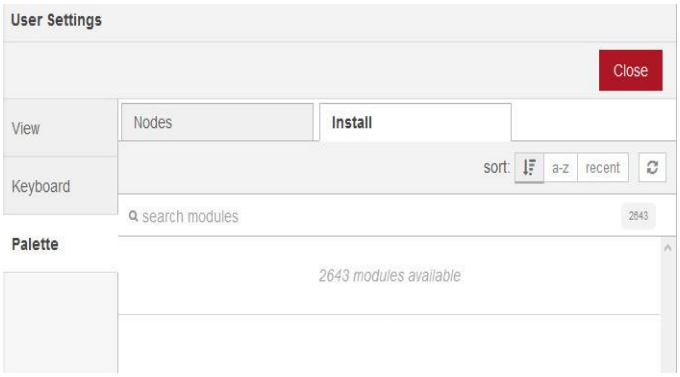
9.11.1 Enable the Node-Red installed in MC100:

Click on "System" in the left menu and then on "Startup".
Search for Initscript: “node-red” and Click the "Enable/Disable" button.

9.11.2 Install extra modules:

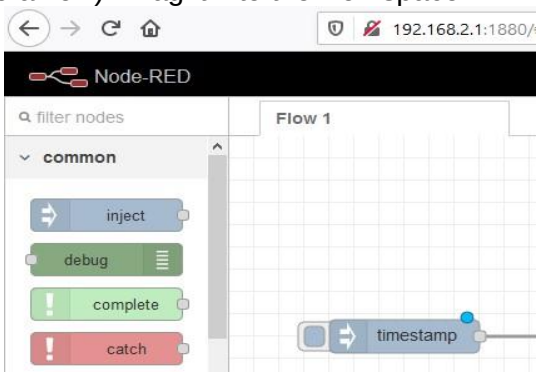
To install modules for Node-RED:

1. Open Node-RED web interface.
2. On the left upper corner, click on the 3 dash button. 
3. Click on “Manage Palette”.
4. In the “Manage Palette” interface, Click on install.
5. Type the modules name in “Search modules”.



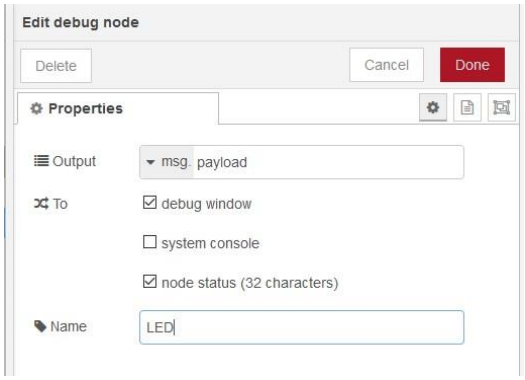
9.11.3 Add a node

Select the node you want to add from the nodes list (i.e. Inject node, allow injecting messages into a flow). Drag it into the workspace.



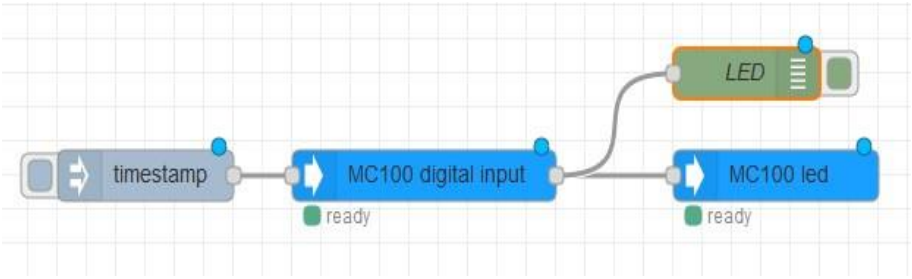
9.11.4 Add a debug node

The debug node displays the payload of the message or the entire message object. It can be renamed from its setting by double clicking on it.



9.11.5 Wire the nodes

After adding all the wanted nodes, wire them together by dragging between the outputs of one to the input of the other.



9.11.6 Deploy

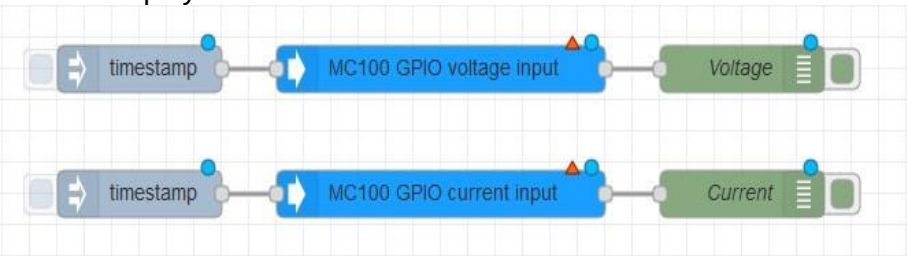
Now click on the Deploy button to deploy the flow to the server.

10 Readout analog and digital IO's via Node-RED website interface

10.1 MC100 GPIO

10.1.1 Analog Inputs (mA/V)

1. Add the inject node
2. Add voltage input/current input node
3. Add Debug node.
4. Wire and Deploy.

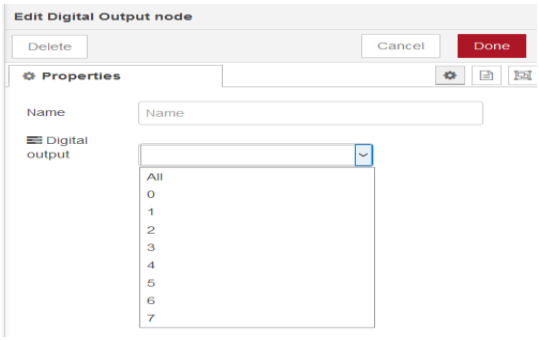


10.1.2 Digital Inputs

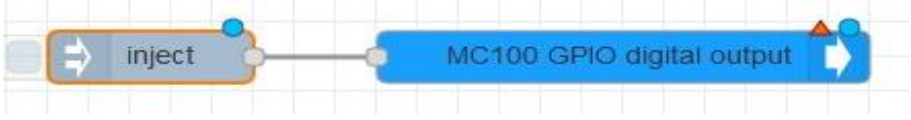
1. Add the inject node.
2. Add the digital inputs node.
3. Double click on MC100 GPIO digital input node to open the settings menu and choose a Digital input.
4. Add Debug node (change name if wanted).
5. Wire and Deploy.

10.1.3 Digital Outputs

1. Add the Inject node.
2. Add the Digital output node.
3. Double click on MC100 GPIO digital output node to open the settings menu and choose a Digital input.

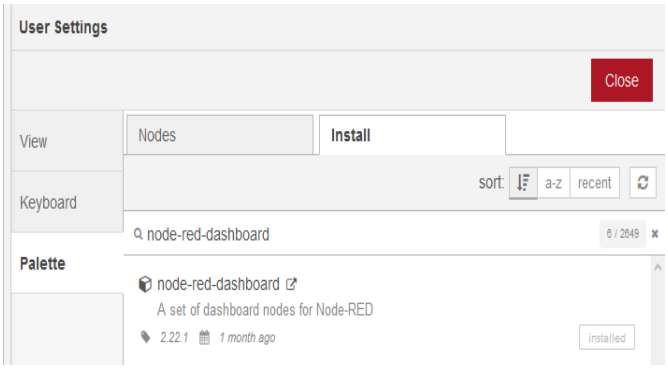


4. Add Debug node (change name if wanted).
5. Wire and Deploy.

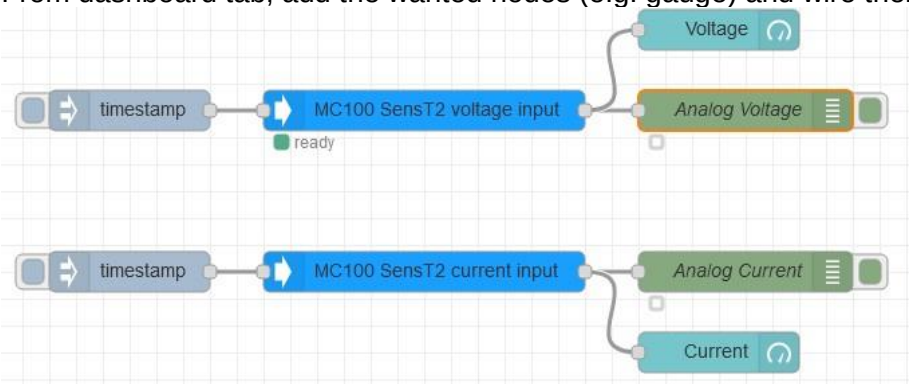


10.1.4 Dashboard

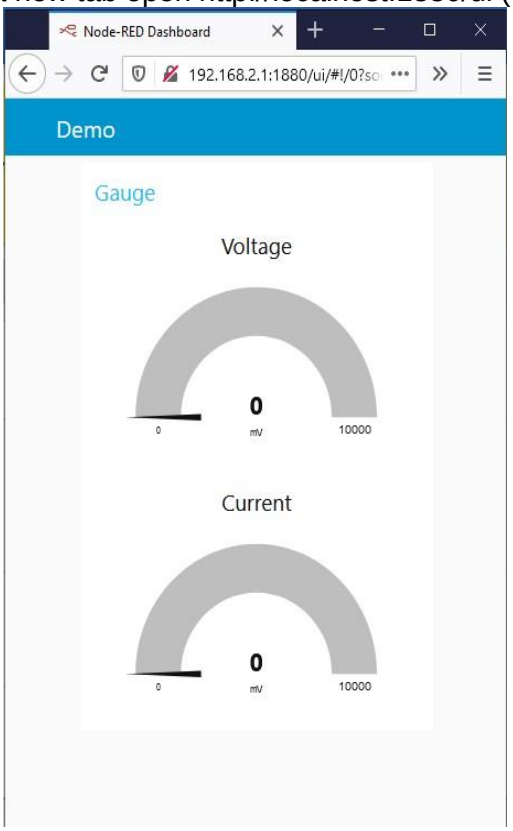
1. Use Menu - Manage palette to search for “node-red-dashboard” and click on install.



2. Restart Node-RED, dashboard tab should appear in the right side panel.
3. From dashboard tab, add the wanted nodes (e.g. gauge) and wire them.



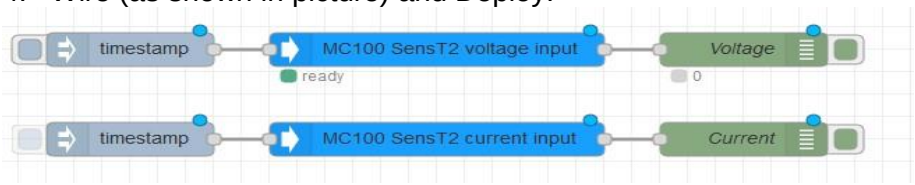
4. Double click on the nodes to change their Properties as wanted.
5. In a new tab open http://localhost:1880/ui (e.i. http://192.168.2.1:1880/ui)



10.2 MC100 SensT2

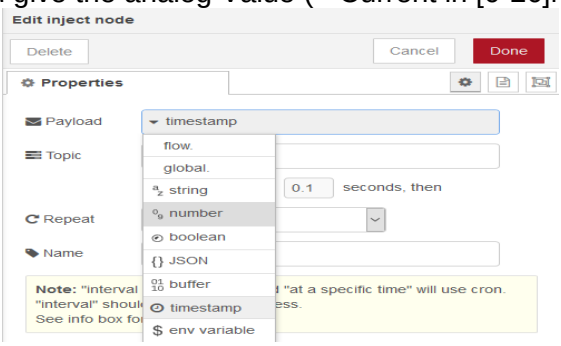
10.2.1 Analog Inputs

1. Add the inject node.
2. Add voltage input/current input node.
3. Add Debug node.
4. Wire (as shown in picture) and Deploy.

