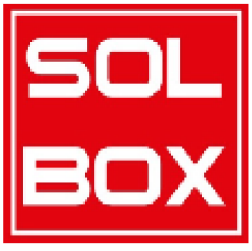
SolBox API



Introduction

In this document the mqtt API for the SolBox device will be explored. The SolBox device IoT capabilities regarding both control and monitoring of the system.

Main json for commands

MQTT topic: *axinar/solbox/MACADD/mainControlJson*

Example: *axinar/solbox/08F9E0E18A6C/mainControlJson*

The commands that can be sent in this topic have the following format: *{“COMMAND”:value}*

Example: *{“pauseCharging”:1}*

Commands can also be sent either grouped together or one by one

{"PWMset":825,"pwmEnable":1,"enableTelemetry":1} --> Set the manual PWM value to 825, enable manual PWM, and also enable telemetry.

{"pwmEnable":0,"enableTelemetry":0} --> Disable manual PWM and disable telemetry.

Pause / Start Function

{“pauseCharging":1}

1 : Starts the function

0 : Stops the function

Manual PWM enable

{“pwmEnable”:1}

1 : Enables manual PWM (default value is 300 / 1023)

0 : Disables manual PWM (enables MPPT function)

Set manual PWM

{“PWMset”:512}

Sets the value for the manual PWM to 512 out of 1023 steps

Enable weekly scheduler

{"enableWeekly":1}

1 : Enables smart schedule function

0 : Disables smart schedule function

Default Value : 1

Enable telemetry

{"enableTelemetry":1}

1 : Enables telemetry sent from the device to the main telemetry json: axinar/solbox/MACADD/jsonTelemetry

0: Disables telemetry sent from the device

Default value : 1

System reboot

{"resetSystem":1}

If the command is sent as is the device will restart in the following 5 seconds.

Force OTA update check

{"forceUpdateCheckFlag":1}

If this command is sent the device will go and check again if there's a newer version of the firmware available (not recommended the device checks for updates on each bootup and also everyday)

Delay wifi attempt to connect

{"delayWifiFlag":1}

1 : WiFi functions begin after the device reaches 38 PWM through normal MPPT function

0 : WiFi functions begin straight away

Manual relay control

{"manualRelayEnable":1}

1 : Turns relay ON (the NO pins are shorted)

0 : Turns relay OFF (the NO pins are not shorted)

Set device name

{"setNickname":"Somename"}

Sets the device's nickname as Somename. Can be any string

Fan control

{"fanControl":1}

1 : Turns fan ON

0 : Turns fan OFF, if the protection temperature is reached then the OFF (0) command is disabled and the fan does not cease operation

Clear load measurement

{“CheckLoad”:1}

Clears the last measured load resistance value

Set telemetry mode

{"telemetryMode":1}

0 : User’s telemetry

1 : Debugging telemetry

2 : Technician’s telemetry

Request 30 day power measurements

{"sendPowerData":1}

If the command is sent, the device will reply with the thirty day power data in the data request topic

axinar/solbox/MACADD/jsonDataSent

Send weekly schedule

{"sendWeekSchedule":1}

If the command is sent the device will reply with the weekly schedule saved on board in the data request topic

axinar/solbox/MACADD/jsonDataSent

Enable bacteria protection

{"checkBacteriaEnable":1}

If the command Is sent, then the device will make sure that the water inside the tank reached a specific temperature for a specific amount of time each month is order to make sure that the water is not contaminated with the legionella bacteria

Set maximum water temperature allowed

{"setMaxWaterTemp":90}

Sets the max water temperature allowed to 90 degrees (for example) this value cannot exceed 90 degrees

Clear today’s production

{"clearProduction":1}

If the command is sent the device will clear the day’s production log

Set device as paired

{"devicePaired":1}

1 : sets the device pairing flag as one = paired

0 : sets the device pairing flag as zero = unpaired

Default value : 0

Set as beta device

{"betaTesting":1}

1 : The device receives versions of the firmware marked as beta

0 : The device only receives versions of the firmware not marked as beta

Default value : 0

Clear 30 day power production log

{"clearProductionHistory":1}

If the command is sent the thirty day log is reset

Set the owner's mail

{"setOwnersMail":"something@aximail.com"}

The value ownersMail is set to something@axinar.com (can be any string)

Set vacation period

{"setVacationPeriod":1,"startDate":[1032024,24062024,99999999],"endDate":[4032024,12072024,99999999]}

Sets 3 vacation periods. 99999999 is the value set when the period is not used.

