Installer kit - basic information and configuration

Document number: PO-216-EN Version: 1.0 Date of publication: April 30, 2024

Description of the main contents of the installer kit

The installer kit comes with a number of different devices from the Ampio offer that allow a new installer to learn the Ampio system, its logic, and configuration. The kit also includes some Ampio gadgets that will not be discussed in this document. This guide will focus solely on explaining the first steps with the Ampio Installer Kit in order for a new installer to grasp the logic of the Ampio system and start implementing basic automation solutions.

M-SERV-s

The M-SERV-s unit is our flagship module packed with the most useful features for all your building automation needs. This module is mostly responsible for IP integration and internet connectivity, hosting the Ampio Designer programming and configuration interface, as well as handling the mobile application. The module also comes with Node-Red preinstalled and has a built-in MQTT broker, SIP server for video intercom handling and supports streaming with the use of the Airplay protocol.

Other than those networking features the M-SERV-s also sports some great standalone functionality, such as:

- A real time clock (RTC) to have access to the current date and time for use in logic
- A multizone temperature controller (bang-bang control with hysteresis)
- 1-Wire digital temperature sensor connection (for up to 6 sensors)
- A built-in One Wire Ampio lighting bus controller for OWA MLED addressable LED light drivers (up to 16 light nodes on one bus connection)
- 10 individual relay outputs, which can be mapped to up to 5 roller blind controllers for controlling motorised gates, curtains, and blinds, with slat blind support
- 4 open-collector (OC) transistor outputs, which output a PWM signal and can be used as is or mapped to an integrated RGBW LED light controller
- 15 binary inputs (14 ground detecting and 1 supply voltage detecting) for dry-contact output integration (motion sensors, switches, buttons, etc.)
- A power supply redundancy connection, which automatically switches to a working supply upon a failure of one and sends a notification on supply failure detections
- An RS-232 communication protocol connection

M-DOT-M18

The M-DOT-M18 is our flagship touch panel, featuring 18 separate touch fields as input, a 1-Wire temperature sensor connection, a controllable buzzer and a scrollable, full-colour, 2” LCD screen. All touch fields have an RGB status LED and an RGBW LED backlight, both of which can be controlled using device parameters and logic. Also, the fields can have custom icons and labels engraved on the top glass part of the touch panel. The touch panel’s look can be customised at kreatorpaneli.ampio.pl.

The panel, because of its large size and LCD display, is uniquely equipped with some distinctive features, e.g.:

- Icon screen – 4 custom icons in display corners with an information field in the middle
- Information screen – 3 lines of information (ex. data from other modules, sensors, date and time)
- Countdown timer screen – screen with an adjustable timer
- Numeric keypad screen – a screen on which a PIN code can be typed in, using the 12 touch fields below the screen as a keypad, to perform an action in the installation
- Ampio alarm control screen – a screen for controlling the M-ALARM family of Ampio alarm systems (arming, disarming, individual user PIN)

M-SENS-LITE

The M-SENS-LITE is a basic environmental parameter sensor, which connects directly to the CAN bus and reports the values of multiple parameters to the system.

The module measures:

- temperature (°C)
- air humidity (%)
- atmospheric pressure (hPa)

M-IN-2p

The M-IN-2p is a miniature input module meant to be mounted in electrical junction boxes and behind switches. The module connects directly to the CAN bus and features 2 ground detecting inputs, a 1-Wire bus connection for digital temperature sensors and a brightness sensor for detecting whether the module has been taken out of its mounting place.

S-LED-U

The S-LED-U is our own LED light driver that works with the One Wire Ampio lighting bus. This voltage driver board acts as a node on the OWA bus, allowing for individual dimming control of the connected voltage-controlled LED light.

DALLAS DS18B20 temperature sensor

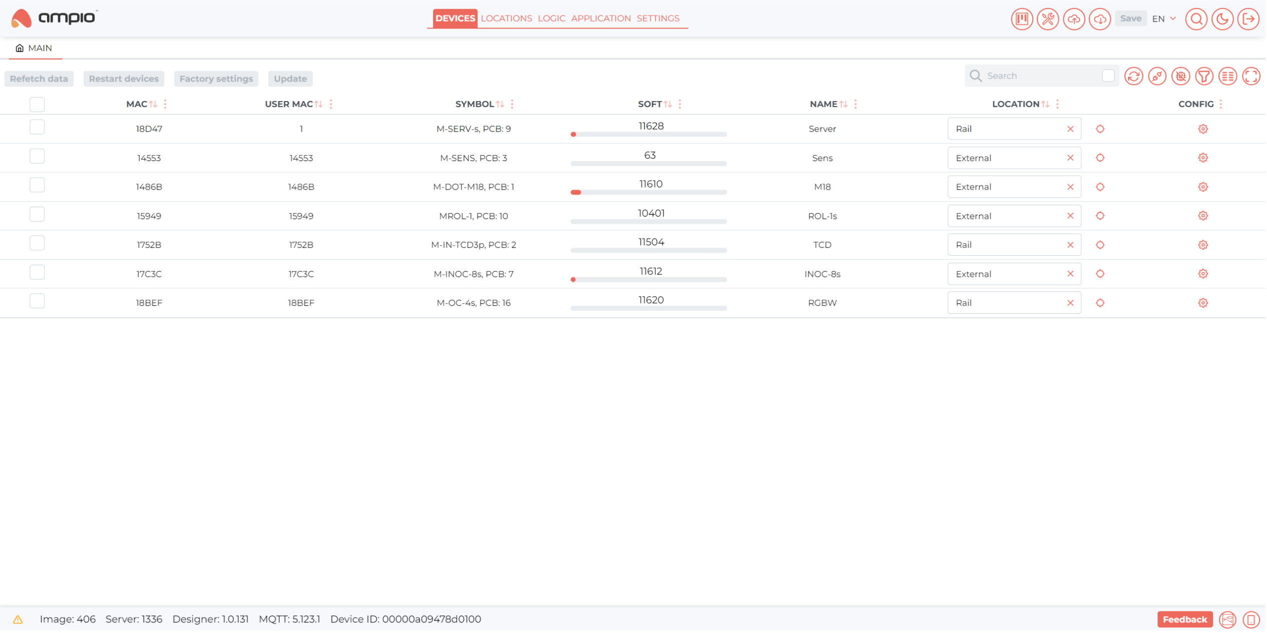
The DALLAS DS18B20 temperature sensor is a simple digital temperature sensor, which connects to the 1-Wire bus and reports its current temperature to it.

Configuration tool interface basics

To fully understand how to use our Ampio Designer configuration software we highly recommend reading the Ampio Designer guide, but for the purposes of this document a quick overview of device configuration and logic will be provided.

Devices tab

In the *Devices* tab all the devices connected to the same CAN network as the M-SERV will be displayed together with some important information relating to them:



	MAC	USER MAC	SYMBOL	SOFT	NAME	LOCATION	CONFIG
□	18D47	1	M-SERV-s, PCB: 9	11628	Server	Rail	
□	14553	14553	M-SENS, PCB: 3	63	Sens	External	
□	1486B	1486B	M-DOT-M18, PCB: 1	11610	M18	External	
□	15949	15949	MROL-1, PCB: 10	10401	ROL-1s	External	
□	1752B	1752B	M-IN-TCD3p, PCB: 2	11504	TCD	Rail	
□	17C3C	17C3C	M-INOC-8s, PCB: 7	11612	INOC-8s	External	
□	18BEF	18BEF	M-OC-4s, PCB: 16	11620	RGBW	Rail	

Image: 406 Server: 1336 Designer: 1.0.131 MQTT: 5.123.1 Device ID: 00000a09478d0100

