User Manual

MC100 Terminal / Gateway
MC100 GPIO
MC100 SensT2
MC100 WMBUS
MC100 SensorBox

Version v2.0.2

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 2.0.0 and later.

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	**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. 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

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
- Phoenix connectors
- Wall mounting

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)

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)

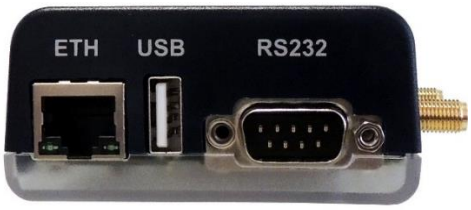
LED 4 shows the statues of the network connection:

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

7 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
LTE	LTE Antenna - SMA socket (main connection)
Div	LTE Diversity Antenna - SMA socket (MIMO)
GPS	GPS antenna – SMA socket

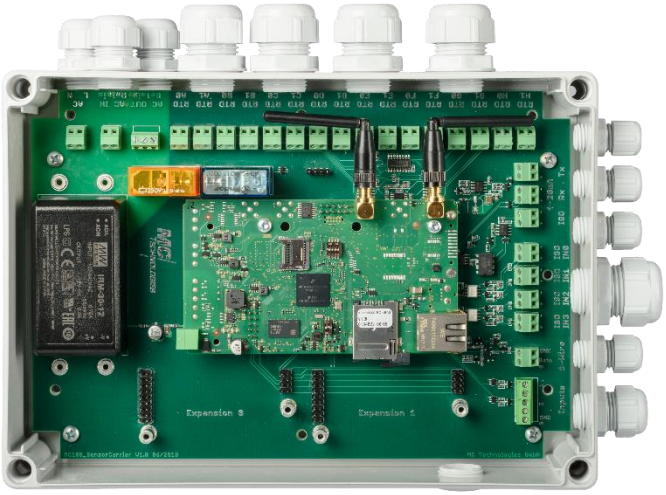
LED	Description
LED 1	Green, Power supply
LED 2	Green, Configurable.
LED 3	Red, Configurable.
LED 4	Long-time Off: Searching for connection. Long-time ON: Connected. Flashing quickly: Packet Data transfer.
LED 5	Blinking: Booting ON: Device is ready to be used.

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 Width 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

MC100
Mobile programmable gateway



Port	Description
Isolated RS485	1 serial RS485 interface (with galvanic isolation)
Isolated Inputs	4 digital inputs
Isolated Outputs	2 digital outputs (solid state relay)
PT100/PT1000	4 x 2 PT100/PT1000 inputs (8 x 2 optional)
4 – 20mA Outputs	1 analog 4 - 20mA outputs
4 – 20mA	1 analog inputs 4 to 20mA
4 – 20mA isolated	1 analog isolated input 4 to 20mA
1-Wire	1 one-wire or single-wire bus input
Input relay	1 220v input relay
Output relay	1 220v Output relay
AC power supply	85 to 264 V AC supply power

8 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 data 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.

8.1 Antenna

Mount the supplied antenna on connector "LTE" of the MC100. Check whether there is sufficient local network coverage from your mobile phone provider.
A2 is for MIMO usage.

8.2 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.

8.3 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).
During the boot sequence LED 5 flashes in heart beating pattern, after a short initialization phase (approx. 40 seconds) the device is ready for operation and LED 5 is always ON.

8.4 Configuration via web interface

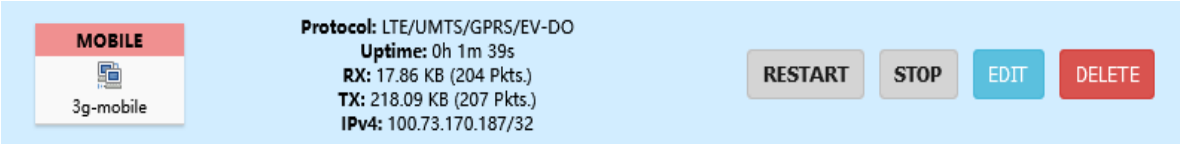
The MC100 can be configured to set mobile phone parameters (SIM pin, APN, etc.) via an integrated web interface. To do this, connect the Ethernet interface of your PC and the Ethernet interfaces of the MC100 (ETH) with a network cable.

- 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 is "root". Please check the product label on the device or the box for a default password. If there is none specified, no default password is set, and it can be left empty. For security reasons, click on "Go to Password Configuration" in the yellow window and change/set the "Password" here.

8.5 Establishing a Mobile Connection

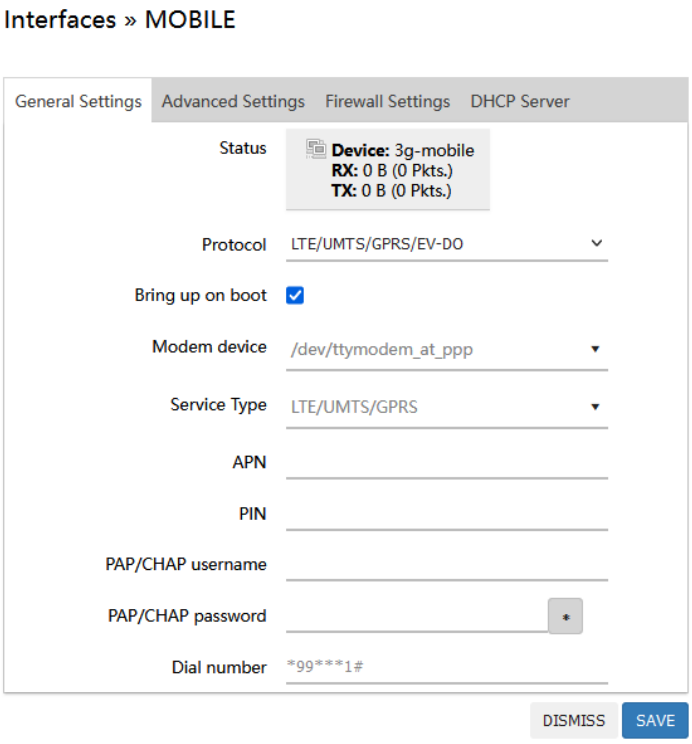
Click on "Network" in the left menu and then on "Interfaces".
Click on the "EDIT" button at the "MOBILE" interface.



Setting APN:

1. Set “Service Type” to: “LTE/UMTS/GPRS”.
2. Enter the access data, APN, user and password for data transmission in your mobile network.
3. Enter the PIN of your SIM card. Leave the entry field empty for a SIM card without a PIN.

You can obtain the data from your mobile phone provider. Many providers do not require



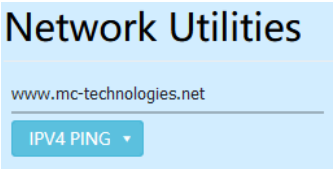
additional entries for user and password.
Confirm your entries by clicking on the "SAVE & APPLY" button.

Establishing a connection takes a few minutes. After that, transmission values for sent and received data should appear under Network -> Interfaces.
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

8.6 Online-Test

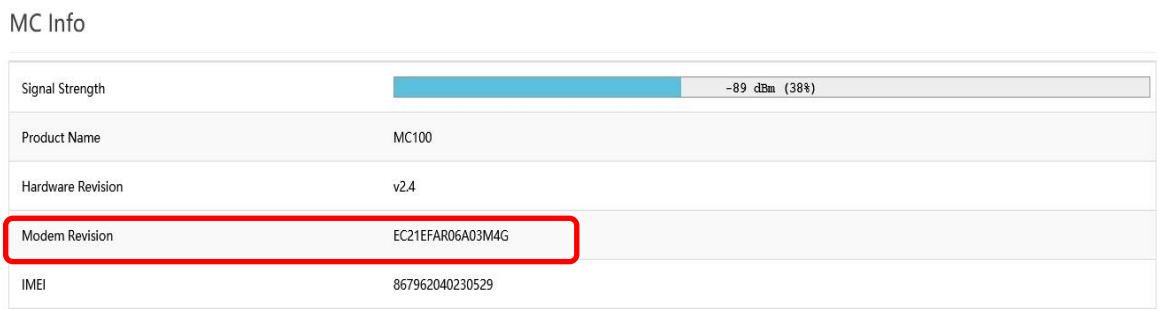
Navigate to the menu item "Network" and "Diagnostics". Click on the "IPV4 PING" button under "Network Utilities". The selected server on the Internet should answer with PING times.



8.7 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.

8.8 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).

8.9 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 with an SSH client. Most Linux distributions already have an SSH client installed. On Windows PuTTY can be used.
The following access data are required:
User: `root`
Password: Please check the product label for the default password (or none if there is no default password specified). If you've changed the password in LuCi, use this changed password here.
To secure your MC100 change the password by entering "passwd" under SSH.