"setVacationPeriod" can be 0 or 1 and it works as a switch to control whether to check for periods or not

First period starts the 1st of March (3rd month) in 2024 and ends on the 4th of March in 2024

Second period starts the 24th of June 2024 and ends the 12th of July ib 2024

Third period is not used

*** In order to send the weekly schedule to the device you have to send the command to a different topic***

Set weekly schedule

****axinar/solbox/MACADD/jsonWeeklyCon****

{"weekday":"Monday","timeStart":[090000,103200,999999],"timeEnd":[103000,125700,999999],"targetTemp":65}

Every Monday between 9:00:00AM and 10:30:00AM and 10:32:00AM and 12:57:00PM the system will try to keep the water temperature at 65 degrees C

Main telemetry json

This json file is sent continuously.


```
{
  "VoltageIn": 70.82,
  "CurremtIn": 11.88,
  "PowerIn": 841.39,
  "WattHours": 3669.47,
  "DutyCycle": 70,
  "PWMduty": 716,
  "loadResistanceCalc": 4.2,
  "WaterTemp": 53.5,
  "HeatSinkTemp": 53.93,
  "TimePassed": 12698,
  "twelveMonitor": 12.03,
  "DisconnectCounter": 0,
  "maxPowerThirty": 7685.94,
  "historicProduction": 736820.38,
  "mesosOrosTrianta": 6573.02,
  "leastPowerThirty": 138.52,
  "minPowerPer": 71,
  "maxPowerPer": 99,
  "ATHwattHours": 7685.94,
  "powerNeedlePer": 47.7,
  "ManualPWMvalue": 300,
  "ManualPwmEnabled": false,
  "lastDayWattHours": 0,
  "waterTempMax": 90,
  "extFanState": false,
  "powerSupplyPresent": true,
  "dayEnded": false,
  "amIbeta": true,
  "DeviceEnabled": true,
  "HeatsinkTempAlert": false,
  "WaterTempAlert": false,
  "heatSinkTrermShorted": false,
  "waterThermShorted": false,
  "heatSinkThermDisconnected": false,
  "waterThermDisconnected": false,
  "RelayEnabled": false,
  "partialLoadFault": false,
  "LoadFaultFlag": 0,
  "delayWifiFlag": true,
  "weeklyScheduleFlag": true,
  "telemetryEnableFlag": true,
  "FirmwareVersion": "0.3.17",
  "DeviceName": "GeorgeHome",
  "DeviceID": "08F9E0E18A6C",
  "ownersMail": "axinar@aximail.com",
  "1acounter+": 2,
  "1bcounter-": 12,
  "2counter+": 15,
  "3counter+": 10,
  "4counter-": 18,
  "5counter++": 0,
  "vacationStart": [
    19072024,
    0,
    0
  ],
  "vacationEnd": [
    23072024,
    0,
    0
  ],
  "Info": [
    "13:5:48",
    "29072024"
  ],
  "pairing": [
    false,
    756332
  ],
  "wanIP": "94.67.95.140"
}
```

Data request json

This json is replied in axinar/solbox/MACADD/jsonDataSent when the {"sendPowerData":1} command is sent to axinar/solbox/MACADD/mainControlJson.


```
{
  "jsonArrayThirtyPower": [
    13462.09473,
    2883.084229,
    264.0133362,
    4939.851563,
    6640.638184,
    8490.116211,
    8299.655273,
    2557.999512,
    19799.50977,
    6055.474121,
    17250.41797,
    17179.15039,
    16956.33789,
    16462.56836,
    4903.162598,
    14168.37207,
    15622.78027,
    16408.81641,
    16558.85938,
    16351.04492,
    16007.18359,
    16383.01563,
    14885.0332,
    14246.20215,
    16280.39551,
    16192.61035,
    0,
    0,
    0,
    0
  ]
}
```


And the following json file is sent again in the /jsonDataSent topic when the {"sendWeekSchedule":1} is sent in the /mainControlJson topic.


```
{
  "timeStartFriday": [
    90000,
    103200,
    999999
  ],
  "timeEndFriday": [
    103000,
    125700,
    999999
  ],
  "timeStartSaturday": [
    90000,
    103200,
    999999
  ],
  "timeEndSaturday": [
    103000,
    125700,
    999999
  ],
  "timeStartSunday": [
    null,
    null,
    null
  ],
  "timeEndSunday": [
```

```
    null,
    null,
    null
  ],
  "timeStartMonday": [
    null,
    null,
    null
  ],
  "timeEndMonday": [
    null,
    null,
    null
  ],
  "timeStartTuesday": [
    null,
    null,
    null
  ],
  "timeEndTuesday": [
    null,
    null,
    null
  ],
  "timeStartWednesday": [
    null,
    null,
    null
  ],
  "timeEndWednesday": [
    null,
    null,
    null
  ],
  "timeStartThursday": [
    null,
    null,
    null
  ],
  "timeEndThursday": [
    null,
    null,
    null
  ]
}
```


Serial commands for network credentials (deprecated)

By sending the following command as text in the serial port with the device, you can manually set the network credentials for the device therefore skipping the WiFi manager function. This has been used in cases where the access point created by the device is not working properly.

Example : {"credentials":["Axinar_Guest","Axinar2017"]}

: { 'command' : ['ssid' , 'password']}