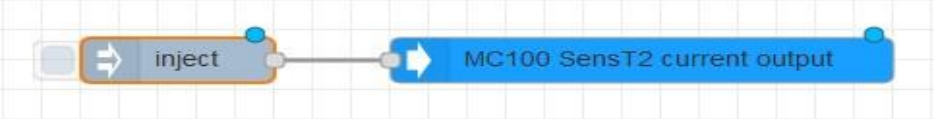


10.2.2 Write Analog Inputs

1. Add Inject node, double click on it and change the payload to Numbers and give the analog Value (= Current in [0-20]mA*1000).

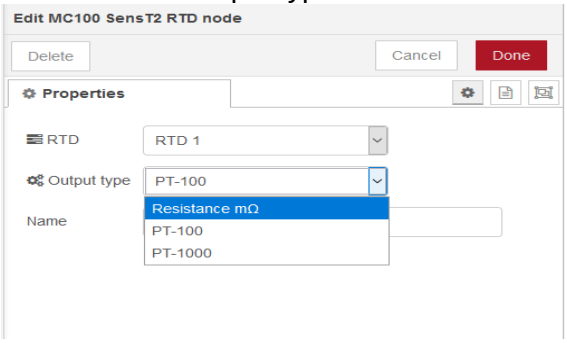


2. Add MC100 SensT2 current output node.
3. Wire and Deploy.



10.2.3 PT100/1000

1. Add the Inject node
2. Add MC100 SensT2 RTD
3. Double click on MC100 SensT2 RTD node to open the settings menu and choose RTD and Output type.

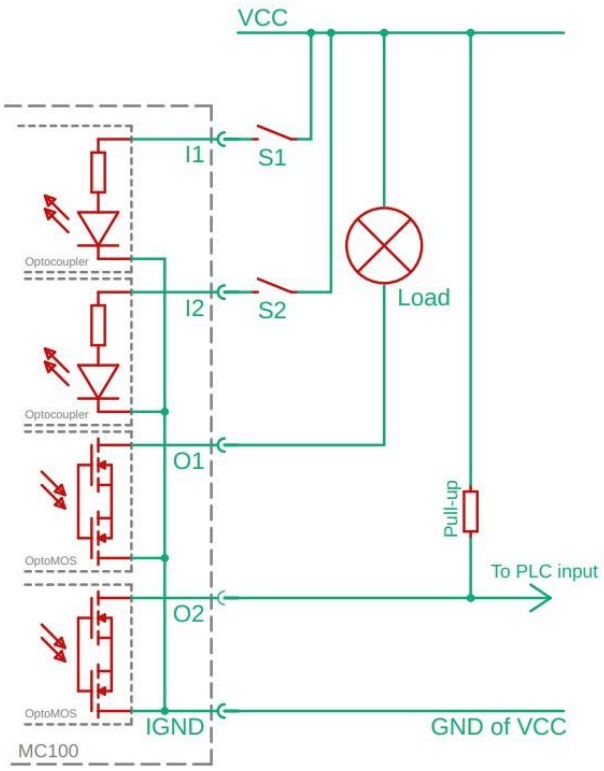


4. Add Debug node(Change name if wanted)
5. Wire and Deploy

11 Contol and readout of analog and digital IO's and LEDs via command line interface

The digital and analog inputs and outputs of the MC100 products can be controlled or read out by commands via command line interface using SSH access or from programming tools. The corresponding commands are shown below with examples.

11.1 MC100 Circuit



Inputs:

- Galvanically isolated inputs with respect to ground.
- High detection from ~5VDC.
- max. 30VDC.

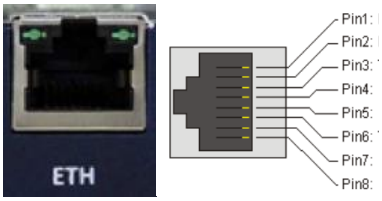
Outputs:

- Galvanically isolated OptoMOS outputs (against ground).
- Open drain circuit.
- max. 30VDC, 120mA RMS/DC.

Example:

- O1: Control of direct load (relay, lamp,….).
- O2: Example with pull-up for control inputs.

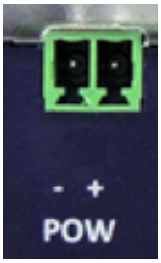
11.2 Ethernet-Interface (All)



Port	Signal	Description
1	TD+	Transmit plus
2	TD-	Transmit minus
3	RD+	receive plus
4	No function	Internal 100nF against GND
5	No function	Internal 100nF against GND
6	RD-	Receive minus
7	nc	
8	Shield	Internal shielding
LED yellow	SPEED	1 Blinks: Led Blinks once = 10Base link 2 Blinks: Led Blinks twice = 100Base link
LED green	ENWAYEN	ON: Network link has been established Blinking: Network activity has been detected

11.3 Power supply (all)

Connection to an AC power supply must be made by using a suitable power supply unit. The DC voltage must be between 8 and 30V.



Port	Description
POW -	Power supply, negative connection
POW +	Power supply, positive connection

11.4 MC100 Gateway

Port	Description
I1	Digital input 1, DC voltage 0 to 30V, switching threshold approx .4V DC
I2	Digital input 2, DC voltage 0 to 30V, switching threshold approx .4V DC
O1	Digital output 1, switching capacity max. 300mA
O2	Digital output 1, switching capacity max. 300mA
IGND	I/O ground, electrically isolated from standard GND of the device

11.4.1 Digital inputs

The MC100 Gateway has two digital inputs. The inputs are high-active.

Function: Read in the digital inputs I1 and I2

Port	Parameter	Read	Write	Path
I1	1,0	x		cat /sys/mc100_gpios/IN1
I2	1,0	x		cat /sys/mc100_gpios/IN2

Example: Reading Input I1
Command: cat /sys/mc100_gpios/IN1
Response: 1# or. 0#

11.4.2 Digital Outputs

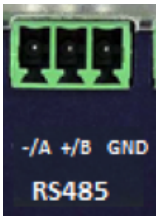
The MC100 Gateway has two digital outputs. The outputs are solid state relays and switch against IGND.

Function: Switching the digital outputs O1 and O2

Port	Parameter	Read	Write	Path
O1	1,0		x	echo 1 > /sys/mc100_gpios/OUT1
				echo 0 > /sys/mc100_gpios/OUT1
O2	1,0		x	echo 1 > /sys/mc100_gpios/OUT2
				echo 0 > /sys/mc100_gpios/OUT2

Example: Switch on O1
Command: echo 1 > /sys/mc100_gpios/OUT1
Response: #

11.4.3 RS485 interface 2-wire bus without galvanic isolation



Port	Description
-/A	Inverted line
+/B	Non-inverted line
GND	Ground / Earth

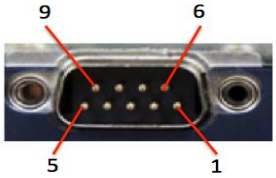
*Note: The RS485 interface is **not** galvanically isolated.*

The RS485 interface can be addressed as /dev/ttymxc4 from the console.
Example: picocom -b 115200 /dev/ttymxc4
Now it is possible to communicate with the RS485 port. RX / TX is switched automatically.
Finally terminate picocom with Ctrl + a + x.

11.4.4 CAN-Interface

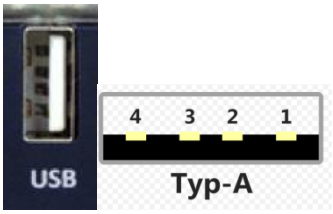
Port	Description
CH	CAN High
CL	CAN Low
GND	Ground / Earth

11.4.5 RS232-Interface



Port	Signal	Description
1	nc	Not connected
2	RXD	receive data
3	TXD	Transmit Data
4	nc	Not connected
5	GND	Ground
6	nc	Not connected
7	RTS	Request To Send
8	CTS	Clear To Send
9	nc	Not connected

11.4.6 USB-Port 2.0 Type A



Port	Signal
1	+
2	D-
3	D+
4	-

11.5 MC100 GPIO

11.5.1 Digital Inputs

Port	Description
DGND1	Digital Ground1, electrically isolated to all (D)GND
I1 bis I4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 4V All input voltages with DGND1 as ground
VPU1	Not supported, please do not connect a signal
DGND2	Digital Ground 2, electrically isolated to all (D)GND
I1 bis I4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 4V All input voltages with DGND2 as ground
VPU2	Not supported, please do not connect a signal

Function: Reading the digital inputs I1 to I8 (v1.0.4 and older)

Port	Parameter	Read	Write	Path
I1	1,0	x		/sys/gpio_board_IO/inputs/input0/value
I2	1,0	x		/sys/gpio_board_IO/inputs/input1/value
I3	1,0	x		/sys/gpio_board_IO/inputs/input2/value
I4	1,0	x		/sys/gpio_board_IO/inputs/input3/value
I5	1,0	x		/sys/gpio_board_IO/inputs/input4value
I6	1,0	x		/sys/gpio_board_IO/inputs/input5/value
I7	1,0	x		/sys/gpio_board_IO/inputs/input6/value
I8	1,0	x		/sys/gpio_board_IO/inputs/input7/value

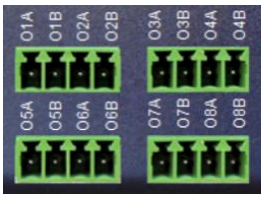
Example: Read out Input I1
Command: cat /sys/gpio_board_IO/inputs/input0/value
Response: e.g. 1# or 0#

Function: Reading the digital inputs I1 to I8 (v1.0.5 and later)

Port	Parameter	Read	Write	Path
I1	1,0	x		/sys/gpio_board/input/1
I2	1,0	x		/sys/gpio_board/input/2
I3	1,0	x		/sys/gpio_board/input/3
I4	1,0	x		/sys/gpio_board/input/4
I5	1,0	x		/sys/gpio_board/input/5
I6	1,0	x		/sys/gpio_board/input/6
I7	1,0	x		/sys/gpio_board/input/7
I8	1,0	x		/sys/gpio_board/input/8

Example: Read out Input I1
Command: cat /sys/gpio_board/input1/value
Response: e.g. 1# or 0#

11.5.2 Digital Outputs



All outputs have an identical behaviour. The description here is only for one output (x = 1 to 8 MC100 GPIO).
All switching outputs are represented by solid state relays. OxA occupies one pin of the switch, OxB the second pin. All outputs are galvanically isolated. The maximum switching current is 300mA. The maximum applied voltage must not exceed 30V.

Function: Switching the digital outputs (**v1.0.4 or older**)

Port	Parameter	Read	Write	Path
O1A,O1B	1,0		X	/sys/gpio_board_IO/outputs/output0/value
O2A,O2B	1,0		X	/sys/gpio_board_IO/outputs/output1/value
O3A,O3B	1,0		X	/sys/gpio_board_IO/outputs/output2/value
O4A,O4B	1,0		X	/sys/gpio_board_IO/outputs/output3/value
O5A,O5B	1,0		X	/sys/gpio_board_IO/outputs/output4/value
O6A,O6B	1,0		X	/sys/gpio_board_IO/outputs/output5/value
O7A,O7B	1,0		X	/sys/gpio_board_IO/outputs/output6/value
O8A,O8B	1,0		X	/sys/gpio_board_IO/outputs/output7/value

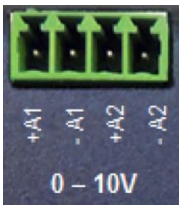
Example: Switch on output O1A,O1B
Command: echo 1 > /sys/gpio_board_IO/outputs/output0/value
Response: #

Function: Switching the digital outputs (**v1.0.5 or later**)

Port	Parameter	Read	Write	Path
O1A,O1B	1,0		X	/sys/gpio_board/output/1
O2A,O2B	1,0		X	/sys/gpio_board/output/2
O3A,O3B	1,0		X	/sys/gpio_board/output/3
O4A,O4B	1,0		X	/sys/gpio_board/output/4
O5A,O5B	1,0		X	/sys/gpio_board/output/5
O6A,O6B	1,0		X	/sys/gpio_board/output/6
O7A,O7B	1,0		X	/sys/gpio_board/output/7
O8A,O8B	1,0		X	/sys/gpio_board/output/8

Example: Switch on output O1A,O1B
Command: echo 1 > /sys/gpio_board/output/value
Response: #

11.5.3 Voltage inputs 0 – 10V



Port	Description
+A1	Positive connection Input 1
-A1	Negative connection Input 1
+A2	Positive connection Input 2
-A2	Negative connection Input 2

The input current at 10V is approx. 2 mA

Note: The applied DC voltage must not exceed 10V.