8.10 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@MC-100:~# 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/ttymodem_reserved)
  -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
```

8.11 Change IP-address

8.11.1 Using Command line

Open the Command Prompt to login your device:

1. Open the configuration file using command: nano /etc/config/network

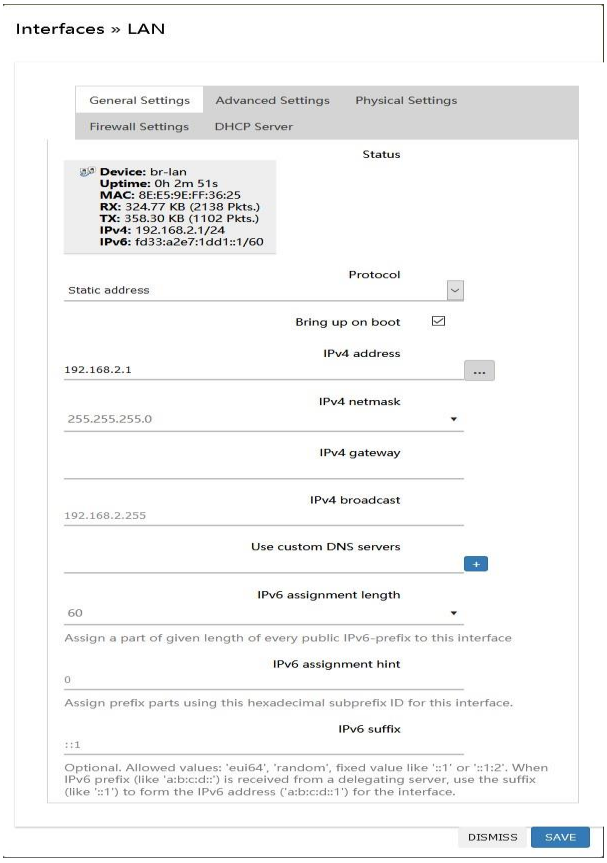
```
config interface 'lan'
        option type 'bridge'
        option ifname 'eth0'
        option proto 'static'
        option netmask '255.255.255.0'
        option ip6assign '60'
        option ipaddr '192.168.2.1'
```

2. Change the line "option ipaddr" in the interface "lan" to the corresponding address.
3. 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).
4. To save the changes, use the key combination Ctrl+o then Ctrl+x to exit.
5. Finally, type service network reload and press Enter.

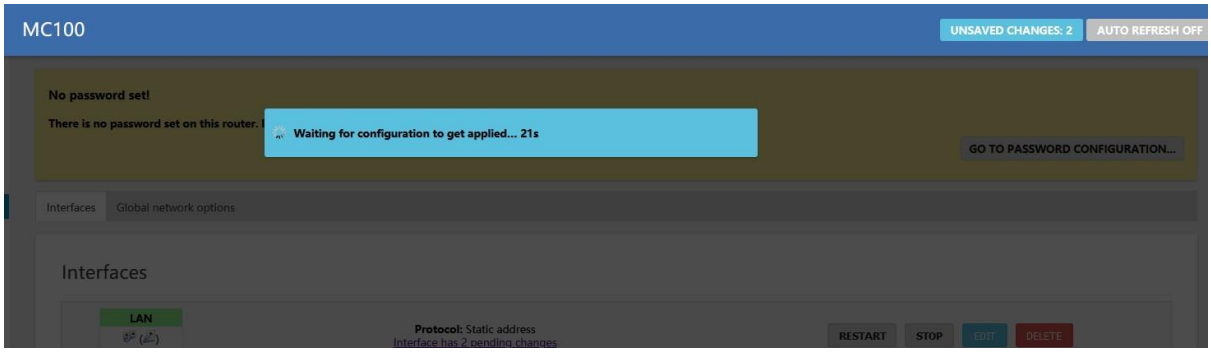
After a few seconds the MC100 has the new IP address. If necessary, the network connection at the host PC must be briefly disconnected and reconnected so that the PC can obtain a new IP.

8.11.2 Using Luci Web interface

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



3. Change IPv4 address to the wanted IP address and click on SAVE.
4. Click on SAVE&APPLY.



5. Login in to LuCi using the new IP address within the given 90 seconds to save the new settings.

8.12 Installation of MC100 to a motor vehicle

This chapter is only valid for MC100-variants which are sold and labelled with E1 marking. There are variants of the MC100 (general purpose) which do not contain the E1 type approval marking.

The installation of the MC100 in a motor vehicle itself, its additional devices such as cables and antennas as well as the connection to the vehicle electronics must be carried out by a qualified vehicle workshop or a specialist workshop.

Attention: Before installation of the MC100 in any car please take care that the serial number plate of the MC100 contains the E1 type approval marking **"E1*10R06/01*9546*00". Otherwise the operating license of the vehicle may expire**

9 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.
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.)

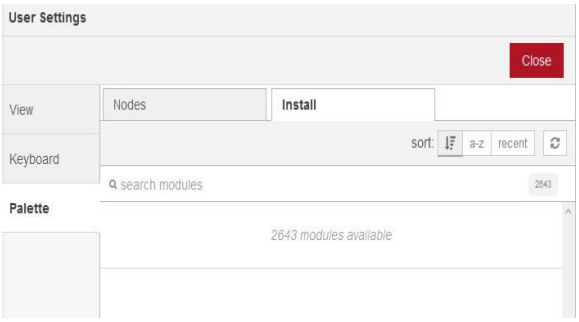
9.1.1 Enable the Node-Red installed in MC100:

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

9.1.2 Install extra modules:

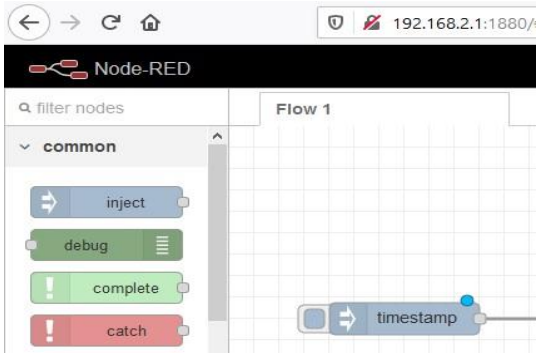
To install modules for Node-RED:

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”.



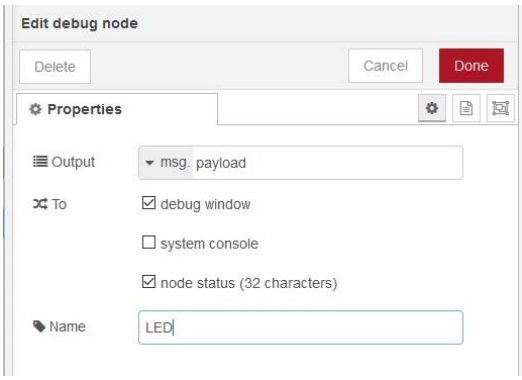
9.1.3 Add a node

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.



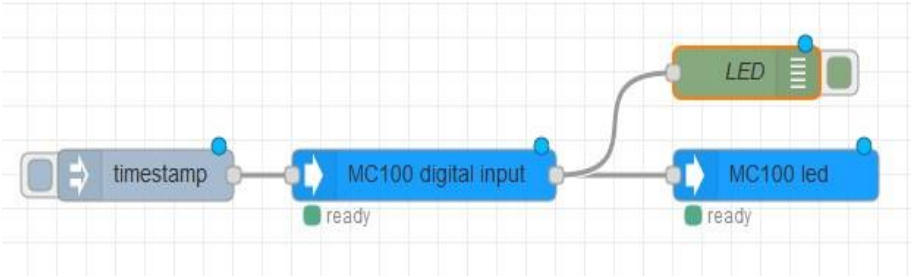
9.1.4 Add a debug node

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.



9.1.5 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.



9.1.6 Deploy

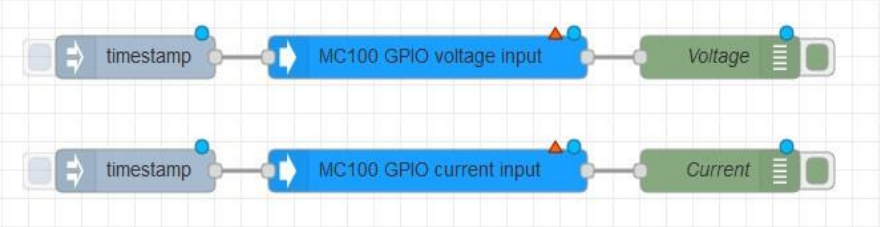
Now click on the Deploy button to deploy the flow to the server.

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.

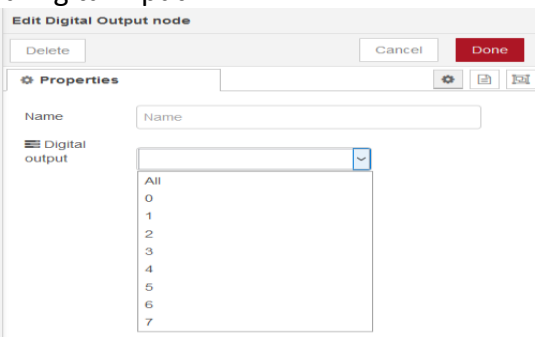


10.1.2 Digital Inputs

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

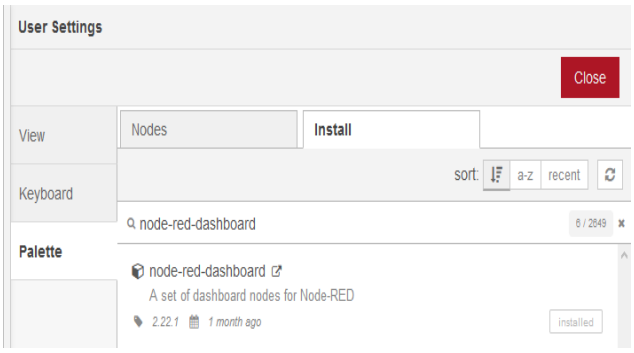
1. Add the Inject node.
2. Add the Digital output node.
3. Double click on MC100 GPIO digital output node to open the settings menu and choose a Digital input.



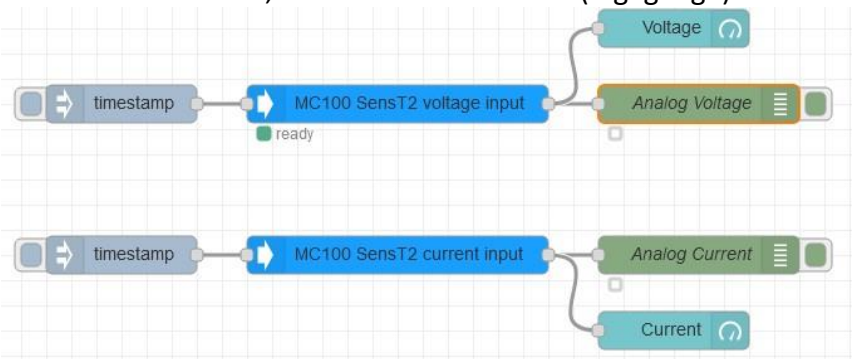
4. Add Debug node (change name if wanted).
5. Wire and Deploy.

10.1.4 Dashboard

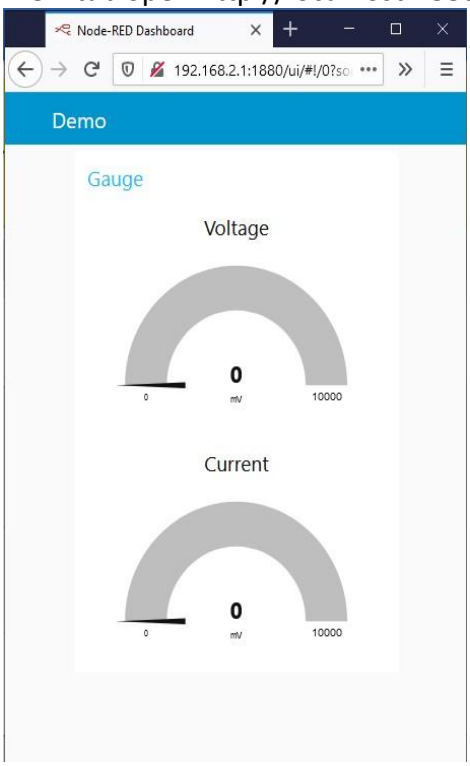
1. Use Menu - Manage palette to search for “node-red-dashboard” and click on install.



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.



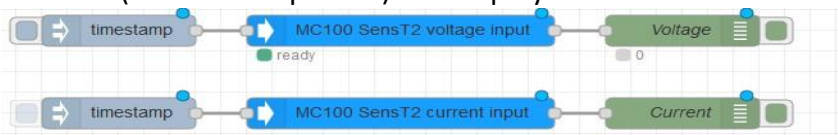
4. Double click on the nodes to change their Properties as wanted.
5. In a new tab open http://localhost:1880/ui (e.i. http://192.168.2.1:1880/ui)



10.2 MC100 SensT2

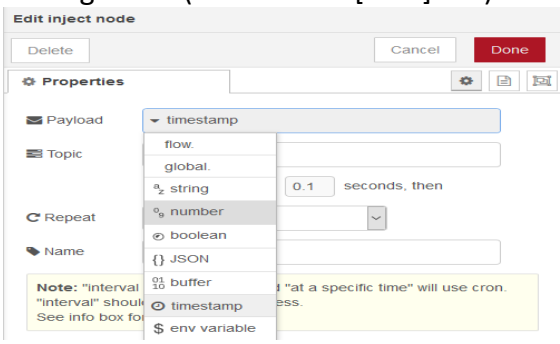
10.2.1 Analog Inputs

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 Output

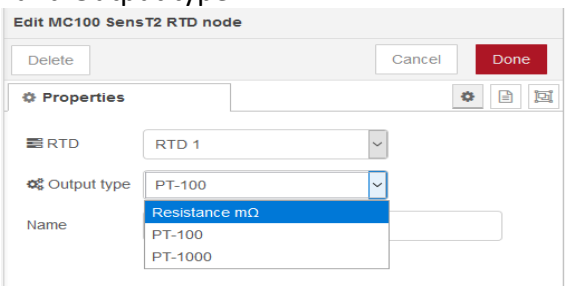
1. Add Inject node, double click on it and change the payload to Numbers and give the analog Value (= Current in [4-20] mA).



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.



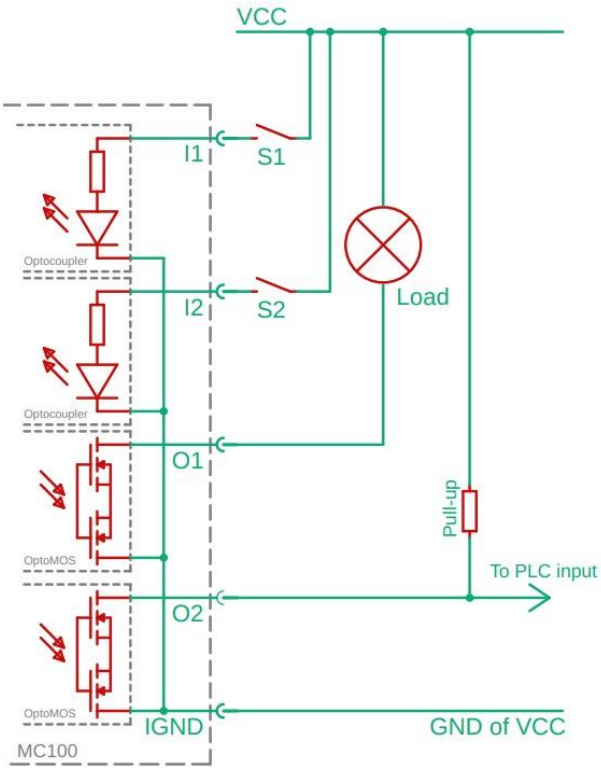
4. Add Debug node (Change name if wanted)

5. Wire and Deploy

11 Control 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

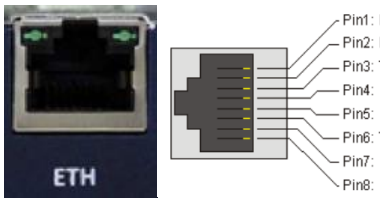


Inputs:
- Galvanically isolated inputs with respect to ground.
- High detection from ~5VDC.
- max. 30VDC.

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.

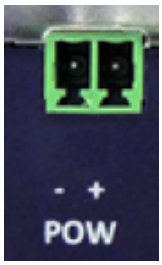
11.2 Ethernet-Interface (All)



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 .6V DC
I2	Digital input 2, DC voltage 0 to 30V, switching threshold approx .6V 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/class/gpio/mc100:in1/value
I2	1,0	x		cat /sys/class/gpio/mc100:in2/value

Example: Reading Input I1
Command: cat /sys/class/gpio/mc100:in1/value
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/class/gpio/mc100:out1/value
				echo 0 > /sys/class/gpio/mc100:out1/value
O2	1,0		x	echo 1 > /sys/class/gpio/mc100:out2/value
				echo 0 > /sys/class/gpio/mc100:out2/value

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

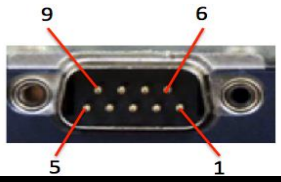
*Note: The RS485 interface is **not** galvanically isolated.*

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

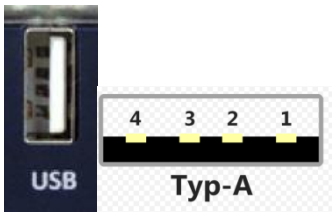
11.4.5 RS232-Interface



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

Note: A hardware handshake is, using RS232 interface, not possible.

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

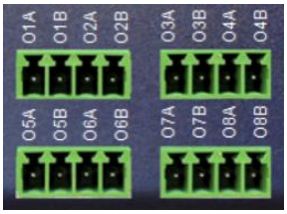
Port	Description
DGND1	Digital Ground1, electrically isolated to all (D)GND
I1 bis I4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 6V All input voltages with DGND1 as ground
VPU1	Not supported, please do not connect a signal
DGND2	Digital Ground 2, electrically isolated to all (D)GND
I1 bis I4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 6V All input voltages with DGND2 as ground
VPU2	Not supported, please do not connect a signal