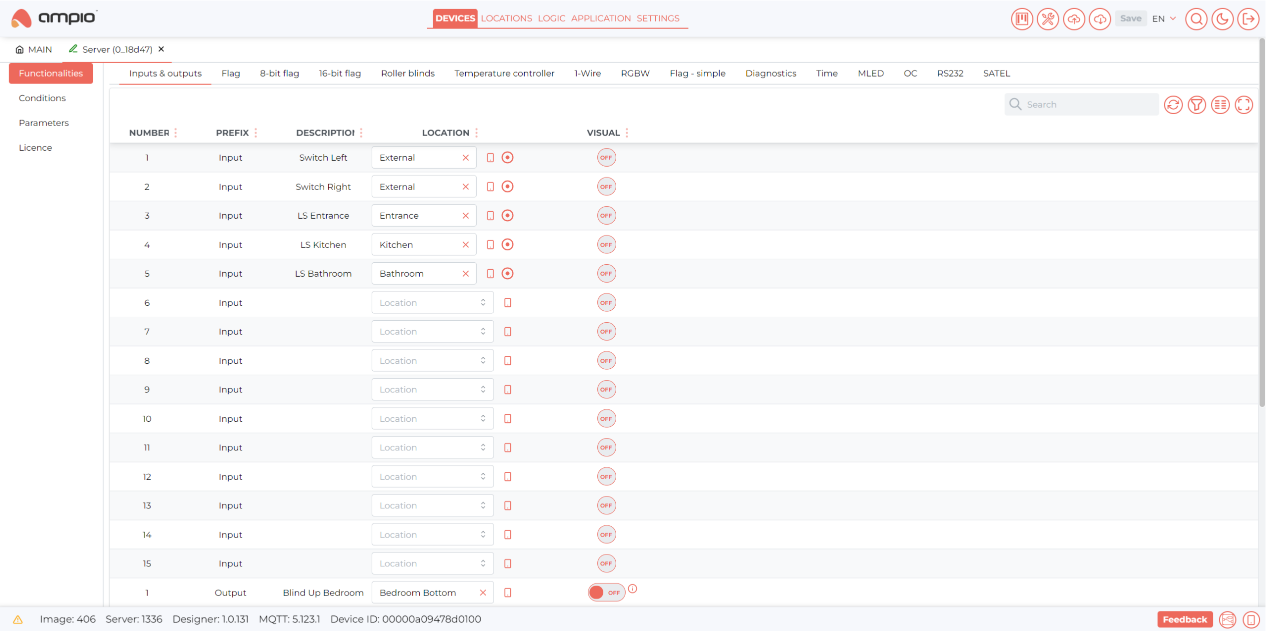
The information is displayed in the following columns:

- MAC – the device's unique factory MAC address
- USER MAC – the device's user-assigned MAC address (useful for compatibility reasons, for example when replacing modules)
- SYMBOL – the type of the device and its PCB version (important for hardware/firmware compatibility)
- SOFT – the current software version that the device is running together with a memory usage bar, updates and software rollbacks can be performed by selecting the software version number and selecting a different software version from the drop-down menu, or by pressing the "Update" button when an update is detected automatically
- NAME – the name given to the device by the user
- LOCATION – where the device is located (most useful for electrical junction box mounted modules)
- CONFIG – device configuration menu

Inside the configuration menu more detailed information about the individual device can be found together with many configuration options.

The most important menus can be found under the *Functionalities* and *Parameters* tabs.

Functionalities subtab



NUMBER	PREFIX	DESCRIPTIOI	LOCATION	VISUAL
1	Input	Switch Left	External	OFF
2	Input	Switch Right	External	OFF
3	Input	LS Entrance	Entrance	OFF
4	Input	LS Kitchen	Kitchen	OFF
5	Input	LS Bathroom	Bathroom	OFF
6	Input		Location	OFF
7	Input		Location	OFF
8	Input		Location	OFF
9	Input		Location	OFF
10	Input		Location	OFF
11	Input		Location	OFF
12	Input		Location	OFF
13	Input		Location	OFF
14	Input		Location	OFF
15	Input		Location	OFF
1	Output	Blind Up Bedroom	Bedroom Bottom	OFF

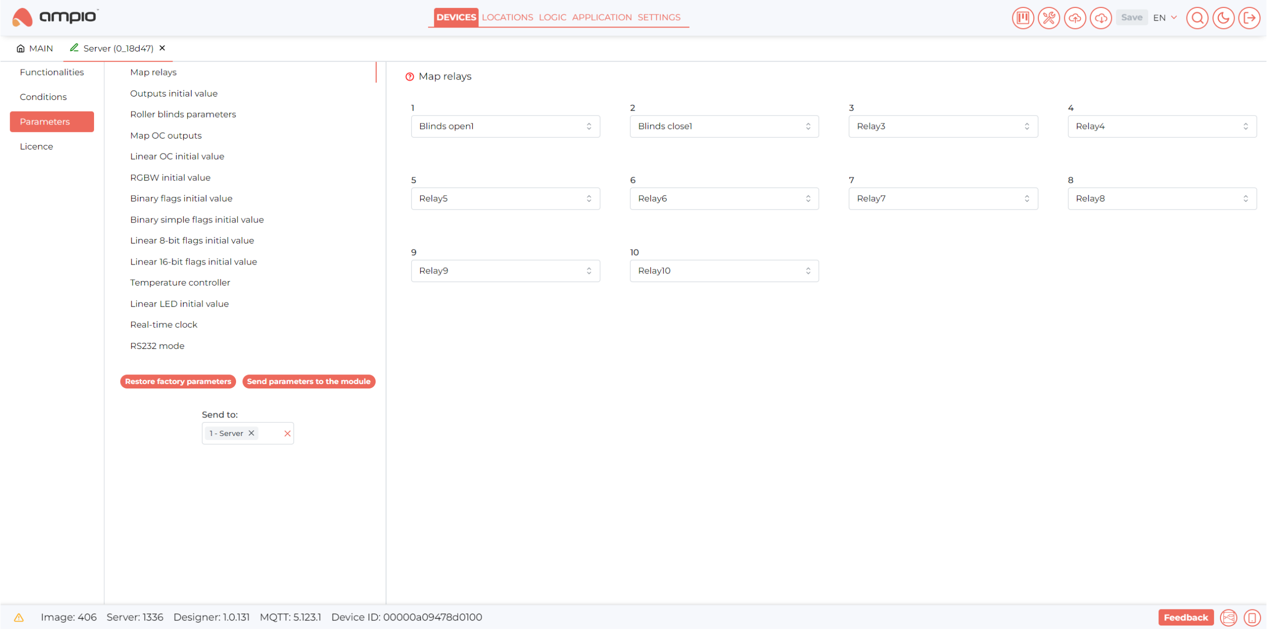
Image: 406 Server: 1336 Designer: 1.0.131 MQTT: 5.123.1 Device ID: 00000a09478d0100

In the *Functionalities* tab you can see an overview of all the functions available in the device.

The different inputs, outputs, flags and special functions like temperature controllers, RGBW drivers, roller blind controllers, etc. can all be named and assigned a location.

You can also see real time visual feedback of the current state of the function as well as manually control them for debugging purposes. After changing anything in the device's functionalities the "Save" button in the top right corner of the interface will light up red, click on it to save your changes.

Parameters subtab



Map relays

1	2	3	4
Blinds open1	Blinds close1	Relay3	Relay4
5	6	7	8
Relay5	Relay6	Relay7	Relay8
9	10		
Relay9	Relay10		

Send to: 1 - Server

Image: 406 Server: 1336 Designer: 1.0.131 MQTT: 5.123.1 Device ID: 00000a09478d0100

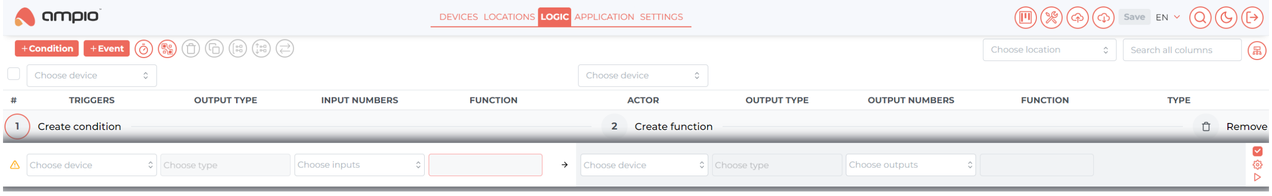
In the *Parameters* tab you can adjust the device's parameters so that it can work exactly as you want it to work. Outputs can be mapped, initial values set, and specific function parameters changed.

After changing any devices parameters click the "Send parameters to the module" below the parameter list to save your changes.

Logic tab

In the *Logic* tab all your logic conditions and functions will be created, examples of such functions will be shown in the sections below.

In this section the basic interface elements need to create those functions and conditions will be presented.



To start creating a condition press the "+ Condition" button in the top right.

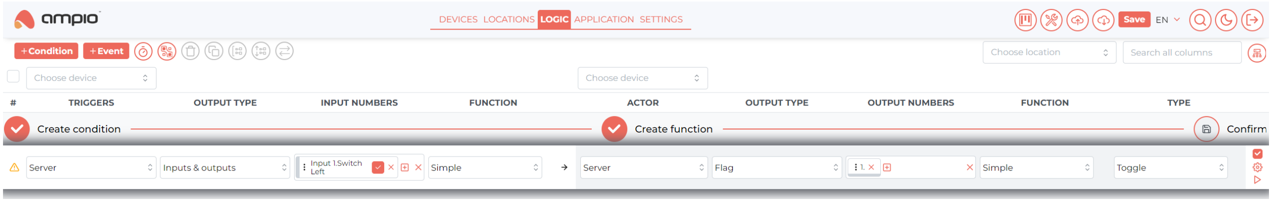
The "+ Event" button works in a similar way to the "+ Condition" button, except that it creates an action in response to an **event broadcast**, which will be discussed further in this guide.