Function: Reading the voltage inputs (**v1.0.4 or older**)

Port	Parameter	Read	Write	Path
+A1,-A1	Value	x		/sys/gpio_board_ADC/voltage/volt0/value
+A2,-A2	Value	x		/sys/gpio_board_ADC/voltage/volt1/value

Example: Read voltage at ADC input +A1,-A1
Command: cat /sys/gpio_board_ADC/voltage/volt0/value
Response: e.g. 6400

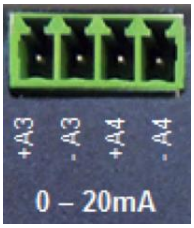
Function: Reading the voltage inputs (**v1.0.5 or later**)

Port	Parameter	Read	Write	Path
+A1,-A1	Value	x		/sys/gpio_board/ voltage_in/1
+A2,-A2	Value	x		/sys/gpio_board/ voltage_in/2

Example: Read voltage at ADC input +A1,-A1
Command: cat /sys/gpio_board/ voltage_in1/value
Response: e.g. 6400

Note: Converting to volts
Formula: Voltage in Volt = value / 1000
Example: Value = 6400 corresponds to 6.4V

11.5.4 Current inputs 0 - 20mA



Port	Description
+A3	Positive connection Input 3
-A3	Negative connection Input 3
+A4	Positive connection Input 4
-A4	Negative connection Input 4

Note: A current of up to 20mA must be fed in from a current source.

Function: Reading the current inputs (**v1.04 and older**)

Port	Parameter	Read	Write	Path
+A3,-A3	Value	x		/sys/gpio_board_ADC/ampere/amp0/value
+A4,-A4	Value	x		/sys/gpio_board_ADC/ampere/amp1/value

Example: Read current at ADC input +A3,-A3
Command: cat /sys/gpio_board_ADC/ampere/amp0/value
Response: e.g. 2000

Function: Reading the current inputs (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
+A3,-A3	Value	x		/sys/gpio_board/current_in/1
+A4,-A4	Value	x		/sys/gpio_board/current_in/2

Example: Read current at ADC input +A3,-A3
Command: cat /sys/gpio_board/current_in1/value
Response: e.g. 2000

Note: Converting to mA
Formula: Current in mA = Value / 200
Example: Value = 2000 corresponds to 10mA

11.5.5 PWM

Port	Description
GND	Ground, Earth
PW2	Open collector digital output.
PW1	Open collector digital output.
PU	Input for internal protection diode (freewheeling diode) for inductive loads. (positive supply voltage)

Function: Switching the digital outputs (v1.0.4 and later)

Port	Parameter	Read	Write	Path
PW2	Prescaler {0,1025}		x	/sys/gpio_board_IO/outputPwm/pwm1/prescalar
	Pulse {0,65535}			/sys/gpio_board_IO/outputPwm/pwm1/pulse
PW1	Prescaler {0,1025}		x	/sys/gpio_board_IO/outputPwm/pwm0/prescalar
	Pulse {0,65535}			/sys/gpio_board_IO/outputPwm/pwm0/pulse

Function: Switching the digital outputs (v1.0.5 and later)

Port	Parameter	Read	Write	Path
PW2	Prescaler {0,1025}		x	/sys/gpio_board/pwm/prescalar
	Pulse {0,65535}			/sys/gpio_board/pwm/1
PW1	Prescaler {0,1025}		x	/sys/gpio_board/pwm/prescalar
	Pulse {0,65535}			/sys/gpio_board/pwm/2
	Period*{0,11}		x	/sys/gpio_board/pwm/period

*Only if MC100 GPIO board supports Period.

11.6 MC100 SensT2

11.6.1 Digital Inputs

Port	Description
DGND1	Digital Ground1, electrically isolated to all (D)GND
I1 bis I4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 4V All input voltages with DGND1 as ground
VPU1	Not supported, please do not connect a signal
DGND2	Digital Ground 2, electrically isolated to all (D)GND
I1 bis I4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 4V All input voltages with DGND2 as ground
VPU2	Not supported, please do not connect a signal

Function: Reading the digital inputs I1 to I8 (**v1.0.4 and older**)

Port	Parameter	Read	Write	Path
I1	1,0	x		/sys/senst2_IO/inputs/input0/value
I2	1,0	x		/sys/senst2_IO /inputs/input1/value
I3	1,0	x		/sys/senst2_IO /inputs/input2/value
I4	1,0	x		/sys/senst2_IO /inputs/input3/value
I5	1,0	x		/sys/senst2_IO /inputs/input4/value
I6	1,0	x		/sys/senst2_IO /inputs/input5/value
I7	1,0	x		/sys/senst2_IO /inputs/input6/value
I8	1,0	x		/sys/senst2_IO /inputs/input7/value

Example: Read out input I1
Command: cat /sys/senst2_IO/inputs/input0/value
Response: e.g. 1# or 0#

Function: Reading the digital inputs I1 to I8 (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
I1	1,0	x		/sys/senst2_board/input/1
I2	1,0	x		/sys/senst2_board/input/2
I3	1,0	x		/sys/senst2_board/input/3
I4	1,0	x		/sys/senst2_board/input/4
I5	1,0	x		/sys/senst2_board/input/5
I6	1,0	x		/sys/senst2_board/input/6
I7	1,0	x		/sys/senst2_board/input/7
I8	1,0	x		/sys/senst2_board/input/8

Example: Read out input I1
Command: cat /sys/senst2_board/input1/value
Response: e.g. 1# or 0#

11.6.2 Digital Outputs

All outputs have an identical behaviour. The description here is only for one output (; x = 1 and 2 MC100 SensT2).
All switching outputs are represented by solid state relays. OxA occupies one pin of the switch, OxB the second pin. All outputs are galvanically isolated. The maximum switching current is 300mA. The maximum applied voltage must not exceed 30V.

Function: Switching the digital outputs (**v1.0.4 and older**)

Port	Parameter	Read	Write	Path
O1A,O1B	1,0		X	/sys/senst2_IO/outputs/output0/value
O2A,O2B	1,0		X	/sys/senst2_IO/outputs/output1/value

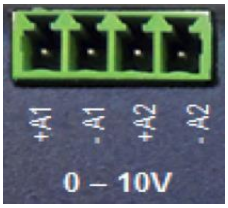
Example: Switch on output O1A,O1B
Command: echo 1 > /sys/senst2_IO/outputs/output0/value
Response: #

Function: Switching the digital outputs (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
O1A,O1B	1,0		X	/sys/senst2_board/current_out/1
O2A,O2B	1,0		X	/sys/senst2_board/current_out/2

Example: Switch on output O1A,O1B
Command: echo 1 > /sys/senst2_board/current_out1/value
Response: #

11.6.3 Voltage inputs 0 – 10V



Port	Description
+A1	Positive connection Input 1
-A1	Negative connection Input 1
+A2	Positive connection Input 2
-A2	Negative connection Input 2

The input current at 10V is approx. 2 mA

Note: The applied DC voltage must not exceed 10V.

Function: Reading the voltage inputs (**v1.0.4 and older**)

Port	Parameter	Read	Write	Path
+A1,-A1	Value	x		/sys/senst2_ADC/voltage/volt0/value
+A2,-A2	Value	x		/sys/senst2_ADC/voltage/volt1/value

Example: Read voltage at ADC input +A1,-A1
Command: cat /sys/senst2_ADC/voltage/volt0/value

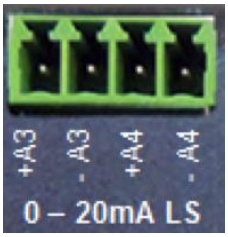
Response: e.g. 6400
Function: Reading the voltage inputs (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
+A1,-A1	Value	x		/sys/senst2_board/voltage_in/1
+A2,-A2	Value	x		/sys/senst2_board/voltage_in/2

Example: Read voltage at ADC input +A1,-A1
Command: cat /sys/senst2_board/voltage_in1/value
Response: e.g. 6400

Note: Conversion to volts
Formula: Voltage in Volt = value / 1000
Example: Value = 6400 corresponds to 6.4V

11.6.4 Current inputs 0 - 20mA



The analog current inputs of the MC100 SensT2 have two operating modes.

1. Supply of a current - current from the sensor - sensor with own power supply
2. Low side shunt - with power supply for the sensor

11.6.4.1 *Einspeisung eines Stroms - Strom aus dem Sensor*

The connected sensor has its own power supply and a current output with a current of 20mA.
The sensor is connected to -A3 or -A4 against ground.

Port	Description
-A3	Positive connection input 3, current output of the sensor
GND	Negative connection Input 3, sensor ground
-A4	Positive connection input 4, current output of the sensor
GND	Negative connection Input 4, sensor ground

11.6.4.2 *Low Side Shunt - power supply for the sensor*

It is possible both to supply the sensor with power and to measure the current it draws over the same two wires. The sensor will be powered from the MC100 SensT2. A voltage is applied to the terminals +A3 or +A4 for this purpose. Connect the sensor to +A3 and A3 or +A4 and -A4.

Port	Description
+A3	Positive connection output 3, sensor supply, approx. 12V to 14V
-A3	Negative connection Input 3
+A4	Positive connection output 4, sensor supply, approx. 12V to 14V
-A4	Negative connection Input 4

Function: Reading the current inputs (**v1.0.4 and older**)

Port	Parameter	Read	Write	Path
+A3,-A3	Value	x		/sys/senst2_ADC/ampere/amp0/value
+A4,-A4	Value	x		/sys/senst2_ADC/ampere/amp1/value

Example: Read current at ADC input +A3,-A3
Command: cat /sys/senst2_ADC/ampere/amp0/value
Response: e.g. 1500

Function: Reading the current inputs (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
+A3,-A3	Value	x		/sys/senst2_board/current_in/1
+A4,-A4	Value	x		/sys/senst2_board/current_in/2

Example: Read current at ADC input +A3,-A3
Command: cat /sys/senst2_board/current_in1/value
Response: e.g. 1500

Note: Conversion to mA
Formula: Current in mA = Value * 20 / 3000
Example: Value = 1500 corresponds to 10mA

11.6.5 Current outputs 0 - 20mA

Function: Output of a current (**v1.0.4 and older**)

Port	Parameter	Read	Write	Path
OI1	Value		x	/sys/senst2_ADC/adc_output/adcoutput0/value
OI2	Value		x	/sys/senst2_ADC/adc_output/adcoutput1/value

Example: Output of 6mA at OI1
Command: echo 6000 > /sys/senst2_ADC/adc_output/adcoutput0/value
Response: none

Function: Output of a current (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
OI1	Value		x	/sys/senst2_board/output/1
OI2	Value		x	/sys/senst2_board/output/2

Note: Conversion from mA to value to be supplied
Formula: Value = Current in mA * 1000
Example: Current = 6mA Corresponds to Value = 6000

11.6.6 PT100 / PT1000 Inputs

Function: Reading a resistance/ temperature value (**v1.0.4 and older**)

Port	Parameter	Read	Write	Path
RTD1	Value	x		/sys/senst2_rtd/rtd0/value
RTD2	Value	x		/sys/senst2_rtd/rtd1/value
RTD3	Value	x		/sys/senst2_rtd/rtd2/value
RTD4	Value	x		/sys/senst2_rtd/rtd3/value

Example: Read the value at RTD1
Command: cat /sys/senst2_rtd/rtd0/value
Response: e.i. 100000

Function: Reading a resistance/ temperature value (**v1.0.5 and later**)

Port	Parameter	Read	Write	Path
RTD1	Value	x		/sys/senst2_board/rtd/1
RTD2	Value	x		/sys/senst2_board/rtd/2
RTD3	Value	x		/sys/senst2_board/rtd/3
RTD4	Value	x		/sys/senst2_board/rtd/4

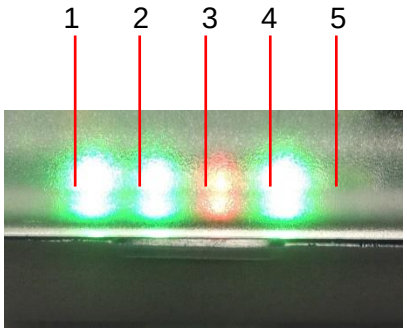
Example: Read the value at RTD1
Command: cat /sys/senst2_board/rtd/1
Response: e.i. 100000

Conversion at port of a resistor in Ohm
Formula: Resistance in Ohm = Value / 100
Example: Value = 100000 corresponds 1000Ohm

Conversion at port of a PT1000 temperature sensor to °Celsius
Formula: Temperature in Grad Celsius = (Value / 100 -1000) / 3,891
Example: Value = 112000 corresponds to +30,84°Celsius
Value = 100000 corresponds to 0°Celsius
Value = 90000 corresponds to -25,5°Celsius

11.7 LEDs

LED	Description
LED 1	Green, Power supply
LED 2	Green, Configurable.
LED 3	Red, Configurable.
LED 4	Long-time Off: Searching for connection. Long-time ON: Connected. Flashing quickly: Packet Data transfer.
LED 5	Green, Reserved for GPS.



