

Let's make Remote Thermometer

an example tutorial

Creating new IoT devices with Click Cloud and G2C is super easy! In order to show you how fast and simple it can be, we've created an example project. Let's make two devices work together via Click Cloud will show you an example of a setup that involves sending the data from a Click board to the Click Cloud, and receiving the data from the Cloud back with another Click board. Let's get started!



To our valued customers

I want to express my thanks to you for being interested in our products and for having confidence in MikroElektronika.

The primary aim of our company is to design and produce high quality electronic products and to constantly improve the performance thereof in order to better suit your needs.

A handwritten signature in black ink, appearing to read 'Nebojsa Matic', with a stylized, cursive script.

Nebojsa Matic
General Manager

Table of Contents

01. Collect your Hardware	4
02. The Preparation	6
03. Thermo 7 device node	7
04. Creating the device in the web application –Thermo 7 click	8
05. Program the Thermo 7 node device application	14
06. Bargraph device node	16
07. Creating the device in the web application – Bargraph click	17
08. Program the Bargraph node device application	18
09. Configuration of the Rule Engine	20
10. Conclusion	25

01. Collect your hardware

This project demonstrates a simple example which shows how the Click Cloud can help in automatization of a certain operation. Let's have 2 device nodes:

- one that will serve as a sensor which will send its measurements to the cloud, and
- the other one as an actuator which will receive the instructions FROM the cloud for performing a certain operation

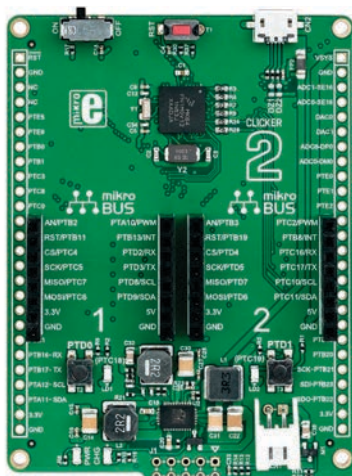
The device nodes will interact through a rule engine, available on the Click Cloud clickcloud.io

The absolutely necessary precondition is that you have already created the account on the Click Cloud.

Out of HARDWARE components, for this Let's make project you'll need:

- 1 x Thermo 7 click
- 1 x BarGraph click
- 2 x G2C click
- 2 x Clicker 2 for Kinetis

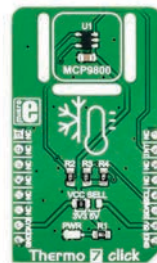
Hardware tools



2x Clicker 2 for Kinetis



2x Go to Cloud
[G2C] click



1x Thermo 7
click



1x Bargraph
click

02. The preparation

For the preparation, you will need to download the libraries for the aforementioned Click boards™ from Libstock:

- [*Go to Cloud \(G2C\) click library*](#)
- [*Thermo 7 click library*](#)
- [*BarGraph click library*](#)

As you are already accustomed, there are standard demo applications which can be used with every click for the sake of testing. If you wish so, you can upload these to the system and test the functionality of the BarGraph click and Thermo 7 click. Beside the standard libraries, you'll need to download and install the already created demo projects:

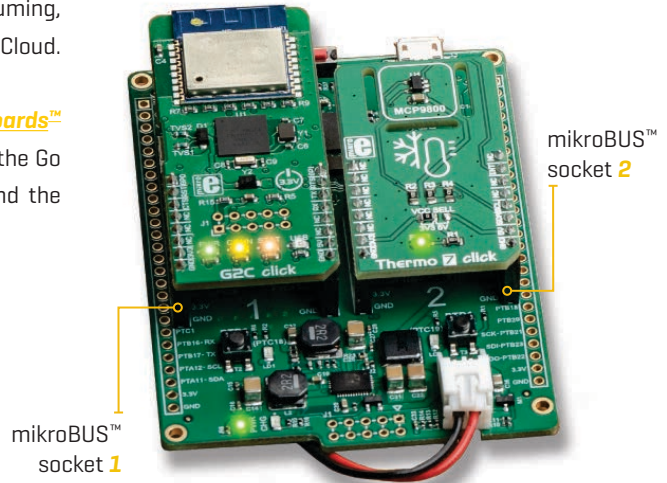
- [*Thermo 7 Cloud demo*](#)
- [*BarGraph Cloud demo*](#)

These demo projects already contain the source code which will make the further steps and the demonstration easier.

