# Domoticz MicroPython Projects
# - Block Diagrams -

<Have Fun/>

Addendum to the Domoticz Homeautomation Workbook.
By rwbl

# Introduction

**Purpose**

To explore how to use the MicroPython programming language running on embedded hardware, the Microcontroller Unit MCU, interfacing with the Domoticz Home Automation System.

The core of the projects uses the Raspberry Pi Pico W, with actuators & sensors, acting as a Web Server to communicate with the Domoticz Home Automation System.

The projects can function as a base for projects or to trigger ideas for use.

The intention is to provide some practical guidance and not to explain Domoticz nor programming languages.

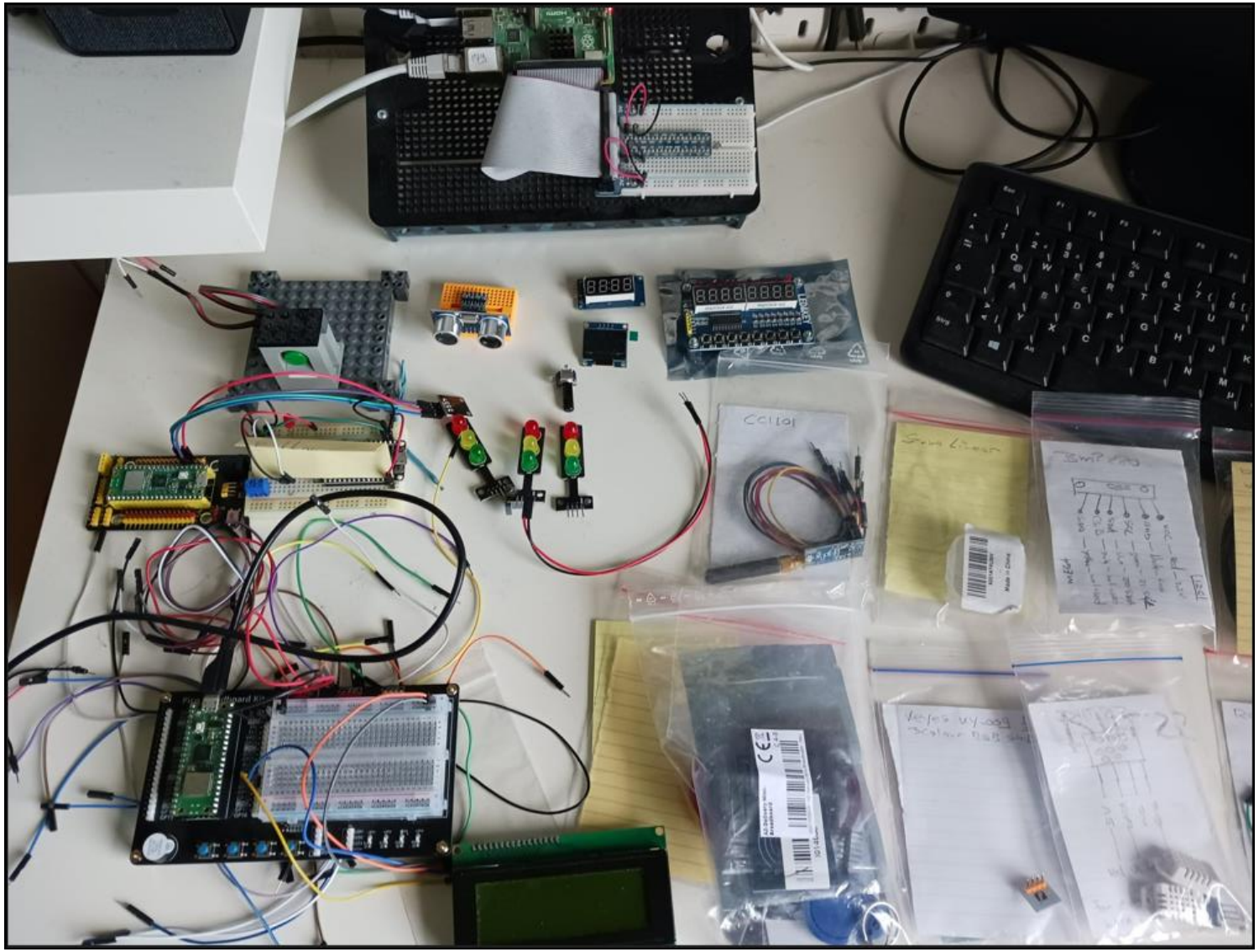
**Prerequisites**

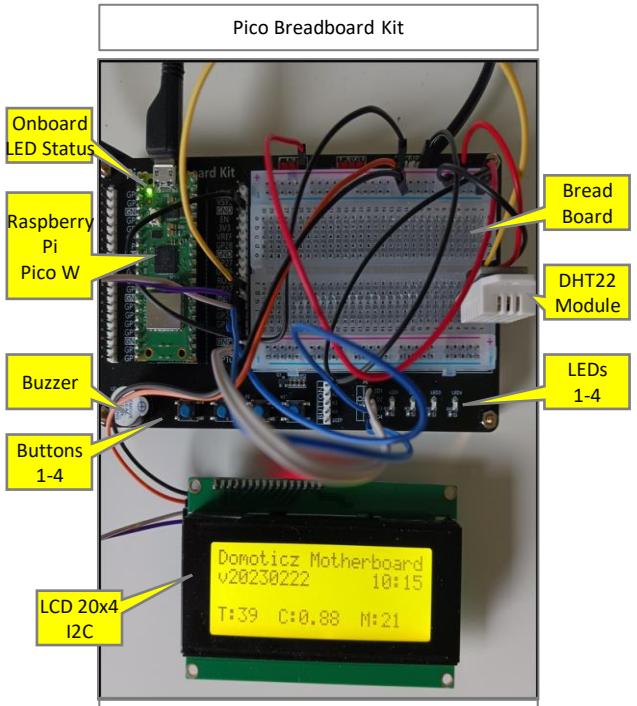
Basic knowledge of

- Domoticz Home Automation System.
- Domoticz Automation Event system dzVents & Lua.
- Programming languages Python and MicroPython.
- Raspberry Pi Pico / Pico W and ESP microcontrollers.
- Thonny Integrated Development Environment.
- JavaScript Object Notation (JSON).
- Message Queuing Telemetry Transport MQTT and MQTT Autodiscover.