To configure the LEDs you need to log in to the MC100 using the Command line.

1. **LED 2:**
   - **Turn ON:**
     ```
     root@MC100: ~#  echo “on”  > /sys/mc100_leds/led2/mode
     ```
   - **Turn OFF:**
     ```
     root@MC100: ~#  echo “off”  > /sys/mc100_leds/led2/mode
     ```
   - **Flash mode:**
     ```
     root@MC100: ~#  echo "flash" >/sys/mc100_leds/led2/mode
     ```
   - **Blink mode:**
     ```
     root@MC100: ~#  echo 50 >/sys/mc100_leds/led2/blinkPeriod
     ```
     Note: “50” is the time between each blinking in ms, can be set to any value.
2. **LED 3:**
   - **Turn ON:**
     ```
     root@MC100: ~#  echo “on”  > /sys/mc100_leds/led1/mode
     ```
   - **Turn OFF:**
     ```
     root@MC100: ~#  echo “off”  > /sys/mc100_leds/led1/mode
     ```
   - **Flash mode:**
     ```
     root@MC100: ~#  echo "flash" >/sys/mc100_leds/led1/mode
     ```
   - **Blink mode:**
     ```
     root@MC100: ~#  echo 50 >/sys/mc100_leds/led1/blinkPeriod
     ```
     Note: “50” is the time between each blinking in ms, can be set to any value.
3. **LED 5:**
   - Enter the following for initialization:
     ```
     root@MC100: ~#  echo 69 > /sys/class/gpio/export
     root@MC100: ~#  echo out > /sys/class/gpio/gpio78/direction
     ```
   - **Turn ON:**
     ```
     root@MC100: ~#  echo 1 > /sys/class/gpio/gpio78/value
     ```
   - **Turn OFF:**
     ```
     root@MC100: ~#  echo 0 > /sys/class/gpio/gpio78/value
     ```

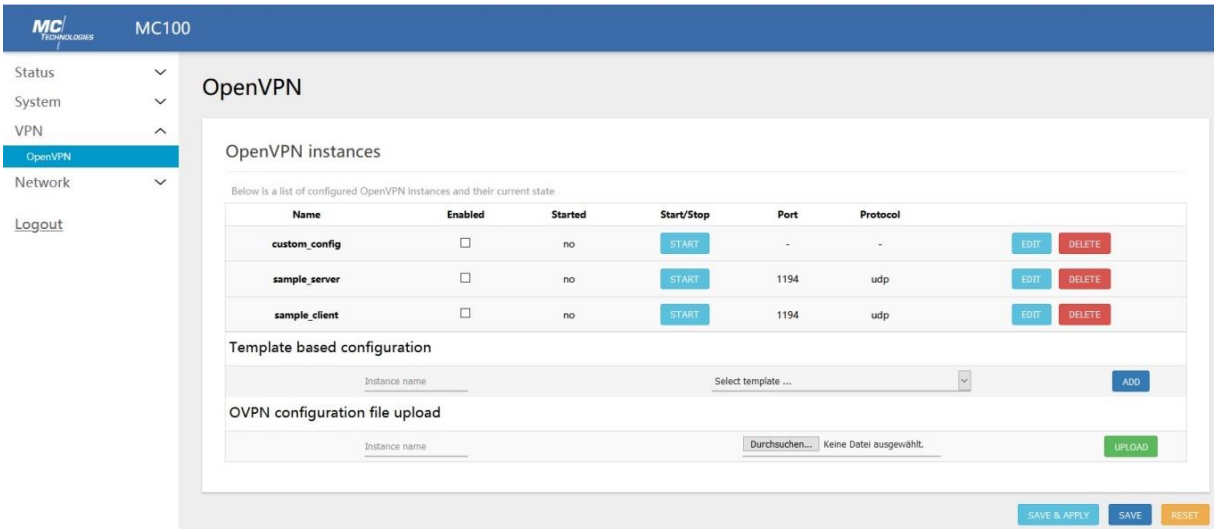
12 OpenVPN

Before starting, create your own Certificate Authority (CA) , certificates and keys for an OpenVPN server and clients.
You will need :

- Certificate Authority (ca.crt)
- Server certificate (server.crt) and server key (server.key)
- Client certificate (client.crt) and client key (client.key)

12.1 Client Configuration

1. After logging to MC100 Luci interface using the IP address (default 192.168.2.1), click on “OpenVPN” under “VPN” tab.



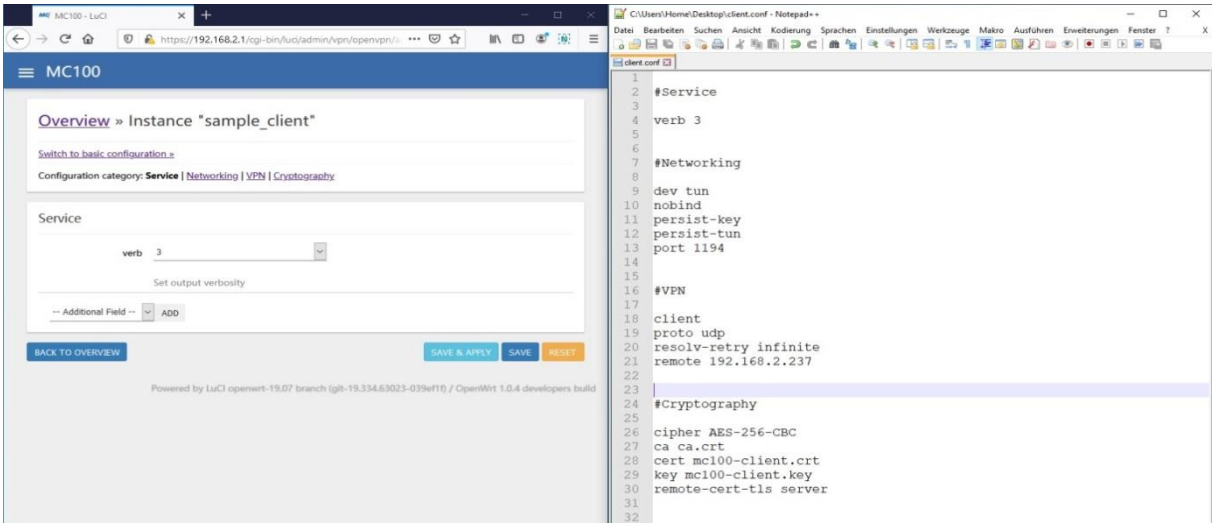
2. Click on “EDIT” at “sample_client” tab.

3. Click on “Switch to advanced configuration”.

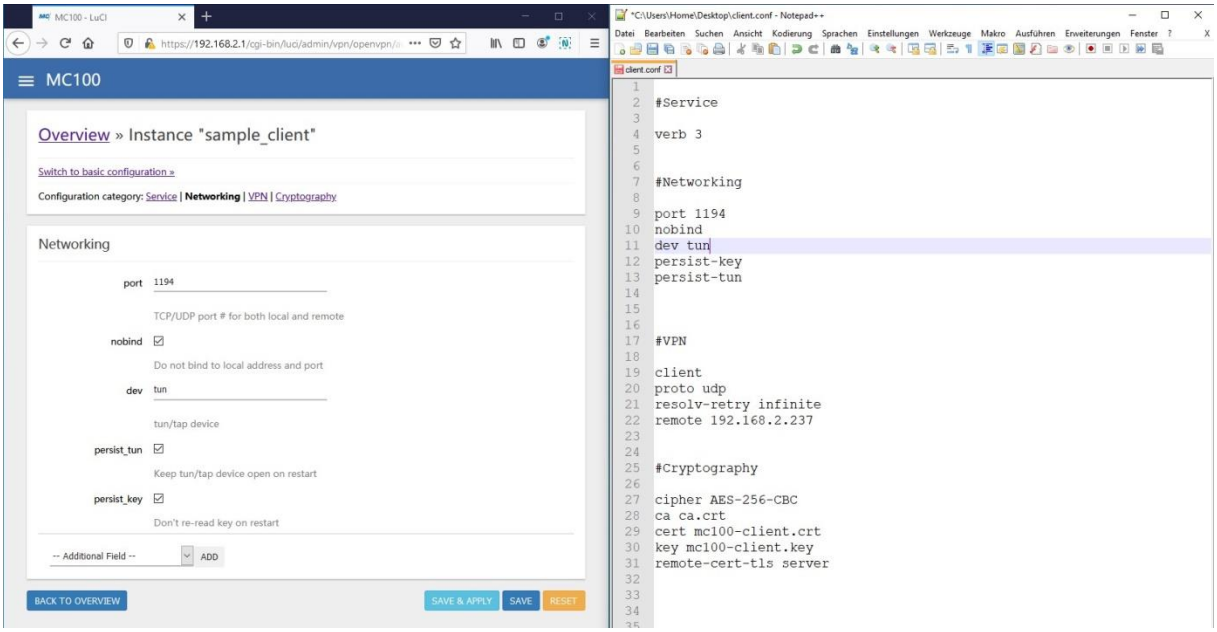
4. Edit the settings as the one in the client.conf file you have.

Example:

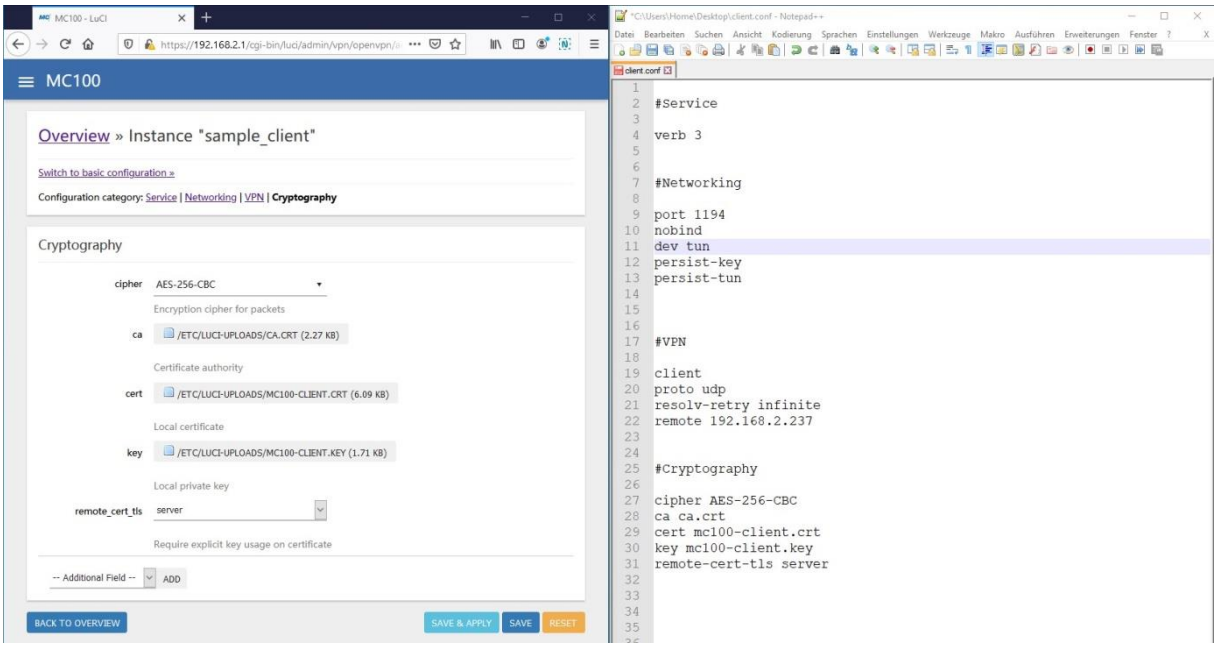
1. Under Service, Change “verb” (verbosity) same as given in the .conf file (here it is 3).