After pressing the button, a condition creation field will appear. It is split in 2 parts:

1. The condition – in here you will create a condition, which will call the created function when evaluated in the specified way. To configure it specify the following things in the presented fields:
   - Triggers – the device on the network to listen to
   - Input type – what function of the device to observe
   - Input numbers – which inputs to consider during evaluation
   - Function – how to interpret the input (simple true/false, counting occurrences for multiple press functions, measuring the duration of the input's state, etc.)

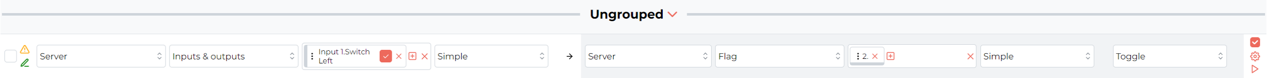
2. The function – what action should be taken upon condition evaluation. To configure it specify the following things in the presented fields:
   - Actor – the device which will perform the action
   - Output type – what kind of action to perform
   - Output numbers – which outputs should perform the action
   - Function – what should the specified outputs do
   - Type – further details regarding the performed action

When the condition is completed correctly ticks (□) will appear above the specific parts of the condition and the "Remove" button will change into a "Confirm" button:



#	TRIGGERS	OUTPUT TYPE	INPUT NUMBERS	FUNCTION		ACTOR	OUTPUT TYPE	OUTPUT NUMBERS	FUNCTION	TYPE
✓	Create condition					✓ Create function				Confirm
⚠	Server	Inputs & outputs	Input 1.Switch Left	Simple	→	Server	Flag	1.	Simple	Toggle

Upon confirming a new condition, a couple of useful user interface elements will be available to further check the functionality of the condition:



Server	Inputs & outputs	Input 1.Switch Left	Simple	→	Server	Flag	2.	Simple	Toggle

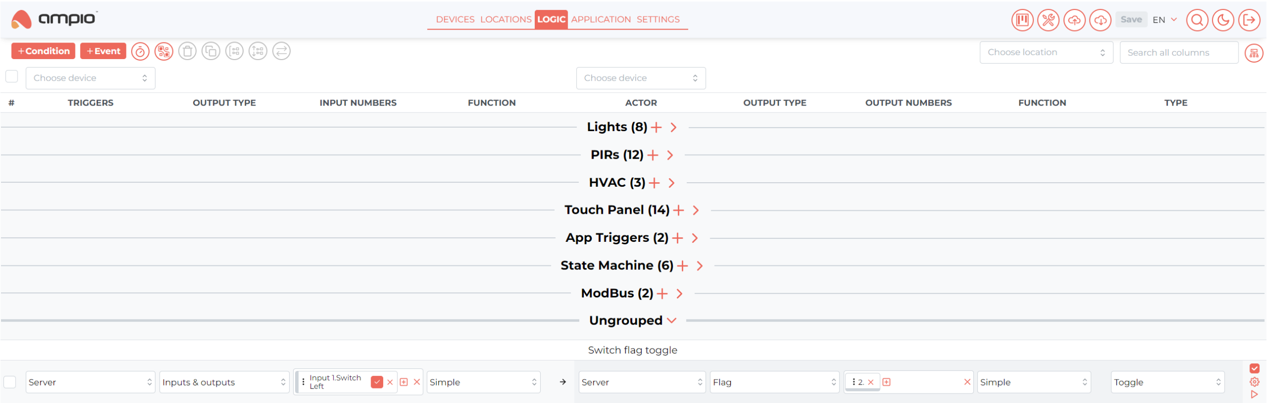
On the left side of the added condition, 2 icons will appear:

- A **yellow warning sign**, which indicates that the condition has no description, by clicking it a text field will appear in which you can describe the condition for readability, confirm the new description by pressing *Enter*.
- A **green pen icon**, which indicates that the condition is different from the conditions saved in the device, so save new conditions click the red "Save" button that will be lit up in the top right corner of the screen.

On the right side of the conditions, 3 buttons are located:

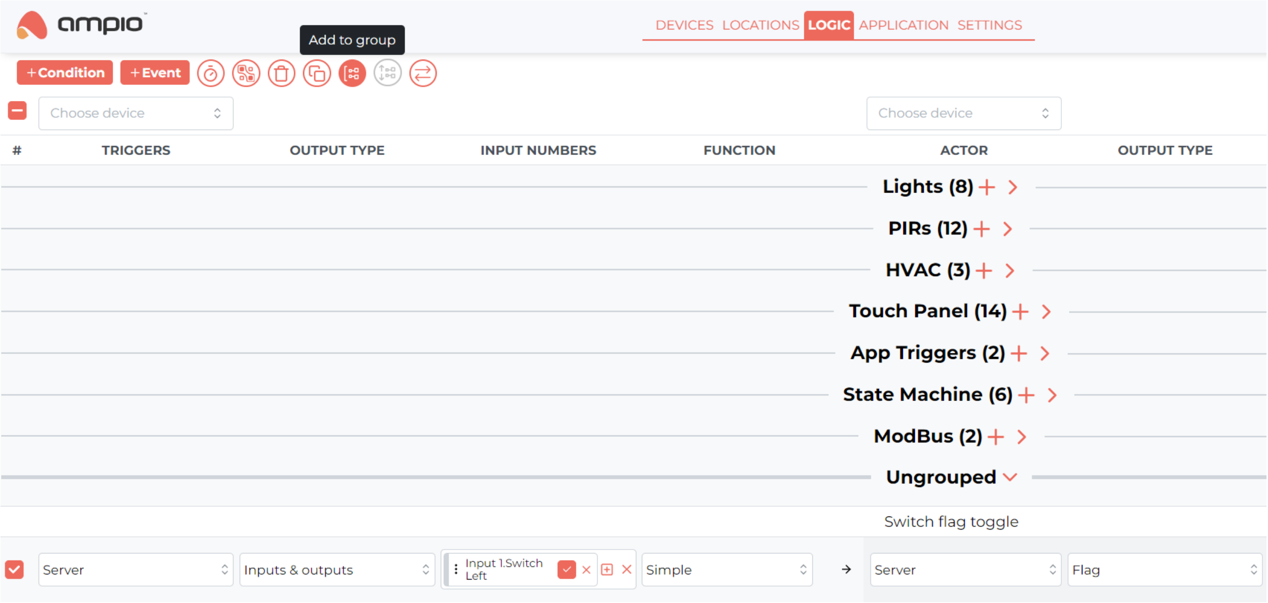
- The **checkbox** indicates whether the condition is active or not, it can be unchecked to disable the condition, it will be greyed out and won't be evaluated, but will still be saved in the device.
- The **gear icon** provides additional action to be performed on the condition, like changing the description, deleting the condition or *adding a linked condition* (an additional true or false check before evaluating the condition).
- The **play icon** allows you to check the condition's function, bypassing the condition check (a manual trigger of the action, useful for debugging).

Multiple conditions can also be selected with the checkbox on the left and added to a **group**, using one of the buttons in the top left corner, for easier condition sorting and management.



Switch flag toggle

Server | Inputs & outputs | Input 1.Switch Left | Simple → Server | Flag | 2. | Simple | Toggle



Switch flag toggle

Server | Inputs & outputs | Input 1.Switch Left | Simple → Server | Flag

We highly recommend creating some conditions and experimenting with different options on your own, but in the following sections we will present a few examples of how to achieve basic building automation functions.

Automation examples

Relay light control

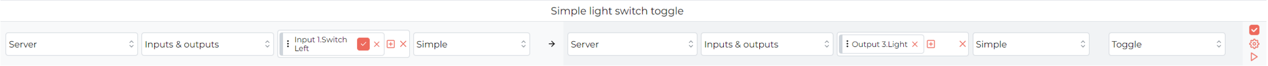
The simplest way to control a light.

Using a regular wall switch

Connect the relay to a light of your choice (incandescent, LED, etc.) and wire up an input switch to either one of the inputs of the M-SERV-s or the M-IN-2p.

Create a simple condition in the Ampio Designer's *Logic* tab:

1. Select the desired input, with the *Simple* input function
2. Select the relay that is connected to your light, with the *Simple Toggle* output function



Simple light switch toggle

Server | Inputs & outputs | Input 1.Switch Left | Simple → Server | Inputs & outputs | Output 3.Light | Simple | Toggle

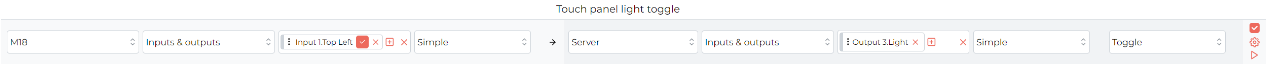
Now, every time the wall switch's button is pressed, the light will switch ON/OFF.

