Switching Monitoring Protection Module

**Introduction**

Switching Monitoring Protection module is a device to switch load & data monitoring. There are 3 data that can be monitored. They are voltage, current and power. In this document we will refer switching monitoring protection module as SMP. SMP implement a REST API as an interface to exchange data thus easily integrate into systems.

**How it works**

When SMP powered up, it attempts to connect into access point with predefined SSID. Once it is connected, SMP will setting up a webserver as a way for other devices to communicate using REST API. There are many features which is provided and each feature has its own endpoint. The usage of each endpoints will be explained later. Once webserver is ready, SMP begin capture and process data from sensor so it can be served.

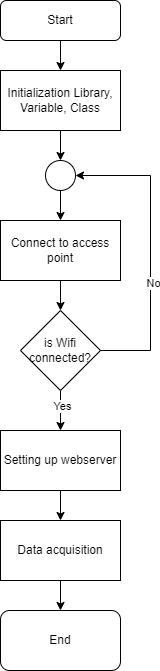


Figure 1. SMP Flowchart

**REST API**

SMP implements a REST API to exchange data. This will make user could easily integrate it into their own systems. SMP implements a HTTP POST and HTTP GET method, and doesn’t support HTTPS right now, make sure to send a request using HTTP and not the later. The list of each endpoints of POST and GET method are described below

**GET Method**

Note : fill the “.” With ip of device (ex : http://192.168.2.143/get-line-data)

Endpoint usage and explanation for GET Method :

1. get-line-data?line=$,$,$,$,$,$
   * To request multiple line data, replace $ with line number each separated with comma. Support up to 12 lines data per request
   * Send a HTTP GET request with parameter line (ex : 192.168.2.143/get-line-data?line=1,2,3,4)
   * SMP will respond with formatted JSON :

|  |
| --- |
| {  "line-data": [  {  "line": 1,  "voltage": 0,  "current": 0,  "power": 0  },  {  "line": 2,  "voltage": 0,  "current": 0,  "power": 0  },  {  "line": 3,  "voltage": 0,  "current": 0,  "power": 0  },  {  "line": 4,  "voltage": 0,  "current": 0,  "power": 0  }  ]  } |

* “line-data” : array of JSON with maximum 12
* “line” : line number, value can be any number between 1 - 64
* “voltage” : voltage data in millivolt(s) (mV)
* “current” : current data in milliamp(s) (mA)
* “power” : power data in watt(s) (W)
* Example above is a GET request for acquiring 4 lines data

**POST Method**



Note : fill the “.” With ip of device (ex : http://192.168.2.143/relay)

Endpoint usage and explanation for POST Method :

1. relay
   * To control relay on SMP, send a HTTP POST with following JSON format:

|  |
| --- |
| {  "data" : [  {  "line" : 1,  "value" : 1  },  {  "line" : 2,  "value" : 0  },  {  "line" : 3,  "value" : 1  },  {  "line" : 4,  "value" : 0  },  {  "line" : 5,  "value" : 1  },  {  "line" : 6,  "value" : 0  },  {  "line" : 7,  "value" : 1  },  {  "line" : 8,  "value" : 0  },  {  "line" : 9,  "value" : 1  },  {  "line" : 10,  "value" : 0  },  {  "line" : 11,  "value" : 1  },  {  "line" : 12,  "value" : 0  }  ]  } |

* + To control multiple relay, append the single JSON which consist of “line” key and “value” key into “data” key. See example above. Single POST request support up to 36 line control
  + “data” : array of JSON with maximum 36
  + “line” : the line that need to be controlled
  + “value” : state of relay. Set 1 to activate relay, set 0 to deactivate relay
  + Example above is a POST request to control 12 relays, which activate relay on line 1, 3, 5, 7, 9,11 while deactivate relay on line 2, 4, 6, 8, 10, 12
  + When POST request successfully received, SMP will respond with JSON format :

|  |
| --- |
| {  "status": 1  } |

* Status 1 indicate that the request is successfully received, any other value indicate abnormality