# Workbench

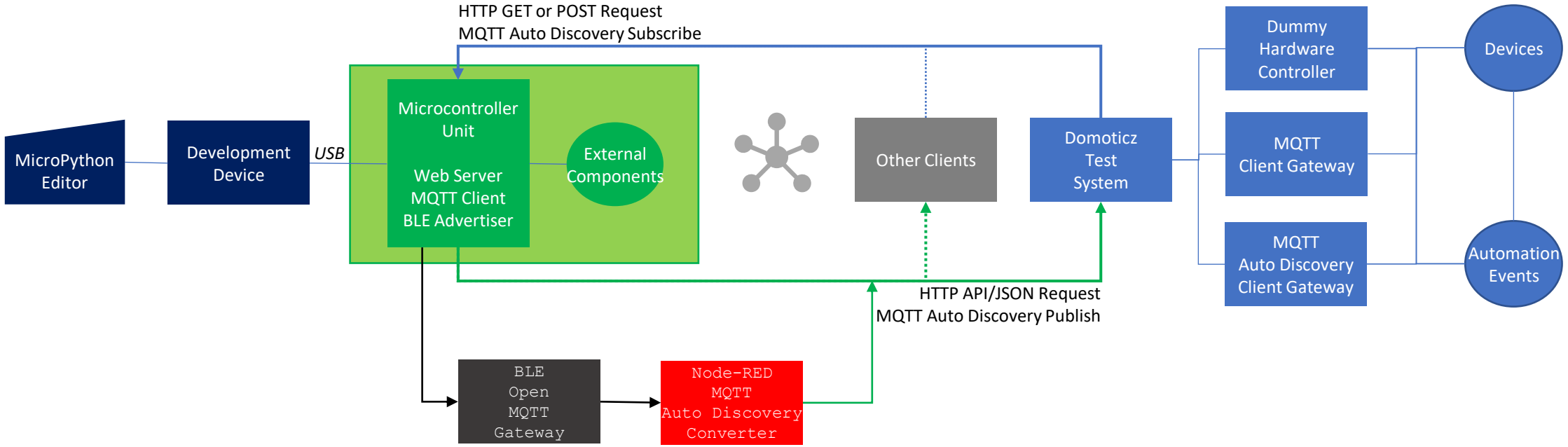
*Overview of the Author's workbench*





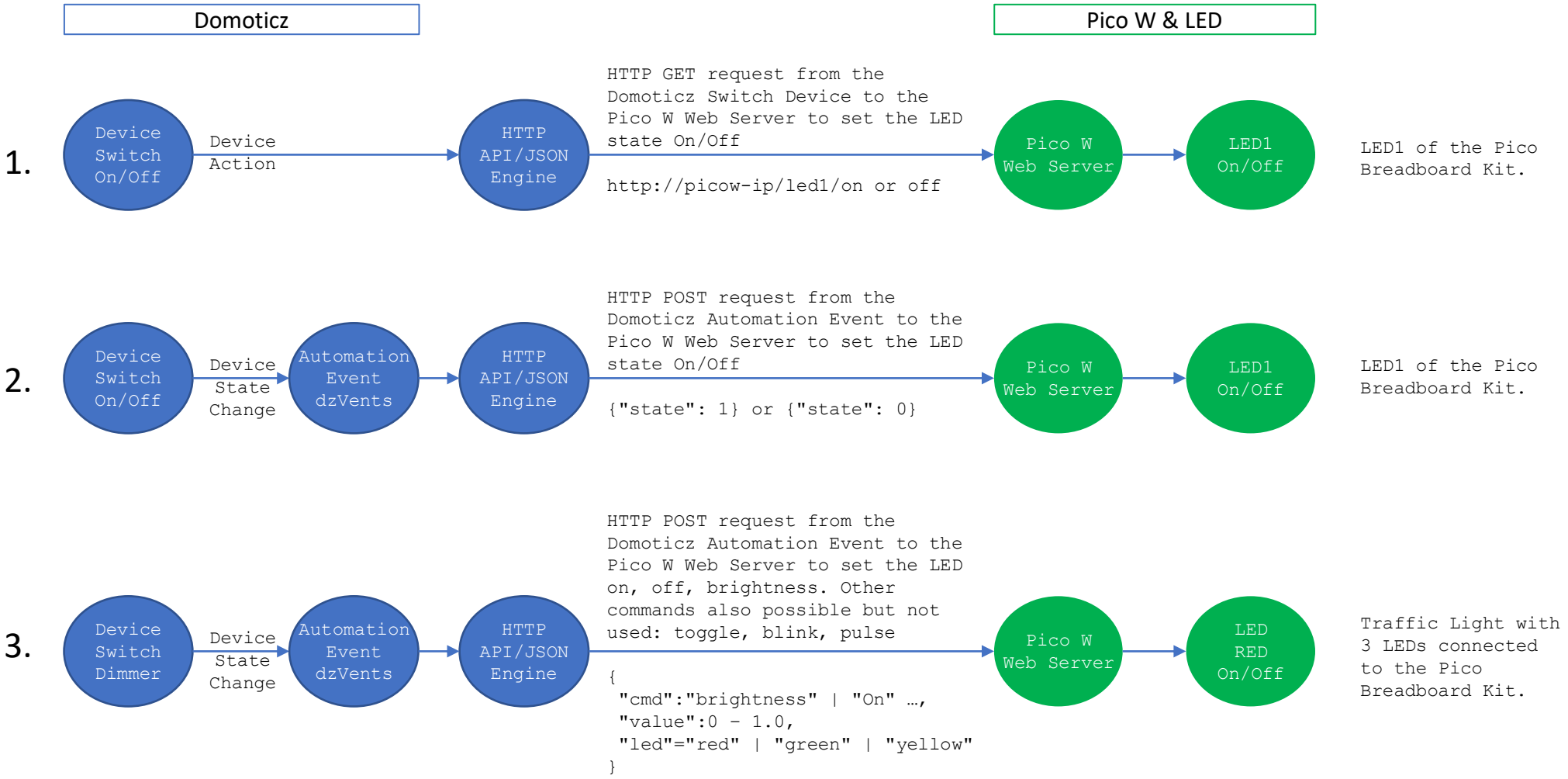

Setup example with modules

# Blockdiagram Setup



# Project LED Control

*Domoticz Switch Device triggers Device On/Off/Brightness Action to set the Pico W LED1 of the Pico Breadboard Kit.*

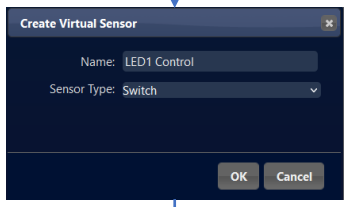


# Project LED Control – Device On/Off Action

*Domoticz Switch Device triggers Device On/Off Action to set the Pico W LED1 of the Pico Breadboard Kit On/Off.*

Domoticz Hardware Dummy

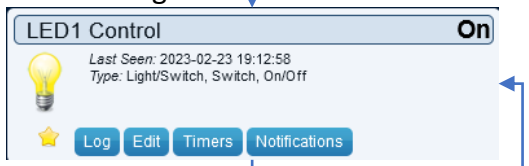
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