03. Thermo 7 device node

First, you will need to assemble the hardware. Let's assemble the Thermo 7 device node which will, as you are assuming, send the temperature measurement data to the Click Cloud.

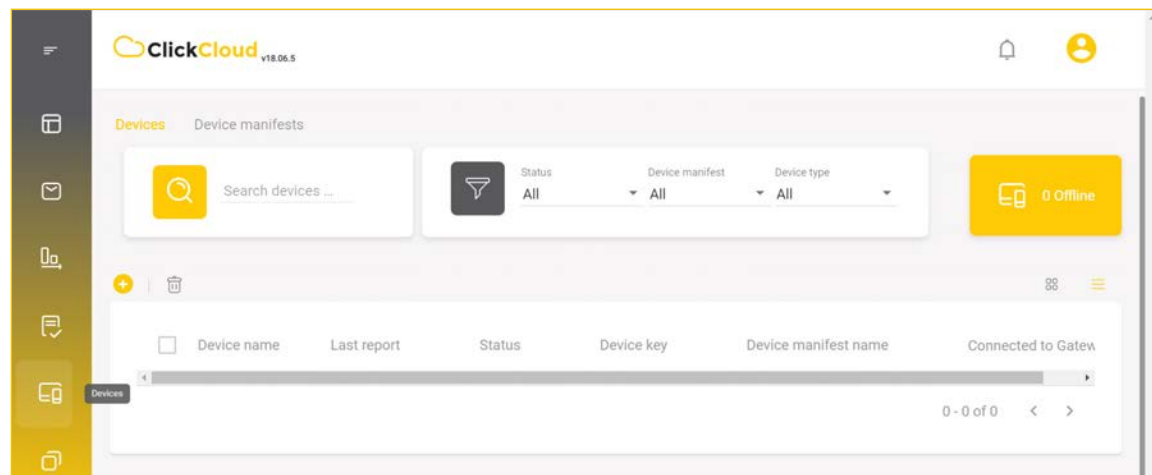
Thanks to the modularity provided by the [Click boards™](#) and the [mikroBUS™](#) socket, all you need to do is set the Go to Cloud [G2C] click to the mikroBUS™ socket **1**, and the Thermo 7 click to the socket **2**.



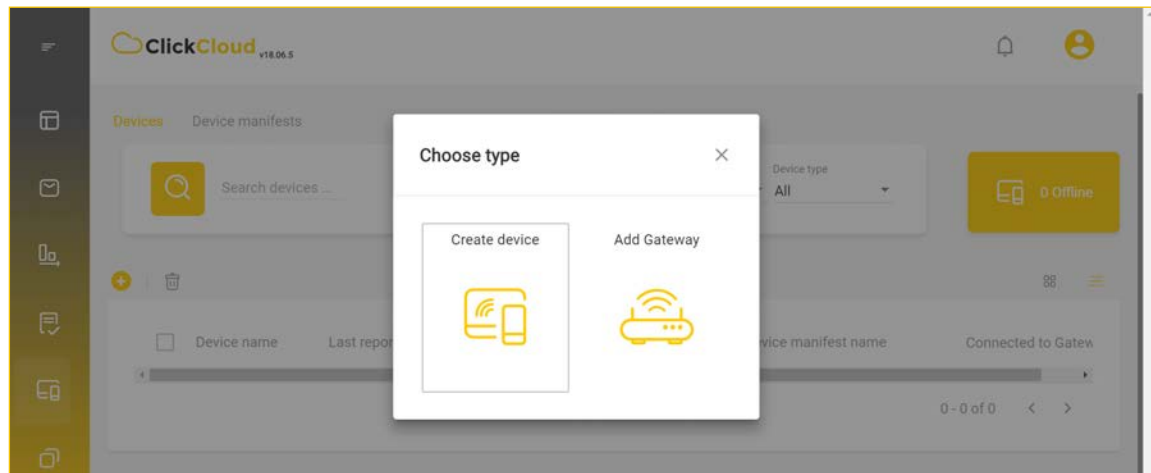
04. Creating the device in the web application