Function: Reading the digital inputs I1 to I8

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



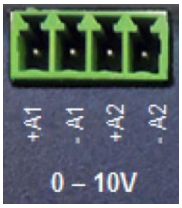
All outputs have an identical behaviour. The description here is only for one output (x = 1 to 8 MC100 GPIO).
All switching outputs are represented by solid state relays. OxA occupies one pin of the switch, OxB the second pin. All outputs are galvanically isolated. The maximum switching current is 300mA.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

Function: Switching the digital outputs

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.

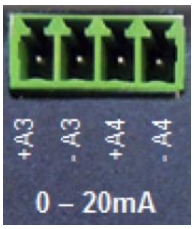
Function: Reading the voltage inputs

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

Port	Parameter	Read	Write	Path
+A3,-A3	Value	x		/sys/gpio_board/current_in/3
+A4,-A4	Value	x		/sys/gpio_board/current_in/4

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

Port	Parameter	Read	Write	Path
PW2	Prescaler {0,1023}		x	/sys/gpio_board/pwm/prescalar
	Pulse {0,65535}			/sys/gpio_board/pwm/1
PW1	Prescaler {0,1023}		x	/sys/gpio_board/pwm/prescalar
	Pulse {0,65535}			/sys/gpio_board/pwm/2
	Period*{0,65535}		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. 6V 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. 6V All input voltages with DGND2 as ground
VPU2	Not supported, please do not connect a signal

Function: Reading the digital inputs I1 to I8

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

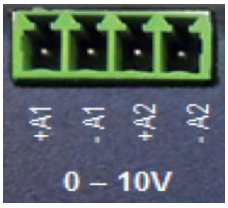
All outputs have an identical behaviour. The description here is only for one output (; x = 1 and 2 MC100 SensT2).
All switching outputs are represented by solid state relays. OxA occupies one pin of the switch, OxB the second pin. All outputs are galvanically isolated. The maximum switching current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

Function: Switching the digital outputs

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

Example: Switch on output O1A,O1B
Command: echo 1 > /sys/senst2_board/output/1
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.

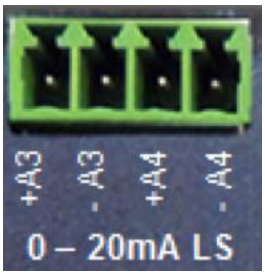
Function: Reading the voltage inputs

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 Supply of a current - current from the sensor

The connected sensor has its own power supply and a current output with a maximal 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

Port	Parameter	Read	Write	Path
+A3,-A3	Value	x		/sys/senst2_board/current_in/3
+A4,-A4	Value	x		/sys/senst2_board/current_in/4

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

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

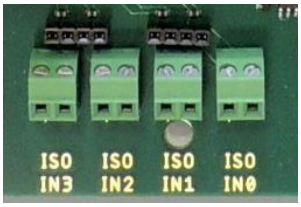
Function: Reading a resistance/temperature value

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 MC100 SensorBox

11.7.1 Digital Inputs



Port	Description
IN0 bis IN4	Digital inputs Input voltage: 0 to 30V Switching threshold: approx. 6V

Function: Reading the digital inputs I1 to I8

Port	Parameter	Read	Write	Path
IN0	1,0	x		/sys/sensor_carrier_board/inputs/isoInput0/value
IN1	1,0	x		/sys/sensor_carrier_board/inputs/isoInput1/value
IN2	1,0	x		/sys/sensor_carrier_board/inputs/isoInput2/value
IN3	1,0	x		/sys/sensor_carrier_board/inputs/isoInput3/value

Example: Read out input IN0
Command: cat /sys/sensor_carrier_board/inputs/isoInput0/value
Response: e.g. 1# or 0#

11.7.2 Digital Outputs

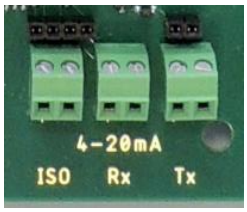
All outputs have an identical behaviour.
All switching outputs are represented by solid state relays. All outputs are galvanically isolated. The maximum switching current is 300mA. The maximum applied voltage must not exceed 30V.

Function: Switching the digital outputs

Port	Parameter	Read	Write	Path
ISO OUT1	1,0		X	/sys/sensor_carrier_board/outputs/output1/value
ISO OUT2	1,0		X	/sys/sensor_carrier_board/outputs/output2/value

Example: Switch on output ISO OUT1
Command: echo 1 > /sys/sensor_carrier_board/outputs/output1/value
Response: #
Example: Value = 6400 corresponds to 6.4V

11.7.3 Current input 0 - 20mA Rx / ISO



The analog current inputs of the MC100 SensorBox 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

11.7.3.1 Supply of a current - current from the sensor

The connected sensor has its own power supply and a current output with a current of 20mA.
The sensor is connected to + and – of Rx.

Port	Description
+	Positive connection input, current output of the sensor
-	Negative connection Input, sensor ground

11.7.3.2 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 SensorBox. A voltage is applied to the terminals + and – of 4-20 mA ISO for this purpose. Connect the sensor to + and – of 4-20 mA ISO.

Function: Reading the current inputs

Port	Parameter	Read	Write	Path
+,- Rx	Value	x		/sys/sensor_carrier_board/adcinputs/adcinput1/value
+,- ISO	Value	x		/sys/sensor_carrier_board/adcinputs/adcinput0/value

Example: Read current at ADC input +,- Rx
Command: cat /sys/sensor_carrier_board/adcinputs/adcinput1/value
Response: e.g. 1500

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

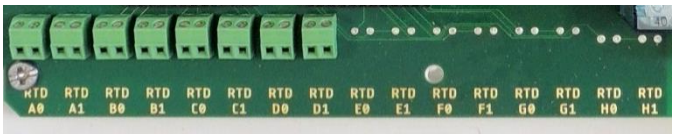
11.7.4 Current outputs 0 - 20mA Tx

Function: Output of a current (**v1.0.8 and older**)

Port	Parameter	Read	Write	Path
+,- Tx	Value		x	/sys/sensor_carrier_board/adcoutputs/adcoutput/value

Example: Output of 6mA at OI1
Command: echo 6000 > /sys/sensor_carrier_board/adcoutputs/adcoutput/value
Response: none

11.7.5 PT100 / PT1000 Inputs



Note: RTD inputs works only as pair (A0A1, B0B1, …). If A0 for example is to be used and A1 not, then A1 need to wire bridged.

Function: Reading a resistance/ temperature value (**v1.0.8 and older**)

Port	Parameter	Read	Write	Path
RTD A0	Value	x		/sys/sensor_carrier_board/rtd/rtd0/value
RTD A1	Value	x		/sys/sensor_carrier_board/rtd/rtd1/value
RTD B0	Value	x		/sys/sensor_carrier_board/rtd/rtd2/value
RTD B1	Value	x		/sys/sensor_carrier_board/rtd/rtd3/value
RTD C0	Value	x		/sys/sensor_carrier_board/rtd/rtd4/value
RTD C1	Value	x		/sys/sensor_carrier_board/rtd/rtd5/value
RTD D0	Value	x		/sys/sensor_carrier_board/rtd/rtd6/value
RTD D1	Value	x		/sys/sensor_carrier_board/rtd/rtd7/value
RTD E0	Value	x		/sys/sensor_carrier_board/rtd/rtd8/value