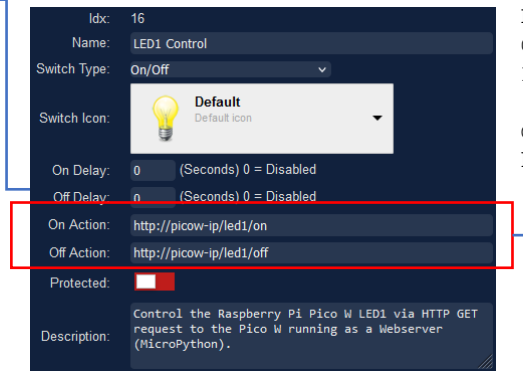



Device created

| Idx | Hardware | ID | Unit | Name | Type | SubType | Data |
|---|---|---|---|---|---|---|---|
| 16 | VirtualSensors | 00014060 | 1 | LED1 Control | Light/Switch | Switch | On |

Device Widget



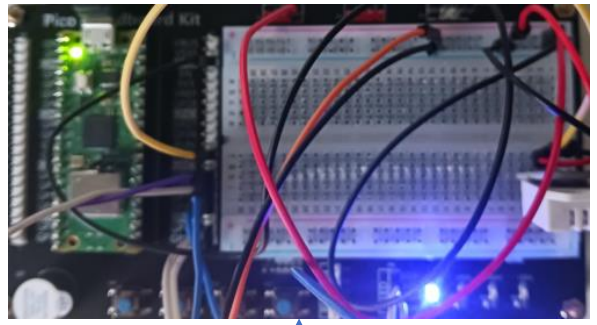




Edit Switch Device Properties with HTTP GET request actions set to switch the Pico W LED1 On / Off

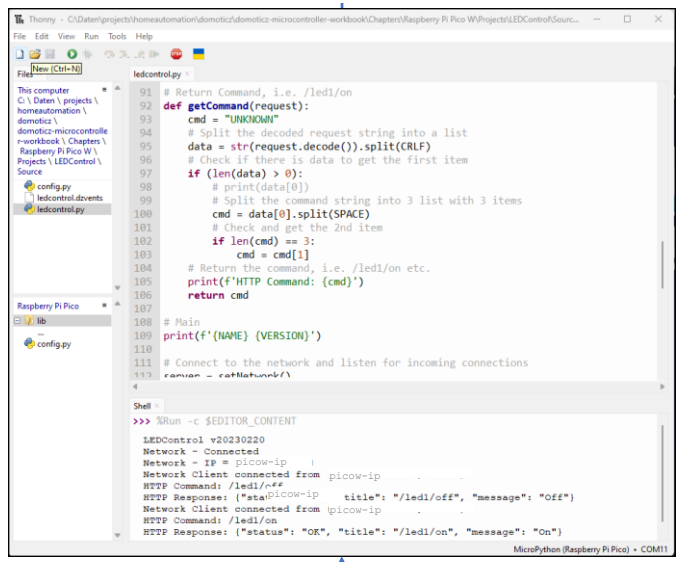
On/Off Action Command:
http://picow-ip/led1/on or off

Device HTTP On / Off Actions to Pico W Web Server

Pico W with Breadboard Kit & LED1 On



Thonny - Pico W Web Server MicroPython Script & Log

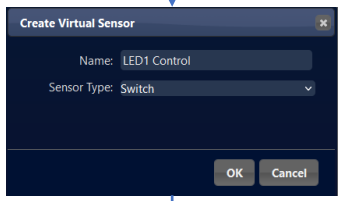


# Project LED Control - Custom Event

*Domoticz Switch Device State Change triggers Automation Event to set the Pico W LED1 of the Pico Breadboard Kit On/Off.*

Domoticz Hardware

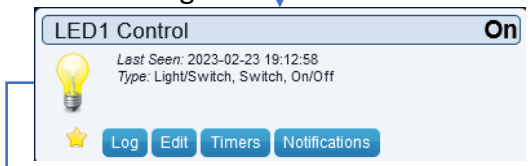
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

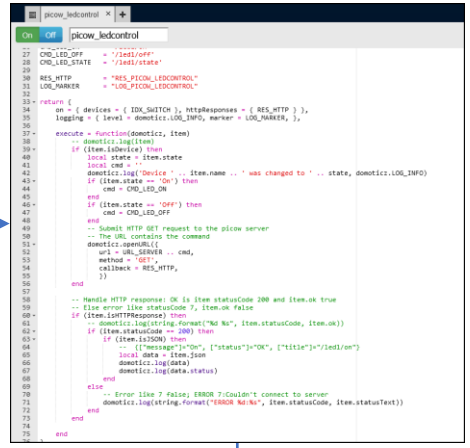



Domoticz Devices

| Idx | Hardware | ID | Unit | Name | Type | SubType | Data |
|---|---|---|---|---|---|---|---|
| 16 | VirtualSensors | 00014060 | 1 | LED1 Control | Light/Switch | Switch | On |