After you've assembled the hardware components you will need to create the device on the cloud application which is equivalent to the hardware you just assembled. After you've logged in to the web application, you'll need to:

1. Go to the Devices in the navigation menu



2. Click on Add device (the + icon), and then on Create Device



3. Choose Thermo 7 click from the drop down list by clicking the Device manifest in the rightside of the window

The screenshot shows the ClickCloud v18.06.5 web interface. On the left is a dark sidebar with icons for navigation. The main header area displays the ClickCloud logo and version. Below the header, the breadcrumb 'Devices > Create' is visible. The primary section is titled '1 Device manifest' and contains a large text input field. To the right of this field is a dropdown menu labeled 'Device manifest'. The dropdown is open, showing a search bar with 'Thermo 7' entered and a list of results, with 'Thermo 7 click' selected. Below the input field, a message reads 'Please choose one of the existing device manifests.' and a 'Next step' button is located at the bottom right. A secondary section at the bottom is titled '2 Create device(s)'.

4. Click on the Next step

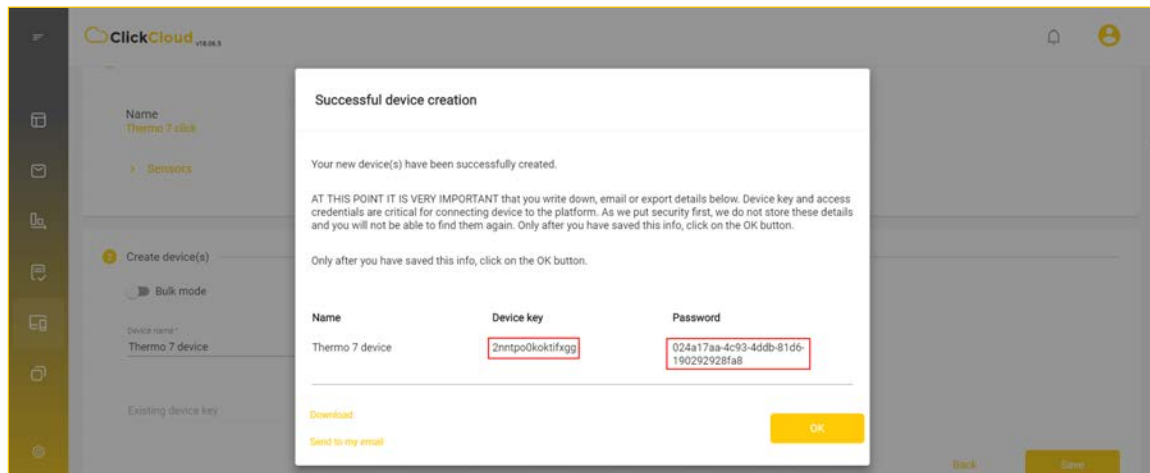
5. Assign the Device Name

6. Click Save

The screenshot shows the ClickCloud v18.06.5 web interface. On the left is a dark sidebar with various icons. The main content area has a header with the ClickCloud logo and version. Below the header, a breadcrumb trail reads 'Devices > Create'. The 'Create' section is divided into two steps. Step 1, 'Device manifest', is active and shows a 'Name' field with the value 'Thermo 7 click' and a '> Sensors' link. Step 2, 'Create device(s)', is below and shows a 'Bulk mode' toggle switch and a 'Device name*' field with the value 'Thermo 7 device'.