RTD E1	Value	x		/sys/sensor_carrier_board/rtd/rtd9/value
RTD F0	Value	x		/sys/sensor_carrier_board/rtd/rtd10/value
RTD F1	Value	x		/sys/sensor_carrier_board/rtd/rtd11/value
RTD G0	Value	x		/sys/sensor_carrier_board/rtd/rtd12/value
RTD G1	Value	x		/sys/sensor_carrier_board/rtd/rtd13/value
RTD H0	Value	x		/sys/sensor_carrier_board/rtd/rtd14/value
RTD H1	Value	x		/sys/sensor_carrier_board/rtd/rtd15/value

Example: Read the value at RTD A0
Command: cat /sys/sensor_carrier_board/rtd/rtd0/value
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

11.7.6 AC Out Relais

Port	Description
NC	Normally Close to COM
COM	Switch
NO	Normally Open to COM

Function: Switch ON, OFF (**v1.0.8 and older**)

Port	Parameter	Read	Write	Path
COM, NO	1,0		x	/sys/sensor_carrier_board/ac/acout/value

Example: Switch ON Relay
Command: echo 1 > /sys/sensor_carrier_board/ac/acout/value
Response: #

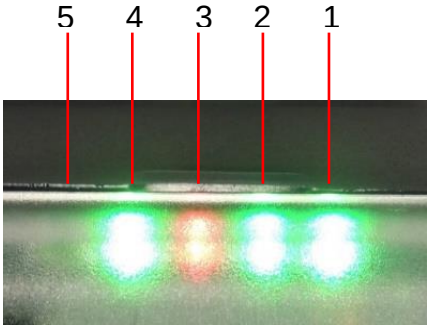
11.7.7 AC IN Relais

Function: Switching the digital outputs (**v1.0.8 and older**)

Port	Parameter	Read	Write	Path
AC IN	1,0		X	/sys/sensor_carrier_board/ac/acin/value

Example: Read out AC input status
Command: cat /sys/sensor_carrier_board/ac/acin/value
Response: 1,0

11.8 LEDs



LED		Description
LED 1	Power	Green, Power supply
LED 2	Info	Green, Configurable.
LED 3	warn	Red, Configurable.
LED 4	Modem	Long-time Off: Searching for connection. Long-time ON: Connected. Flashing quickly: Packet Data transfer.
LED 5	status	Green, Configurable. Default: Blinking: Booting ON: Device is ready to be used.

MC100 has three LEDs that can be configured: info, status and warn.
To configure the LEDS:

1. Login to MC100 web interface.
2. Click on System -> LED Configuration.
3. Click on “ADD LED ACTION”.

All possible LED configuration options:

Name: Name of the LED configuration.

LED Name: Colour and name of the LED.

Default state of the LED: On/Off.

Trigger: One of the various triggers can be assigned to an LED to changes its states.

Possible values are shown in a table below.

Trigger type	Description
heartbeat	Simulating actual heart beats
Always on	LED always stays on
Always off	LED always stays off
Custome flash interval	Blinking according to predefined timer pattern
netdev	Flashes according to link status and send/receive activity

Device: Network interface which is going to be tracked.

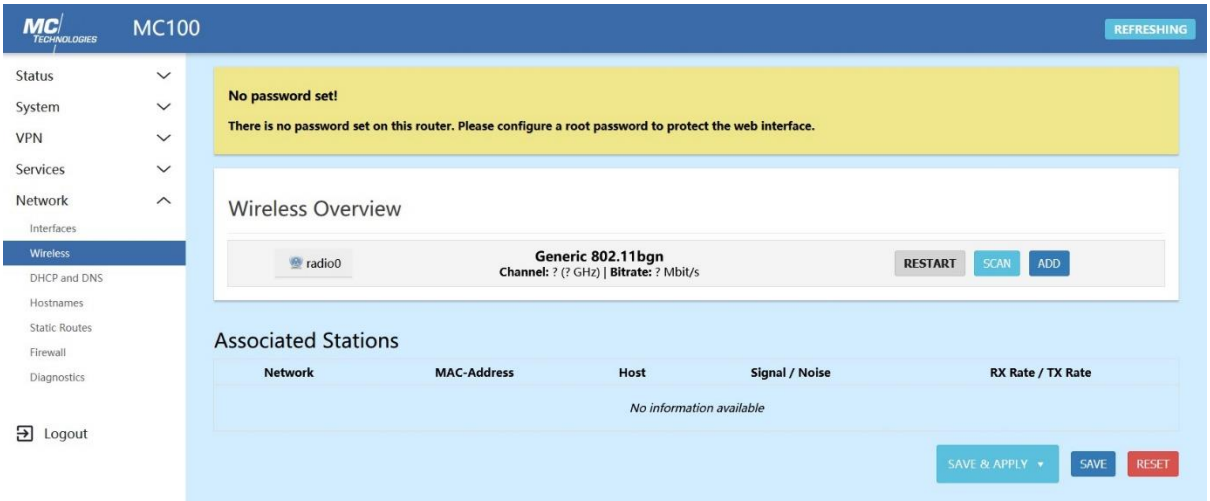
12 Setting Up Wifi

This section is only for MC100 devices that support wireless connection (Wifi).

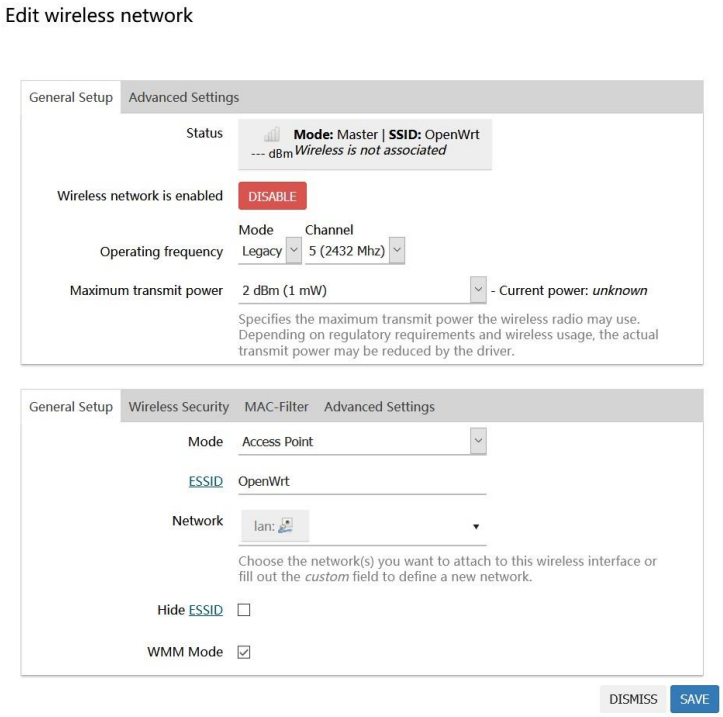
12.1.1 MC100 in Access Point mode

MC100 can be used as an access point for other devices to wirelessly connect to it.
To set MC100 in Access point mode:

1. In LuCI, click on Network then on Wireless.
2. Under Wireless Overview, click on ADD.



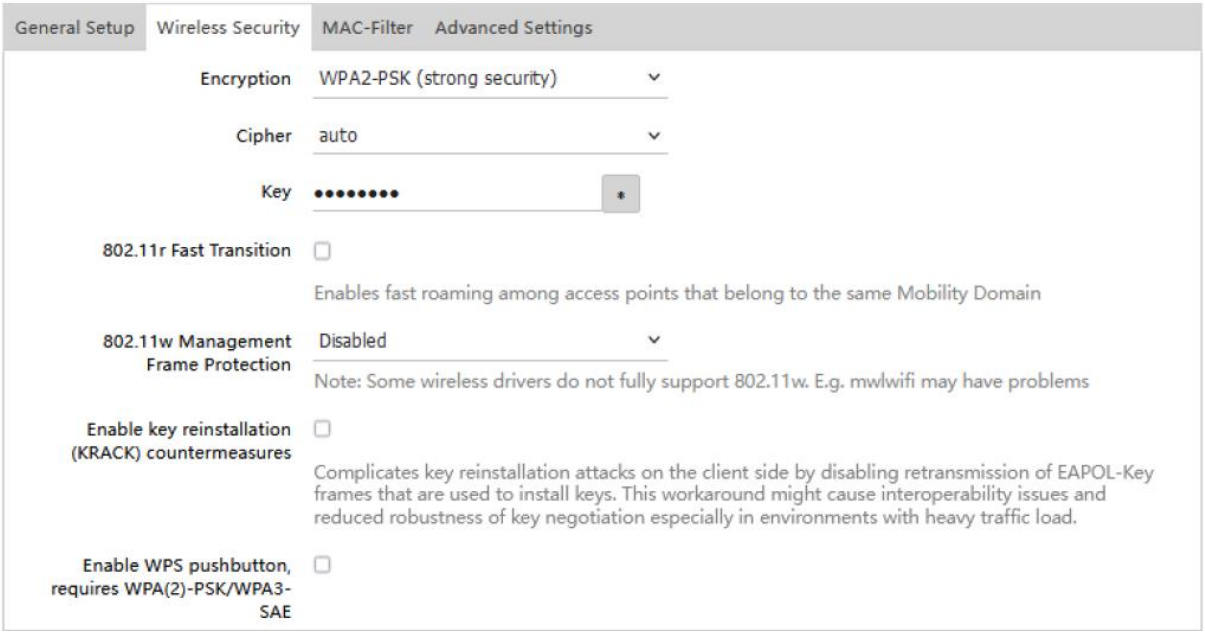
3. In the tab General Setup, set Operating frequency to Legacy.
4. In the tab General Setup, enter an ESSID to become the name of your Wi-Fi



network.

5. In the tab Wireless Security, select an Encryption method (e.g. “WPA2-PSK”, recommended for home/small office networks).

6. In the tab Wireless Security, select a Cipher (e.g. “auto” is recommended).
7. In the tab Wireless Security, enter a Key, a secret phrase which client must use to connect to your Wi-Fi network. NB: With WPA2-PSK, the Key must be at least eight



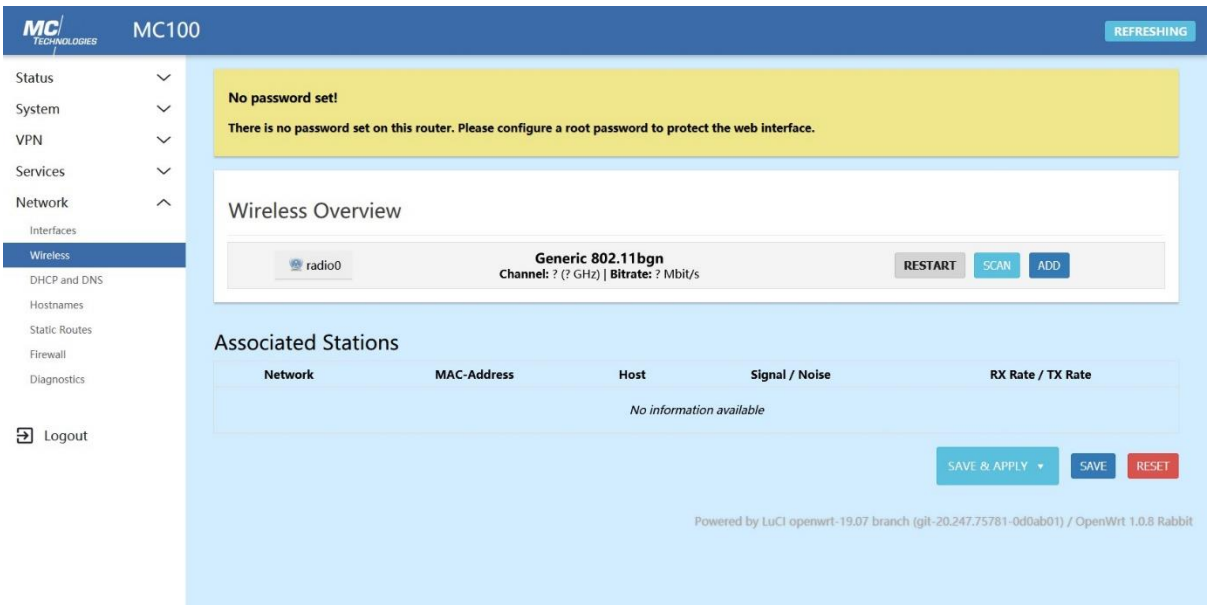
(8) characters.

8. If needed, configure further settings according to your needs. Other default settings ordinarily correct for general usage.
9. Click on SAVE, then SAVE&APPLY

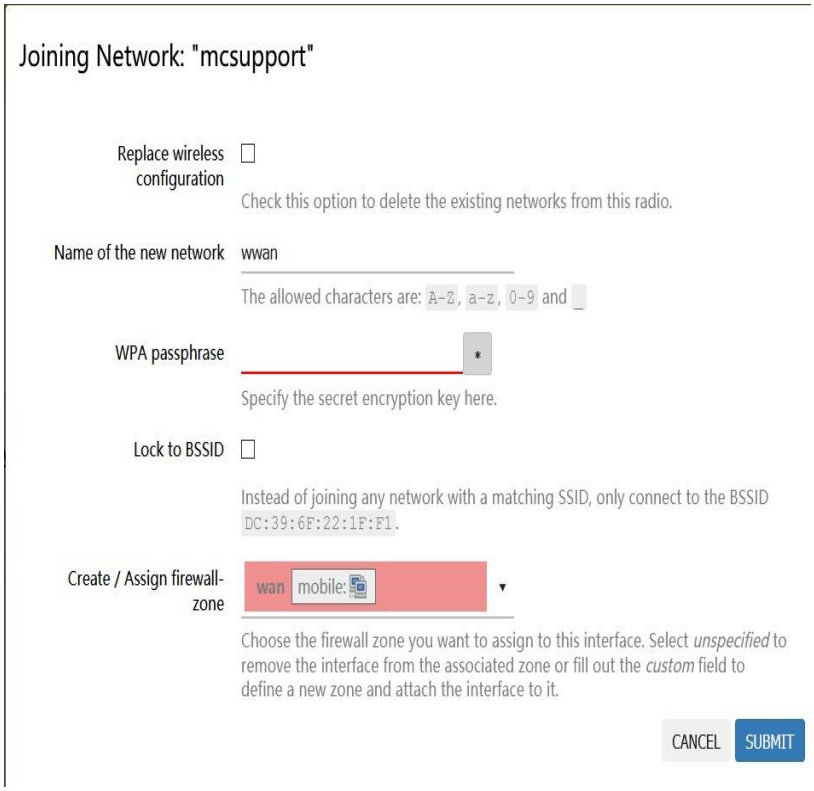
12.1.2 MC100 as Wifi Client

MC100 can be used as a Wifi client and connect to access point in its range.
To set MC100 in Clinet mode:

1. In LuCI, click on Network then on Wireless.
2. Under Wireless Overview, click on SCAN.



3. Choose wanted network and click on JOIN.
4. Type wifi password under WPA passphrase.



5. Click on Submit, then SAVE.
6. SAVE&APPLY.

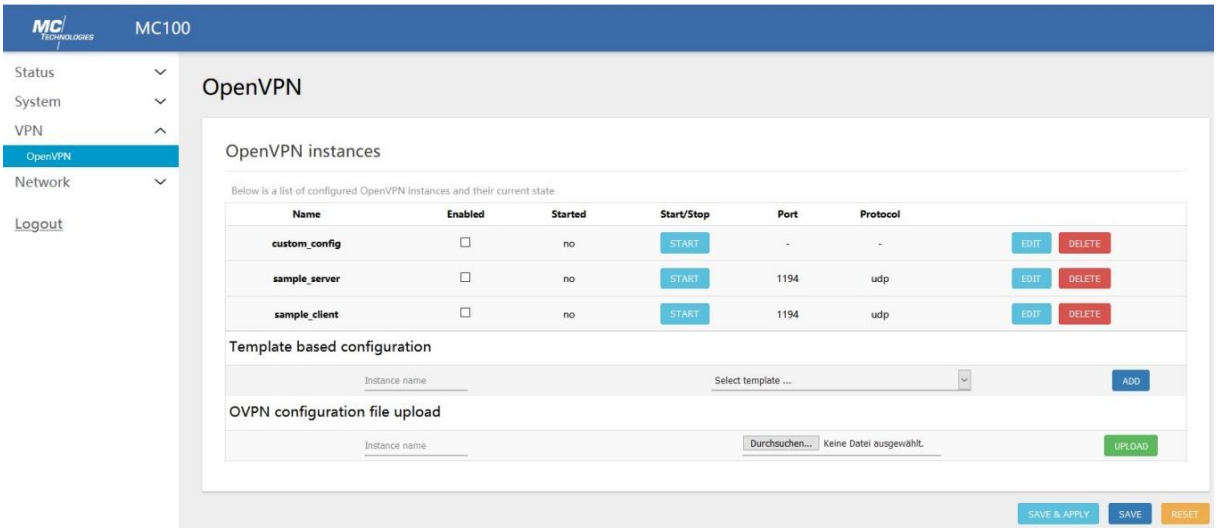
13 OpenVPN

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)

13.1 Client Configuration

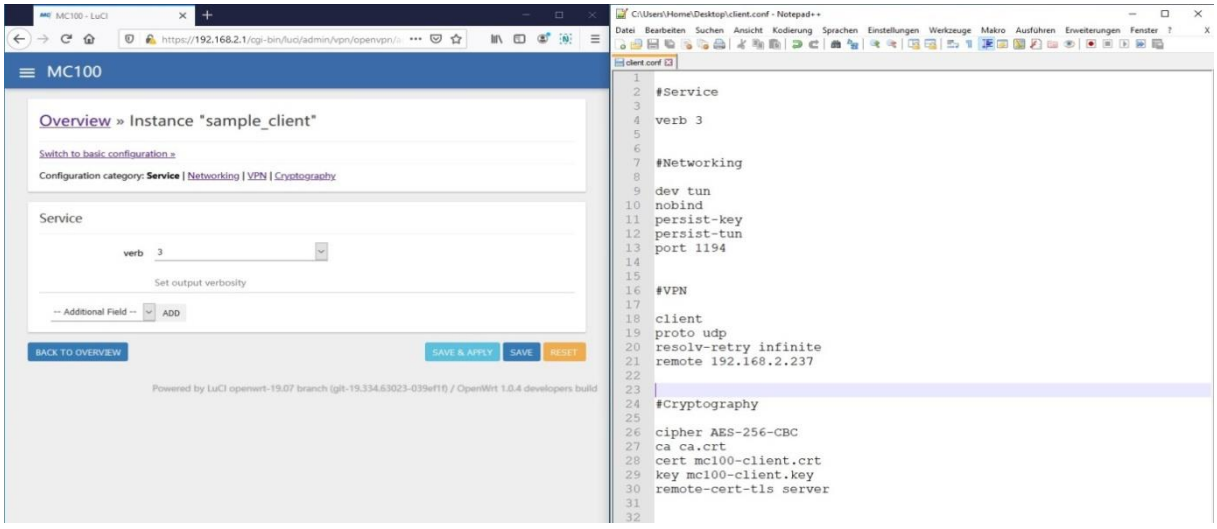
1. After logging to MC100 Luci interface using the IP address (default 192.168.2.1), click on „OpenVPN“ under „VPN“ tab.



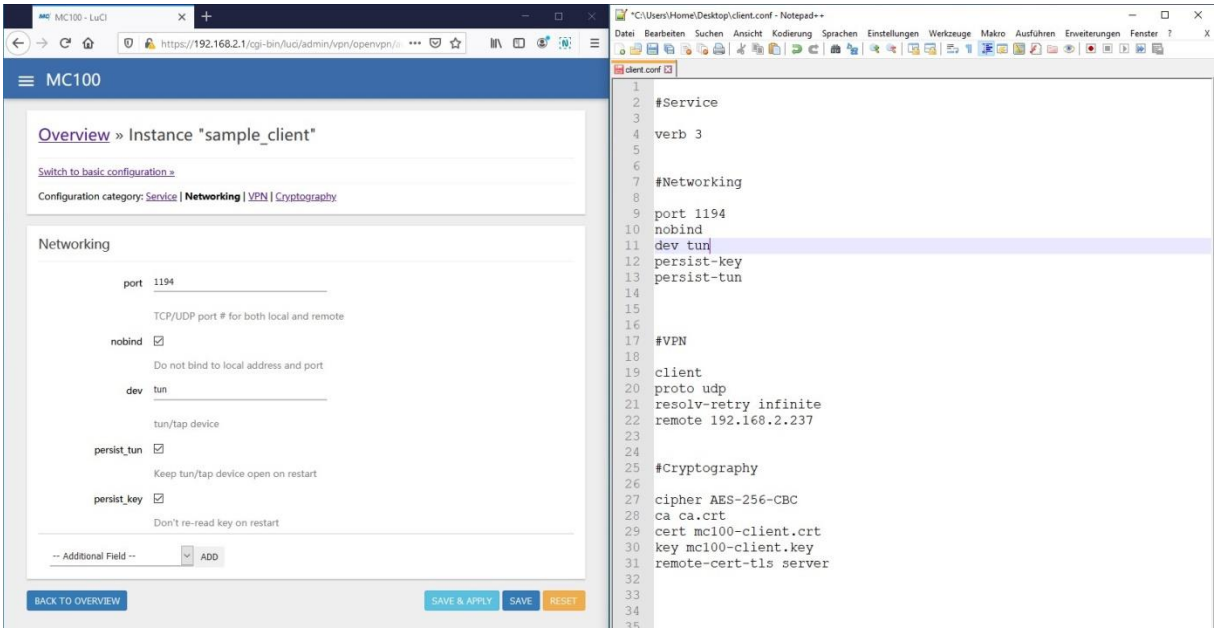
2. Click on „EDIT“ at „sample_client“ tab.
3. Click on „Switch to advanced configuration“.
4. Edit the settings as the one in the client.conf file you have.

Example:

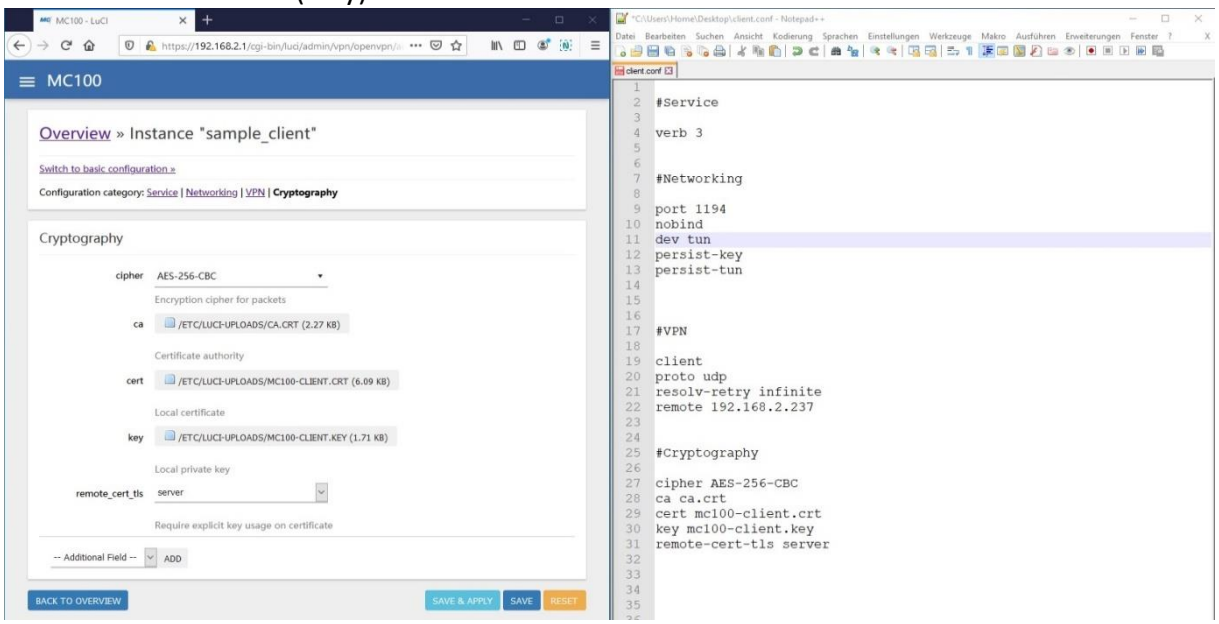
1. Under Service, Change „verb“ (verbosity) same as given in the .conf file (here it is 3).



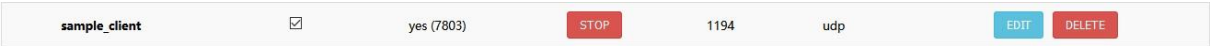
2. Under Networking, change every setting as same as given in the .conf file.



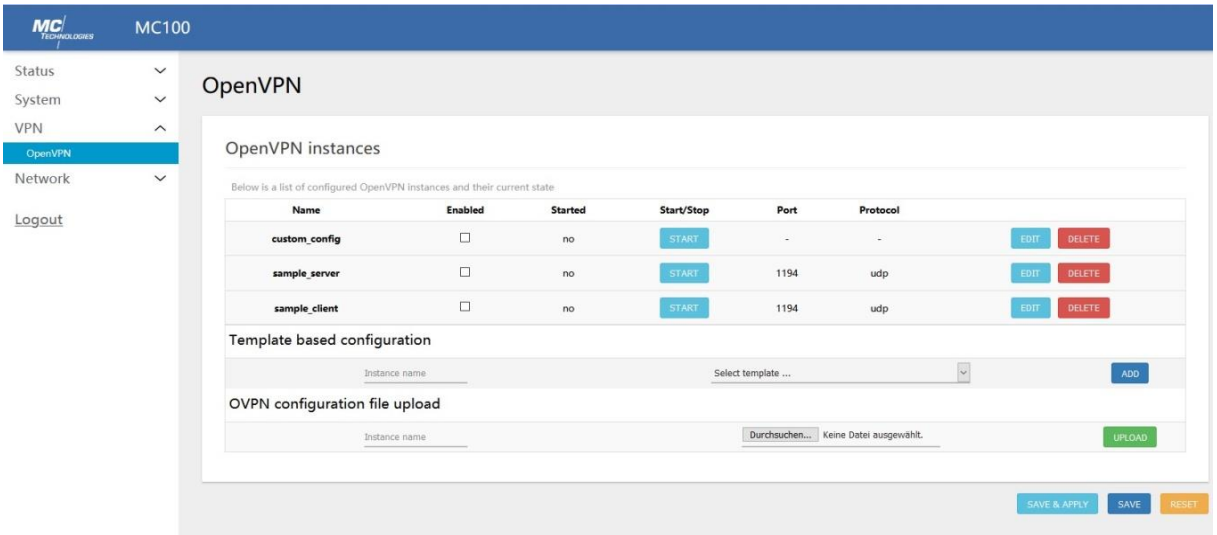
3. Same for VPN section.
4. Under Cryptography:
   a. Change the cipher type to the one given in the .conf file.
   b. In the path /ETC/LUCI-UPLOADS/ upload the Certificate authority „ca“ file(.CRT), local certificate „cert“ (.CRT), and the local private key „key“ (.key).



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



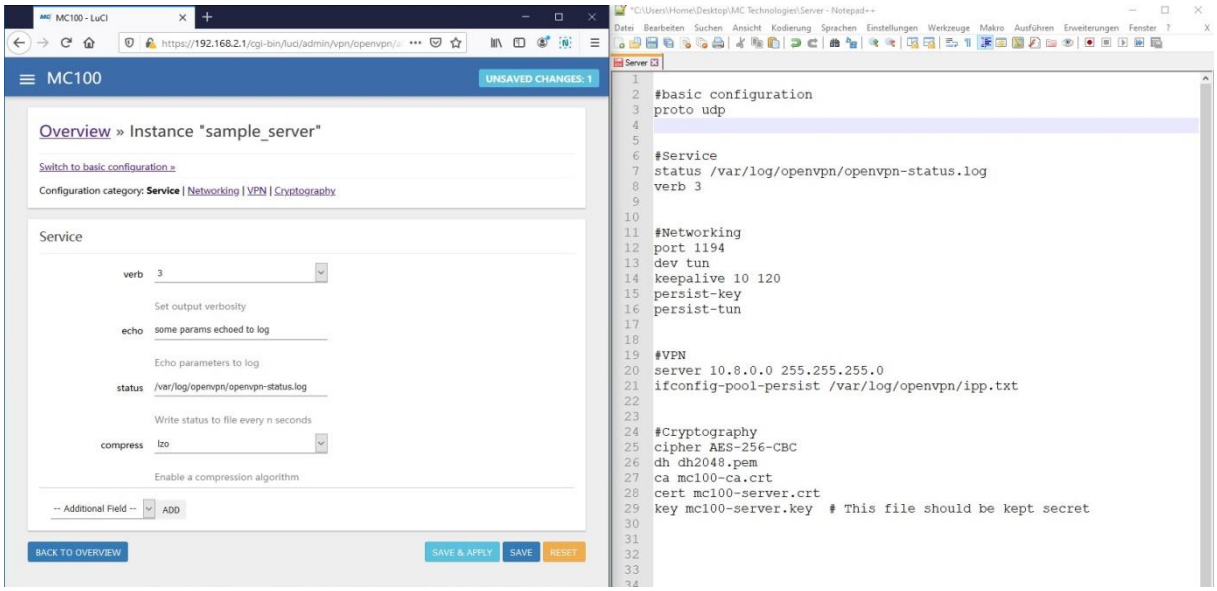
13.2 Server Configuration



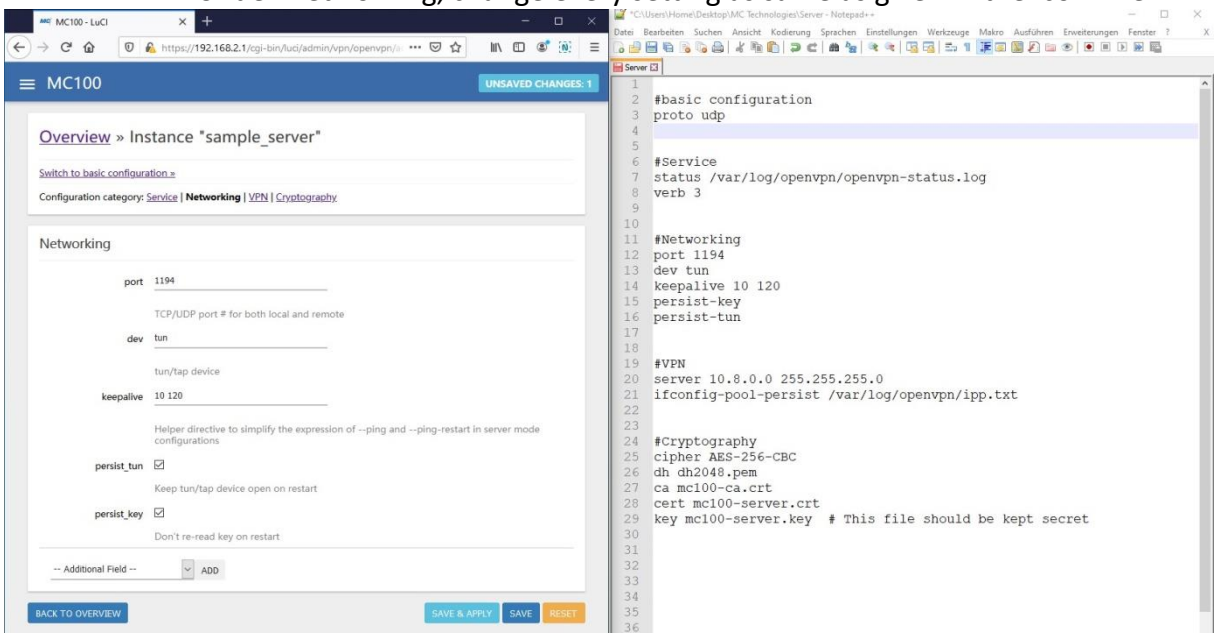
1. After logging to MC100 Luci interface using the IP address (default 192.168.2.1), click on “OpenVPN” under “VPN” tab.
2. Click on “EDIT” at “sample_server” tab.
3. Click on “Switch to advanced configuration”.
4. Edit the settings as the one in the server.conf file you have.

Example:

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



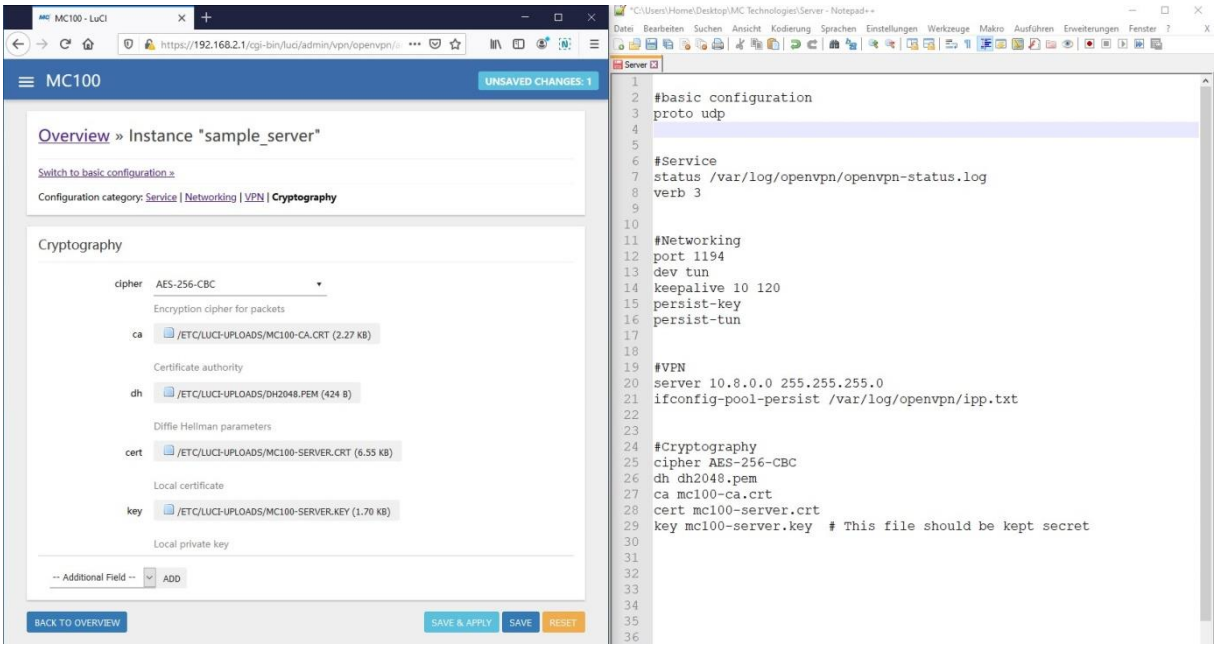
2. Under Networking, change every setting as same as given in the .conf file.



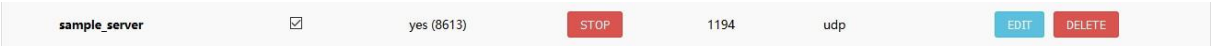
3. Same for VPN section.

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”.

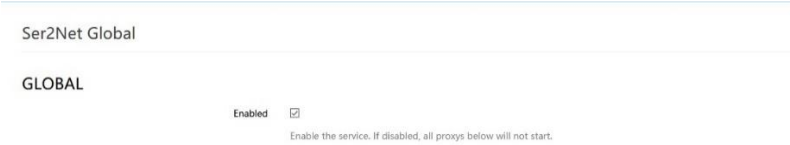


14 Ser2net

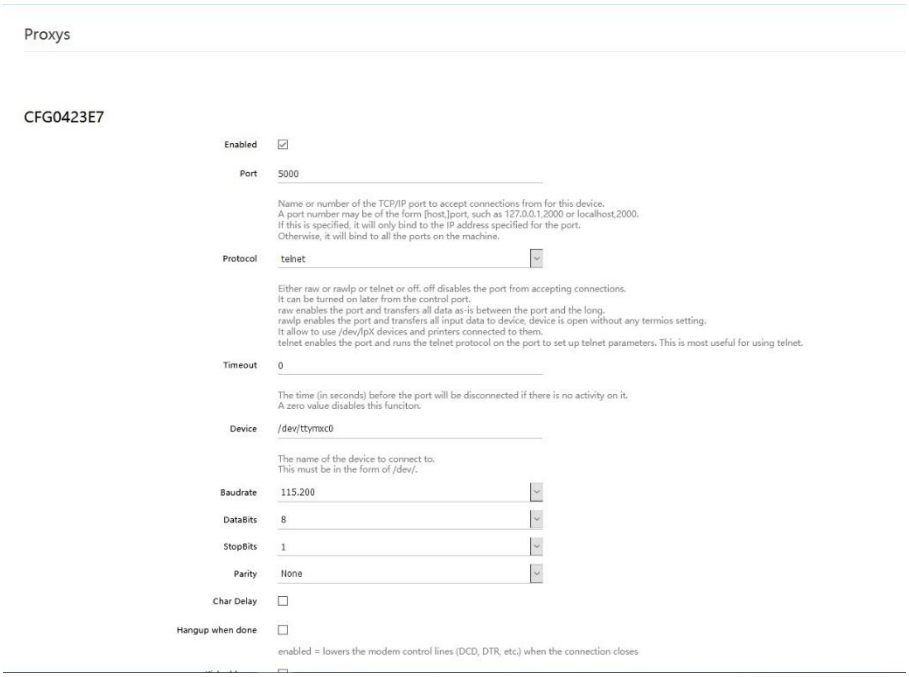
Note: A hardware handshake is, using RS232 interface, not possible.

Ser2net can be enabled and configured using LuCI web Interface:

1. In LuCI web Interface, Click on "Services" in the left menu and then on "Ser2net".
2. Enable “**GLOBAL**”.



3. Enable one of the “Proxys”, ex. CFG0423E7.
4. Set wanted “Port” and type of “Protocol”.
