MyController.org

The Open Source Controller

Author : Jeeva Kandasamy

Doc Version : d0.3
Product Version : 0.0.2-Alpha4
Date : September 2015

Table of Contents

1.0 Introduction....4
1.1 Technology....4
2.0 Installation....4
2.1 System Requirements....4
2.2 Supported Platforms....4
2.3 Download....5
2.4 Configuration....5
2.4.1 Select your gateway....5
2.4.2 Serial Gateway Configuration....5
2.4.3 Ethernet Gateway....5
2.4.4 MQTT Gateway....6
2.4.5 Database Configuration....6
2.4.6 Web server configuration....6
2.4.7 Logger configuration....6
2.4.8 Start/Stop Server....6
2.5 Menus....7
3.0 Nodes and Sensors....7
3.1 Nodes....7
3.1.1 Auto Discover....7
3.1.2 Add Manually....8
Navigate to node page, by clicking icon on top of right side corner of the node table you can add new node....8
3.1.3 Reboot....8
3.1.4 Erase EEPROM....8
3.1.5 Edit....8
3.1.6 Delete....8
3.1.7 Nodes Battery Usage Report....8
3.1.7 Mapping Firmware....9
3.2 Sensors....10
3.2.1 Add Sensor....10
3.2.2 Edit Sensor....10
3.2.3 Delete Sensor....10
3.2.4 Filter....10
4.0 Action Board....11
4.1 Filter....11
4.1.1 By Node....11
4.1.2 By Text....11
4.2 Refresh....12
4.3 ON/OFF....12
4.4 Send Payload....12
4.5 Graphical Report....12
4.5.1 Last 1 hour report....13
4.5.2 Last 24 hours....13
4.5.3 Last 30 days....14
4.5.4 All available data....14

4.6 Timer....15
4.6.1 Normal....15
4.6.2 Cron....15
4.6.2 Before Sunrise, After Sunrise, Before Sunset and After Sunset....16
4.6.3 Frequency....16
4.6.4 Time....16
4.6.5 Payload....16
4.6.6 Validity....16
4.6.7 Delete....16
4.7 Alarm....16
4.7.1 Conditions....17
4.7.2 Notifications....17
4.7.2 Edit....17
4.7.3 Delete....17
4.7.4 Special Operations....18
4.8 Forward Payload....18
4.9 Logs....18
4.10 Edit....19
5.0 UID Tag....19
5.1 Arduino code structure....19
5.1.1 Query Payload....19
5.1.2 Send Payload....20
6.0 Firmware....20
6.1 Adding Firmware....21
6.1.1 Add Firmware Type....21
6.1.2 Add Firmware Version....21
6.1.3 Add Firmware....21
7.0 Users....22
7.1 Manage Users....22
8.0 Logs....22
9.0 System Status....23
10.0 System Settings....24
10.1 Sunrise/Sunset....24
10.2 Node Defaults....24
10.2.1 Auto Node Id....24
10.2.2 Default Firmware....24
10.2.3 Requested firmware absences....24
10.3 Plivo SMS Gateway....25
10.4 Email Gateway....25
10.5 Units/Version Information....25
10.6 Other Settings....26
11.0 Log In/Out....26
12.0 Source Code....27
12.1 Issue management....27
12.2 License....27

1.0 Introduction

MyController.org is a controller for MySensors.org. Only focusing for MySensors.org. MyController.org is a light weight Open Source controller.

You can control/monitor various MySensors.org sensors with this controller. As this controller is a web application you can access from mars also if you have internet connection. It has various features like, firmware control for nodes, alarms, timers., etc. For further details dig into each section.

1.1 Technology

MyController.org is Java based web application server.

- Web Server : TJWS
- REST-API : Jboss RestEasy
- Database : H2DB
- Front-end : AngularJS

2.0 Installation

2.1 System Requirements

MyController.org server is very lightweight, It required very less resource, considered system resources in mind on development. Even you can run it on Raspberry PI (256 MB RAM) very first model.

Required,

- Disk : 30 MB
- RAM : 40 MB
- Java SE : 1.8 or later

NOTE: Test done up to 5 nodes and 30 sensors with the above configuration.

2.2 Supported Platforms

We can run it in any platform which supports Java. So far it has been tested in the following platforms, Kindly share your success stories on other platforms we can add it here.

- Linux
- Windows
- Raspberry PI (Oracle Java recommend)

2.3 Download

Executable download is available in two formats, zip and tar.gz. You can download suitable compressed flavor for you.

Kindly visit releases page of MyController.org to get latest version.

Releases Page

2.4 Configuration

Extract downloaded bundle where exactly do you want to run. Configuration file is located in mycontroller/conf
File name: `mycontroller.properties`

2.4.1 Select your gateway

Three type of gateway supported by MyController.org.

- Serial Gateway
- Ethernet Gateway
- MQTT Gateway (Limited Support)

Available Options: *`serial, ethernet, mqtt`*

Select your preferred gateway by changing the line '`mcc.ethernet.gateway.type=serial`'

2.4.2 Serial Gateway Configuration

```
mcc.serialport.driver.type=auto
mcc.serialport.name=/dev/ttyUSB0
mcc.serialport.baud.rate=115200
```

- **mcc.serialport.driver.type:** Available driver types: pi4j, jssc, jserialcomm, auto. It is recommended to keep in auto. For now jssc driver is not stable with MyController.org
- **mcc.serialport.name :** will vary based on platform and number of ports. Find it on your computer and update
- **mcc.serialport.baud.rate :** by default Serial Gateway uses 115200. If you changed on your setup you have to update your custom value here.

Note: *To connect NRF24L01+ device directly with Raspberry PI module, Execute steps from https://github.com/mysensors/Raspberry(Discussions: http://forum.mysensors.org/topic/1151/tutorial-raspberry-pi-nrf24l01-direct-connection). This will create new serial port like **/dev/ttyMySensorsGateway** you can use this serial port in serial gateway configuration.*

2.4.3 Ethernet Gateway

```
mcc.ethernet.gateway.host=192.168.178.6
mcc.ethernet.gateway.port=5003
mcc.ethernet.gateway.keep.alive.frequency=60
```

- **mcc.ethernet.gateway.host:** Host name/ip of your Ethernet Gateway
- **mcc.ethernet.gateway.port:** Ethernet Gateway port
- **mcc.ethernet.gateway.keep.alive.frequency:** This field is used to monitor Ethernet Gateway. If Connection goes try to reconnect automatically. Value in seconds.