7. Copy and paste the Device key and Password which will appear on the screen, or use some of the options offered in the dialogue [Download/Send via email]

8. Click OK



Then, you will be taken back to the list of devices where you can see the newly - created device with the name you assigned in the step no. 5. Just for the test, you can click on it to open it and to see the details about the device. You'll notice the Data tab. If you click on it you'll see that the Last Value field is empty at the moment, which is expected as the device has just been set and it has never before received any data.

Devices window view

The screenshot displays the ClickCloud v18.06.5 interface. The top header features the ClickCloud logo and a user profile icon. The main content area is divided into two tabs: "Devices" (active) and "Device manifests". Below the tabs, there is a search bar with a magnifying glass icon and the text "Search devices by device name and...". To the right of the search bar are three filter dropdowns: "Status" (set to "All"), "Device manifest" (set to "All"), and "Device type" (set to "All"). Further right is a yellow button with a lock icon and the text "2 Offline". Below the filters is a table with the following columns: "Device name", "Last report", "Status", "Device key", "Device manifest name", and "Connected to Gateway". The table contains two rows of data:

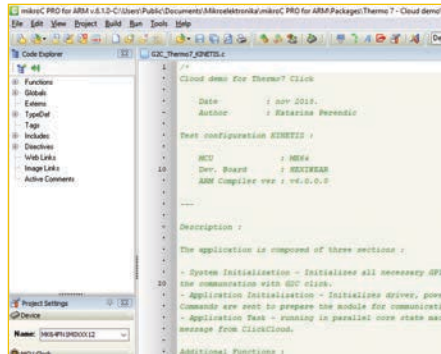
<input type="checkbox"/>	Device name	Last report	Status	Device key	Device manifest name	Connected to Gateway
<input type="checkbox"/>	BarGraph device	-	Offline	syqj2g7mxgfjsyd	Bar Graph click	No
<input type="checkbox"/>	Thermo 7 device	-	Offline	2nntpo0koktifxgg	Thermo 7 click	No

At the bottom right of the table, there is a pagination indicator "1 - 2 of 2" with left and right arrow icons.

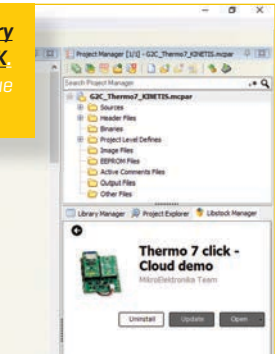
05. Program the Thermo 7 node device application

In this step, you will create the device node embedded application, which will send some data toward the web app (more precisely, to the device we've just created in the previous step) for the first time. All the steps we've enlisted here are performed from IDE only.

1. Install the Thermo 7 Cloud demo package which contains the code for the demo application

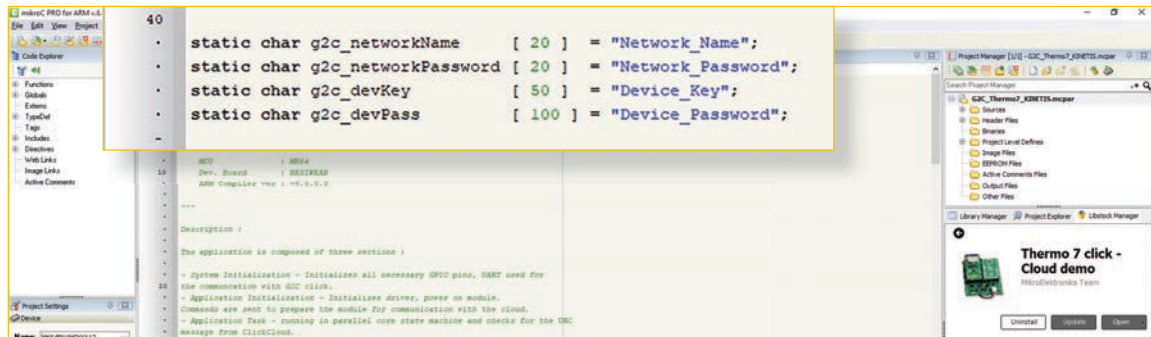


NOTE: You have to pre-install **Thermo 7 click library** and **Go to cloud [G2C] click** and **mikroSDK**. With this devices you can easily continue with your project!