5. Type wanted device (ex. /dev/ttymcx0 for Serial port RS232).



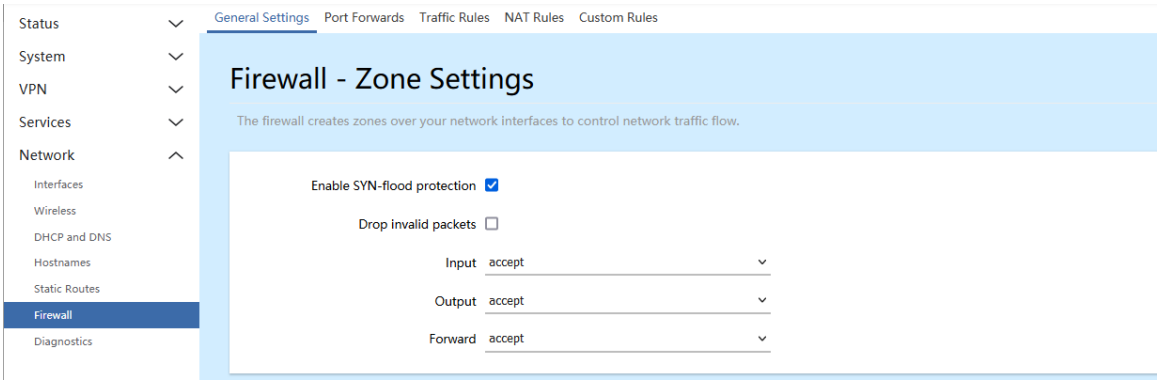
6. Click on “SAVE” then “SAVE&APPLY”.

15 Firewall

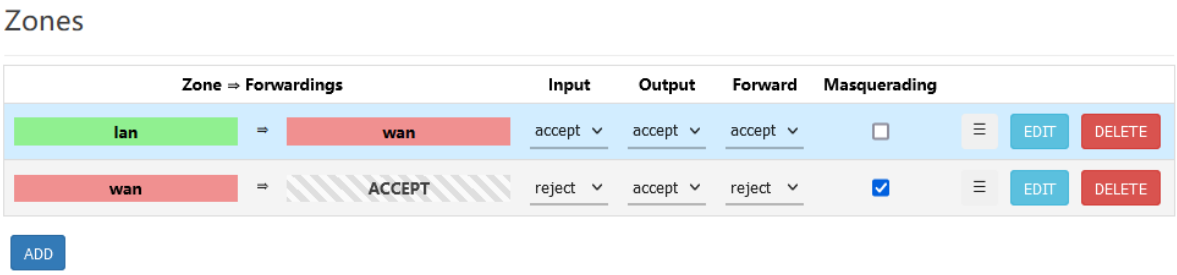
This subsection is divided into four categories: general settings, port forwards, traffic rules and custom rules.

15.1 General settings

General Settings for firewall can be changed in *General Settings* screen. These settings are defined as follows:



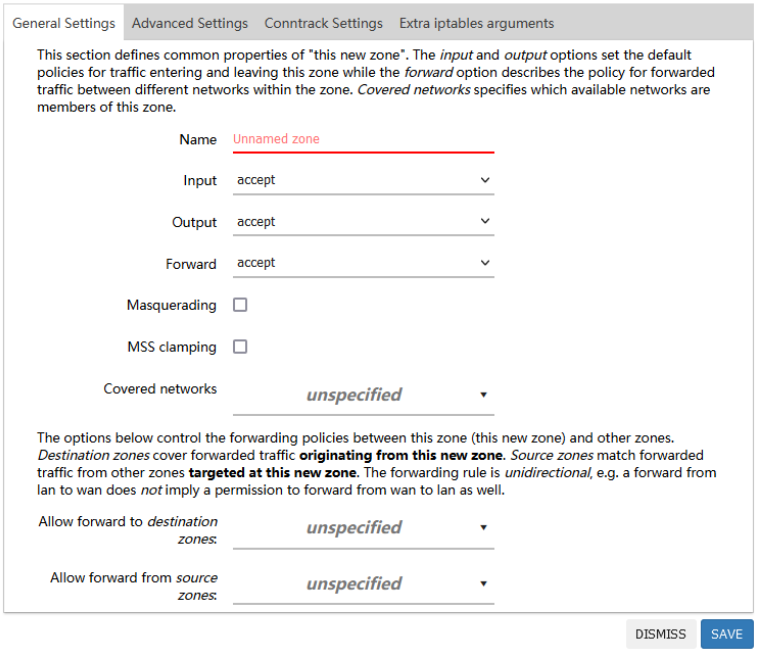
Input: All incoming packets can be: accepted, rejected, dropped.
Output: All outgoing packets can be: accepted, rejected, dropped.
Forward: All packets being sent to another device can be: accepted, rejected, dropped.
Additional zones for firewall can be created, edited or deleted.



Zone => Forwardings: Defines zones and their traffic flow.
Input: All incoming packets can be: accepted, rejected, dropped.
Output: All outgoing packets can be: accepted, rejected, dropped.
Forward: All packets being sent to another device can be: accepted, rejected, dropped.
Masquerading: Allows one or more devices in a zones network without assigned IP addresses to communicate with the Internet.

Additional actions can be performed with zones: add, edit, delete.
Common properties of newly created or edited zones can be edited in this panel.

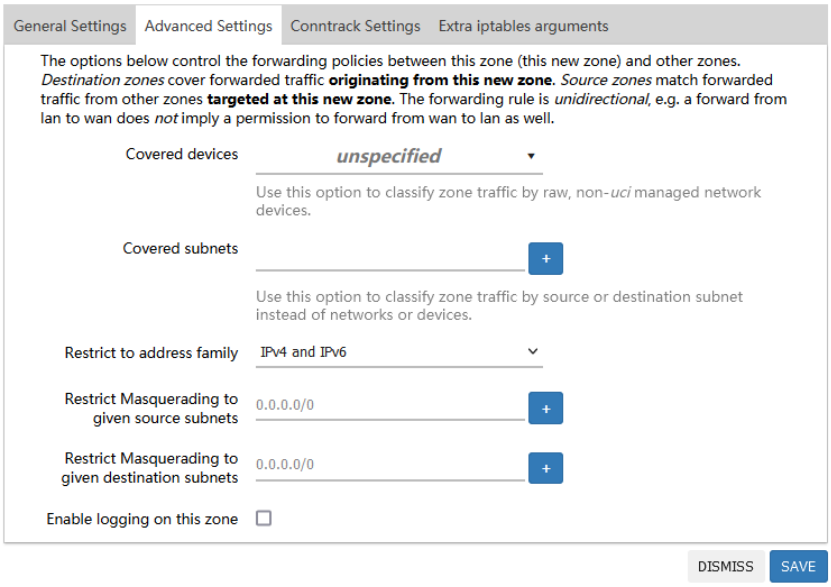
Firewall - Zone Settings



The input and output options set the default policies for traffic entering and leaving this zone while the forward option describes the policy for forwarded traffic between different networks within the zone. Covered networks specify which available networks are members of this zone.
MSS clamping: Change the maximum segment size (MSS) of all TCP connections passing through this zone with MTU lower than the Ethernet default of 1500.

Advanced settings of new created or edited zone.

Firewall - Zone Settings



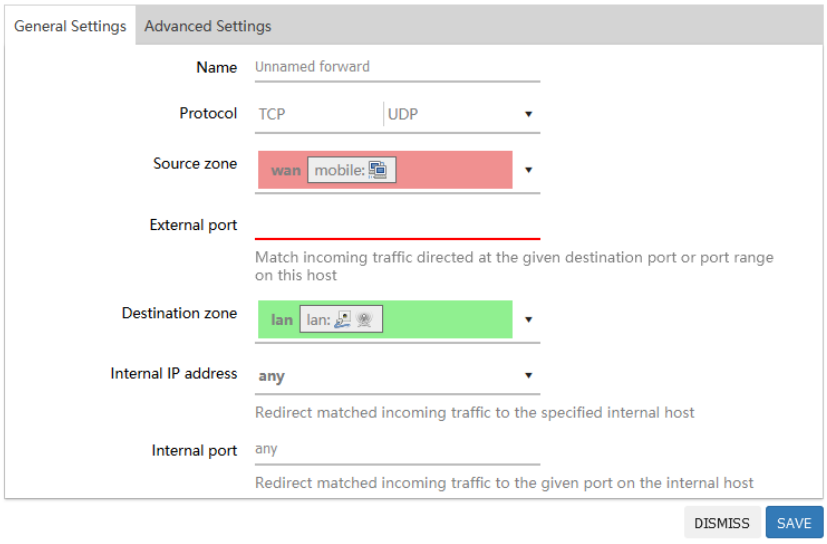
Restrict to address family option defines to what IP families the zone belongs to IPv4, IPv6 or both. Restrict masquerading to given source/destination subnets defines one or more

subnets for which the masquerading option is applied to. Connection tracking and logging options enable additional information gathering on the zone.
Controls of the forwarding policies between new/edited zone and other zones. Destination zones cover forwarded traffic originating from the new/edited zone. Source zones match forwarded traffic from other zones targeted at the new/edited zone. The forwarding rule is unidirectional, e.g. a forward from LAN to WAN does not imply a permission to forward from WAN to LAN as well.

15.2 Port forwards

Port forwarding allows remote computers on the Internet to connect to a specific computer or service within the private LAN. It is done in a way of routing network packets within a private network created by the device. Settings for the port forwarding of the device are defined as follows:

Firewall - Port Forwards - Unnamed forward

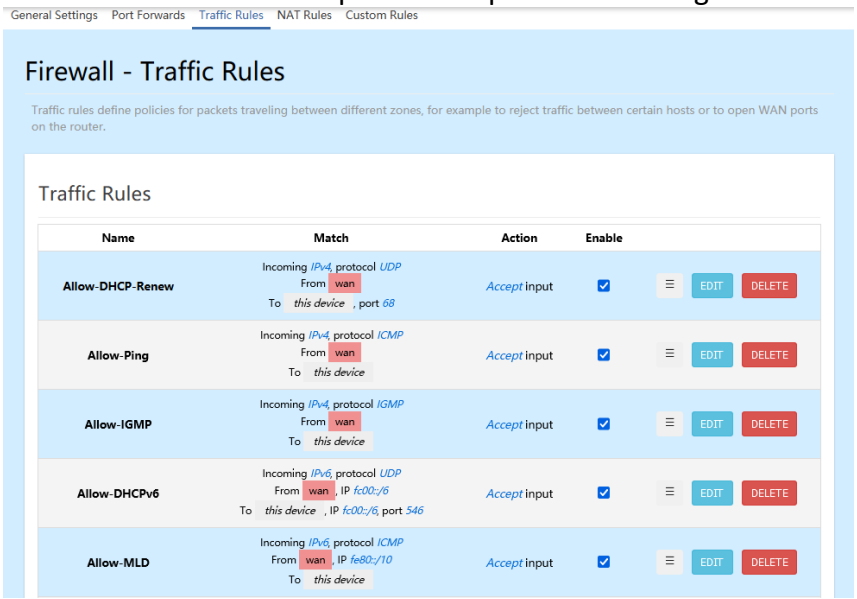


Name: The name of the port forwarding rule.
Protocol: Used protocol (Any/TCP/UDP/ICMP)
Source Zone: Informs which interface forward is matched to.
External port: Informs what port forward is matched to.
Destination Zone: Informs which interface is forwarded to
Forward to: Informs where the port is forwarded to.
Internal IP address: Redirect matched incoming traffic to the specified internal host.
Internal port: Redirect matched incoming traffic to the given port on the internal host.

The user can add, edit or delete port forwarding rules.

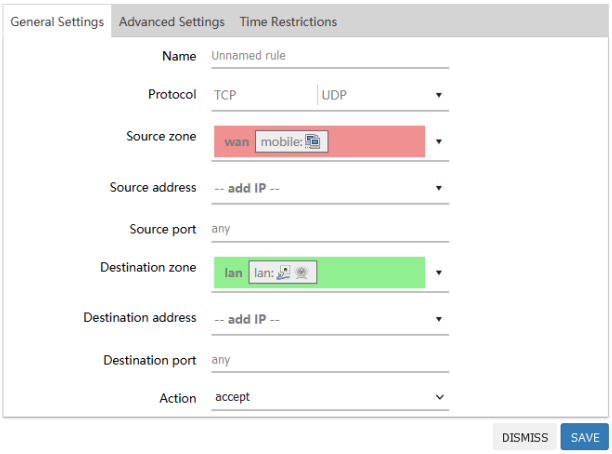
15.3 Traffic rules

Traffic rules which define policies for packets traveling between different zones.



Name: The name of the traffic rule.
Match: Informs what ICMP types are matched.
Action: Informs what action would be performed.
Enable: Enable (checked) or disable the rule.
The user can add, edit or delete traffic rules.

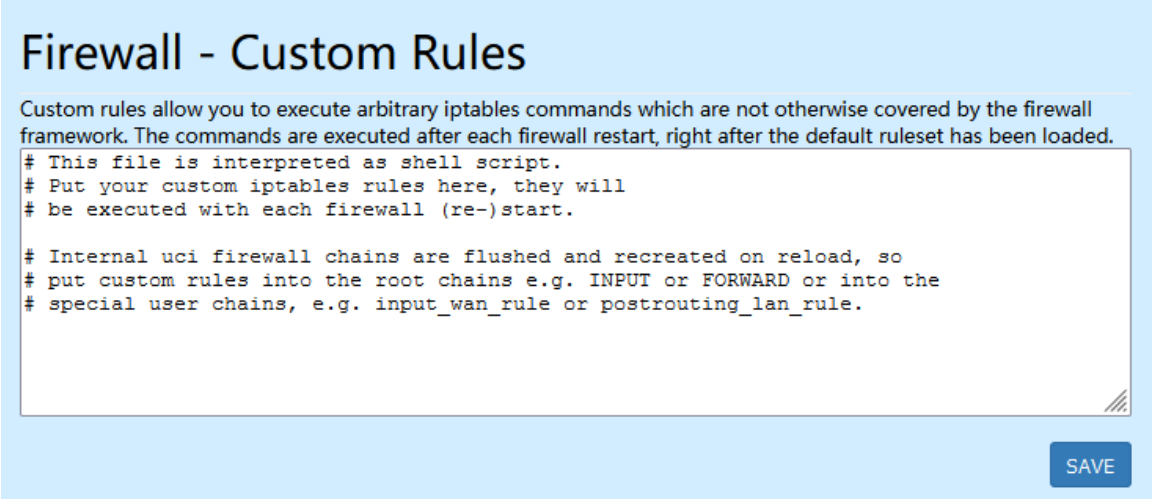
Firewall - Traffic Rules - Unnamed rule



For every rule can be defined these options under general settings: name, protocol, source and destination zones, source IP addresses and port, destination IP address and port, and action.
The following setting can be defined under advanced settings: match device, restrict to address family, source MAC, Match helper, Match mark, Match DSCP, Limit matching and extra arguments.
Source NAT, which is a specific form of masquerading which allows fine grained control over the source IP used for outgoing traffic, for the example to map multiple WAN addresses to internal subnets.
The user can add, edit or delete source NAT rules. For every rule can be defined these options:

name, protocol, source and destination zones, source, destination, SNAT IP addresses, ports, extra arguments, month and weekdays for which rule will apply, start/stop dates and times, time in UTC.

15.4 Custom rules



Custom rules allow to executing arbitrary iptables commands which are not otherwise covered by
the firewall framework. The commands are executed after each firewall restart, right after the default ruleset has been loaded.

16 WMBus

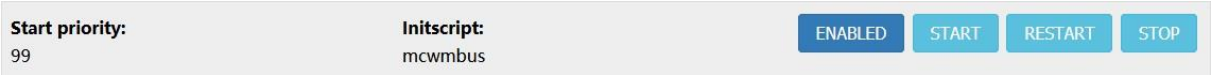
16.1 General information

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

16.2 Setting mcwmbus tool using LuCI

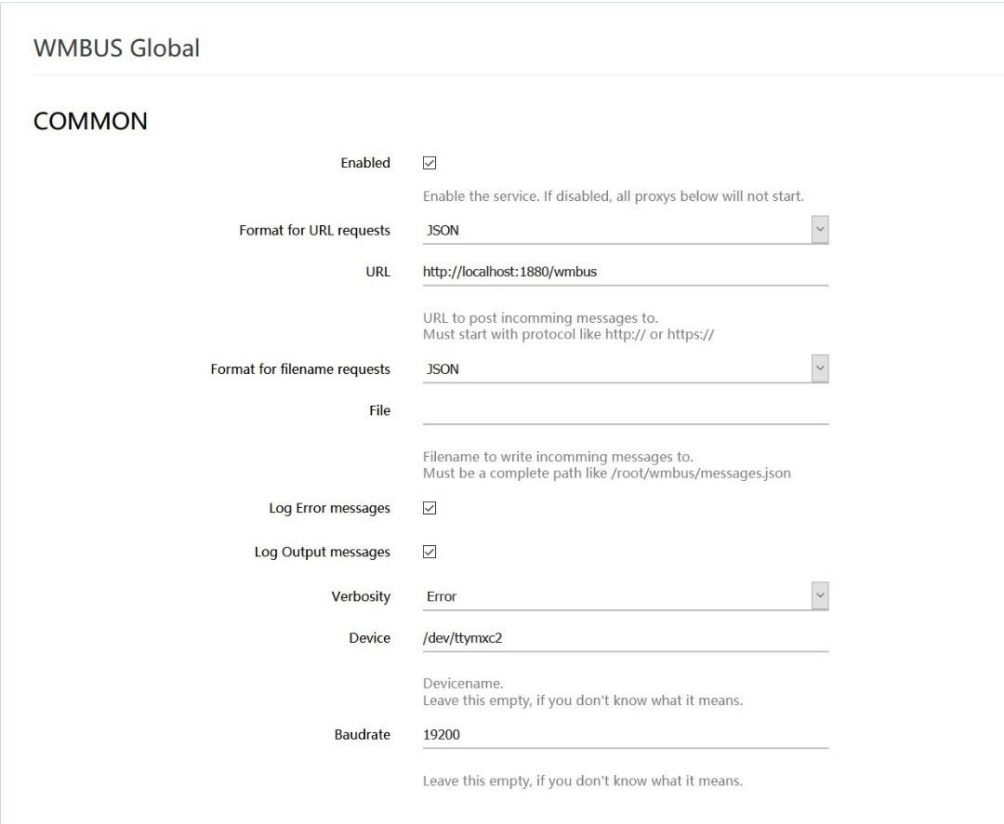
16.2.1 Enable the mcwmbus installed in MC100:

Click on "System" in the left menu and then on "Startup". Search for Initscript: “mcwmbus” and click the "Enable/Disable" button (If it is enabled then no need to do this step).



16.2.2 Setting WmBus

1. Under Services tab, click on WMBus.



2. Click on Enable.
3. Choose the Output format to use to POST on URL under Format for URL requests
4. Type wanted URL. (If not needed leave it empty).
5. Choose the Output format to use to save output data as file under Format for file name requests
6. Type wanted path and file name. (If not needed leave it empty).
7. Click on SAVE&APPLY.

16.3 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.

16.4 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 communication (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
```