Using a touch panel

Connect the relay to a light of your choice (incandescent, LED, etc.) and select a touch field of your choice (if one of the 4 corner fields around the LCD display is selected an icon with status notification can be configured).

Create a simple condition in the Ampio Designer’s *Logic* tab:

1. Select the desired input, with the *Simple* input function
2. Select the relay that is connected to your light, with the *Simple Toggle* output function



Touch panel light toggle

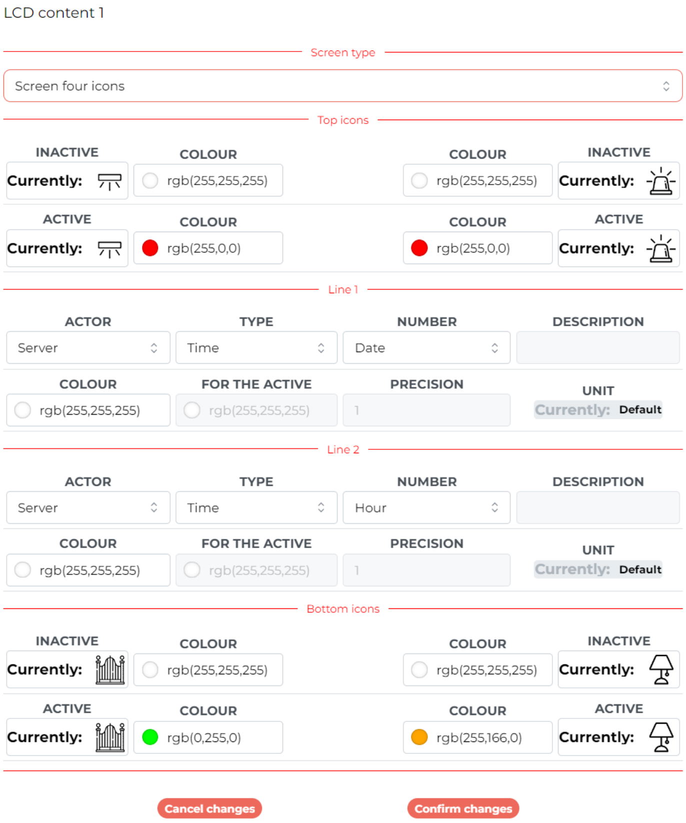
M18 | Inputs & outputs | Input 1.Top Left | Simple → Server | Inputs & outputs | Output 3.Light | Simple | Toggle

Now, every time the selected touch field is tapped, the light will switch ON/OFF.

Setting up an LCD icon and changing its colour

First, in the M-DOT-M18 configuration menu, select the *Parameter -> LCD* tab, select the screen you want to configure, and select the “Screen four icons” screen type and set up the desired icons.

An example of a 4-icon screen configuration with the date and time displayed is presented below:



LCD content 1

Screen type

Screen four icons

Top icons

INACTIVE	COLOUR	COLOUR	INACTIVE
Currently:	rgb(255,255,255)	rgb(255,255,255)	Currently:
ACTIVE	**COLOUR**	**COLOUR**	**ACTIVE**
Currently:	● rgb(255,0,0)	● rgb(255,0,0)	Currently:

Line 1

ACTOR	TYPE	NUMBER	DESCRIPTION
Server	Time	Date	
COLOUR	**FOR THE ACTIVE**	**PRECISION**	**UNIT**
rgb(255,255,255)	rgb(255,255,255)	1	Currently: Default

Line 2

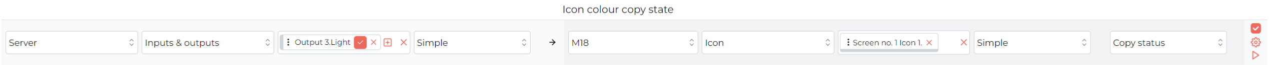
ACTOR	TYPE	NUMBER	DESCRIPTION
Server	Time	Hour	
COLOUR	**FOR THE ACTIVE**	**PRECISION**	**UNIT**
rgb(255,255,255)	rgb(255,255,255)	1	Currently: Default

Bottom icons

INACTIVE	COLOUR	COLOUR	INACTIVE
Currently:	rgb(255,255,255)	rgb(255,255,255)	Currently:
ACTIVE	**COLOUR**	**COLOUR**	**ACTIVE**
Currently:	● rgb(0,255,0)	● rgb(255,166,0)	Currently:

To change the colour of an icon in response to the state of the light, a simple condition can be made to copy the status of the output to the status of the icon:

1. Select the relay connected to the light, with the *Simple* input function
2. Select the M-DOT-M18 with the *Icon* output type and select the icon to be targeted
3. Select the *Simple Copy* status output function



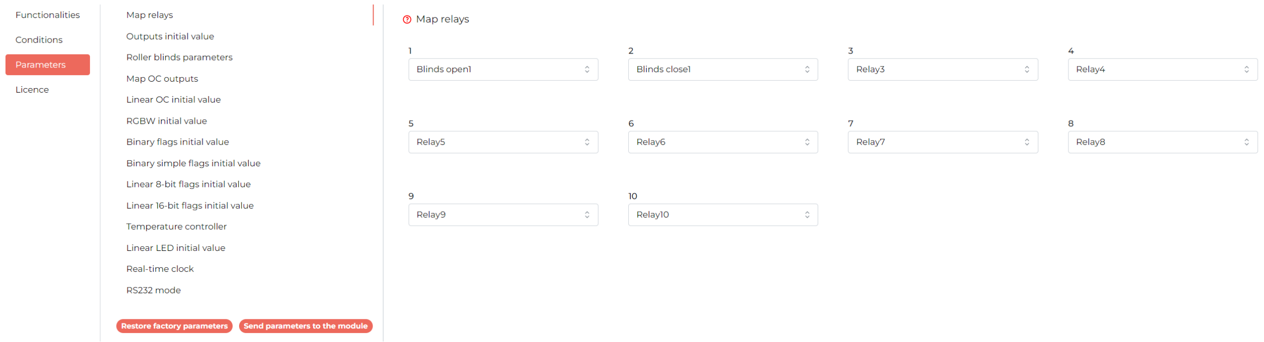
Icon colour copy state

Server | Inputs & outputs | Output 3.Light | Simple → M18 | Icon | Screen no. 1 Icon 1. | Simple | Copy status

Now, the icon should change the colour when the light is ON.

Roller blind control

In most of Ampio's modules which are equipped with relay outputs there is a "roller blind controller" built in. To use this feature relays, which operate the blind's motor, need to be mapped as *Blind Open* and *Blind Close*, this can be done in device parameters:

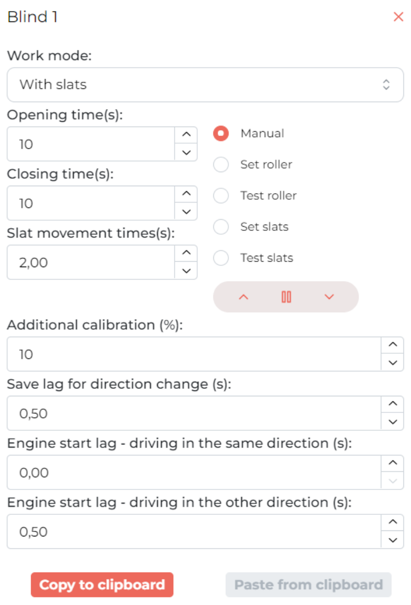


Map relays

1	2	3	4
Blinds open1	Blinds close1	Relay3	Relay4
5	**6**	**7**	**8**
Relay5	Relay6	Relay7	Relay8
9	**10**		
Relay9	Relay10		

Here the relays 1 and 2 of the M-SERV-s are mapped to opening and closing a blind under the blind controller number 1.

Next, the actual controller parameters need to be configured, to do that move to the *Roller blinds parameters* tab. The roller blind control is time-based, meaning all parameters relating to the blind movement are in seconds of motion:



Blind 1

Work mode: With slats

Opening time(s): 10

Closing time(s): 10

Slat movement times(s): 2,00

● Manual
○ Set roller
○ Test roller
○ Set slats
○ Test slats

Additional calibration (%): 10

Save lag for direction change (s): 0,50

Engine start lag - driving in the same direction (s): 0,00

Engine start lag - driving in the other direction (s): 0,50

A couple of important parameters can be configured in this menu:

- Work mode: either *Standard* or *With slats*
- Opening time: how long it takes for the blind to open
- Closing time: how long it takes for the blind to close
- Additional calibration: additional % of motion to perform in case of slack or backlash in the mechanism

- Save lag for direction change: delay when the blind is ordered to change direction in the middle of motion to protect the motor from current surges
- Engine start lag – driving in the same direction: delay when ordered to start moving in the same direction
- Engine start lag – driving in the other direction: delay when ordered to start moving in the other direction
- Additional parameters for blinds with slats:
  - Slat movement time: how long it takes for the slats to fully open/close

Here, standard blinds are configured, which open and close in 10 seconds, stop for 0.5 seconds when changing directions and perform a 10% calibration movement.