2. Open the project from the Examples tab

3. Change the parameters for the Wi-Fi network in the src code of the demo application [g2c_networkName, g2c_networkPassword]
4. Paste the credentials which we've received in the step no. 7 of the previous section [g2c_devKey, g2c_devPass]



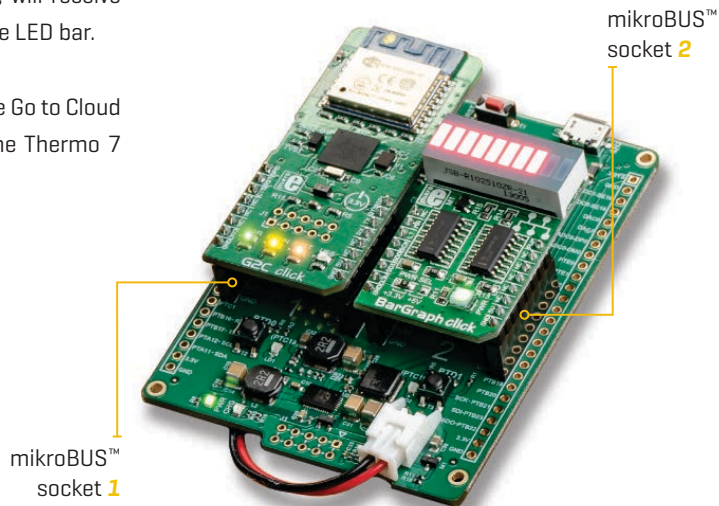
5. Rebuild the demo by clicking the "rebuild all" (Alt + F9)
6. Program the MCU on the system (F11)

After this, the first data will start arriving at the cloud. If you go back to the web browser and repeat the operations from the last paragraph of the previous section, you'll see within the details about the device its online status, and that now in the Data field there is some data. These are the measurements from the Thermo 7 node.

06. Bargraph device node

This device node, as we've already mentioned, will receive the data from the cloud and display them on the LED bar.

The device needs to be prepared by inserting the Go to Cloud [G2C] click to the mikroBUS™ socket **1**, and the Thermo 7 click to the socket **2**.



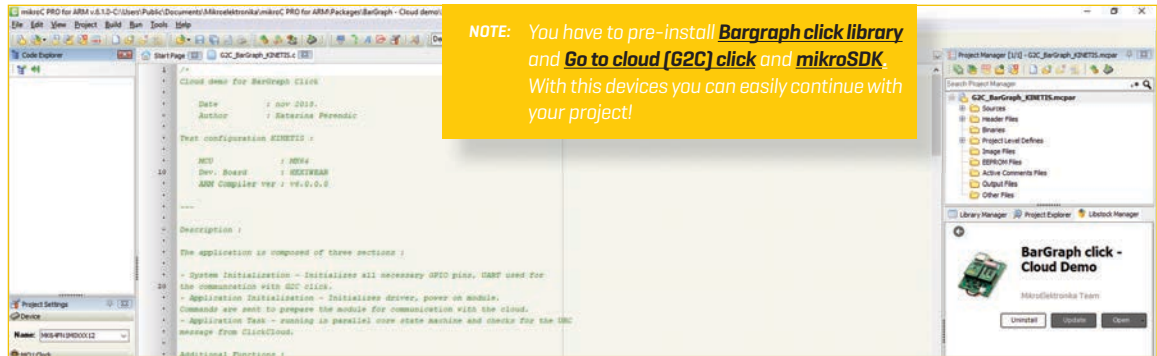
07. Creating the device in the web application

The steps for creating the device are the same as for the Thermo 7 node, with the difference being that you need to choose the BarGraph click from the drop down list and assign a name to the device which is different from the previous device [Thermo 7 node]. After creating the device and returning to the devices list, if you look at the details about the device by clicking on it, you'll notice that there is no Data tab, contrary to the Thermo 7 node. The reason for this is that this device is not sending any measurements to the cloud, but instead the cloud controls the behavior of this device – we will define the behavior on the section titled **Configuration of the Rule Engine**.