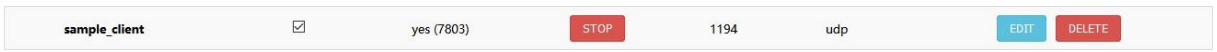
2. Under Networking, change every setting as same as given in the .conf file.



3. Same for VPN section.
4. Under Cryptography:
   a. Change the cipher type to the one given in the .conf file.
   b. In the path /ETC/LUCI-UPLOADS/ upload the Certificate authority “ca” file(.CRT), local certificate “cert” (.CRT), and the local private key “key” (.key).

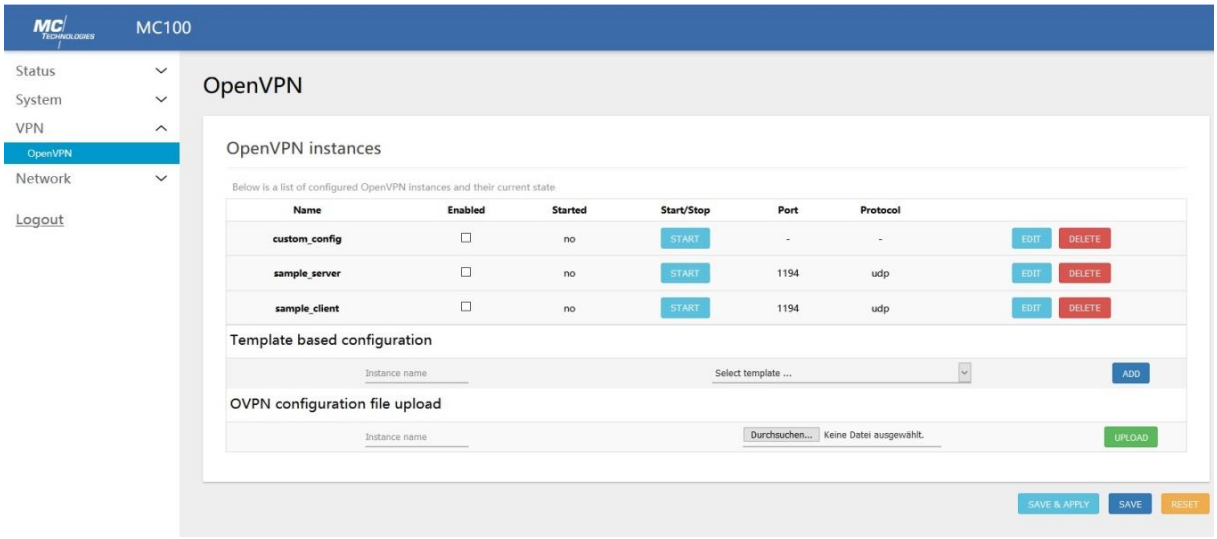


5. Click on “SAVE & APPLY” and then “BACK TO OVERVIEW”.
6. Enable the configured instance, then click on “SAVE AND APPLY”, then “START”.



12.2 Server Configuration

1. After logging to MC100 Luci interface using the IP address (default 192.168.2.1), click on “OpenVPN” under “VPN” tab.



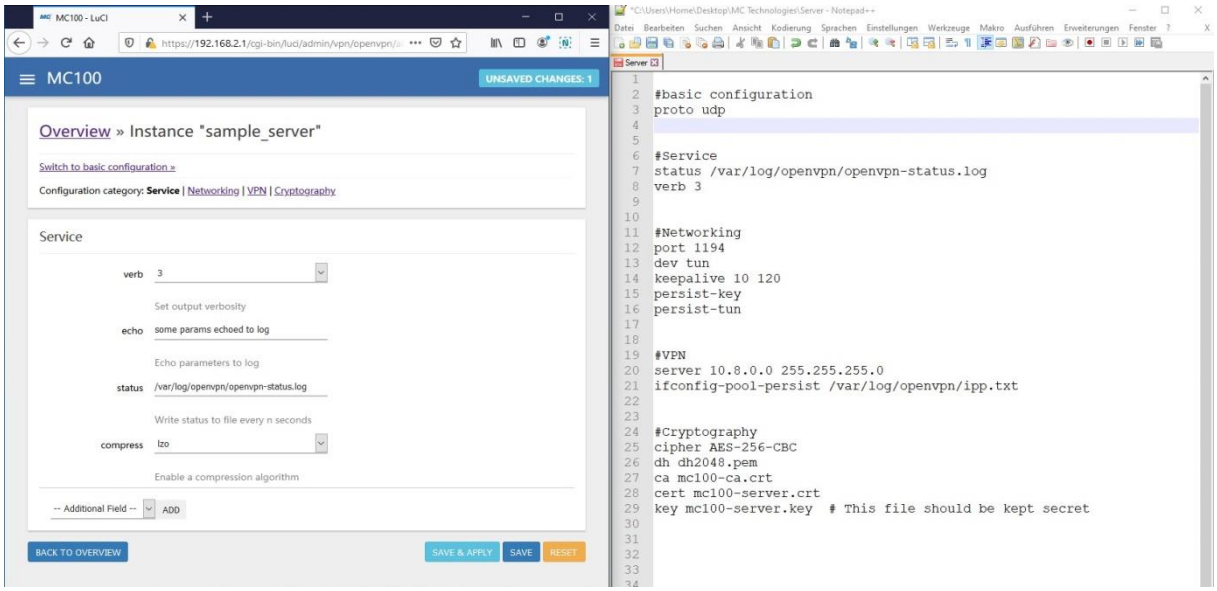
2. Click on “EDIT” at “sample_server” tab.

3. Click on “Switch to advanced configuration”.

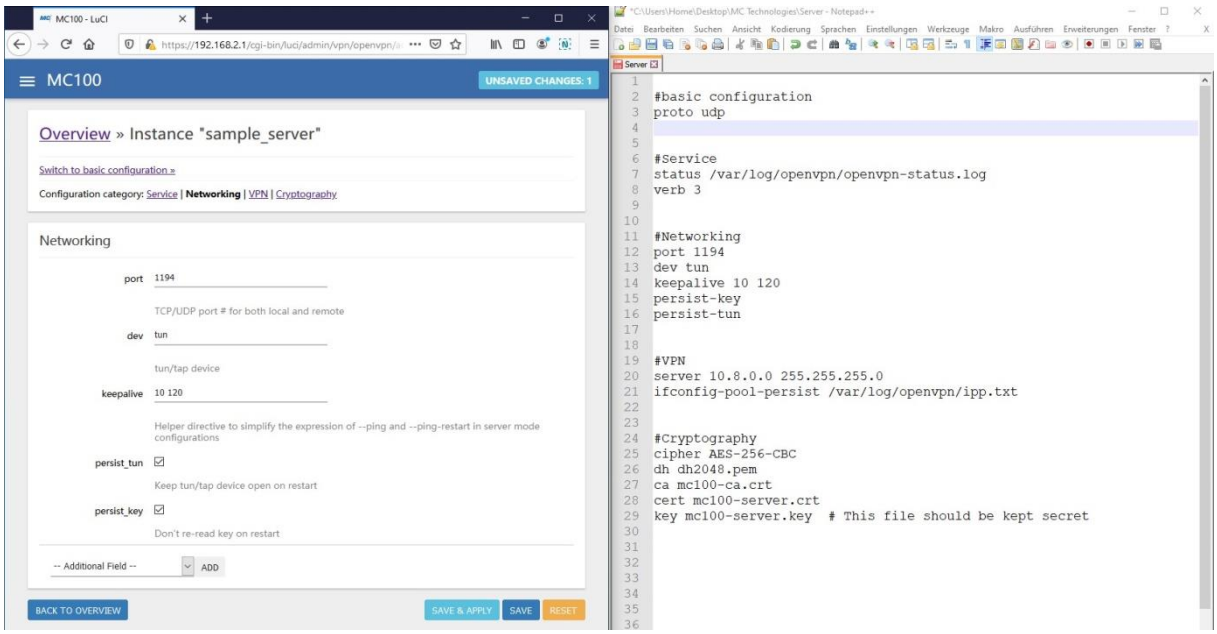
4. Edit the settings as the one in the server.conf file you have.

Example:

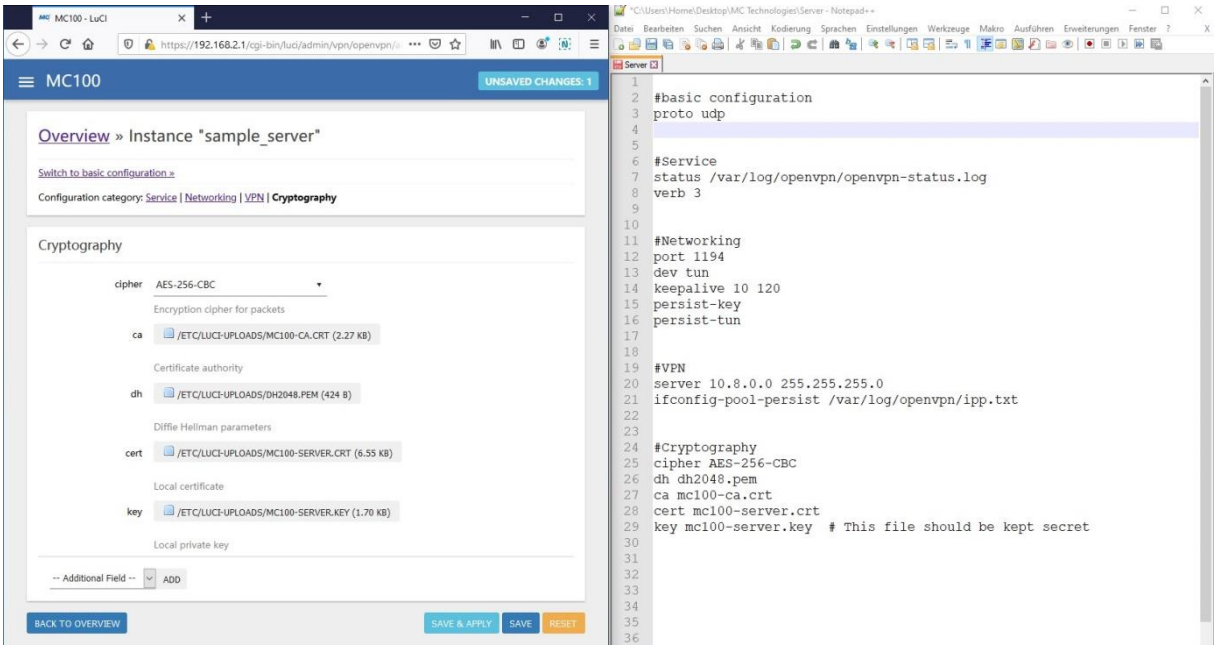
1. Under Service, Change “verb” (verbosity) same as given in the .conf file (here it is 3).