2.4.4 MQTT Gateway

```
mcc.mqtt.gateway.broker.host=192.168.178.6
mcc.mqtt.gateway.broker.port=1883
mcc.mqtt.gateway.broker.root.topic=MyMQTT
```

- **mcc.mqtt.gateway.broker.host:** MQTT Gateway Hostname/IP
- **mcc.mqtt.gateway.broker.port:** MQTT Gateway port
- **mcc.mqtt.gateway.broker.root.topic:** Topic name, Should be same name that we used in MQTT Gateway(MQTT_BROKER_PREFIX)

2.4.5 Database Configuration

```
mcc.h2db.location=../conf/mycontroller
```

You can change default location and file name. File will be stored with the file extension .h2.db. Do not add file extension.

2.4.6 Web server configuration

```
enable.https=true
http.port=8443
ssl.keystore.file=../conf/keystore.jks
ssl.keystore.password=password
ssl.keystore.type=JKS
```

- **enable.https:** Enable/disable https. Only one protocol supported at a time.
- **http.port:** Port number you want to access MyController.org server
- **ssl.keystore.file, ssl.keystore.password, ssl.keystore.type:** If https is enabled these fields are mandatory.

Default URL: https://<ip>:8443 (ex: https://localhost:8443)
Note: *Default username/password:* admin/admin

Important: Change default ssl.keystore.file and ssl.keystore.password and https protocol is recommended

2.4.7 Logger configuration

Configuration File Name: `logback.xml`
Default log file location: `logs/mycontroller.log`

2.4.8 Start/Stop Server

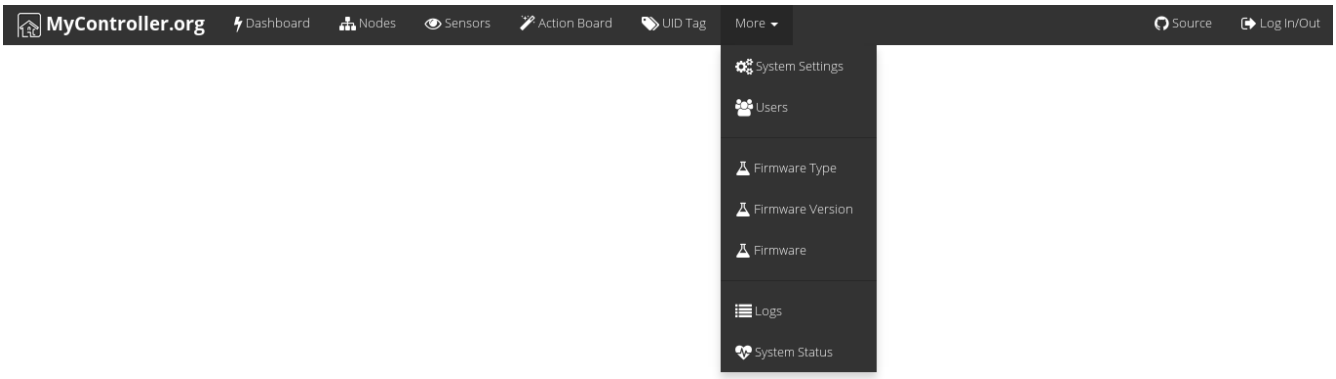
Executable scripts are located in mycontroller/bin/

- Linux
  - `Start    : ./start.sh`
  - `Stop     : ./stop.sh`

- Windows
  - `Start : Double click on start.bat`
  - `Stop  : Ctrl+C`

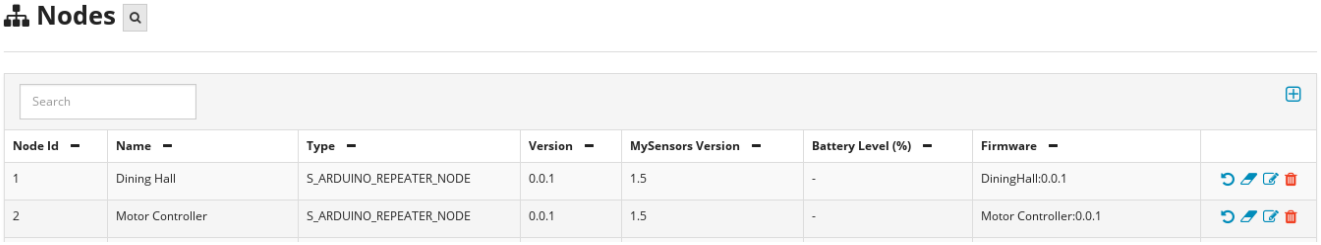
- Other Platforms
  - `navigate to 'mycontroller/bin/'`
  - `java -Xms8m -Xmx40m -Dlogback.configurationFile=../conf/logback.xml -Dmc.conf.file=../conf/mycontroller.properties -jar ../lib/mycontroller-standalone-0.0.1-single.jar`

2.5 Menus



3.0 Nodes and Sensors

3.1 Nodes



Under this page you can see node information's like, Node Id, Node Name, Node Type, Firmware Version, MySensors Version, Battery Level, Assigned firmware.

3.1.1 Auto Discover

Nodes will be added in to MyController.org when you restart your nodes. Also created an entry for node when we receive a sensor data from any node.

Also you have an option to discover all the nodes on the network by clicking icon right side of header(Nodes)
Note: *"Discover Nodes" will send REBOOT command to all the nodes (1 to 254).*

3.1.2 Add Manually

Navigate to node page, by clicking icon on top of right side corner of the node table you can add new node.

3.1.3 Reboot

Click on the icon Then confirm Reboot. Selected node will be rebooted.

3.1.4 Erase EEPROM

Click on the icon and confirm. Selected node configuration will be erased.
NOTE: 'Erase EEPROM' supports only if your node running with MYSBootloader

3.1.5 Edit

If you thing you want to change node name, type, version etc., click on the icon and follow the pop-up screen.

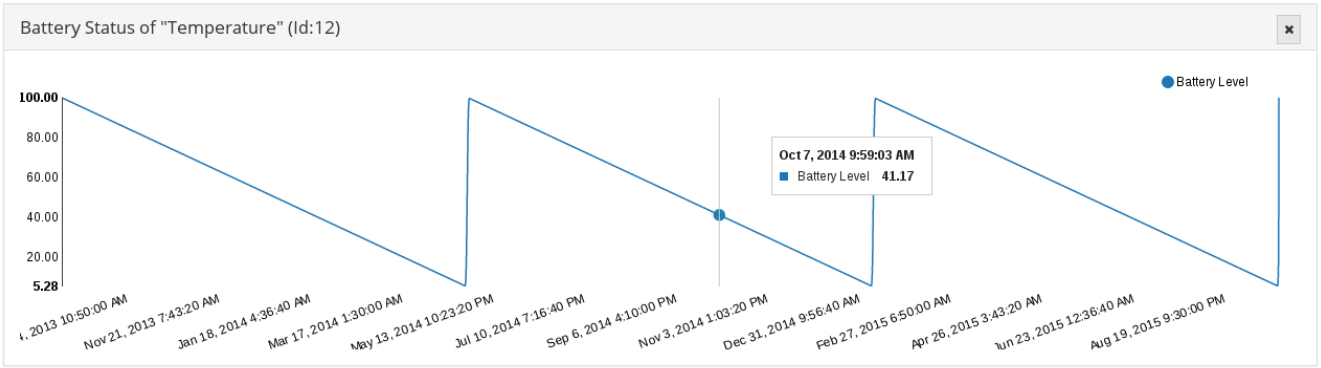
3.1.6 Delete

By clicking the icon of selected node, you can delete that node from MyController.org.
Note: *Deletion node will delete all the data relevance to that node. We cannot recover it back.*