16.5 Output formats

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

16.5.1 hex

Hex produces the message in hexadecimal. One message per line.
Example on hex Output:

```
1644AF4C02000041011B7A980000000266E8000266E900
```

16.5.2 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": "Senseco 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"
}
```

16.5.3 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>
```

16.5.4 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"}
```

16.6 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@   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.

16.7 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:

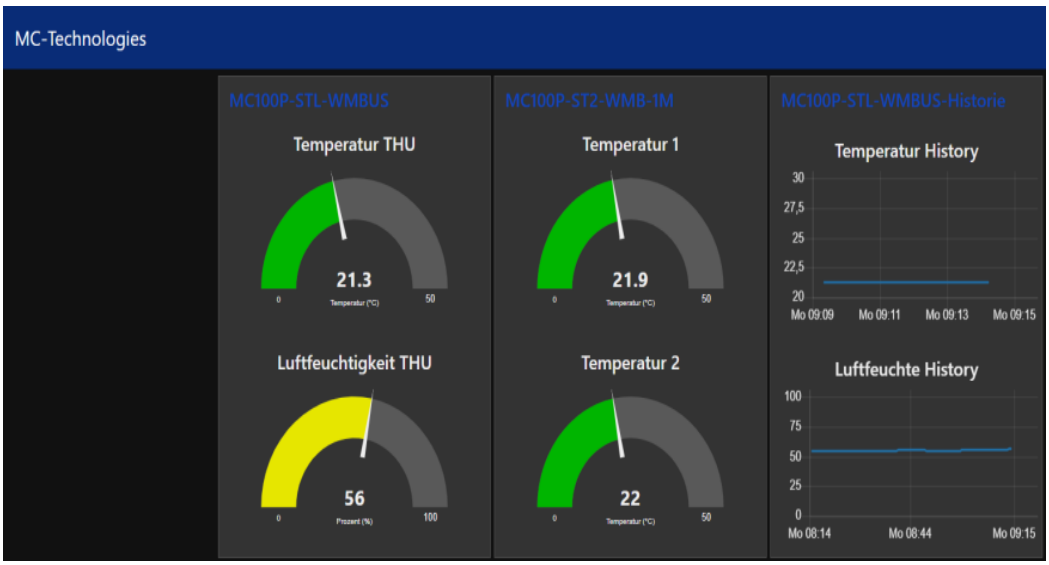
```
root@MC100:~# mcwmbus -f wmbus_messages/%M/%U.json
```

This gives us the nice opportunity of preselecting manufacturer and device.
Example file-tree for *„mcwmbus* -f wmbus_messages/%M/%U.json“

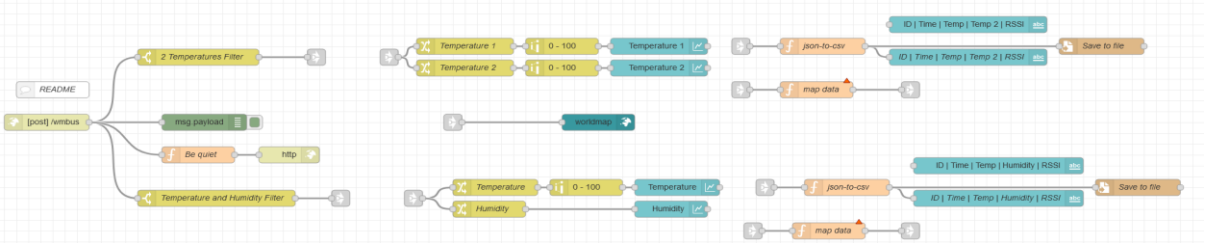
```
wmbus_messages
+-- 18c4
|   +-- 17949.json
+-- 25c5
|   +-- 33100007.json
+-- 4caf
|   +-- 30000007.json
|   +-- 41000002.json
|   +-- 41000292.json
+-- 8d7
    +-- 570536.json
    +-- 570542.json
    +-- 570663.json
```

16.8 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:



Node Red flow example:



Command:

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

Exported Flow from Node RED example:


```
[{"id":"2d7d91ec.eb8846","type":"tab","label":"Wmbus","disabled":false,"info":""},{"id":"a4663c25.fa8f98","type":"switch","z":"2d7d91ec.eb8846","name":"2 Temperatures
Filter","property":"payload.SlaveInformation.Id","propertyType":"msg","rules":[{"t":"eq","v":"41000018","vt":"num"}],"checkall":"true","repair":true,"outputs":1,"x":340,"y":960,"wires":[["ea217aa1.348558"]
]},{"id":"c31e2cda.f9d7e","type":"http
in","z":"2d7d91ec.eb8846","name":"","url":"/wmbus","method":"post","upload":false,"swaggerDoc":"","x":90,"y":1080,"wires":[["a4663c25.fa8f98","ff357041.e0fc98","ef2bf7f.5136188","b051727d.3aa3a8"]]},{"id":
"33fa8aec.51c216","type":"comment","z":"2d7d91ec.eb8846","name":"README","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\": \"1644AF4C02000041011B7AE10000000266F5000266F600\"\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":100,"y":1020,"wires":[]},{"id":"ff357041.e0fc98","type":"debug","z":"2d7d91ec.eb8846","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"false","x":350,"y"
:1080,"wires":[]},{"id":"1ac3dc7c.7a5114","type":"change","z":"2d7d91ec.eb8846","name":"Temperature
1","rules":[{"t":"move","p":"payload.DataRecords[0].Value","pt":"msg","to":"payload","tot":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":780,"y":940,"wires":[["77aa58c.31664a8"]]},{"i
d":"ac6f27a5.b2baf8","type":"change","z":"2d7d91ec.eb8846","name":"Temperature
2","rules":[{"t":"move","p":"payload.DataRecords[1].Value","pt":"msg","to":"payload","tot":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":780,"y":980,"wires":[["244fbd87.84b42a"]]},{"i
d":"d0c214cb.c425","type":"ui_chart","z":"2d7d91ec.eb8846","name":"","group":"2b1712f4.c90846","order":2,"width":0,"height":0,"label":"Temperature
1","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":true,"ymin":"","ymax":"","removeOlder":1,"removeOlderPoints":"","removeOlderUnit":"604800","cutout":0,"u
seOneColor":false,"useUTC":false,"colors":["#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"outputs":1,"x":1100,"y":940,"wires":[[]]},{"id":"c
d4e2dac.d1457","type":"ui_chart","z":"2d7d91ec.eb8846","name":"","group":"2b1712f4.c90846","order":2,"width":0,"height":0,"label":"Temperature
2","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":true,"ymin":"","ymax":"","removeOlder":1,"removeOlderPoints":"","removeOlderUnit":"604800","cutout":0,"u
seOneColor":false,"useUTC":false,"colors":["#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"outputs":1,"x":1100,"y":980,"wires":[[]]},{"id":"a
b52af69.2de7d","type":"ui_text","z":"2d7d91ec.eb8846","group":"d45c3f1b.23739","order":4,"width":30,"height":2,"name":"ID | Time | Temp | Temp 2 | RSSI","label":"","format":"{{msg.payload}}","layout":"col-
center","x":1610,"y":960,"wires":[]},{"id":"df2314df.1e1bd","type":"function","z":"2d7d91ec.eb8846","name":"json-to-csv","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\nid=
msg.payload.SlaveInformation.Id\ntemp= msg.payload.DataRecords[0].Value /10.0;\ntemp2= msg.payload.DataRecords[1].Value / 10.0;\ntime= msg.payload.DataRecords[1].Timestamp\nrssi = msg.payload.RSSI
\n\nmsg.payload=id+\" ,\"+time+\" ,\"+temp+\" ,\"+temp2+\" ,\"+rssi\nreturn
msg;\n","outputs":1,"noerr":0,"x":1370,"y":940,"wires":[["ab52af69.2de7d","7be6b356.4bea54"]]},{"id":"7be6b356.4bea54","type":"file","z":"2d7d91ec.eb8846","name":"Save to
file","filename":"/sensors/41000018.csv","appendNewline":true,"createDir":true,"overwriteFile":"false","x":1830,"y":940,"wires":[[]]},{"id":"2df41e95.80746a","type":"ui_text","z":"2d7d91ec.eb8846","group":
"d45c3f1b.23739","order":1,"width":"30","height":2,"name":"","label":"ID | Time | Temp | Temp 2 | RSSI","format":"{{msg.payload}}","layout":"col-
center","x":1610,"y":900,"wires":[]},{"id":"ea217aa1.348558","type":"link
out","z":"2d7d91ec.eb8846","name":"41000018/In","links":["56517074.898d88","253baead.5638ea","c1f3e626.b000d"],"x":535,"y":960,"wires":[]},{"id":"253baead.5638ea","type":"link
in","z":"2d7d91ec.eb8846","name":"41000018/Out1","links":["ea217aa1.348558"],"x":1235,"y":940,"wires":[["df2314df.1e1bd"]]},{"id":"56517074.898d88","type":"link
in","z":"2d7d91ec.eb8846","name":"41000018/Out","links":["ea217aa1.348558"],"x":655,"y":960,"wires":[["1ac3dc7c.7a5114","ac6f27a5.b2baf8"]]},{"id":"be5603e3.b8fb98","type":"function","z":"2d7d91ec.eb8846",
"name":"map data","func":"var time = msg.payload.DataRecords[1].Timestamp;\nvar Temp1= msg.payload.DataRecords[0].Value/10.0;\nvar Temp2= msg.payload.DataRecords[1].Value/10.0;\nvar id=
msg.payload.SlaveInformation.Id;\n\nmsg.payload = {};\n\nmsg.payload = {\n    \"name\": id,\n    \"time\": time,\n    \"Temp 1\": Temp1,\n    \"Temp 2\": Temp2,\n    \"id\": id ,\n    \"status\":
\"reviewed\",\n    \"locationSource\": \"nn\",\n    \"magSource\": \"nn\",\n    \"lon\": 000,\n    \"lat\": 000,\n    \"icon\": \"globe\",\n    \"iconColor\": \"orange\",\n
\"layer\":\"wmbus\",\n}\ndelete msg.payload.status;\ndelete msg.payload.locationSource;\t\ndelete msg.payload.magSource;\ndelete msg.payload.id;\n\nreturn
msg;","outputs":1,"noerr":4,"x":1360,"y":1020,"wires":[["659e3d0f.b9ced4"]]},{"id":"c1f3e626.b000d","type":"link
in","z":"2d7d91ec.eb8846","name":"41000018/Out2","links":["ea217aa1.348558"],"x":1235,"y":1020,"wires":[["be5603e3.b8fb98"]]},{"id":"ef2bf7f.5136188","type":"switch","z":"2d7d91ec.eb8846","name":"Temperatu
re and Humidity
Filter","property":"payload.SlaveInformation.Id","propertyType":"msg","rules":[{"t":"eq","v":"63000001","vt":"num"}],"checkall":"true","repair":true,"outputs":1,"x":370,"y":1220,"wires":[["5c2a3f23.d8b84"]
]},{"id":"95da81a5.910068","type":"change","z":"2d7d91ec.eb8846","name":"Temperature","rules":[{"t":"move","p":"payload.DataRecords[0].Value","pt":"msg","to":"payload","tot":"msg"}],"action":"","property":
"","from":"","to":"","reg":false,"x":830,"y":1200,"wires":[["1cc10cb8.788533"]]},{"id":"decdcbb5.0ff8","type":"change","z":"2d7d91ec.eb8846","name":"Humidity","rules":[{"t":"move","p":"payload.DataRecords[
1].Value","pt":"msg","to":"payload","tot":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":820,"y":1240,"wires":[["53f3cca1.ecb224"]]},{"id":"5c2a3f23.d8b84","type":"link
out","z":"2d7d91ec.eb8846","name":"61000001/In","links":["bbe60719.6551c8","f32f2d3d.1a96d","3fc40fe7.bd2c88"],"x":575,"y":1220,"wires":[]},{"id":"bbe60719.6551c8","type":"link
in","z":"2d7d91ec.eb8846","name":"63000001/Out","links":["5c2a3f23.d8b84"],"x":695,"y":1220,"wires":[["95da81a5.910068","decdcbb5.0ff8"]]},{"id":"f32f2d3d.1a96d","type":"link
in","z":"2d7d91ec.eb8846","name":"63000001/Out","links":["5c2a3f23.d8b84"],"x":1275,"y":1200,"wires":[["b875b5a0.e15b28"]]},{"id":"4512209e.633028","type":"file","z":"2d7d91ec.eb8846","name":"Save to
file","filename":"/sensors/63000001.csv","appendNewline":true,"createDir":true,"overwriteFile":"false","x":1890,"y":1200,"wires":[[]]},{"id":"b00c0cbd.4da31","type":"ui_text","z":"2d7d91ec.eb8846","group":
"356e6fbe.368018","order":5,"width":30,"height":1,"name":"ID | Time | Temp | Humidity | RSSI","label":"","format":"{{msg.payload}}","layout":"col-
center","x":1660,"y":1220,"wires":[[]],{"id":"b875b5a0.e15b28","type":"function","z":"2d7d91ec.eb8846","name":"json-to-csv","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\nid=
msg.payload.SlaveInformation.Id\ntemp= msg.payload.DataRecords[0].Value /10.0;\nhum= msg.payload.DataRecords[1].Value \ntime= msg.payload.DataRecords[1].Timestamp\nrssi = msg.payload.RSSI
\n\nmsg.payload=id+\" ,\"+temp+\" ,\"+hum+\" ,\"+time+\" ,\"+rssi\nreturn
msg;\n","outputs":1,"noerr":0,"x":1410,"y":1200,"wires":[["b00c0cbd.4da31","4512209e.633028"]]},{"id":"25749ef5.ae691a","type":"ui_text","z":"2d7d91ec.eb8846","group":"356e6fbe.368018","order":1,"width":30
,"height":1,"name":"","label":"ID | Time | Temp | Humidity | RSSI","format":"{{msg.payload}}","layout":"col-center","x":1660,"y":1160,"wires":[[]],{"id":"3fc40fe7.bd2c88","type":"link
in","z":"2d7d91ec.eb8846","name":"63000001/Out","links":["5c2a3f23.d8b84"],"x":1255,"y":1280,"wires":[["7c9bb07b.6f2ca"]]},{"id":"7c9bb07b.6f2ca","type":"function","z":"2d7d91ec.eb8846","name":"map
data","func":"var time = msg.payload.DataRecords[1].Timestamp;\nvar Temp1= msg.payload.DataRecords[0].Value/10.0;\nvar Humidity= msg.payload.DataRecords[1].Value;\nvar id=
msg.payload.SlaveInformation.Id;\n\nmsg.payload = {};\n\nmsg.payload = {\n    \"name\": id,\n    \"time\": time,\n    \"Temp1\": Temp1,\n    \"Humidity\": Humidity,\n    \"id\": id ,\n    \"status\":
\"reviewed\",\n    \"locationSource\": \"nn\",\n    \"magSource\": \"nn\",\n    \"lon\": 00000,\n    \"lat\": 00000,\n    \"icon\": \"globe\",\n    \"iconColor\": \"orange\",\n
\"layer\":\"wmbus\",\n}\ndelete msg.payload.status;\ndelete msg.payload.locationSource;\t\ndelete msg.payload.magSource;\ndelete msg.payload.id;\n\nreturn
msg;","outputs":1,"noerr":4,"x":1380,"y":1280,"wires":[["a2214636.71d7d"]]},{"id":"9a5d7873.d2e8d8","type":"ui_worldmap","z":"2d7d91ec.eb8846","group":"df02e013.0749a","order":1,"width":"20","height":"20",
"name":"","lat":"52.4126344","lon":"9.7326637","zoom":"18","layer":"OSM","cluster":"4","maxage":"1200","usermenu":"hide","layers":"hide","panit":"false","panlock":"false","zoomlock":"false","hiderightclick
":"true","coords":"none","showgrid":"false","path":"/worldmap","x":1000,"y":1080,"wires":[[]],{"id":"979840e5.0c08a8","type":"ui_chart","z":"2d7d91ec.eb8846","name":"","group":"6586f11c.31c43","order":1,"wi
dth":0,"height":0,"label":"Temperature","chartType":"line","legend":"false","xformat":"HH:mm","interpolate":"linear","nodata":"","dot":true,"ymin":"0","ymax":"100","removeOlder":1,"removeOlderPoints":"","r
emoveOlderUnit":"604800","cutout":0,"useOneColor":false,"useUTC":false,"colors":["#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"outputs":1,"
x":1130,"y":1200,"wires":[[]]},{"id":"53f3cca1.ecb224","type":"ui_chart","z":"2d7d91ec.eb8846","name":"","group":"6586f11c.31c43","order":3,"width":0,"height":0,"label":"Humidity","chartType":"line","legen
d":"false","xformat":"HH:mm","interpolate":"linear","nodata":"","dot":true,"ymin":"","ymax":"","removeOlder":1,"removeOlderPoints":"","removeOlderUnit":"604800","cutout":0,"useOneColor":false,"useUTC":fals
e,"colors":["#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"outputs":1,"x":1120,"y":1240,"wires":[[]]},{"id":"e2c4ed7d.b50bb8","type":"http
response","z":"2d7d91ec.eb8846","name":"","statusCode":"","headers":{},"x":490,"y":1140,"wires":[]},{"id":"b051727d.3aa3a8","type":"function","z":"2d7d91ec.eb8846","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":340,"y":1140,"wires":[["e2c4ed7d.b50bb8"]]},{"id":"1cc10cb8.788533","type":"range","z":"2d7d91ec.eb8846","minin":"0","maxin":"1000","minout":"0","maxout":"100","action":"sca
le","round":false,"property":"payload","name":"","x":980,"y":1200,"wires":[["979840e5.0c08a8"]]},{"id":"77aa58c.31664a8","type":"range","z":"2d7d91ec.eb8846","minin":"0","maxin":"1000","minout":"0","maxout
":"100","action":"scale","round":false,"property":"payload","name":"","x":940,"y":940,"wires":[["d0c214cb.c425"]]},{"id":"244fbd87.84b42a","type":"range","z":"2d7d91ec.eb8846","minin":"0","maxin":"1000","m
inout":"0","maxout":"100","action":"scale","round":false,"property":"payload","name":"","x":940,"y":980,"wires":[["cd4e2dac.d1457"]]},{"id":"a2214636.71d7d","type":"link
out","z":"2d7d91ec.eb8846","name":"map 169MHz output","links":["e02fa5aa.f6abf8"],"x":1515,"y":1280,"wires":[[]],{"id":"659e3d0f.b9ced4","type":"link out","z":"2d7d91ec.eb8846","name":"map 833
Output","links":["e02fa5aa.f6abf8"],"x":1515,"y":1020,"wires":[[]],{"id":"e02fa5aa.f6abf8","type":"link in","z":"2d7d91ec.eb8846","name":"map
Input","links":["659e3d0f.b9ced4","a2214636.71d7d"],"x":760,"y":1080,"wires":[["9a5d7873.d2e8d8"]]},{"id":"2b1712f4.c90846","type":"ui_group","z":"","name":"41000018","tab":"fbb0f1e5.e1a2e","order":2,"disp
":true,"width":"7","collapse":false},{"id":"d45c3f1b.23739","type":"ui_group","z":"","name":"Last
Update","tab":"fbb0f1e5.e1a2e","order":1,"disp":true,"width":"30","collapse":false},{"id":"356e6fbe.368018","type":"ui_group","z":"","name":"Last
update","tab":"439b7598.b25cc4","order":1,"disp":true,"width":"30","collapse":false},{"id":"df02e013.0749a","type":"ui_group","z":"","name":"Map","tab":"3e66a237.a71a76","order":1,"disp":true,"width":"20",
"collapse":false},{"id":"6586f11c.31c43","type":"ui_group","z":"","name":"63000001","tab":"439b7598.b25cc4","order":2,"disp":true,"width":7,"collapse":false},{"id":"fbb0f1e5.e1a2e","type":"ui_tab","z":"",
name":"2 Temperature Sensors","icon":"dashboard","disabled":false,"hidden":false},{"id":"439b7598.b25cc4","type":"ui_tab","z":"","name":"Temp & Humidity
Sensors","icon":"dashboard","disabled":false,"hidden":false},{"id":"3e66a237.a71a76","type":"ui_tab","z":"","name":"Map","icon":"dashboard","disabled":false,"hidden":false}]
```

16.9 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
```

16.10 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.