Finally let us create some logic to control the blind, there are a couple of ways to do that.

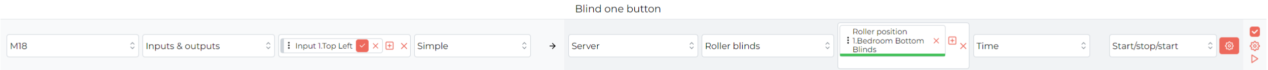
One-button control

One-button approach utilises only a single input, which can be a little bit restrictive.

Any input function can be used (ex. M-SERV-s or M-IN-2p physical inputs), here the M-DOT-M18 touch panel's input will be used.

With a single input a blind can be set at a position or be controlled in a start/stop/start manner, which will be shown here:

1. Select the desired input, with the *Simple* input function
2. Select the appropriate roller blinds controller, with the *Time Start/stop/start* output function



Blind one button

M18 | Inputs & outputs | Input 1.Top Left | Simple → Server | Roller blinds | Roller position 1.Bedroom Bottom Blinds | Time | Start/stop/start

Now, when an input is detected, the blind will start moving, when detected again, the blind will stop, and when detected again, the blind will start moving in the other direction.

Two-button control

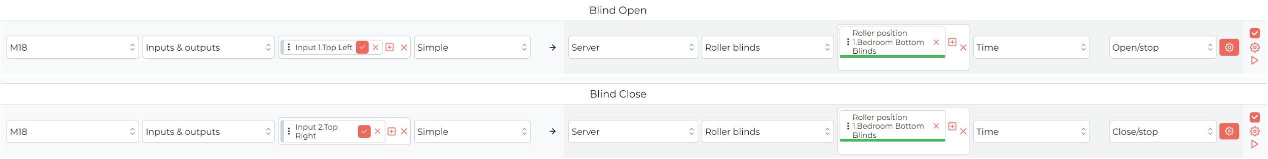
With two inputs a much greater level of control can be achieved.

Any input function can be used (ex. M-SERV-s or M-IN-2p physical inputs), here the M-DOT-M18 touch panel's input will be used.

With two inputs the blind position can be incremented and decremented by a certain step or continuously controlled, two conditions need to be created, one for each input:

1. Select the desired input, with the *Simple* input function
2. Select the appropriate roller blinds controller, with the *Time* output function and type of:

- *Close/stop* and *Open/stop*: will open/close the blind when triggered and stop the blind when triggered again
- *Close on hold* or *Open on hold*: will open/close the blind if the input is true, will stop when the input is false (effectively moves as long as the "button" is pressed)



Blind Open

M18 | Inputs & outputs | Input 1.Top Left | Simple → Server | Roller blinds | Roller position 1.Bedroom Bottom Blinds | Time | Open/stop

Blind Close

M18 | Inputs & outputs | Input 2.Top Right | Simple → Server | Roller blinds | Roller position 1.Bedroom Bottom Blinds | Time | Close/stop

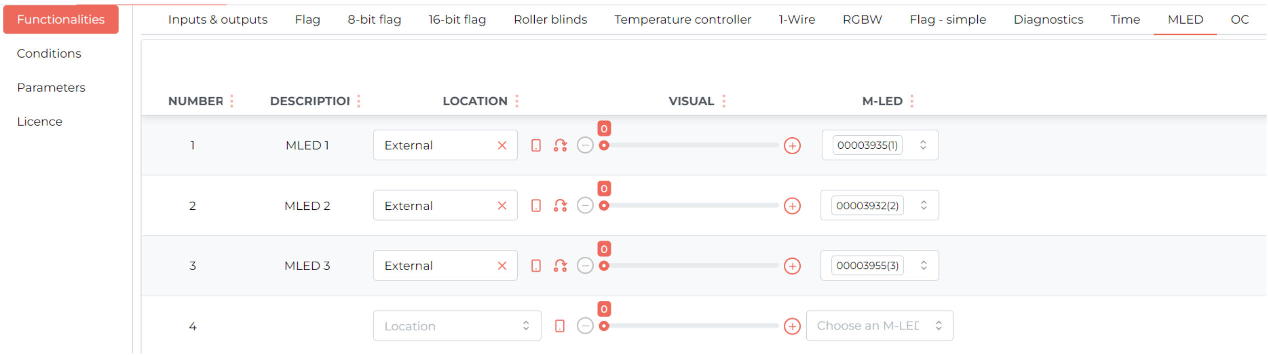
In the example above when the top-left touch field is touched the blind will start opening or when the top-right field is touched the blind will start closing. If any of the fields is touched while the blind is in motion it will stop.

OWA LED light dimmer

The One Wire Ampio lighting bus is a very convenient and cost-effective solution for your lighting needs, allowing you to connect up to 16 individually addressable lighting nodes with just 3 wires, one for supply voltage, one for data and one for common ground connection.

> Most of the features presented for One Wire Ampio lighting and the M-LED lighting driver are also applicable to Open Collector (OC) outputs.

Once connected the recognised lights should be available in the M-LED driver's *Functionalities* tab in the device's configuration. For this demonstration, the M-LED driver of the M-SERV-s module will be used:



NUMBER	DESCRIPTIOI	LOCATION	VISUAL	M-LED
1	MLED 1	External	0	00003935(1)
2	MLED 2	External	0	00003932(2)
3	MLED 3	External	0	00003955(3)
4		Location	0	Choose an M-LEI

Here 3 lighting nodes are connected, their addresses are assigned to numbers 1, 2 and 3, respectively. Multiple addresses can be assigned to one number for group control.

Since only one MLED driver board is provided in this kit further instruction will be considering only single light functionalities.

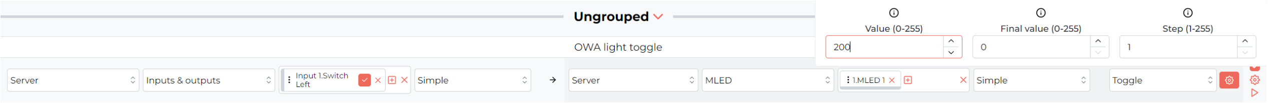
All lights connected to the bus have dimming capability, so they can not only be turned ON and OFF, but also their brightness can be easily adjusted.

Dimming can be achieved in many ways, for thw purpose of this document a simple binary input, like a switch, button, touch panel field, etc. will be used. In the examples to follow, momentary switches connected to the inputs of the M-SERV-s will be used.

One-button control

A single input is very convenient for everyday lighting needs and with some logic both toggling the light and dimming with the same button.

A simple light toggle:



OWA light toggle

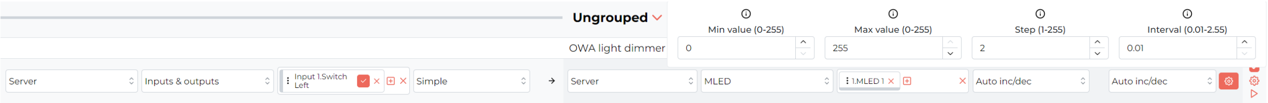
Server | Inputs & outputs | Input 1.Switch Left | Simple → Server | MLED | 1.MLED 1 | Simple | Toggle

Value (0-255): 200
Final value (0-255): 0
Step (1-255): 1

The max brightness was adjusted below the device maximum. Also, *Step* can be adjusted to control the brightness ramp up and down speed when switching the light.

Now when the button is pressed the light will toggle ON and OFF.

Let us add the dimming functionality, this can be done by changing the output function to *Auto inc/dec* (automatic increment / decrement):



Ungrouped

OWA light dimmer

Min value (0-255): 0

Max value (0-255): 255

Step (1-255): 2

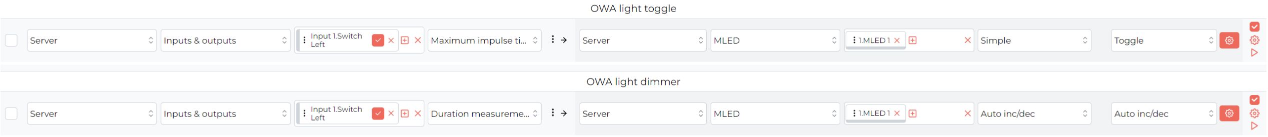
Interval (0.01-2.55): 0.01

Server | Inputs & outputs | Input 1.Switch Left | Simple → Server | MLED | 1.MLED 1 | Auto inc/dec | Auto inc/dec

Now when the button is held, the light will slowly increase in brightness until the button is released or the light reaches the configured max brightness, then, when the button is pushed again the light will decrease in brightness until the button is released or it reaches the configured minimum brightness.