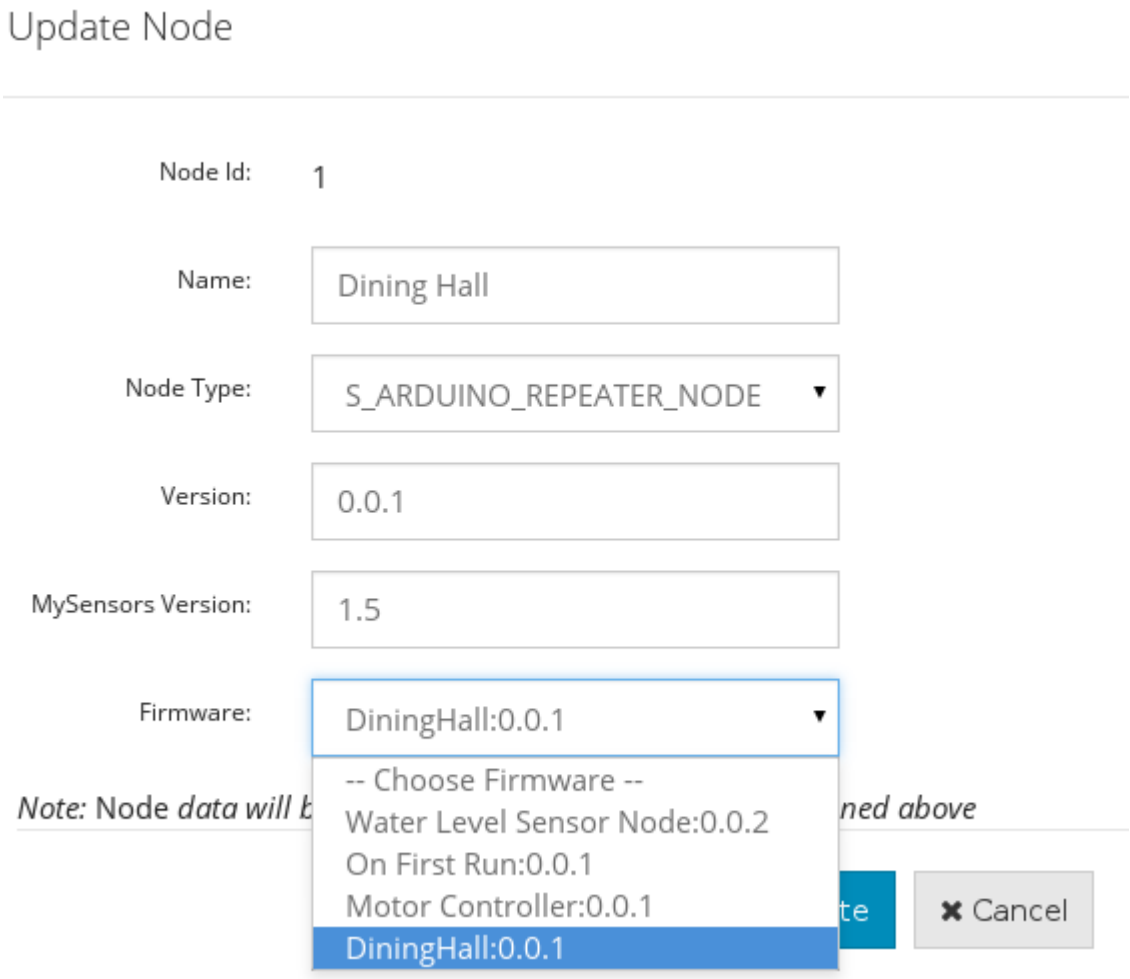
3.1.7 Nodes Battery Usage Report

Node Id	Name	Type	Version	MySensors Version	Battery Level (%)	Firmware	
12	Temperature	S_ARDUINO_NODE	0.0.3	1.5	100.0 %		

If your node sends battery level to MyController.org server you can see current battery percentage under *"Battery Level"* column. Also it supports to view change history as graphical report. By clicking the icon you can view graphical chart on pop up screen as like below.

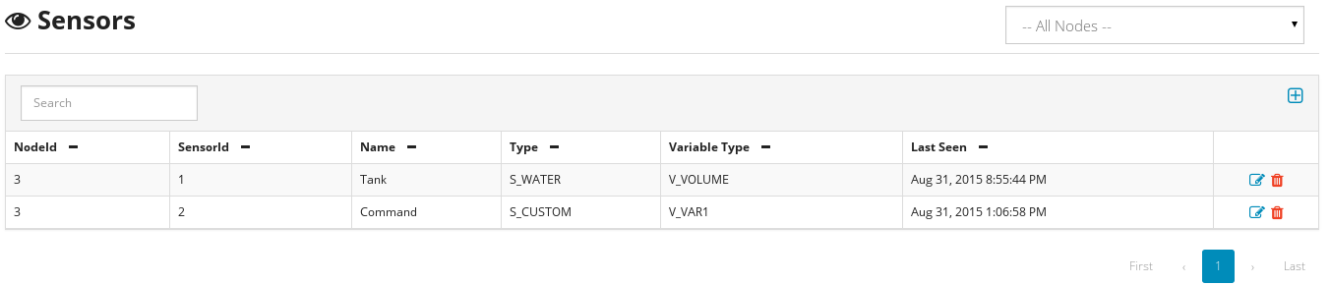


3.1.7 Mapping Firmware



Before doing this action, firmware should be added in MyController.org. Refer ***6.0 Firmware***
3.1.5 Edit and set firmware. To update in to selected node perform ***3.1.3 Reboot***

3.2 Sensors



Navigate to sensors Page. In this page you see sensor details like Node Id, Sensor Id, Name, Type, Variable Type, Last seen. You can add, edit and delete sensors from here.