Device Widget

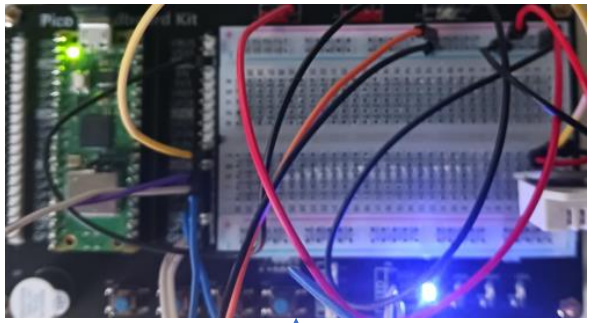


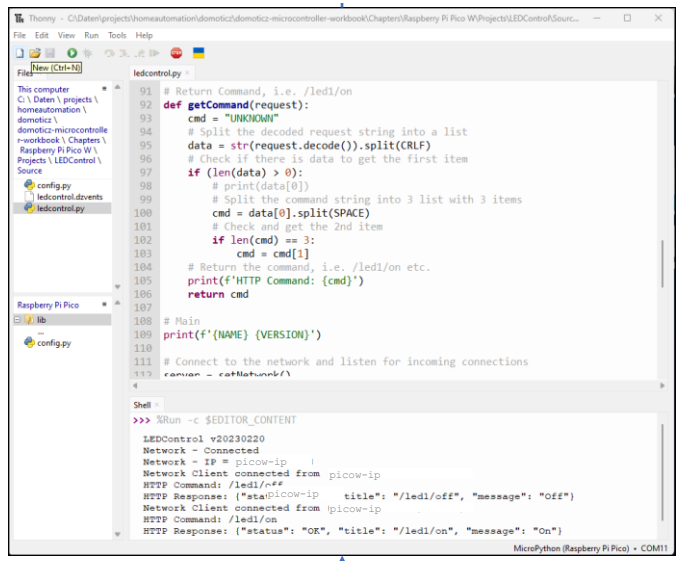
Domoticz Automation Event dzVents triggered by state change of the Domoticz Switch device (IDX=16).

The event submits a HTTP GET request to the Pico W Web Server to switch the LED1 On/Off.

Pico W with Breadboard Kit & LED1 On



Thonny - Pico W Web Server MicroPython Script & Log



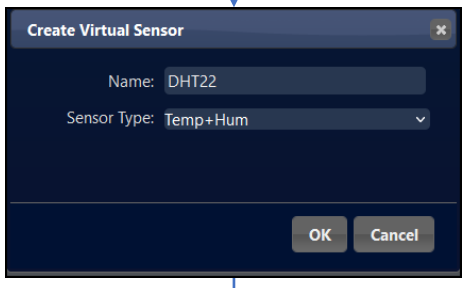
Device HTTP On / Off Actions to the Pico W Web Server

# Project DHT22 - HTTP API/JSON

*Pico W samples DHT22 sensor data and triggers updating the Domoticz Temp + Humidity device.*

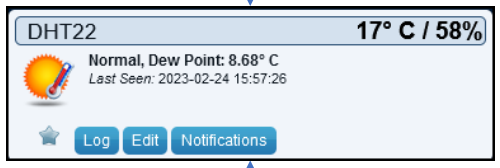
Domoticz Hardware

| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

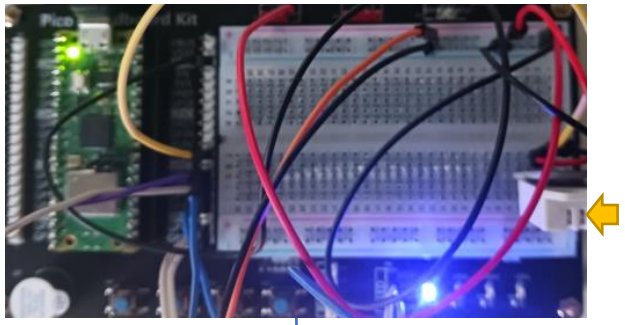



Domoticz Devices

| Idx | Hardware | ID | Unit | Name | Type | SubType | Data |
|---|---|---|---|---|---|---|---|
| 15 | VirtualSensors | 1405F | 1 | DHT22 | Temp + Humidity | THGN122/123/132, THGR122/228 /238/268 | 17.0 C, 58 % |