08. Program the Bargraph node device application

In this step, we will create the device node embedded application which will receive some data from the web app. All the enlisted steps are performed from IDE only.

1. Install the **Bargraph Cloud demo package** which contains the code for the demo application



2. Open the project from the Examples tab

3. Change the parameters for the Wi-Fi network in the src code of the demo application [g2c_networkName, g2c_networkPassword]
4. Paste the credentials which we've received in the step no. 7 of the previous section [g2c_devKey, g2c_devPass]



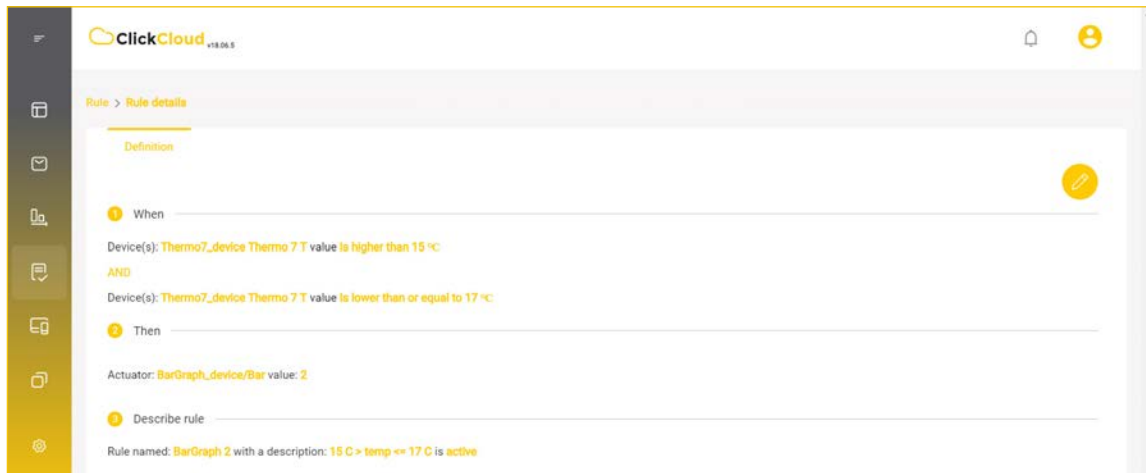
5. Rebuild the demo by clicking the "rebuild all" (Alt + F9)
6. Program the MCU on the system (F11)

After this, if you check out the details about the device on the web application, you'll see that the device is now online, which means it's ready to accept the actuation from the cloud.

09. Configuration of the Rule Engine

After entering the Rule Engine by clicking on Rules in the Navigation menu, after you click on Add Rule +icon, you will enter the Create Rule dialogue. You need to define the condition under which a certain operation will be performed – in this case, the change of the value which the BarGraph shows depending on the temperature measured. The definition of the conditions is done in the **When** section by the following:

1. *Click on the New condition – Add*
2. *Choose the Thermo 7 device by the name which you assigned in the step no. 5 while creating the Thermo 7 node*
3. *For the field Property choose the Thermo 7*
4. *Set the operator to the Lower than*
5. *Static value*
6. *Type in 17 [degrees]*



This way you can set up that a certain operation will be performed in case that the temperature on the Thermo 7 device drops under 17°C. In the next step we will show which is the operation we just mentioned. Click on the Next step. After this, a section titled Then will open, where you will do as follows:

1. Choose “Set / Actuate” for the Action Type
2. Choose the device by the name assigned while creating the BarGraph device
3. Choose the BarGraph actuator
4. Set the value 2