3.2.1 Add Sensor

We can add sensors in two methods, via node and manually. If node sends any data related to sensors and if the sensor detail is not available in MyController.org new sensor will added automatically.
To add new sensor manually click on the icon follow pop-up screen.
Note: *If MyController.org receives any data related to sensor that you have added already will be overwritten.*

3.2.2 Edit Sensor

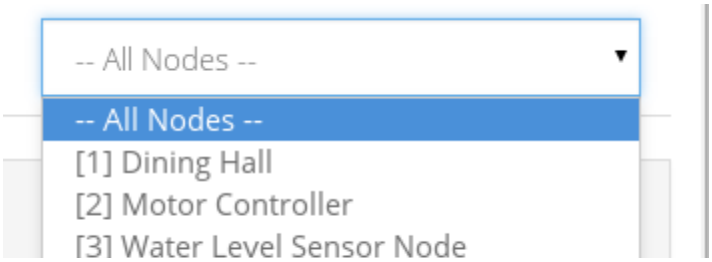
You can edit sensor details by clicking icon you can update sensor Name, Type and variable type.

3.2.3 Delete Sensor

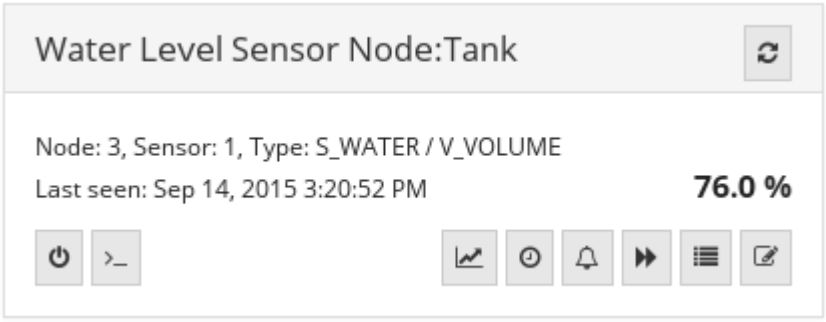
By clicking the icon of selected sensor will be deleted permanently.
Note: *Deletion sensor will delete all the data relevance to that sensor. We cannot recover it back.*

3.2.4 Filter

Two types of filters are available.
1. Filter by Node, Select particular node to filter by node. 2. Filter by text, on left top of



4.0 Action Board

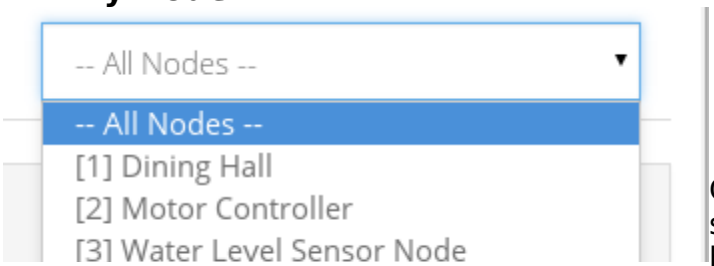


Action board is a very important page of MyController.org. After your initial setup you will spend most of the time here. All the sensor related actions and monitoring will be performed on this page.

4.1 Filter

We can filter sensors by two options. By node and by text

4.1.1 By Node



Click on “-- All Nodes --” drop down and select your favorite node. Sensors will be displayed only on that node.

4.1.2 By Text

Suppose you want to filter only switch type sensors, type status on filter box.

4.2 Refresh

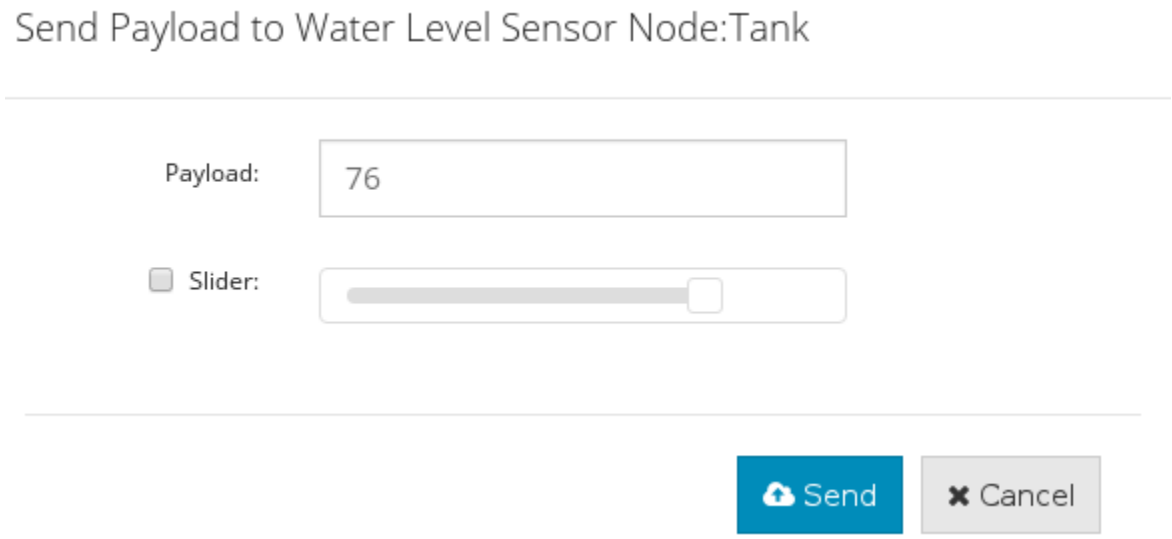
Particular Sensor data will be refreshed by clicking refresh icon on right side top corner of the action board.

4.3 ON/OFF

ON will send payload '1' and OFF will send payload '0'. Check the current status and proceed.

4.4 Send Payload

You may want to send payload other than 1 and 0. in that case you have to use send payload option. Click on the icon you will get pop-up.



You can specify any number, string or anything then click on “Send” button. That's it you data will be sent to your sensor.
If you want use slider check on “Slider” check box it will allow you send only 0 to 100 number. This is mainly used for dimmers. When you release slider tracking button data will be sent immediately.