The *Step* and *Interval* parameters can be adjusted to control the brightness ramp up and down speed.

Since the dimming works when the button is held, to avoid condition interference between the functions let’s change the toggle’s input function to *Maximum impulse time* and *Duration measurement *for the dimmer, this will stop the functions from running together, toggle will work for short presses and dimmer will trigger only after a certain time threshold (here the threshold is 0.5s for both input functions):



OWA light toggle

Server | Inputs & outputs | Input 1.Switch Left | Maximum impulse ti... → Server | MLED | 1.MLED 1 | Simple | Toggle

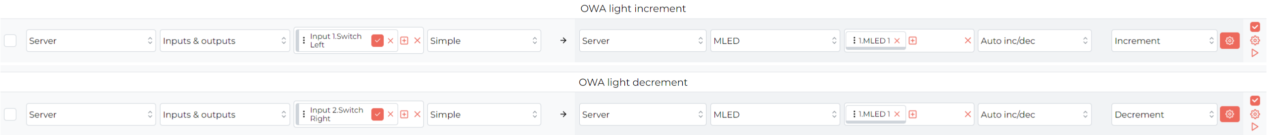
OWA light dimmer

Server | Inputs & outputs | Input 1.Switch Left | Duration measureme... → Server | MLED | 1.MLED 1 | Auto inc/dec | Auto inc/dec

Two-button control

With the *Auto inc/dec* function sometimes it can be hard to know whether the light will get brighter or dimmer when the button is pressed. With two buttons the brightness ramp up and down functions can be split between them for finer control of lights, whose brightness needs to be accurately set.

2 conditions need to be created, one for incrementing and one for decrementing the brightness value:



OWA light increment

Server | Inputs & outputs | Input 1.Switch Left | Simple → Server | MLED | 1.MLED 1 | Auto inc/dec | Increment

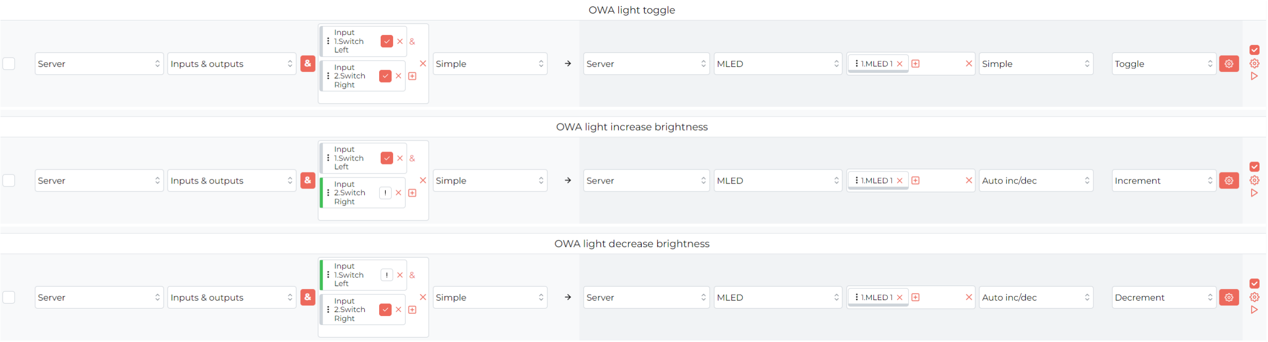
OWA light decrement

Server | Inputs & outputs | Input 2.Switch Right | Simple → Server | MLED | 1.MLED 1 | Auto inc/dec | Decrement

Now *Switch Left* (input 1) will increase the light’s brightness and *Switch Right* (input 2) will decrease it.

To allow light toggling without time restrictions like the ones used with the single button control style the *AND* function will be used to only trigger the toggle when both buttons are pressed at the same time.

To avoid condition collisions an interlock will be used on the dimmers to make them work only when **one** of the buttons is pressed, this will also utilise the *AND* function, but with one of the switches’ inputs inverted to prevent the function from working when **both** buttons are pressed:



OWA light toggle

Server | Inputs & outputs | & | Input 1.Switch Left & Input 2.Switch Right | Simple → Server | MLED | 1.MLED 1 | Simple | Toggle

OWA light increase brightness

Server | Inputs & outputs | & | Input 1.Switch Left & Input 2.Switch Right ! | Simple → Server | MLED | 1.MLED 1 | Auto inc/dec | Increment

OWA light decrease brightness

Server | Inputs & outputs | & | Input 1.Switch Left ! & Input 2.Switch Right | Simple → Server | MLED | 1.MLED 1 | Auto inc/dec | Decrement

Now when only *Switch Left* is held the light will get brighter, when only the *Switch Right* is held the light will get dimmer and when both are pressed at the same time the light will toggle ON and OFF.

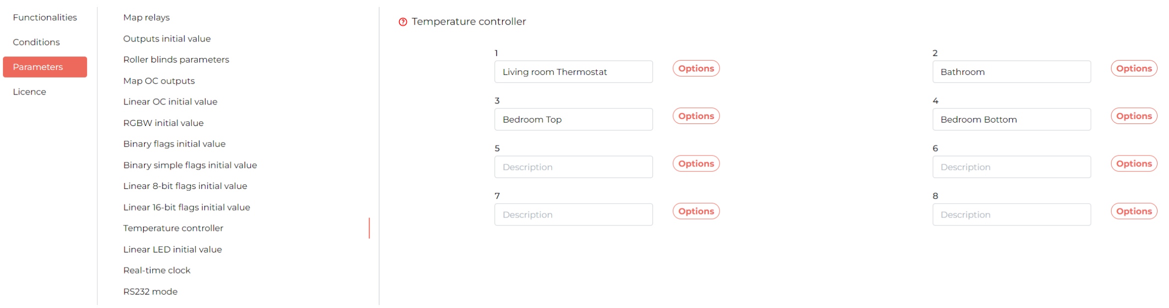
Temperature controller

A *temperature controller* is the universal solution for keeping your rooms at exactly the temperature you want.

The M-SERV devices have a built-in *temperature controller* which can handle from 1 to 32 individual heating zones (depending on the license purchased). Together with the fact that most of our devices support an input for 1-Wire temperature sensors, setting up temperature control with our *temperature controller* is very convenient.

Setting up the controller's parameters

To configure how the temperature controller should behave, enter the M-SERV's device configuration and in *Parameters -> Temperature controller*:



Temperature controller

1 Living room Thermostat

2 Bathroom

3 Bedroom Top

4 Bedroom Bottom

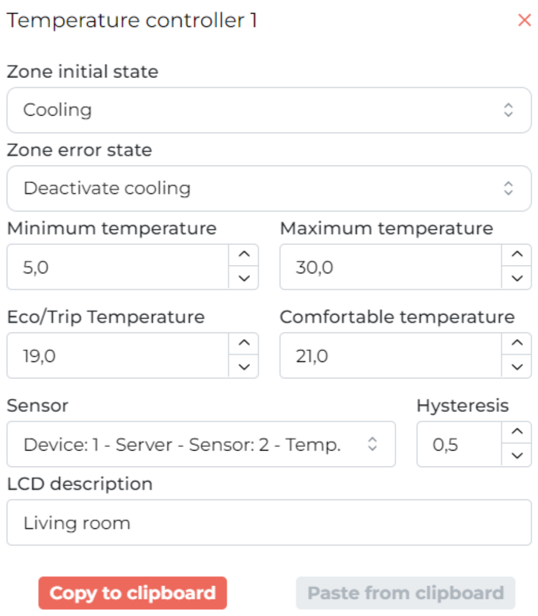
5 Description

6 Description

7 Description

8 Description

Here select one of the available zones for configuration, for the purpose of this example the *Living room Thermostat* will be configured:



Temperature controller 1

Zone initial state: Cooling

Zone error state: Deactivate cooling

Minimum temperature: 5,0

Maximum temperature: 30,0

Eco/Trip Temperature: 19,0

Comfortable temperature: 21,0

Sensor: Device: 1 - Server - Sensor: 2 - Temp.