ClickCloud v18.06.5

+ New condition

1 Then

Action type *

Set / actuate

Devices * Actuator * Number between 0 and 10 *

BarGraph_device BarGraph_device/Bar 2

+ New action

2 Describe rule

Rule name * Description * Rule status

BarGraph 2 15 C > temp <= 17 C

Cancel Save

You've just defined the operation which will be performed in case the temperature drops below 17°C, which is that the value of the BarGraph is set to 2, i.e. to illuminate only two field. After clicking on the Next button, a dialogue will open in which you will need to:

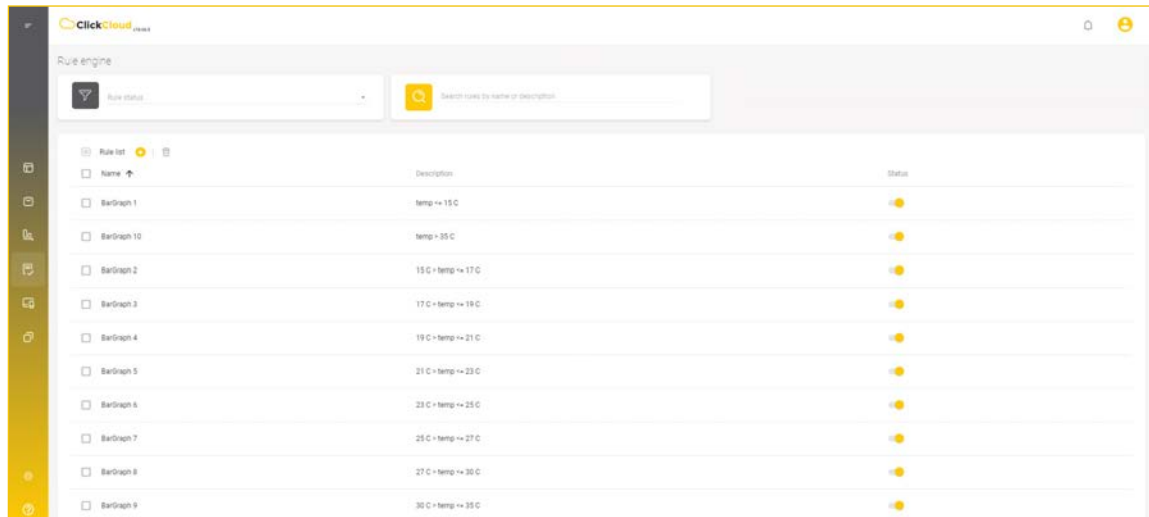
1. **Describe the Rule – for example, “Turn on 2 Bar”**
2. **Write the short description about the rule**
3. **Turn on the rule**
4. **Click on the Create Rule**

With this, you've completed the operation of creating one rule.

The screenshot displays the ClickCloud v19.06.5 interface for editing a rule. The left sidebar contains navigation icons for Home, Alerts, Rules, and Settings. The main content area is titled 'Rule > Rule details' and shows the 'Definition' tab. The rule is defined as follows:

- When:** Device(s): Thermo7_device Thermo 7 T value is higher than 15 °C
- AND**
- When:** Device(s): Thermo7_device Thermo 7 T value is lower than or equal to 17 °C
- Then:** Actuator: BarGraph_device/Bar value: 2
- Describe rule:** Rule named: BarGraph 2 with a description: 15 C > temp <= 17 C is active

In the same way, you can define the rest of the ranges. For example, you can create another rule for which you can set up so that in case the temperature is - or is over 10°C, but lower than 15°C, there should be 2 LED bars on the BarGraph click. The bottom line would be that you need to create the rule for all the 10 statuses which the BarGraph can display. It's up to you which temperature ranges you will set.



The screenshot displays the ClickCloud Rule engine interface. At the top, there's a header with the ClickCloud logo and a 'Rule engine' title. Below the header, there's a search bar with a magnifying glass icon and the text 'Search rules by name or description'. The main area shows a table of rules. The table has three columns: 'Name', 'Description', and 'Status'. There are 10 rules listed, each with a checkbox in the 'Name' column. The rules are for BarGraph 1 through BarGraph 9, and BarGraph 10. The descriptions show temperature ranges in Celsius. The status column shows a yellow circle with a dot inside, indicating the current status.