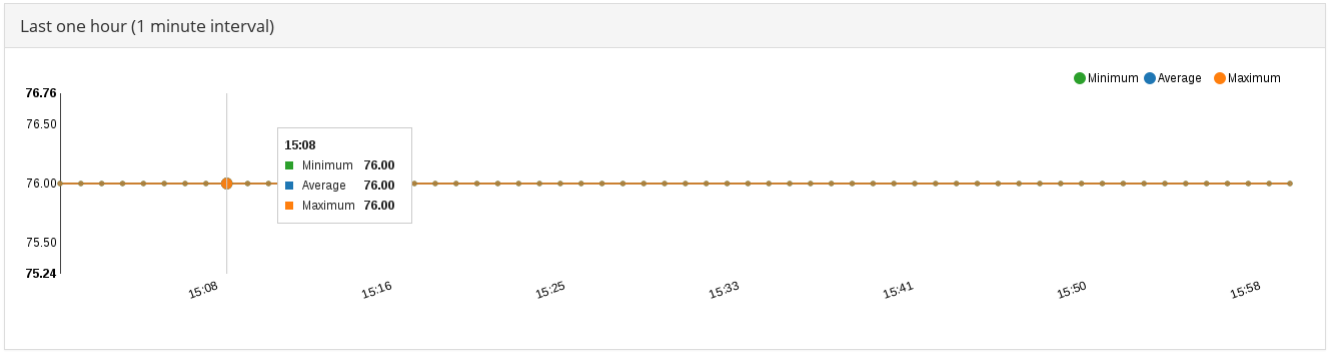
4.5 Graphical Report

MyController.org supports four type of duration's in graph.

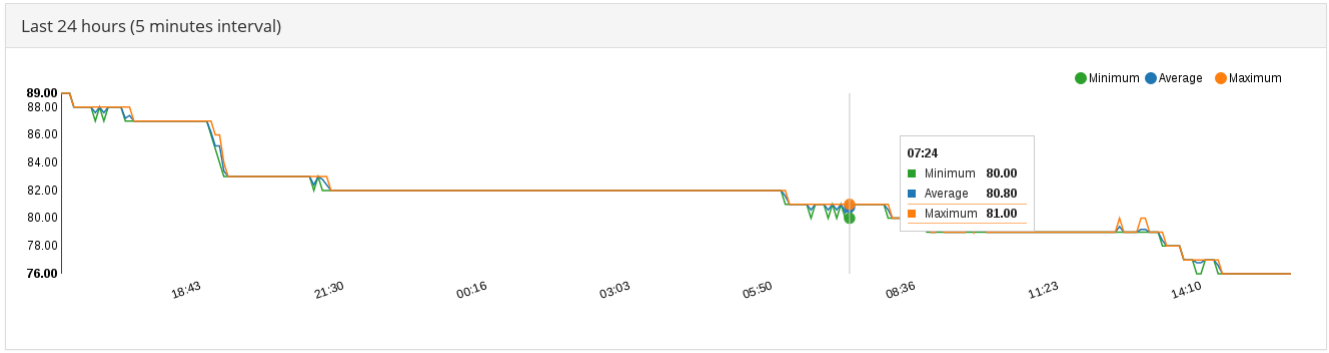
1. Last 1 hour
2. Last 24 hours
3. Last 30 days
4. All available data

For switch type sensors there is no aggregation performed hence data will be shown as is. But other than switch type data (example: temperature, voltage, current, etc.,) will be aggregated stored. For last 1 hour data - 1 minutes interval, last 24 hours – 5 minutes interval, last 30 days – 1 hour interval, and all available data – 1 day interval.
Clicking on the icon  you see graphical report of the sensor.

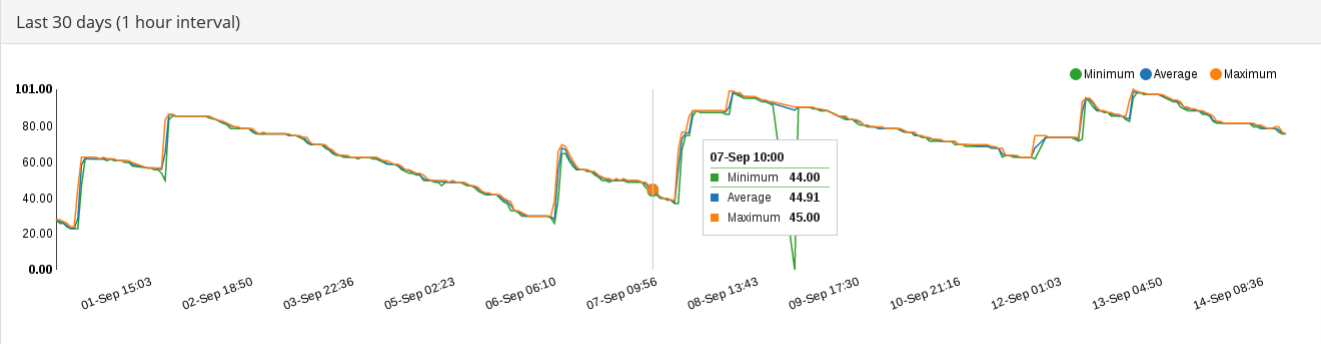
4.5.1 Last 1 hour report



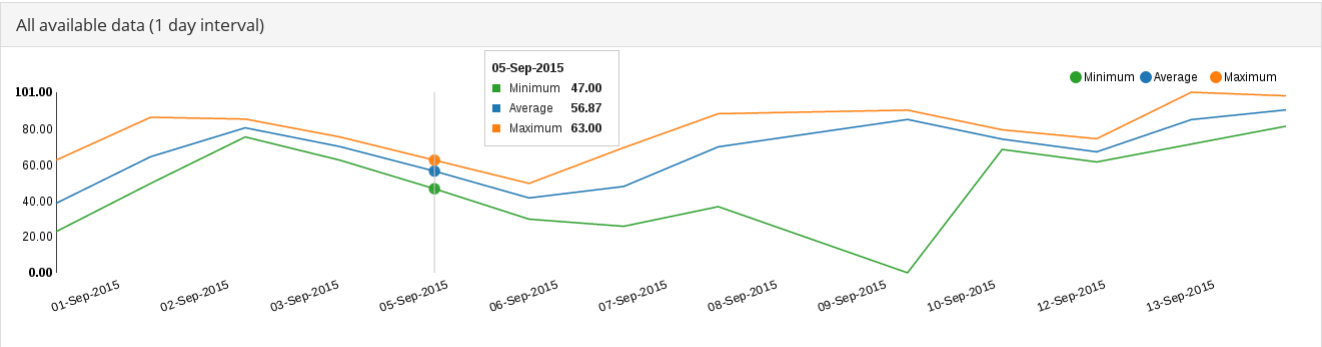
4.5.2 Last 24 hours



4.5.3 Last 30 days

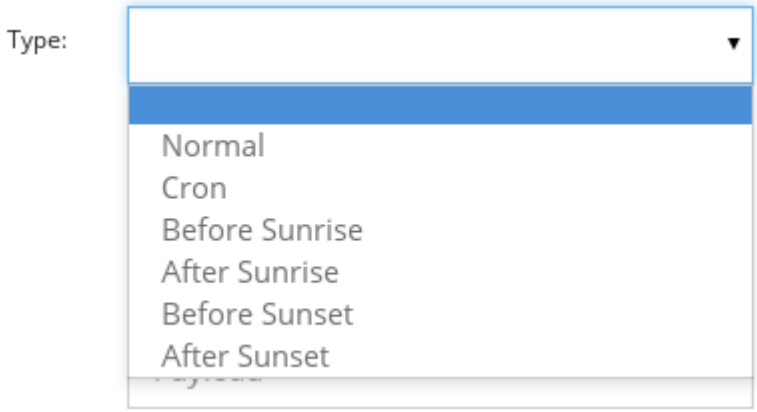


4.5.4 All available data



4.6 Timer

With the timer you can schedule your sensors task and you can forget about that sensors. Different types of timers are supported by MyController.org. ***You can schedule a timer till seconds.*** *Means you can schedule a task for time 21:45:23 like that.*
By Clicking  icon you will be taken to timer page for that particular sensor. Once you have entered in to timer page you can add any number of new timers  by clicking on icon