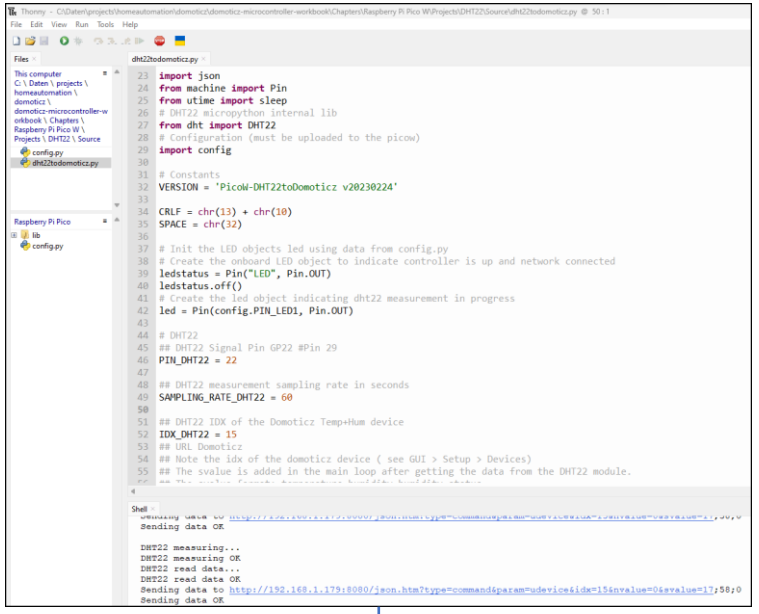



Pico W with Breadboard Kit & DHT22



Get DHT22 Data

Thonny - Pico W Web Server MicroPython Script & Log



```
import json
from machine import Pin
from utime import sleep
# DHT22 micropython internal lib
from dht import DHT22
# Configuration (must be uploaded to the picow)
import config

# Constants
VERSION = 'PicoW-DHT22toDomoticz v20230224'

CRLF = chr(13) + chr(10)
SPACE = chr(32)

# Init the LED objects led using data from config.py
# Create the onboard LED object to indicate controller is up and network connected
ledstatus = Pin("LED", Pin.OUT)
ledstatus.off()
# Create the led object indicating dht22 measurement in progress
led = Pin(config.PIN_LED1, Pin.OUT)

# DHT22
## DHT22 Signal Pin GP22 #Pin 29
PIN_DHT22 = 22

## DHT22 measurement sampling rate in seconds
SAMPLING_RATE_DHT22 = 60

## DHT22 IDX of the Domoticz Temp+Hum device
IDX_DHT22 = 15
## URL Domoticz
## Note the idx of the domoticz device ( see GUI > Setup > Devices)
## The svalue is added in the main loop after getting the data from the DHT22 module.
```

```
Sending data OK

DHT22 measuring...
DHT22 measuring OK
DHT22 read data...
DHT22 read data OK
Sending data to http://192.168.1.179:8080/json.htm?type=command&param=udevice&idx=15&nvalue=0&svalue=17;58;0
Sending data OK
```

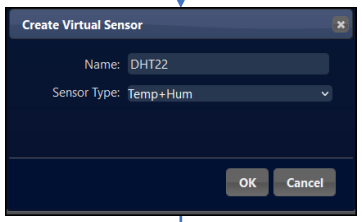
```
Pico W Web Server HTTP API/JSON GET request to Domoticz to update the device.
Example HTTP URL:
http://domoticz-ip:8080/json.htm?type=command&param=udevice&idx=15&nvalue=0&svalue=17;58;0
```

# Project DHT22 - Custom Event

*Pico W samples DHT22 sensor data and triggers Automation Event to update the Domoticz Temp + Humidity device.*

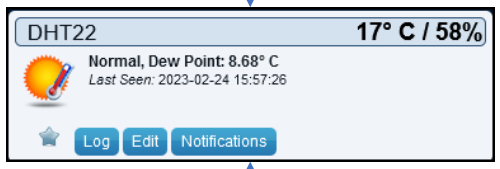
Domoticz Hardware

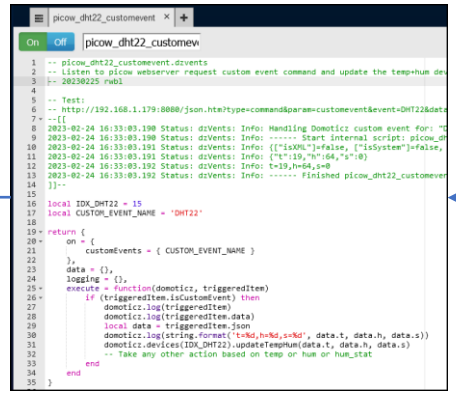
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |




Domoticz Devices

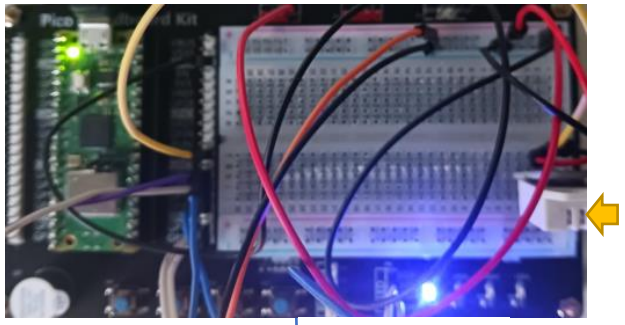
| Idx | Hardware | ID | Unit | Name | Type | SubType | Data |
|---|---|---|---|---|---|---|---|
| 15 | VirtualSensors | 1405F | 1 | DHT22 | Temp + Humidity | THGN122/123/132, THGR122/228 /238/268 | 17.0 C, 58 % |



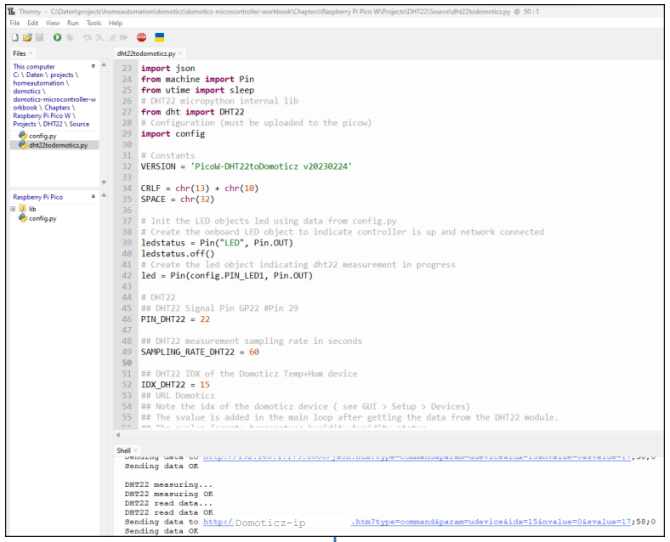

Domoticz Automation Event dzVents
Trigger: Custom Event

Pico W with Breadboard Kit & DHT22



Get DHT22 Data

Thonny - Pico W Web Server MicroPython Script & Log



Pico W Web Server HTTP API/JSON POST request to Domoticz Automation Event to update the device.
Example HTTP URL:
http://domoticz-ip:8080/json.htm?type=command&param=customevent&event=DHT22&data={"h": 58, "t": 17, "s": 0}

# Project DHT22 - MQTT Autodiscover

*Pico W publishes MQTT Config & State Messages to Domoticz MQTT Auto Discovery Client Gateway.*