Hysteresis: 0,5

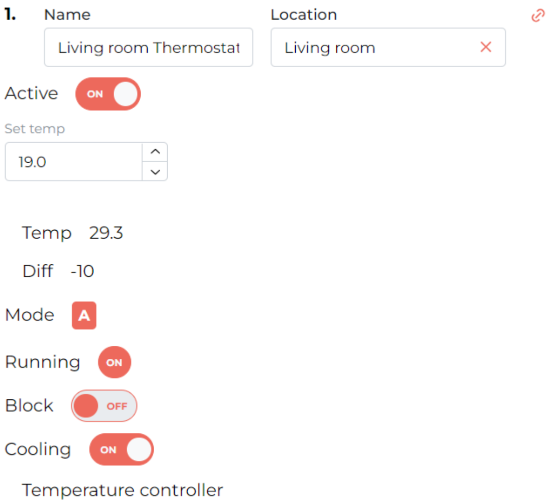
LCD description: Living room

A couple of parameters can be configured:

- Zone initial state – the mode the controller will enter after recovering from power loss: Heating or Cooling
- Zone error state – the operating state the controller will enter after temperature data vanishes from the network: Activate/Deactivate heating/cooling
- Minimum and Maximum temperature – the hard temperature limits the controller will keep, setting a temperature outside those limits will not work and if the temperature falls outside of them the controller will be forced on to compensate

- Eco/Trip temperature – the temperature preset for not being present in the space, usually a value set to save power
- Comfortable temperature – the default temperature preset
- Sensor – the temperature reading selected as input for the controller
- Hysteresis – the “dead zone” set so that the controller will not toggle its state for insignificant temperature changes
- LCD description – the text displayed on a touch panel’s LCD screen when the temperature controller is selected as the display source

To see the current state of the temperature controller can be seen in the *Functionalities* tab:

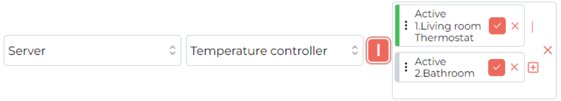


1. Name: Living room Thermostat
Location: Living room
Active: ON
Set temp: 19.0
Temp 29.3
Diff -10
Mode A
Running ON
Block OFF
Cooling ON
Temperature controller

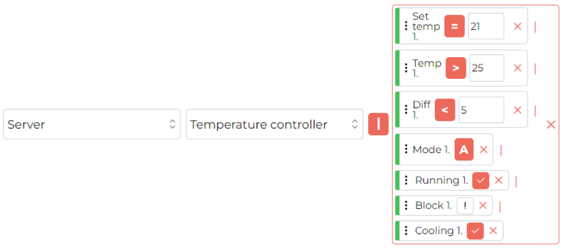
Working with the temperature controller

To utilise the temperature controller, logical conditions need to be created using the different parameters that the temperature controller provides for each individual zone.

Here is a list of the parameters that the controller makes available in conditions:



Server | Temperature controller | I
Active 1.Living room Thermostat ✓
Active 2.Bathroom ✓



Server | Temperature controller | I
Set temp 1. = 21
Temp 1. > 25
Diff 1. < 5
Mode 1. A
Running 1. ✓
Block 1. !
Cooling 1. ✓

- Active – whether a zone is active or not (multiple zones can be selected for the condition, but no additional parameters)
- Parameters in use for only 1 zone at a time:
  - Running – whether the zone is running (ordering temperature change)
  - Cooling – whether the zone is in cooling (true) or heating (false) mode
  - Set temp – the set temperature
  - Temp – the current temperature reading
  - Diff – the temperature difference between the set and the current temperature
  - Mode – in which mode is the controller operating:

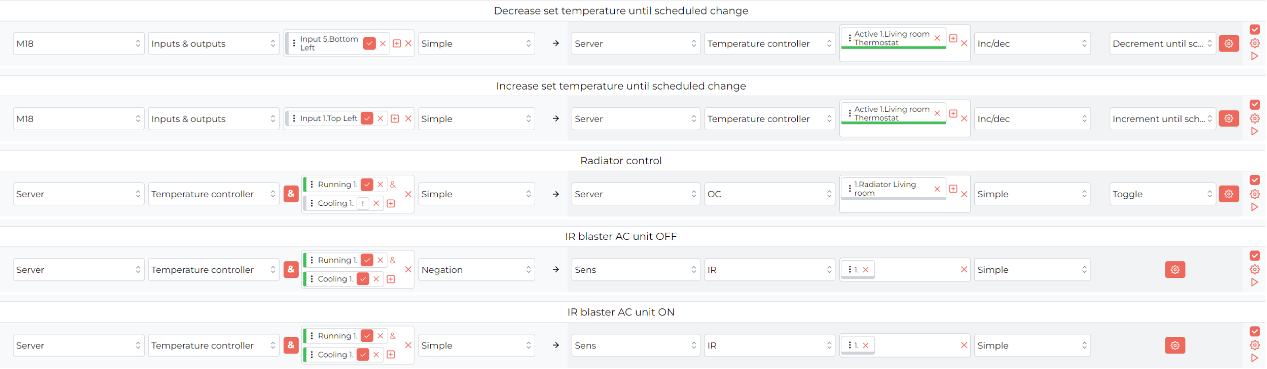
  - ⋆ A – automatic, according to schedule
  - ⋆ S – semi-automatic, set manually until scheduled change
  - ⋆ M – manual, user set temperature
  - ⋆ H – holidays, permanently in Eco/Trip temperature
- Block – whether the zone is in blocking mode (will stop the controller from running)

Using those conditions integrated heaters, radiators and AC units can be controlled.

The states and parameters of the temperature controller can also be set and adjusted using logic, we recommend looking through the different options available and testing them out.

Heating and cooling

With a few conditions a simple temperature control system can be set up for a room. When combined with one of our M-DOT series touch panels as a display and for control purposes, a great all-in-one temperature control package can be created:



Decrease set temperature until scheduled change

M18	Inputs & outputs	Input 5.Bottom Left	Simple	→	Server	Temperature controller	Active 1.Living room Thermostat	Inc/dec	Decrement until sc...

Increase set temperature until scheduled change

M18	Inputs & outputs	Input 1.Top Left	Simple	→	Server	Temperature controller	Active 1.Living room Thermostat	Inc/dec	Increment until sch...

Radiator control

Server	Temperature controller	&	Running 1. & Cooling 1. !	Simple	→	Server	OC	1.Radiator Living room	Simple	Toggle

IR blaster AC unit OFF

Server	Temperature controller	&	Running 1. & Cooling 1.	Negation	→	Sens	IR	1.	Simple

IR blaster AC unit ON

Server	Temperature controller	&	Running 1. & Cooling 1.	Simple	→	Sens	IR	1.	Simple

In this example the touch panel with the temperature controller display is used to adjust the set temperature.

When the zone is running **and not** in cooling mode the radiator will be ON.

When the zone is running **and** in cooling mode the *Simple* input function sends an IR code from the M-SENS unit to turn ON the AC on the *rising edge *(when the condition is evaluated **true**) and the *Negation* input function sends an IR code from the M-SENS unit to turn OFF the AC on the *falling edge* (when the condition is evaluated **false**).

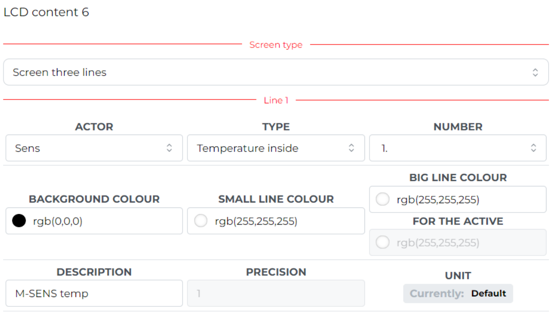
Displaying the M-SENS-LITE reported values

The M-SENS-LITE reports many useful parameters, which can be displayed to the user on a touch panel's LCD display.

The values can be selected in the M-DOT-M18 device parameters under the LCD tab.

Select the screen you want to use and the screen type of "Screen four icons" if used with additional controls in mind or "Screen three lines" for more space to display data.

Here is an example of displaying the temperature measurement from the M-SENS-LITE module on the panel's LCD screen using "Screen three lines":



LCD content 6

Screen type

Screen three lines

Line 1

ACTOR	TYPE	NUMBER
Sens	Temperature inside	1.

BACKGROUND COLOUR	SMALL LINE COLOUR	BIG LINE COLOUR
● rgb(0,0,0)	○ rgb(255,255,255)	○ rgb(255,255,255)
		FOR THE ACTIVE
		○ rgb(255,255,255)

DESCRIPTION	PRECISION	UNIT
M-SENS temp	1	Currently: Default

Using the touch panel timer

The *timer screen* is a particularly useful and simple tool available within the touch panel.

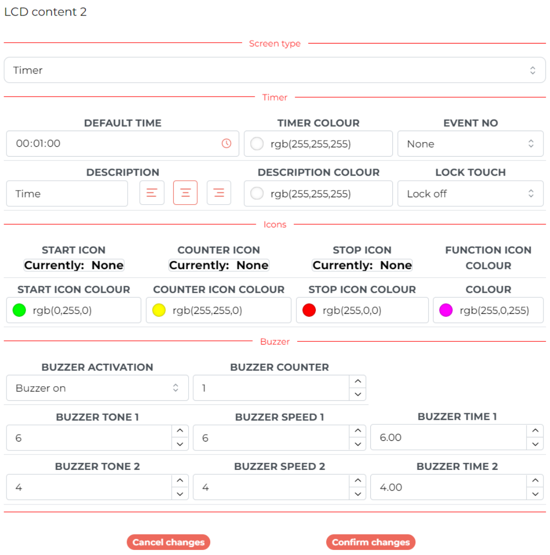
This screen allows the user to start an adjustable timer, which will trigger the panel's buzzer, or an *event*, when it runs out.