4.6.1 Normal

Normal is a very basic and classic timer. You can select “Normal” in the type drop down.

4.6.2 Cron

Cron is for advanced configuration. It is simple and easy. After selected “Cron” if you put cursor on cron expression text box, you will get pop-up with detailed tips. Here we are using quartz-scheduler cron tab, visit Quartz-Scheduler page for further detailed configuration
Note: *Quartz-Scheduler* cron supports from seconds.

4.6.2 Before Sunrise, After Sunrise, Before Sunset and After Sunset

If your task based on Sunrise and Sunset you have to go with this option. Say you want to control your garden light based on your sun rise and sun set time. You can use this option.

4.6.3 Frequency

3 types of frequencies supported by MyController.org

1. Daily – you can select all the 7 days or day(s) only you want to run
2. Weekly – Select a day in week
3. Monthly – Select a day in month

4.6.4 Time

```
Time format: HH:mm:ss
HH – hour in 24 hours format (0~23)
mm – minutes (0~59)
ss – seconds (0~59)
```
Note: *For sunrise and sunset options “Time” will react differently. If you select After Sunrise and After Sunset time will be added with “Time” mentioned in task + “Sunrise” or “Sunset” time. If you select “Before Sunrise or Before Sunset” “Time” mentioned in task will be subtracted from actual “Sunrise or Sunset” time.*

4.6.5 Payload

Set payload for that timer. When your timer triggers payload entered here will sent to selected sensor. Payload supports special operations also, have look for detailed information ***4.7.4 Special Operations***

4.6.6 Validity

You may feel you do not want to run this job all the time and want to run only on particular window period. In that case you can select validity. You can select “Validity From” and “Validity To” or only either or nothing. If you do not select any validity that job will be treating like never end. If you select only “Validity From” job will run from that date and never end. If you select only “Validity To” that job will start immediately and will end on the specified date.

4.6.7 Delete

You can delete your timer at anytime by clicking on icon.
Note: *There is no option to edit your timer, if you thing you want to edit delete your timer first and recreate newly with your changes.*

4.7 Alarm

To get alarm page click on the icon . With alarm feature you can control your sensors or you can get notified. Once you have entered in to alarm page you can add any number of new alarms by clicking on icon

4.7.1 Conditions

Alarm supported for >, >=, <, <=, = and != you can select one of option from drop down. Threshold value will be anything integer, string or double.
Note: *for ON/OFF device 0 and 1 should be a threshold value. If you enter ON or OFF will treat as string and it never fire.*
Ignore Duplicate: By default Ignore Duplicate will be enabled. It means if you receive same data again and again do not trigger alarm again and again. But for some cases which is required, in that case you have to uncheck “Ignore Duplicate”.
Dampening: With dampening you can control your alarm, Three modes are available in dampening.
1. None
2. Consecutive
3. Last N Evaluations

None: If you select “None”, alarm will triggers on first occurrence.
Consecutive: On this mode threshold data should receive continuously on X time(s), if it happens alarm will triggers.
Last N Evaluations: On this mode you can specify number of evaluations and number of hits. Based on this alarm will triggers.

4.7.2 Notifications

Three types of notifications are supported by MyController.org alarm.

- SMS

- Email
- Send Payload

SMS: For SMS MyController.org is using Plivo SMS API. When you create trail account you will get some credit. To use SMS notifications you should configured Plivo SMS API settings under “System Settings”. Give destination numbers with '+' with country code then mobile number without any space. If you want to give more than one number use comma(,)

Email: You can specify email address(s) with comma separated.

Send Payload: this is powerful mode. Using this mode you can control self/other sensor directly by send payload when specified alarm triggers.

4.7.2 Edit

You can edit alarm details by clicking icon then you can update.

4.7.3 Delete

By clicking the icon of selected alarm will be deleted permanently.

4.7.4 Special Operations

While defining payload you can assign following special operations, All the special operation reads last received/sent value from target senor and doing this operation on top of that value and sends to target sensor.

Invert: By assigning the value: **!** You can select this operation. It is doing invert operation. This will be useful for BINARY devices. Example is switch is ON it will be turned OFF vise versa.

Increment: By assigning the value: **++** You can select this operation. Adding 1 with the value. Example: last rx/tx value is 45, on resulting this operation will send 46 to target sensor.

Decrements: By assigning the value: **--** You can select this operation. Subtracting 1 with the value. Example: last rx/tx value is 45, on resulting this operation will send 44 to target sensor.

Addition: By assigning the value: **+3** You can select this operation. Here **+** meant for addition and **3** is the value should add. Example: last rx/tx value is 45, on resulting this operation will send 48 to target sensor.

Subtraction: By assigning the value: **-4** You can select this operation. Here **-** meant for addition and 3 is the value should add. Example: last rx/tx value is 45, on resulting this operation will send 48 to target sensor.

Multiplication: By assigning the value: ***2** You can select this operation. Here * meant for multiplication and 2 is the value should multiple. Example: last rx/tx value is 45, on resulting this operation will send 90 to target sensor.

Division: By assigning the value: **/3** You can select this operation. Here / meant for division and 3 is the value should divide by. Example: last rx/tx value is 45, on resulting this operation will send 15 to target sensor.

Modulus: By assigning the value: **%4** You can select this operation. Here % meant for modulus and 3 is the value used for modulus. Example: last rx/tx value is 45, on resulting this operation will send 1 to target sensor.

Reboot: By assigning the value: **reboot** You can select this operation. On this operation target sensor node will be rebooted.

4.8 Forward Payload

By clicking icon  you will get in to forward payload page.

You can forward the data received by this sensor to another sensor directly without any condition. This will be useful when you want to send your sensor date to multiple sensors. No need to do any code change on sensor side. Simply add an entry in MyController.org and be happy MyController.org will take care.