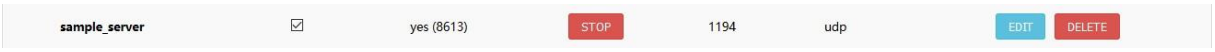
2. Under Networking, change every setting as same as given in the .conf file.



3. Same for VPN section.
4. Under Cryptography:
   a. Change the cipher type to the one given in the .conf file.
   b. In the path /ETC/LUCI-UPLOADS/ upload the Certificate authority “ca” file(.CRT), Diffie Hellman parameters “dh” (.PEM), local certificate “cert” (.CRT), and the local private key “key” (.key).



5. Click on “SAVE & APPLY” and then “BACK TO OVERVIEW”.
6. Enable the configured instance, and then click on “SAVE AND APPLY”, then “START”.



13 WMBus

13.1 General information

The MC100 Terminal/Gateway can be extended with an extension board for Wireless M-Bus packages.

13.2 Installing Command line tool mcwmbus

To interact with the WM-Bus extension, mcwmbus command line tool is needed.
It can be installed using opkg:

```
root@MC100: ~# opkg update
root@MC100: ~# opkg install mcwmbus
```

Update the tool to ensure that you get the latest features and stability updates.

13.3 Basic functionality

The tool supports the -h parameter to print information about its usage.
The output might look like this:

```
Usage: mcwmbus [options]
Copyright (C) 2020 MC-Technologies GmbH
Options:
-h        Show this help message and exit.
-V        Show version information and exit.
-v        Print verbose debug information
-d        DEVICE tty device (default: /dev/ttymxc2)
-b        BAUD Baud rate for comunication (default: 19200)
-f        FILENAME Write data to file.
-u        URL Send Data via POST Request to URL
-c        Print Data on commandline
-F        FORMAT Data format for files (default: hex)
-U        FORMAT Data format for POST data (default: json)
-C        FORMAT Data format for command-line data (default: json)
          FORMAT must be one of the following:
                    hex - RAW-Data HEX encoded
                    json - Data interpreted as JSON
                    json_hex - Data as JSON, uninterpreted
                    xml - Data interpreted as XML

       FILENAME and URL allow the following special characters,
which will be replaced:
                    %M - Manufacturer ID
                    %D - Device Type / Medium
                    %U - Ident Nr.
                    %V - Version
```

13.4 Output formats

The tool supports different output formats, which can be individually set for each output channel (URL, FILE, Command line).

13.4.1 hex

Hex produces the message in hexadecimal. One message per line.

Example on hex Output:

```
1644AF4C02000041011B7A980000000266E8000266E900
```

13.4.2 json

Json interprets the message and gives as much information as possible. It also contains the raw message as a hexadecimal string.

Example on json Output:


```
{
 "SlaveInformation": {
  "Id": 41000002,
  "Manufacturer": "SEO",
  "Version": 1,
  "ProductName": "Sensoco Wireless M¬Bus 2 NTC Temperature
  Sensor",
  "Medium": "Ambient Sensor",
  "AccessNumber": 152,
  "Status": "00",
  "Signature": "0000"
 },
 "DataRecords": [
  {
   "id": 0,
   "Function": "Instantaneous value",
   "StorageNumber": 0,
   "VIF": 102,
   "VIFE": 0,
   "Unit": "External temperature (1e-1 deg C)",
   "Value": "232",
   "Timestamp": "2020-06-09T07:51:08Z"
  },
  {
   "id": 1,
   "Function": "Instantaneous value",
   "StorageNumber": 0,
   "VIF": 102,
   "VIFE": 0,
   "Unit": "External temperature (1e-1 deg C)",
   "Value": "233",
   "Timestamp": "2020-06-09T07:51:08Z"
   }
 ],
 "RawMessage": "1644AF4C02000041011B7A980000000266E8000266E900"
}
```

13.4.3 XML

XML output interprets the message and produces an XML output.

Example on xml output:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MBusData>

  <SlaveInformation>
        <Id>41000002</Id>
        <Manufacturer>SEO</Manufacturer>
        <Version>1</Version>
        <ProductName>Sensoco Wireless M¬Bus 2 NTC Temperature
Sensor</ProductName>
        <Medium>Ambient Sensor</Medium>
        <AccessNumber>157</AccessNumber>
        <Status>00</Status>
        <Signature>0000</Signature>
  </SlaveInformation>

  <DataRecord id="0">
        <Function>Instantaneous value</Function>
        <StorageNumber>0</StorageNumber>
        <Unit>External temperature (1e-1 deg C)</Unit>
        <Value>232</Value>
        <Timestamp>2020-06-09T07:52:03Z</Timestamp>
  </DataRecord>

  <DataRecord id="1">
        <Function>Instantaneous value</Function>
        <StorageNumber>0</StorageNumber>
        <Unit>External temperature (1e-1 deg C)</Unit>
        <Value>233</Value>
        <Timestamp>2020-06-09T07:52:03Z</Timestamp>
  </DataRecord>

</MBusData>
```

13.4.4 json_hex

It is also possible to get the data in a json format that is easier to read than pure hex

Example on *json_hex* output

```
{"timestamp":"2020-06-09T09:53:47.295+0200","LEN": "22","C":
"44","MAN": "4caf","UID": "41000002","VER": "01","DEV": "1b","CI":
"7a","COUNTER": "a7","STATUS": "00","ENCRYPTION": "0000","DATA":
"0266e8000266e800"}
```

13.5 Send Messages to REST API

It is possible to send the messages to a REST-API, by passing the command line “parameter –u”. For example, this can be used to send messages to the internal node-red server on the mc100:

```
root@MC100: ~# mcwmbus -u http://localhost:1880/wmbus
```

Or it can be used in combination with the integrated modem, to send messages to a server running in the cloud to have real time data available on your workstation.

13.6 Writing on disk

When writing data to disk, it is possible to make parts of the path dynamic, by using the special characters.

```
%M - Manufacturer ID
%D - Device Type / Medium
%U - Ident Nr.
%V - Version
```

The following file-tree has been created by

```
root@MC100: ~# mcwmbus –f wmbus_messages/%M/%U.json
```

This gives us the nice opportunity of preselecting manufacturer and device.

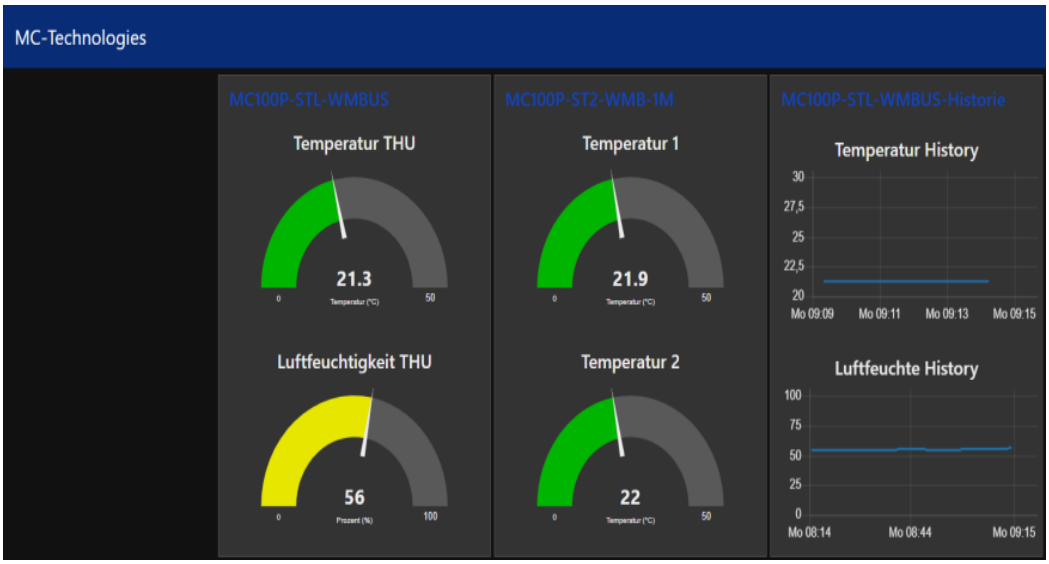
Example file-tree for “ mcwmbus -f wmbus_messages/%M/%U.json”

```
wmbus_messages
+-- 18c4
|   +-- 17949.json
+-- 25c5
|   +-- 33100007.json
+-- 4caf
|   +-- 30000007.json
|   +-- 41000002.json
|   +-- 41000292.json
+-- 8d7
    +-- 570536.json
    +-- 570542.json
    +-- 570663.json
```

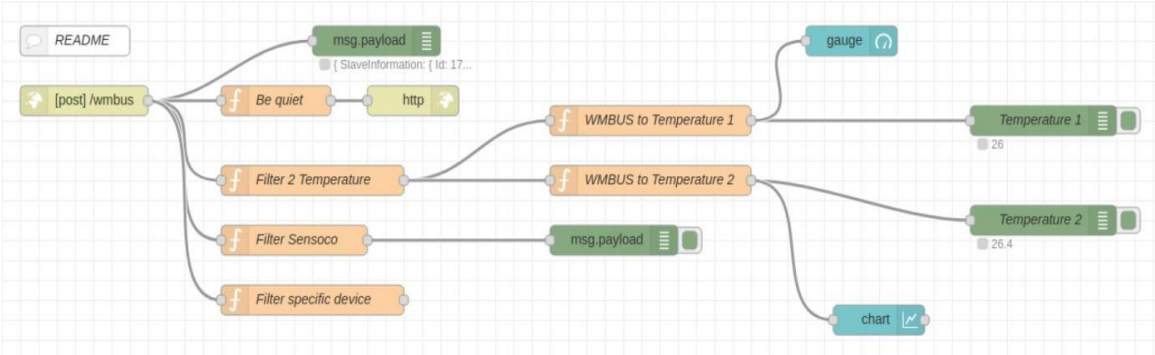
13.7 View live information in Node-RED

To view live information in Node-RED, we can create a flow that has a rest-server and displays the information using the dashboard plugin for node-red.

The output might look like this:



The Node Red flow for example:



Exported Flow from Node RED for example:

```
root@MC100: ~# mcwmbus -u http://localhost:1880/wmbus -c -C json
```


```
[{"id":"71bf261.6ea3858","type":"tab","label":"Flow 1","disabled":false,"info":""},{"id":"27a6f8dc.c998c8","type":"http
in","z":"71bf261.6ea3858","name":"","url":"/wmbus","method":"post", "upload":false,"swaggerDoc":"","x":650,"y":160,"wires":[["bb092b4c.
860858","3fd4f2f3.1075ae","c672490.a3a49b8","2ed266b0.b6c772","36c0f277.3b4abe"]]},{"id":"4fcc7e5f.03c2a8","type":"http
response","z":"71bf261.6ea3858","name":"","statusCode":"","headers":{},"x":1010,"y":160,"wires":[]},{"id":"bb092b4c.860858","type":"de
bug","z":"71bf261.6ea3858","name":"","active":true,"tosidebar":false,"console":false,"tostatus":true,"complete":"payload","targetType"
:"msg","x":970,"y":100,"wires":[]},{"id":"3fd4f2f3.1075ae","type":"function","z":"71bf261.6ea3858","name":"Be
quiet","func":"\nmsg.payload= \"\";\n\n// This node will remove the message from the answer to curl.\n// Whatever is given here as the
payload, will be displayed by the command line tool.\n\nreturn msg;","outputs":1,"noerr":0,"x":860,"y":160,"wires":[["4fcc7e5f.03c
2a8"]]},{"id":"c672490.a3a49b8","type":"function","z":"71bf261.6ea3858","name":"Filter Sensoco","func":"\n// This function will filter
out all messages not from Sensoca (SEO)\n\nif (msg.payload.SlaveInformation.Manufacturer == \"SEO\")\n return msg;\n
\n","outputs":1,"noerr":0,"x":880,"y":300,"wires":[["7d0db0ce.f418e"]]},{"id":"7d0db0ce.f418e","type":"debug","z":"71bf261.6ea3858","n
ame":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","x":1230,"y":300,"wires":[]},{"id":"2ed266b
0.b6c772","type":"function","z":"71bf261.6ea3858","name":"Filter 2 Temperature","func":"\n// This function will filter messages from 2
NTC\n\nif (msg.payload.SlaveInformation.ProductName == \"Sensoco Wireless M-Bus 2 NTC Temperature Sensor\")\n return msg;\n
\n","outputs":1,"noerr":0,"x":900,"y":240,"wires":[["71bc028d.ae4ecc","57c4db21.0eea74"]]},{"id":"71bc028d.ae4ecc","type":"function","
z":"71bf261.6ea3858","name":"WMBUS to Temperature 1","func":"var temperature_id = 0\n\nvar obj =
msg.payload.DataRecords[temperature_id];\n\nif(obj.VIF== 102)\n msg.payload = obj.Value / 10.0;\n\n\nreturn
msg;","outputs":1,"noerr":0,"x":1270,"y":180,"wires":[["c3bf54ec.e25ee","e0b96c34.3ae11"]]},{"id":"c3bf54ec.e25ee","type":"debug","z":
"71bf261.6ea3858","name":"Temperature 1","active":true,"tosidebar":true,"console":false,"tostatus":true,"
complete":"payload","targetType":"msg","x":1700,"y":180,"wires":[]},{"id":"de4c75d9.a1432","type":"debug","z":"71bf261.6ea3858","name"
:"Temperature 2","active":true,"tosidebar":true,"console":false,"tostatus":true,"complete":"payload","targetType":"msg","x":1700,"y":280,"wires":[]}
,{"id":"57c4db21.0eea74","type":"function","z":"71bf261.6ea3858","name":"WMBUS to Temperature 2","func":"var temperature_id = 1 // The
objects are counted from 0, so 2 is 1\n\nvar obj =msg.payload.DataRecords[temperature_id];\n\nif(obj.VIF== 102)\n msg.payload = obj.Value / 10.0;\n\n\nreturn
msg;","outputs":1,"noerr":0,"x":1270,"y":240,"wires":[["de4c75d9.a1432","b228c32e.65e678"]]},{"id":"e0b96c34.3ae11","type":"ui_gauge",
"z":"71bf261.6ea3858","name":"","group":"9013b113.158438","order":1,"width":0,"height":0,"gtype":"gage","title":"gauge","label":"units"
,"format":"{{value}}","min":0,"max":"100","colors":["#39a9c1","#e67700","#ca3838"],"seg1":"19","seg2":"35","x":1490,"y":100,"wires":[
]},{"id":"36c0f277.3b4abe","type":"function","z":"71bf261.6ea3858","name":"Filter specific device","func":"\n// This function will
messages, based on ID and Manufacturer\n\n// ID and Manufacturer are unique for all devices.\n// The device ID used here must be
replaced with the device you have.\n// Please use the Debug Window to find out your device ID\n\nif
(msg.payload.SlaveInformation.Manufacturer == \"SEO\" && \n msg.payload.SlaveInformation.Id == 41000002)\n return
msg;\n","outputs":1,"noerr":0,"x":900,"y":360,"wires":[[]]},{"id":"554b79ab.7be15","type":"comment","z":"71bf261.6ea3858","name":"READ
ME","info":"# WMBus message filter and display\n\nThis Flow is here to demonstrate how to work with WM-Bus messages received by the
MC100.\n\nA command line tool will receive and parse the messages and post them to an url, which gets provided by node red.\n\nThis
way every message can be parsed by node red.\n\nOn the command line call\n\n```\nmcwmbus -U json -u
http://localhost:1880/wmbus\n```\n\nto start receiving messages.\n\nWe tent to provide an init script, that starts the
command on boot, as well as node-red.\nIf this is not the case, feel free to contact us.\n\n## Message Format\n\nHere is an example
messages, as received by node-red:\n\n```\n{\n\"SlaveInformation\": {\n \"Id\": 41000002,\n
\"Manufacturer\": \"SEO\",\n \"Version\": 1,\n\"ProductName\": \"Sensoco Wireless M-Bus 2 NTC Temperature
Sensor\",\n \"Medium\": \"Ambient Sensor\",\n\"AccessNumber\": 225,\n \"Status\": \"00\",\n \"Signature\":
\"0000\"\n },\n \"DataRecords\": [\n {\n \"id\": 0,\n \"Function\": \"Instantaneous value\",\n \"StorageNumber\":
0,\n \"VIF\": 102,\n \"VIFE\": 0,\n \"Unit\":\"External temperature (1e-1 deg C)\",\n \"Value\":
\"245\",\n \"Timestamp\": \"2020-06-08T13:41:42Z\"\n },\n {\n \"id\": 1,\n \"Function\": \"Instantaneous value\",\n
\"StorageNumber\": 0,\n \"VIF\": 102,\n \"VIFE\": 0,\n\"Unit\": \"External temperature (1e-1 deg C)\",\n \"Value\":
\"246\",\n \"Timestamp\": \"2020-06-08T13:41:42Z\"\n }\n],\n \"RawMessage\":
\"1644AF4C020000041011B7AE10000000266F5000266F600\"\n}\n```\n\nThe message itself is in json and the fieldnames should not change, but
the number of DataRecords might, depending on your sensor.\n","x":640,"y":100,"wires":[[]]},{"id":"b228c32e.65e678","type"
:"ui_chart","z":"71bf261.6ea3858","name":"","group":"9013b113.158438","order":1,"width":0,"height":0,"label":"chart","chartType":"line"
,"legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":false,"ymin":"","ymax":"","removeOlder":1,"removeOl
derPoints":"","removeOlderUnit":"3600","cutout":0,"useOneColor":false,"useUTC":false,"colors":["#1f77b4","#aec7e8","#ff7f0e","#2ca02c"
,"#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"outputs":1,"x":1520,"y":380,"wires":[[]]},{"id":"9013b113.158
438","type":"ui_group","z":"","name":"Default","tab":"863748e2.a86768","order":1,"disp":true,"width":"6","collapse":false},{"id":"8637
48e2.a86768","type":"ui_tab","z":"","name":"Home","icon":"dashboard","disabled":false,"hidden":false}]
```


13.8 Combine data for 1h, 6h, 1day and send via FTP/SCP

Collect messages for the whole day as interpreted json, but only send messages from manufacturer SEO to a server at the end of the day in a zip file.

```
while true;
do
mcwmbus -F json -f "wmbus_messages/%M.json" &
sleep 86400 # = 60*60*24 = 24 hours
killall mcwmbus
zip SEO.zip wmbus_messages/4caf.json
scp SEO.zip 192.168.1.1:/data/SEO-`date "+%Y-%m-%d"`.zip
rm SEO.zip
done
```

13.9 Support

Please execute "`mcwmbus -vV`" and sent the output together with your error description and any error message to support@mc-technologies.net
It will be helpful if you run the command that produced the error with "`-vvvvvv`" to turn on the maximal debug information.

14 MODBUS

14.1 MODBUS for Node-RED

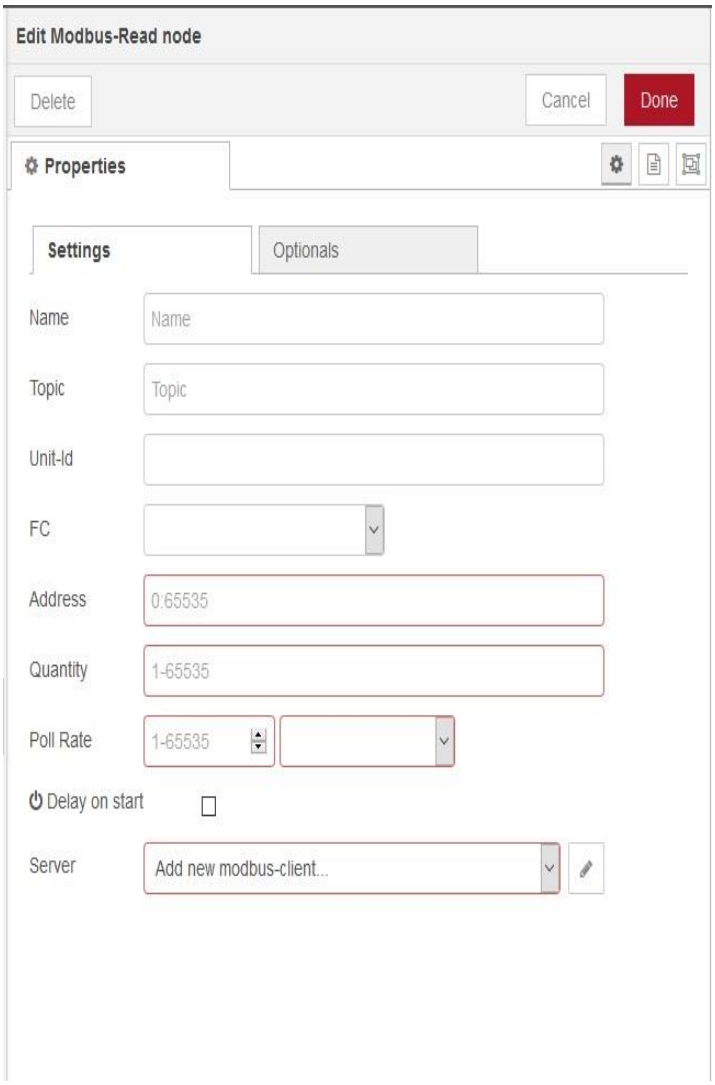
To install the Node-RED package, copy all folders from the node-red-modbus directory to /usr/lib/node/node-red/node_modules/

Then, restart node red via /etc/init.d/node-red restart

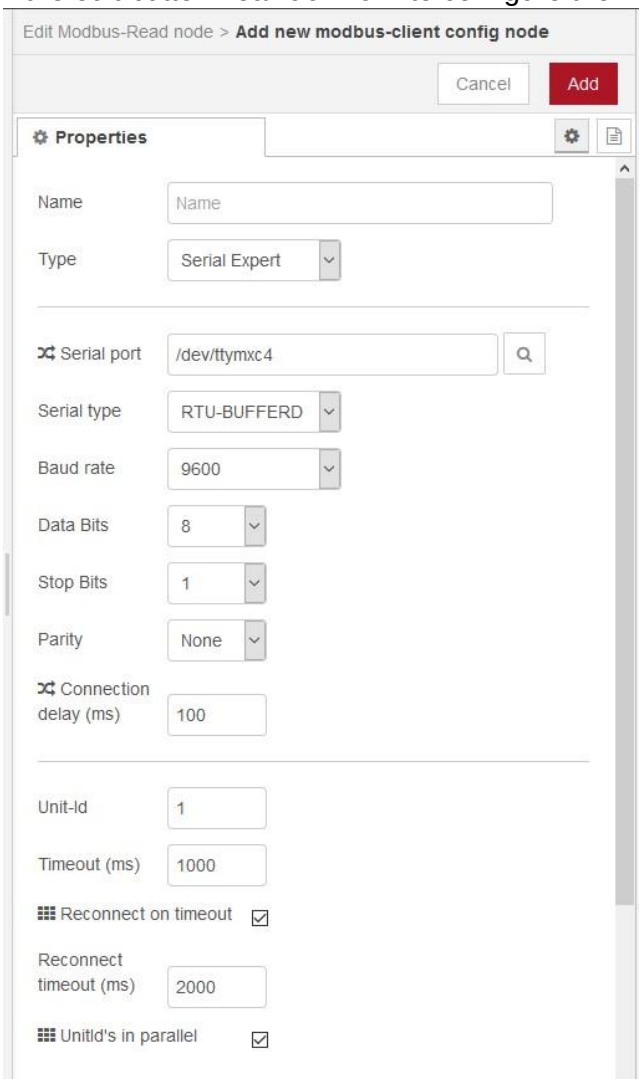
Do not try to install the node-red-contrib-modbus package from within Node-RED. This will fail and leave you with an inconsistent state.
Use our method to get a working version.

14.1.1 Creating a first flow

1. Open Node-Red in your browser.
2. Add Debug node.
3. Add Modbus read node.
4. Wire them.
5. Double click on Modbus read node to configure it.
6. Change the settings depending on the device you want to read out.



7. Click on the edit button near `Server` to configure the Modbus device.



8. Change `Type` to `Serial Expert`.
9. Change `Serial port` to `/dev/ttymxc4`
10. Make sure, that the `Serial Type` is `RTU-BUFFERED` and not `RTU`.
11. Deploy.

By clicking on `Debug`, messages fetched from the device should be seen.

14.2 Modbus command line tool

14.2.1 Installation

To install the modbus command line tool you have to install the ‘mcmodbus.ipk’ from the MC-Technologies update server.

14.3 Usage and Command lines

➢ Read register 0 on slave 1:
• mcmodbus -a 0

➢ Print debug information during execution:
• mcmodbus -v -a 0
• mcmodbus -vv -a 0

➢ Show help message:
• mcmodbus
• mcmodbus –h

➢ Read register 0 on slave 17:
• mcmodbus -s 17

➢ Set the output of slave 17 for the I/O Pins 4,5,6 to 1 0 1:
• mcmodbus -o wb - a 4 -s 17 1 0 1

➢ Use another device with a baud rate of 115200:
• mcmodbus -d /dev/ttyUSB10 -b 115200

➢ Setting digital output at address 0x34 to ON:
• mcmodbus -o wib -a 0x34 1
• mcmodbus -o wib -a 064 1
• mcmodbus -o wib -a 52 1

➢ Reading analog input at address 0x20 and 0x21:
• mcmodbus -o rir -a 0x20 -n 2

➢ Setting register 8 to 0x4563:
• mcmodbus -o wr -a 0x08 0x4563
• mcmodbus -a 8 -o wr 17763

15 MBus

The MC100 terminal/gateway can be used with an additional module to receive M-Bus packets from various sensors (heat cost allocators, electricity meters ...).

15.1 Setting up the MBUS module

The internal radio module for the wireless M-Bus is connected to a serial interface of the MC100 via a multiplexer.

Communication specifications

Technological feature	Description
Interface	/dev/ttymxc2
Speed	depending on the device, Probably 2400bd
Configuration	8 data bits, no parity, 1 stop bit

For MBUS the PMIC must activate the power supply of the M-Bus module.
This is done at start-up by the script in /etc/rc.d/s90pmic
The most common source of error for a non-functioning M-BUS is the lack of this script.

15.2 Using libmbus

On the MC100, libmbus and certain programs for communication with Mbus are installed. Those are:

- mbus-serial-request-data-multipli-reply
- mbus-serial-request-data
- mbus-serial-scan
- mbus-serial-scan-secondary
- mbus-serial-set-address
- mbus-serial-select-secondary
- mbus-serial-switch-baudrate

Using the parameter -h, short user information is shown for each of the programs and the parameters are explained.

Function	Command line
A list of all connected devices	mbus-serial-scan -b 2400 /dev/ttymxc2
Requesting data from device 10:	mbus-serial-request-data -b 2400 /dev/ttymxc2 10

16 1-wire

The 1-wire driver is loaded and started during the boot process. The driver provides the user with a command-option with 1-wire devices.

16.1 For v1.0.4 or older

16.1.1 Using 1-wire driver

After the 1-wire driver has been loaded, the user functions are located in the following folder:

/sys/class/1w/

To enter the folder using Command prompt, use the command:

```
root@MC100: ~#cat /sys/class/1w/
```

There are the following files which act as interface for the user:

- probe_bus
- read
- write
- reset

16.1.2 Function files

- probe_bus

This scans the 1-wire bus and registers all detected 1-wire devices.

The following command executes the scan:

```
root@MC100:~# echo 1 > /sys/class/1w/probe_bus
```

Please note that this command can take a while before completion.

- read

This reads the content of the 1-wire bus.

Example:

```
root@MC100:~# cat /sys/class/1w/read
288694aa0b00009dff
root@MC100:~
```

- write

This is used to write commands to the bus:
Please note the following format:

<expected bytes to read\>:<command\>

Example:

```
root@MC100:~#  echo 9:be > /sys/class/1w/write
root@MC100:~#
```

- reset

This file rests the bus.

Example:

```
root@MC100:~# echo 1 > /sys/class/1w/reset
root@MC100:~#
```

16.1.3 Select device:

A separate folder is created for each device detected during scanning.

Example:

```
root@MC100:~# ls /sys/class/1w/
100702A30308004e/  105B94A303080036/    probe_bus      read     reset    write
```

For selecting a specific device, the user has to write to the select file in the device folder.

Example:

```
root@MC100:~# echo 1 > /sys/class/1w/282E12AD0B0000b1/select
root@MC100:~#
```

16.2 For v1.0.5 (future release)

OWFS 1-Wire file system is an easy way to use the powerful 1-wire system. It is factory installed option in MC100 with version v1.0.5 or later.

16.2.1 Using OWFS file system

The user functions are located in the following folder:

/mnt/owfs/

1. To enter the folder using Command prompt, use the following command and press print:

```
root@MC100: ~# ls /mnt/owfs/
10.0702A3030800/  alarm/        settings/      statistics/     system/
10.5B94A3030800/  bus.0/        simultaneous/  structure/      uncached/
```

2. Choose the wanted sensor and use the following command.
   (In this example temperature sensor is used)

```
root@MC100:~# ls /mnt/owfs/10.0702A3030800/
address   crc8      family    latesttemp  power     r_id      scratchpad  temphigh  type
alias     errata    id        locator     r_address r_locator temperature templow
```

3. To read the contents, type:

```
root@MC100:~# ls /mnt/owfs/10.0702A3030800/temperature
24.5625
```

16.2.2 OWhttpd

OWFS provides a small webserver that shows the Dallas/Maxim 1-Wire bus attached to a serial port. The main page shows the devices found; you can then navigate to individual devices, and view/change their properties.

This can be enabled using the following steps:

1. Type the following command and press print:

```
root@MC100:~# vi /etc/config/owfs
```

2. Press “i” to enter the input mode, and change "option enabled" to 1.
3. Then press Escape to save the file, and type :wq and press Enter.
4. Then, type the following command and press print:

```
root@MC100:~# vi /etc/config/owserver
```

5. Change "option enabled" to 1.
6. Then press Escape to save the file, and type :wq and press Enter.
7. Then, type the following command and press print:

```
root@MC100:~# vi /etc/config/owhttpd
```

8. Change "option enabled" to 1.
9. Then press Escape to save the file, and type :wq and press Enter.

You can reach the owhttpd web interface with the same IP address as the standard web interface and the specification of the port (e.g. 3001)
Example with default address: http://192.168.2.1:3001

More information and tutorials can be found on:
(https://owfs.org/index_php_page_quickstart-guide.html)

17 Updating MC100 1.0.3rc10 to1.0.4

17.1 Using Command line

1. Replace the content of /etc/opkg/distfeed with the following:

```
src/gz openwrt_core https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/targets/imx6/generic/packages
src/gz openwrt_base https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-a9_neon/base
src/gz openwrt_luci https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-a9_neon/luci
src/gz openwrt_mctPackages
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortexa9_neon/mctPackages
src/gz openwrt_node https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-a9_neon/node
src/gz openwrt_packages https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortexa9_neon/packages
src/gz openwrt_routing https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortexa9_neon/routing
src/gz openwrt_telephony https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/package/arm_cortexa9_neon/telephony
```

When copying from this PDF, the line breaks should be fixed.

2. Now execute the following commands in the terminal of mc100:

```
root@MC100:~#  opkg update --no-check-certificate
root@MC100:~# opkg install --no-check-certificate ca-certificates
root@MC100:~# opkg install mc100-all
root@MC100:~# opkg list-upgradable
```

3. Execute the following command to update all packages on your system to the latest version:
4. Restart the device to get full effect.

Under no circumstances should you try and update the base-files package! Doing this will corrupt your system and make it unusable!

```
root@MC100:~# opkg list-upgradable | cut -f 1 -d ' ' | grep -v base- root@MC100:~# files > update-packages.txt
root@MC100:~# cat update-packages.txt | xargs –n 1 opkg u+pgrade
```

17.2 Using Luci Web Interface

17.2.1 Update Feeds

1. Open Luci and click on `System` and then `Software`
2. Click on `CONFIGURE OPKG`
3. replace the content of `opkg/distfeed` with the following:

```
src/gz openwrt_core
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/targets/imx6/generic/packages
src/gz openwrt_base
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-
a9_neon/base
src/gz openwrt_luci
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-
a9_neon/luci
src/gz openwrt_mctPackages
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortexa9_
neon/mctPackages
src/gz openwrt_node
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-
a9_neon/node
src/gz openwrt_packages
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortexa9_
neon/packages
src/gz openwrt_routing
https://downloads.mctechnologies.net/MC100/v2.4/1.0.4/packages/arm_cortexa9_
```

When copying from this PDF, the line breaks should be fixed.

17.2.2 Installing Certificates

1. Download the certificates IPK from **https://downloads.mc-technologies.net/MC100/v2.4/1.0.4/packages/arm_cortex-a9_neon/base/cacertificates_20190110-1_all.ipk**
2. Click on `Upload package.`
3. Select the package you just downloaded.
4. Confirm by clicking `Upload` and afterwards `Install.`

17.2.3 Updating the system

1. Click on Update Lists
2. Enter `mc100-all` into `Download and install package`
3. Click on install
4. Click on `Updates`
5. Install all except base-files
6. Restart your device to get full effect.

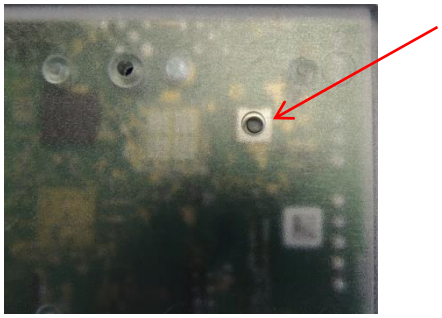
We recommend installing the `mc100-all` package as it will install all newly developed features for the mc100.

18 Factory reset

WARNING:
All customized setting will be reset to factory. Make sure you are directly connected to the MC100.

To reset the MC100 (approximatly 5 minutes):

- Locate the Reset button on the back of your MC100 device.



- Use the pointed end of a paperclip or similar object to press and hold the Reset button.
- Then while still holding the button, connect the power.
- After approximately 5 seconds the red LED will start flashing, and only the green LED 5 will be lit.
- After 2-3 minutes the red light will flash and then the green LED 5 will lit again.
- Wait for the device to fully reset and restart.
- At the end of the procedure the green LED 5 will turn off and the green LED 4 will start flashing

The default factory settings have been restored.

19 Maintenance, repair and troubleshooting

19.1 Maintenance

The product is maintenance-free and requires no special regular maintenance.

19.2 Troubleshooting

If a fault occurs during operation of the product and you need assistance, please contact MC Technologies support. You can reach our support department by e-mail at support@mc-technologies.net.

19.3 Repair

Send defective products with a detailed error description to:

MC Technologies GmbH, Kabelkamp 2, 30179 Hannover

Before shipping the device:

- Call our support team and ask for an RMA(Return to Manufacturer Authorisation) number.
- Remove possibly inserted SIM cards.
- Back up the configurations on the device and any other data stored on it.
- Back up any applications running on the device.

It is not permitted to open the product for repair work or modifications.

20 Disposal

20.1 Return of the old equipment

In accordance with WEEE regulations, the return and recycling of old MC Technologies equipment for our customers is regulated as follows:

Please send your old devices carriage paid to the following address:

MC Technologies GmbH
-Entsorgung-
Kabelkamp 2
30179 Hannover