Events are signals broadcast to the entire network, to which other devices can respond, they can be defined under *Resources* in the *Settings* tab.

The timer's user interface is quite simple:

- **Top buttons** (*decrease time* and *increase time*) - Used to adjust the timer outside of its default time
- **Counter** (middle) - Displays the timer's description, time remaining and a progress bar
- **Bottom buttons** (*stop timer* and *start/pause timer*) - The *start timer* button (right) will begin the countdown, during which it will change to the *pause timer* button, which will pause the countdown at its current state, the *stop timer* button (left) will reset the timer

To add a timer screen, go to the M-DOT-M18 configuration menu, select the *Parameter -> LCD* tab, select the screen you want and select the "Timer" screen type. There are a couple of parameters to configure:



LCD content 2

Screen type: Timer

Timer

DEFAULT TIME: 00:01:00
TIMER COLOUR: rgb(255,255,255)
EVENT NO: None
DESCRIPTION: Time
DESCRIPTION COLOUR: rgb(255,255,255)
LOCK TOUCH: Lock off

Icons

START ICON Currently: None
COUNTER ICON Currently: None
STOP ICON Currently: None
FUNCTION ICON COLOUR
START ICON COLOUR: rgb(0,255,0)
COUNTER ICON COLOUR: rgb(255,255,0)
STOP ICON COLOUR: rgb(255,0,0)
COLOUR: rgb(255,0,255)

Buzzer

BUZZER ACTIVATION: Buzzer on
BUZZER COUNTER: 1
BUZZER TONE 1: 6
BUZZER SPEED 1: 6
BUZZER TIME 1: 6.00
BUZZER TONE 2: 4
BUZZER SPEED 2: 4
BUZZER TIME 2: 4.00

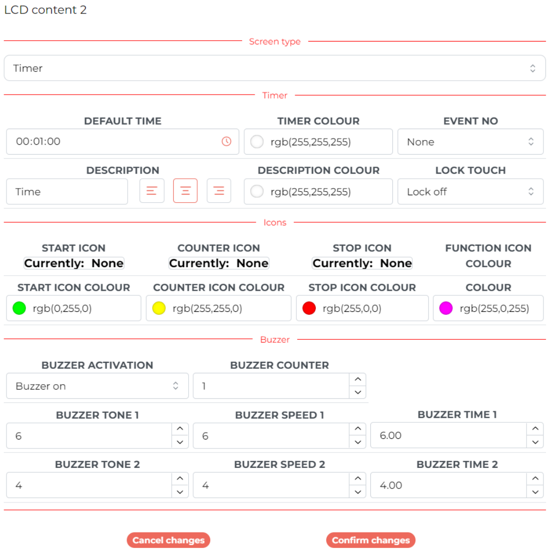
- **Default time**: The time that will be set when first entering the screen and its colour
- **Description**: The text displayed above the timer, the colour and text alignment can also be selected
- **Event No**: The event to trigger when time tuns out
- **Lock Touch**: Whether to lock the touch screen during the timer's operation or not
- **Timer icon** – The icon between the top two touch fields, if no specific icon is selected, it will be a clock:
  - **Start icon**: The icon before the timer starts and its colour
  - **Counter icon**: The icon during the timer's countdown and its colour
  - **Stop icon**: The icon when the time runs out and its colour
- **Function colour**: The colour of the stop and start buttons
- **Buzzer** – When the ***buzzer is active***, upon the time running out it will switch back and forth between two tones, this will loop for repeat the number of times specified by the Buzzer Counter parameter:
  - **Buzzer Tone**: The buzzer's pitch
  - **Buzzer Speed**: The ramp up/down speed when switching tones
  - **Buzzer Time**: For how long the tone will be held

After configuring the timer how you want *confirm changes* and *send parameters to the module*. Now the timer will be available on one of the touch panel's screens.

PIN-enabled relay

The M-DOT-M18 touch panel has the unique ability to broadcast events upon entering a predefined PIN code.

As mentioned in the previous section, *events* are signals broadcast to the entire network, to which other devices can respond. For the purpose of this example, an event “PIN 1” was created.



LCD content 2

Screen type: Timer

Timer

DEFAULT TIME: 00:01:00
TIMER COLOUR: rgb(255,255,255)
EVENT NO: None
DESCRIPTION: Time
DESCRIPTION COLOUR: rgb(255,255,255)
LOCK TOUCH: Lock off

Icons

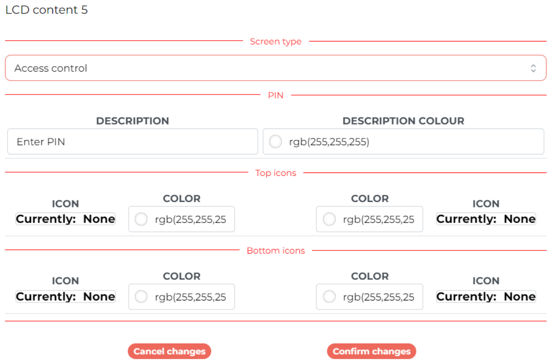
START ICON Currently: None
COUNTER ICON Currently: None
STOP ICON Currently: None
FUNCTION ICON COLOUR
START ICON COLOUR: rgb(0,255,0)
COUNTER ICON COLOUR: rgb(255,255,0)
STOP ICON COLOUR: rgb(255,0,0)
COLOUR: rgb(255,0,255)

Buzzer

BUZZER ACTIVATION: Buzzer on
BUZZER COUNTER: 1
BUZZER TONE 1: 6
BUZZER SPEED 1: 6
BUZZER TIME 1: 6.00
BUZZER TONE 2: 4
BUZZER SPEED 2: 4
BUZZER TIME 2: 4.00

Next, in the M-DOT-M18 configuration menu, select the *Parameter -> LCD* tab, select the screen you want and select the ”Access control” screen type, type in the ”Description” field a message you want to display above the PIN input field.

In addition, icons can be selected in a similar fashion to the “Screen four icons”, this will be omitted here.



LCD content 5

Screen type: Access control

PIN

DESCRIPTION: Enter PIN
DESCRIPTION COLOUR: rgb(255,255,255)

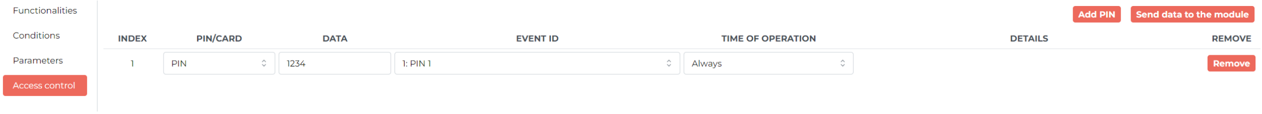
Top icons

ICON Currently: None
COLOR: rgb(255,255,25
COLOR: rgb(255,255,25
ICON Currently: None

Bottom icons

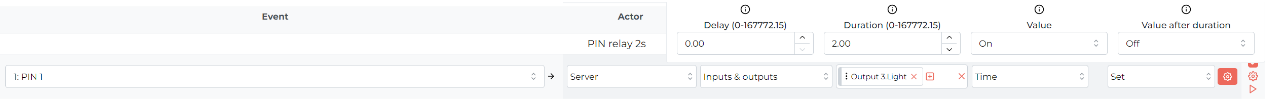
ICON Currently: None
COLOR: rgb(255,255,25
COLOR: rgb(255,255,25
ICON Currently: None

Next, in the *Access control* tab add a new PIN code and select the event to broadcast upon entering the PIN code.



INDEX	PIN/CARD	DATA	EVENT ID	TIME OF OPERATION	DETAILS	REMOVE
1	PIN	1234	1: PIN 1	Always		Remove

Finally, a piece of logic is needed to use the *event call*, here a relay will be switched ON for 2 seconds upon entering the PIN code:



Event	Actor	Delay (0-167772.15)	Duration (0-167772.15)	Value	Value after duration
	PIN relay 2s	0.00	2.00	On	Off
1: PIN 1 →	Server Inputs & outputs	Output 3.Light		Time	Set

Now every time a PIN of 1234 is entered into the ”Access control” screen, the relay will turn ON for 2 seconds.

The action called does not necessarily have to be a relay trip and can trigger any action you programme in the *Logic tab*.