4.9 Logs

By clicking icon  you will get in to logs page.
In this page you can understand whats going on about a particular sensor.

4.10 Edit

By clicking icon  you will get in to Edit pop-up.
You can add/modify sensors unit here.

5.0 UID Tag

Do you want to control all the sensors from one sensor? for this you have to remember each node id and sensor id and you have to add tones of Arduino code in a control sensor which is very difficult for you? To make life easier MyController.org has UID Tag support. ***Unique Identification Tag*** for each sensor. With this UID Tag you can assign unique ID for each sensor and you can forget actual node and sensor id :)

Click on the icon  you can add new UID for a sensor. You will get a pop-up to enter sensor and UID details. You can assign any number, but should be unique. Only one number per sensor.

5.1 Arduino code structure

To use this UID Tag feature you should do some minimal code in MySensors node. From a node you can control all other sensors with UID.
Use case: You can assign unique id's all the switches in your home and from a node you can control either remote control or key pad.
UID tags struct in MySensor (Arduino) code.

```
typedef struct {
    uint16_t uid;
    uint16_t status;
    uint16_t payload;
} UID_STRUCT;
```

uid : unique id that you created in MyController.org

status : whether you want to query or you want to send payload?
Payload : set payload

5.1.1 Query Payload

To query a sensor status with UID, “status” should be 0.

```
gw.present(SEN_UID_ID, S_CUSTOM , "UID Sensor");
MyMessage msg(SEN_UID_ID, V_VAR5);
UID_STRUCT uid;

uid.uid = 100;
uid.status = 0; //Query
uid.payload = 0;

gw.send(msg.set(&uid, sizeof(UID_STRUCT))); //Query status from MyController.org

//Receive response from MyController.org
void incomingMessage(const MyMessage &message) {
   if (message.sensor == SEN_UID_ID) {
      uint8_t bvalue[sizeof(UID_STRUCT)];
      setHexValue(bvalue, message.getString());
      memcpy(&uid, &bvalue, sizeof(UID_STRUCT));
   }
}
//--------------- Other methods in above reference code ------------------
void setHexValue(uint8_t bvalue[], const char *stringHex) {
    uint8_t blen = 0;
    uint8_t val;
    while (*stringHex) {
        val = h2i(*stringHex++) << 4;
        val += h2i(*stringHex++);
        bvalue[blen] = val;
        blen++;
    }
}

uint8_t h2i(char c) {
    uint8_t i = 0;
    if (c <= '9')
        i += c - '0';
    else if (c >= 'a')
        i += c - 'a' + 10;
    else
        i += c - 'A' + 10;
    return i;
}
//-----------------------------------------------------------------------
```

5.1.2 Send Payload

To send payload you have to follow the same code mentioned in Query Payload, only one modification you have to do is

```
uid.status = 1; // Send Payload
```

and you have to set Payload
```
uid.payload = 20;
```

uid.status = 0

- When you send from node it will be considered query mode.
- When you receive from MyController.org which means specified uid is not available in MyController.org.

uid.status = 1

- When you send from node it will be considered set payload mode
- When you receive from MyController.org, specified uid is available in MyController.org and *uid.payload* is sent successfully to the specified UID.

6.0 Firmware

You can control node firmware with the help of MYSBootloader

What you can do?

- Upload new firmware
- Change existing firmware to new latest
- Maintain nodes firmware up to date

6.1 Adding Firmware

By clicking More on top menu you can see list of options for firmware.

6.1.1 Add Firmware Type

Click on Firmware Type you will get firmware type page. Clicking you can add new firmware type.

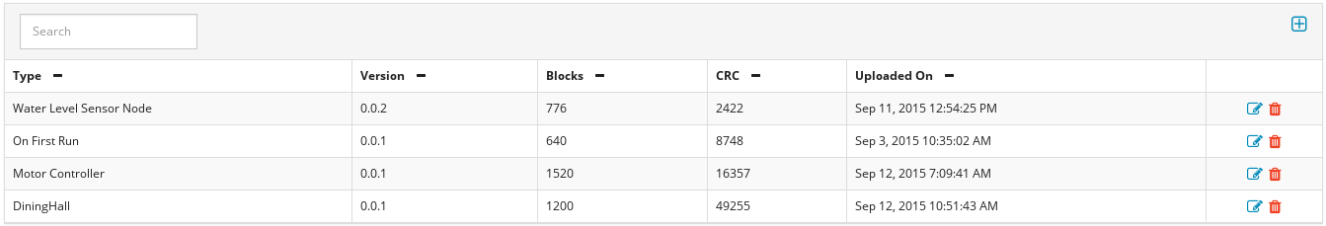
Note: *Id for the firmware type will be generated automatically. If you leave blank 'Type Id' field.*

6.1.2 Add Firmware Version

Click on Firmware Version you will get firmware version page. Clicking you can add new firmware version. While adding firmware version you can leave 'Version Id' as blank. Id will generated automatically. If you have any requirement for specified id add it.

Note: *Id for the firmware version will be generated automatically. If you leave blank 'Version Id' field.*

6.1.3 Add Firmware



Click on Firmware you will get firmware page. Clicking you can add new firmware. Select Firmware type and version from the pop-up and select .hex from your local computer. Finally click Add. Your firmware is ready for action.

7.0 Users

You can manage users under users page.
By clicking More on top menu you can see Users

7.1 Manage Users

You can perform add edit delete options in users page.

- Add new user

- Edit User

- Delete user

Note: *To change your logged in user password navigate to users page and click edit and update password. Once password updated successfully, GUI will throws error, you should Sign Out and Sign In again.*

8.0 Logs

To reach Logs page, click on More and click on Logs

In this page all the logs will be shown from gateway/sensors.

Search

Time	Type	Status	Log
Sep 15, 2015 9:04:52 AM	Sensor Internal	Sent	[I_CONFIG], [NodeId:3, SensorId:255, PayLoad: M]
Sep 15, 2015 9:04:52 AM	Sensor Internal	Received	[I_CONFIG], [NodeId:3, SensorId:255, PayLoad: 0]
Sep 15, 2015 9:04:52 AM	Sensor Presentation	Received	[S_ARDUINO_REPEATER_NODE], [NodeId:3, SensorId:255, PayLoad: 1.5]
Sep 15, 2015 9:04:51 AM	Sensor Stream	Sent	[ST_FIRMWARE_CONFIG_RESPONSE], [NodeId:3, SensorId:255, PayLoad: 0100020008037609]
Sep 15, 2015 9:04:51 AM	Sensor Stream	Received	[ST_FIRMWARE_CONFIG_REQUEST], [NodeId:3, SensorId:255, PayLoad: 0100020008037609 0101]
Sep 15, 2015 9:04:50 AM	Sensor Internal	Sent	[I_REBOOT], [NodeId:3, SensorId:255, PayLoad:]
Sep 15, 2015 9:04:36 AM	Sensor Internal	Received	[I_GATEWAY_READY], [NodeId:0, SensorId:0, PayLoad: Gateway startup complete.]
Sep 15, 2015 9:03:52 AM	Sensor	Sent	Motor Controller:Display, [NodeId:2, SensorId:1, PayLoad: 70]
Sep 15, 2015 9:03:52 AM	Sensor	Received	Water Level Sensor Node:Tank, [NodeId:3, SensorId:1, PayLoad: 70.0]
Sep 15, 2015 9:02:52 AM	Sensor	Sent	Motor Controller:Display, [NodeId:2, SensorId:1, PayLoad: 70]

9.0 System Status

To reach this page, click on More and click on System Status
You can see more information's about your current system.