17 MODBUS

Modbus messaging protocol is used to establish client-server (master-slave) communication between devices.

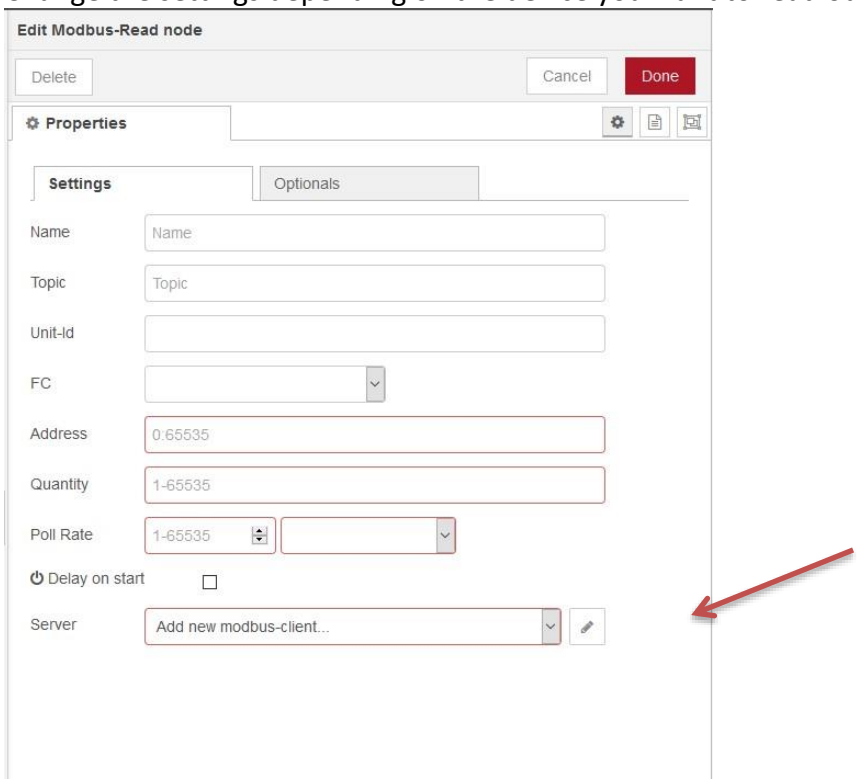
MC100 can be used either as a master device (server) or slave device (client).

17.1 MODBUS for Node-RED

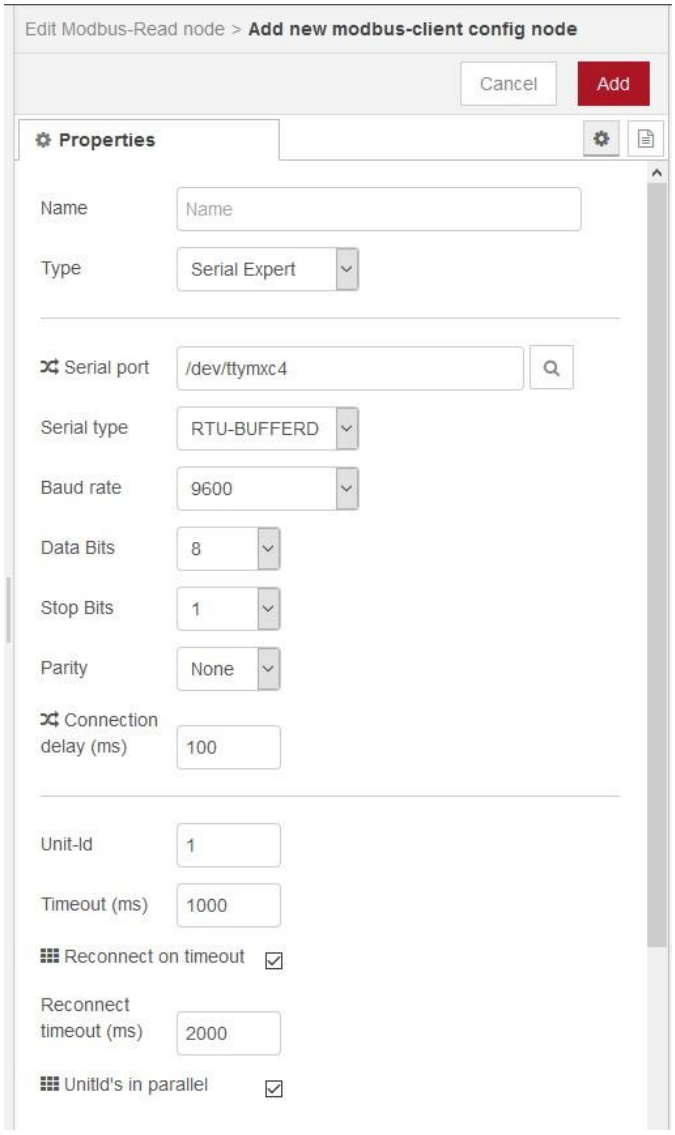
node-red-modbus is factory installed on MC100.
Do not try to install the node-red-contrib-modbus package from within Node-RED.

17.1.1 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. 

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

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

17.2 Modbus Master Command line Tool

MC100 can be used as master device to read/write data command for up to 247 slave devices.

17.2.1 Installation

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

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

17.2.2 Usage and Command lines

- Read register 0 on slave 1:

```
root@MC100:~# mcmodbus -a 0
```

- Print debug information during execution:

```
root@MC100:~# mcmodbus -v -a 0
root@MC100:~# mcmodbus -vv -a 0
```

- Show help message:

```
root@MC100:~# mcmodbus –h
```

- Read register 0 on slave 17:

```
root@MC100:~# mcmodbus -s 17
```

- Set the output of slave 17 for the I/O Pins 4,5,6 to 1 0 1:

```
root@MC100:~# mcmodbus -o wb - a 4 -s 17 1 0 1
```

- Use another device with a baud rate of 115200:

```
root@MC100:~#mcmodbus -d /dev/ttyUSB10 -b 115200
```

- Setting digital output at address 0x34 to ON:

```
root@MC100:~# mcmodbus -o wib -a 0x34 1
root@MC100:~# mcmodbus -o wib -a 064 1
root@MC100:~# mcmodbus -o wib -a 52 1
```

- Reading analog input at address 0x20 and 0x21:

```
root@MC100:~# mcmodbus -o rir -a 0x20 -n 2
```

- Setting register 8 to 0x4563:

```
root@MC100:~# mcmodbus -o wr -a 0x08 0x4563
root@MC100:~# mcmodbus -a 8 -o wr 17763
```

Note: MC100 was tested to query data using Modbus RTU trouble free in frequencies between 20 to 40 Hz.

17.3 Modbus Slave Command line Tool

17.3.1 Install:

17.3.2 Using MC100 as Slave device:

```
Usage: mcmodbus-slave [OPTIONS]
Options:
  -h,--help                  Print this help message and exit
  -c,--config-file TEXT      Json config for the address mappings. Default: ./mappings.json
  -d,--device-file TEXT      Serial device for modbus RTU
  -p,--port UINT             Port for modbus TCP.
  -b,--baud-rate UINT Baud rate for the serial device. Default: 115200
  -v,--verbose        Activate debug input
```

17.3.3 Using ModBus RTU

Command: mcmodbus-slave -d <Serial-port-Device-file> -b <baud-rate> -c <Json Config>
For Example (RS485):

```
root@MC100:~# mcmodbus-slave -d /dev/ttymxc4 -b 115200
```

17.3.4 Using ModBus TCP

Command: mcmodbus-slave -p <port> -b <baud-rate> -c <Json Config>
For Example:

```
root@MC100:~# mcmodbus-slave -p 502
```

17.3.5 MC100 default Json mapping.

XXXX

First Digit:
1: Digital Input
2: Digital Output
3: Current Input
4: Current Output
5: Voltage Input
6: RTD
7: PWM

Second Digit:
0: MC100 Gateway
1: SensT2 expansion board
2: GPIO expansion board
3: SensorBox expansion board

Last 2 Digits:
00-01: MC100 Gateway Inputs
00-01: MC100 Gateway Outputs
01-08: Digital Inputs
01-08: Digital Outputs
03-04: Current Inputs
01-02: Current Outputs (Sens)
01-02: Voltage Inputs
{00,01}, ... , {06,07}, ...: RTD (2 registers each).
00: PWM Prescalar
01-02: PWM Pulse
03: AC Input Relais (SensorBox)
00: Current Output (SensorBox)

MC100 ModBus addressing map:

Example: The address of Digital Input 4 on MC100 GPIO is: 1204

The address of RTD 2 on MC100 SensT2 is: {6104,6105}

17.3.6 Json Configuration file

Example:

```
  {
    "type": "file",
    "address": 4000,
    "num_addresses":  1,
    "register": true,
    "output": true,
    "filename": "
/sys/sensor_carrier_board/adcoutputs/adcoutput/value",
    "factor": 1000,
    "isfloat": true,
    "MinValue": 4000,
    "MaxValue": 20000
  },
```

Modbus data is most often read and written as "registers" which are 16-bit pieces of data. If a 32-bit integer is required, these values are actually read as a pair of registers.

	Description
type	Type of mapping used. file: Contents of a file are mapped. exe: Execute a command
address	ModBus Address used to be called by master device.
num_addresses	Number of registers used to map the file. [Default 1] 1 : 16-bit integer 2 : 32-bit integer
register	True: read / write a 16-bit integer False: read / write 1 or 0 [Default]
Output	True: write False: Read [Default]
filename	Path to the Input / Output file.
isfloat	True, if the input/output is float. [Default False]
factor	Factor used to transform float to integer. [Default 1]
MinValue	Minimum input value
MaxValue	Maximum input value
Command	Command to be executed. (Only when **Type** is „exe“)

18 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 ...).

18.1 Setting up the MBUS module

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

Communication specifications

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

18.2 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

19 1-wire

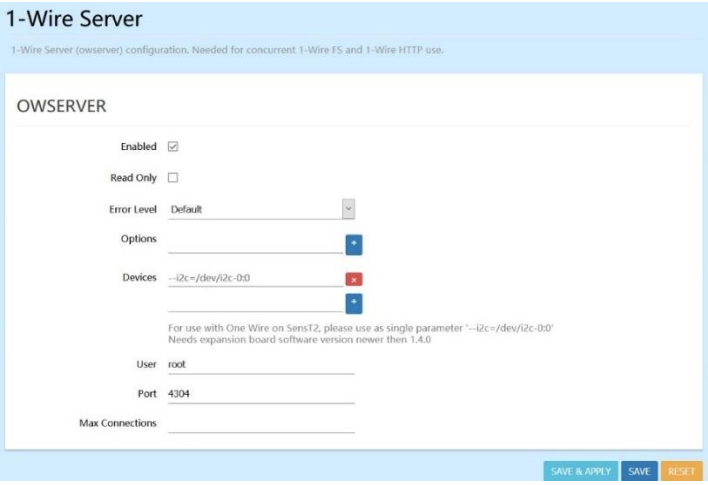
The 1-wire driver is loaded and started during the boot process. The driver provides the user with a command-option with 1-wire devices.
Note: *Needs expansion board software version newer then 1.4.0.*

OWFS 1-Wire file system is an easy way to use the powerful 1-wire system.

19.1 OWServer

OWserver can be Enabled using LuCI web Interface:

1. In LuCI web Interface, under Services click on 1-Wire Server.
2. Check Enabled check-box
3. Set Devices to: --i2c=/dev/i2c-0:0

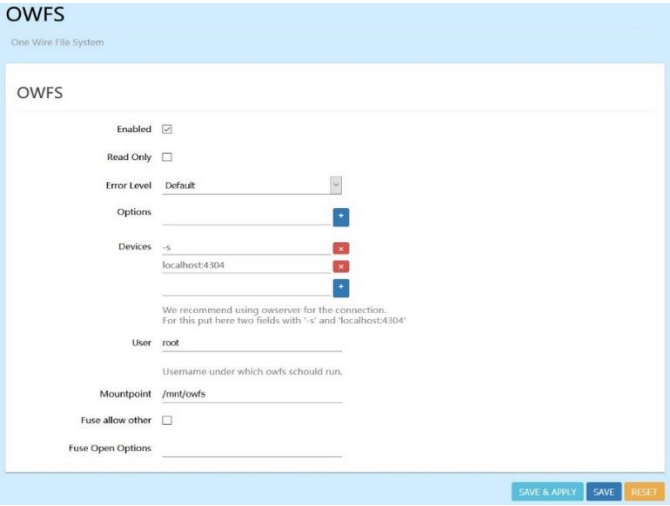


4. Port can be changed as wanted, default is 4304.
5. SAVE&APPLY

19.2 Using OWFS file system

OWFS can be Enabled using LuCI web Interface:

1. In LuCI web Interface, under Services click on 1-Wire FS.
2. Check Enabled check-box



3. Mountpoint can be changed as wanted, default is /mnt/owfs
4. SAVE&APPLY

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
```

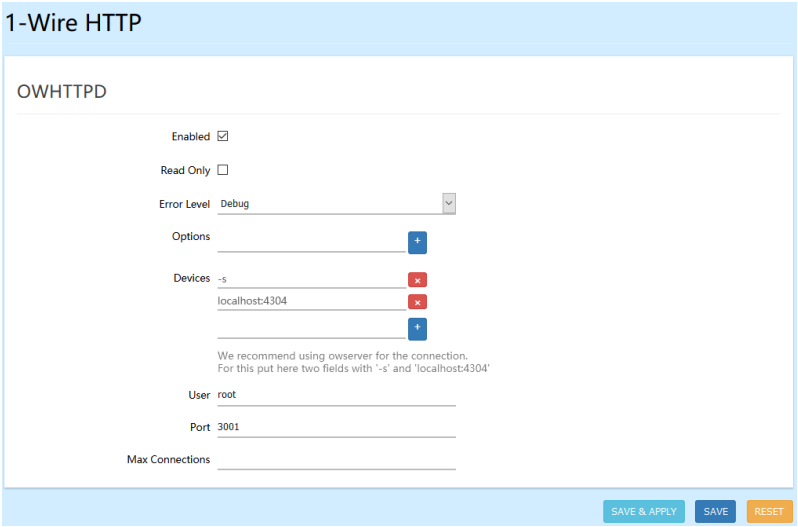
3. To read the contents, type:

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

19.3 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. In LuCI web Interface, under Services click on 1-Wire HTTP.
2. Check Enabled check-box



3. Port can be changed as wanted, default is 3001.
4. SAVE&APPLY

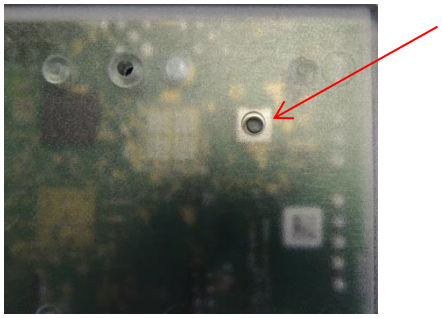
You can reach the owhttpd web interface with the same IP address as the standard web interface and the specification of the port (e.g. 3001).
Example with default address: http://192.168.2.1:3001

20 Factory reset

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

To reset the MC100:

- 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, the green LED (info) will light up.
- After approximately 5 seconds the green LED (info) will start flashing for approx. 1second.
- When the green LED (info) stops flashing you can release the button.
- After 10 seconds, the green LED 5 will start flashing indicating that the reset process is running.
- Wait for the green LED (status) to stop flashing. That means the device is fully reset and is ready to be used.

The default factory settings have been restored.

21 Maintenance, repair and troubleshooting

21.1 Maintenance

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

21.2 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.

21.3 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.

22 Disposal

22.1 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