Name	Description	Status
<input type="checkbox"/> BarGraph 1	temp <= 15 C	0
<input type="checkbox"/> BarGraph 10	temp > 35 C	0
<input type="checkbox"/> BarGraph 2	15 C < temp <= 17 C	1
<input type="checkbox"/> BarGraph 3	17 C < temp <= 19 C	2
<input type="checkbox"/> BarGraph 4	19 C < temp <= 21 C	3
<input type="checkbox"/> BarGraph 5	21 C < temp <= 23 C	4
<input type="checkbox"/> BarGraph 6	23 C < temp <= 25 C	5
<input type="checkbox"/> BarGraph 7	25 C < temp <= 27 C	6
<input type="checkbox"/> BarGraph 8	27 C < temp <= 30 C	7
<input type="checkbox"/> BarGraph 9	30 C < temp <= 35 C	8

10. Conclusion

The final result will be the BarGraph click which shows the momentary measurements of the temperature at the location where the Thermo 7 device is set. The precondition is, of course, availability of a Wi-Fi network at the location where you're measuring the temperature. During this Let's make project, you've probably noticed other options such as choosing another actions to be performed – sending an email for example, or a push notification. It's also important to stress that it's possible to define custom devices by creating the manifests of your choice.

It's also important to emphasize that thanks to the [mikroSDK™](#), the already created projects can be used for the development systems by your choice, by simply selecting the system from the [mikroBUS™](#) API.

We've chosen clicker 2 system because of the design of that system and the jack for the Lithium-Ionic battery which provide a degree of liberty, in the sense that the device doesn't have to be near the power supply source.

DISCLAIMER

All the products owned by MikroElektronika are protected by copyright law and international copyright treaty. Therefore, this manual is to be treated as any other copyright material. No part of this manual, including product and software described herein, may be reproduced, stored in a retrieval system, translated or transmitted in any form or by any means, without the prior written permission of MikroElektronika. The manual PDF edition can be printed for private or local use, but not for distribution. Any modification of this manual is prohibited. MikroElektronika provides this manual 'as is' without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose. MikroElektronika shall assume no responsibility or liability for any errors, omissions and inaccuracies that may appear in this manual. In no event shall MikroElektronika, its directors, officers, employees or distributors be liable for any indirect, specific, incidental or consequential damages (including damages for loss of business profits and business information, business interruption or any other pecuniary loss) arising out of the use of this manual or product, even if MikroElektronika has been advised of the possibility of such damages. MikroElektronika reserves the right to change information contained in this manual at any time without prior notice, if necessary.

HIGH RISK ACTIVITIES

The products of MikroElektronika are not fault – tolerant nor designed, manufactured or intended for use or resale as on – line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of Software could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). MikroElektronika and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

TRADEMARKS

The MikroElektronika name and logo, mikroC, mikroBasic, mikroPascal, Visual TFT, Visual GLCD, mikroProg, Ready, MINI, mikroBUS™, EasyPIC, EasyAVR, Easy8051, Click boards™ and mikromedia are trademarks of MikroElektronika. All other trademarks mentioned herein are property of their respective companies. All other product and corporate names appearing in this manual may or may not be registered trademarks or copyrights of their respective companies, and are only used for identification or explanation and to the owners' benefit, with no intent to infringe.

The FTDI Chip® and Windows® logos and product names are trademarks of FTDI Chip and Microsoft® in the U.S.A. and other countries.

Copyright © 2019 MikroElektronika. All Rights Reserved.



If you want to learn more about our products, please visit our website at www.mikroe.com

If you are experiencing some problems with any of our products or just need additional information, please place your ticket at www.mikroe.com/support

If you have any questions, comments or business proposals, do not hesitate to contact us at office@mikroe.com

MIKROE