Operating System:

Operating System	Linux
Architecture	arm
Version/Kernel	3.18.6-v7+
Available Process	4
System Load Average	72.00 %
System Cpu Load	0.00 %
Process Cpu	Load: 0.00 %, Time: 1858720 milliseconds
Physical Memory (RAM)	Free: 486 MB, Total: 974 MB
Swap Space	Free: 199 MB, Total: 199 MB
Committed Virtual Memory Size	145 MB
User working directory	/root/mycontroller/bin

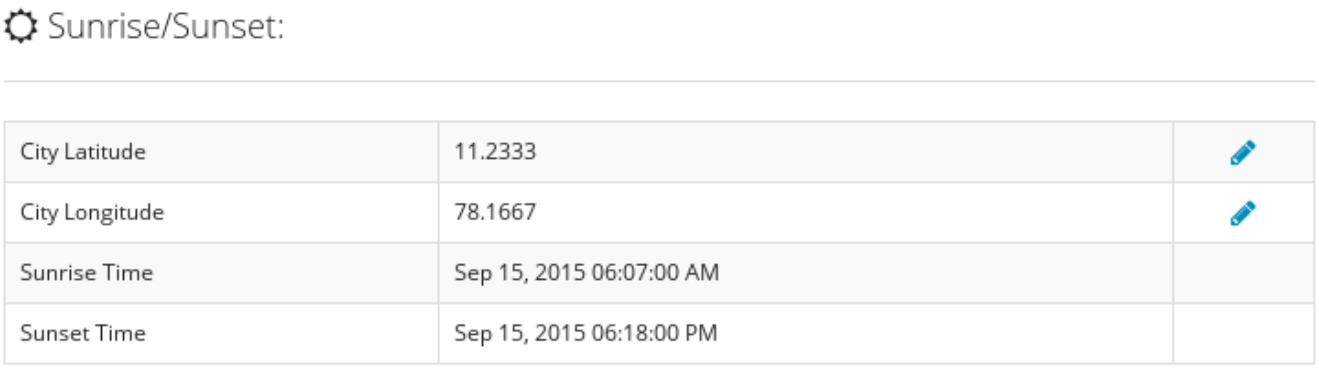
Java Virtual Machine Specification:

Java Home	/root/jdk1.8.0_60/jre
Java Vendor Url	http://java.oracle.com/
VM Name	Java HotSpot(TM) Client VM
VM Vendor	Oracle Corporation
VM Version	25.60-b23
Specification Vendor	Oracle Corporation
Specification Version	1.8
Management Specification Version	1.2
Up Time	1 day 11 hours 24 minutes 59 seconds
Heap Memory	Used: 26 MB, Committed: 32 MB, Init: 8 MB, Max: 38 MB
Non Heap Memory	Used: 19 MB, Committed: 20 MB, Init: 0 MB, Max: n/a
Threads Count	Current: 30, Peak: 31, Total started: 2170, Daemon: 14
Class Loading Details	Loaded: 4527, Unloaded: 57, Total loaded: 4584
Garbage Collector - Copy	Collection time: 3224, Collection count: 204, Last GC duration: 27
Garbage Collector - MarkSweepCompact	Collection time: 1503, Collection count: 6, Last GC duration: 347

10.0 System Settings

To reach this page, click on  and click on 

10.1 Sunrise/Sunset



Sunrise and Sunset time is used in ***4.6 Timer*** you should set Latitude and Longitude of your location to get proper Sunrise/Sunset time.

10.2 Node Defaults

10.2.1 Auto Node Id

Auto Node Id is used to assign node id for MySensors nodes. You can update current reserved id, if you set 0 it will start from 1 and so.
Also you can know the status of assigned id from this location.

10.2.2 Default Firmware

Default Firmware will be used on first run of MySensors node. You should add your default ***6.0 Firmware*** and you can select it here.

10.2.3 Requested firmware absences

When a node request some firmware version and which is not available in MyController.org. That time if you want to redirect to default firmware set this option as 'true'

10.3 Plivo SMS Gateway

Plivo SMS Gateway:

Auth Id	MA[redacted]	
Auth Token	*****	
From phone number	+91[redacted]	

Create an account in Plivo and update details here. This will be used to send SMS. Trail account is available with some free credit as on September 2015.

10.4 Email Gateway

Email Gateway:

SMTP Host	smtp.gmail.com	
SMTP Port	465	
From address	[redacted]@gmail.com	
Username	[redacted]@gmail.com	
Password	*****	
Enable SSL	true	

You can update any SMTP email gateway here. This will be used to send email notifications.

10.5 Units/Version Information

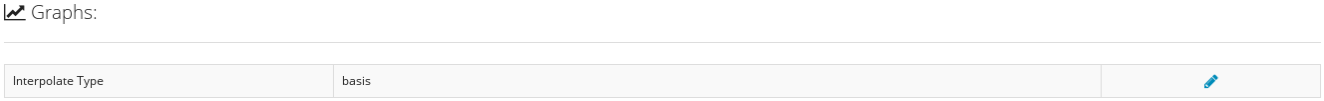
In this block you can see what is default units will be assigned and version information of MyController.org

Units:

Temperature	°C	
Distance	cm	
Percentage	%	

10.6 Other Settings

If you want to change line graph Interpolate options you can change it here.



Click edit icon of ***Interpolate Type*** on the pop-up you have option to select your favorite interpolate type.

11.0 Log In/Out

To *Log In* have to click 

you will be redirected to Log In page.

To *Log Out* you have to click 

icon and you will be redirected to Log In page.

12.0 Source Code

MyController.org is Open Source project. You can contribute/download source code on [Github repository]

12.1 Issue management

Do you want to file issues or new request/enhancement?
You are welcome to **[Github Issues page]**

12.2 License

Apache License version 2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.