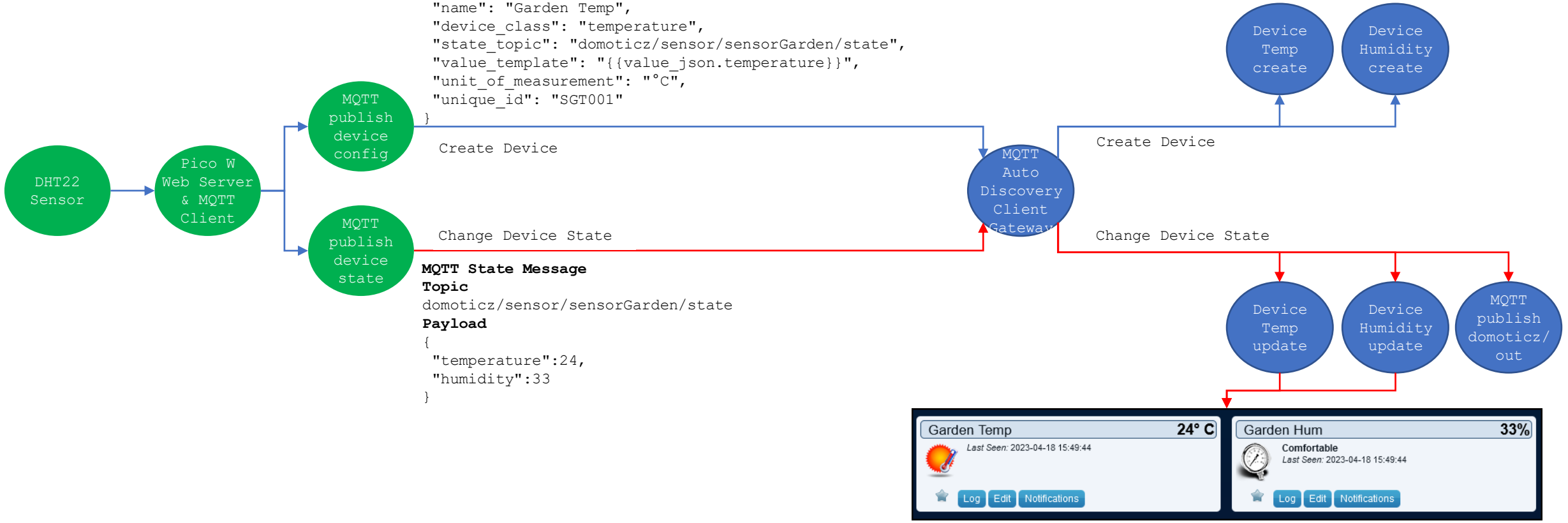
**Pico W**

**MQTT Configuration Message**
**Example Temperature**
**Topic**
domoticz/sensor/sensorGardenT/config
**Payload**

```
{
 "name": "Garden Temp",
 "device_class": "temperature",
 "state_topic": "domoticz/sensor/sensorGarden/state",
 "value_template": "{{value_json.temperature}}",
 "unit_of_measurement": "°C",
 "unique_id": "SGT001"
}
```

**MQTT State Message**
**Topic**
domoticz/sensor/sensorGarden/state
**Payload**

```
{
 "temperature":24,
 "humidity":33
}
```

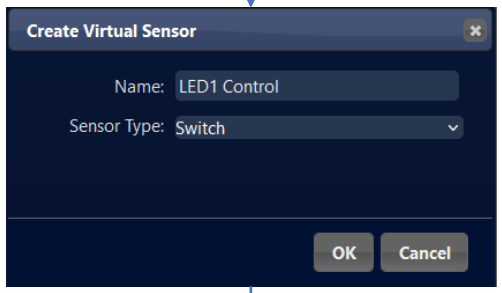


# Project LCD LED Control

*Domoticz Switch Device State Change triggers Automation Event to set the Pico W LED1 & LCD2004 display.*

Domoticz Hardware

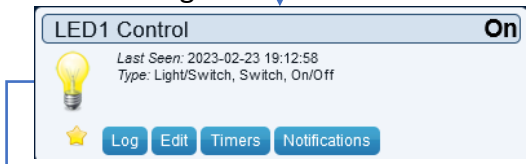
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

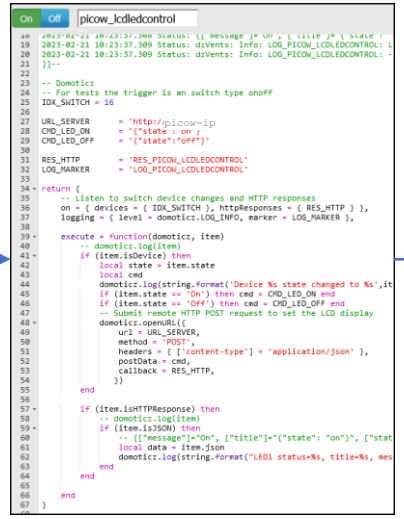



Domoticz Devices

| Idx | Hardware | ID | Unit | Name | Type | SubType | Data |
|---|---|---|---|---|---|---|---|
| 16 | VirtualSensors | 00014060 | 1 | LED1 Control | Light/Switch | Switch | On |

Device Widget



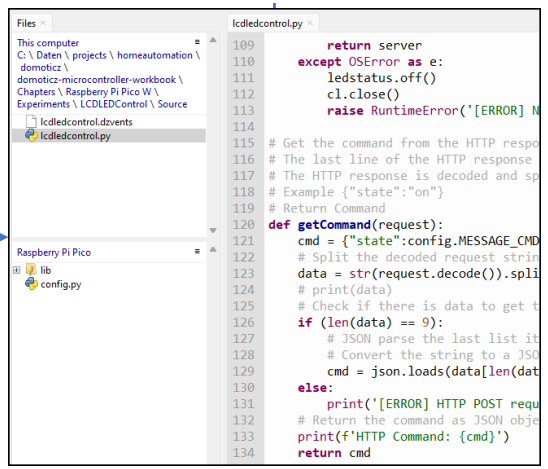




Domoticz Automation Event dzVents triggered by state change of the Domoticz Switch device (IDX=16).

The event submits a HTTP POST request to the Pico W Web Server to switch the LED1 On/Off and set the state on the LCD2004.

Pico W with Breadboard Kit & LCD 20x4 & LED1 On

Thonny - Pico W Web Server Script & Log



# Project LCD Motherboard

*Domoticz sends RPi Motherboard sensor data to set the LCD 20x4 I2C display connected to the Pico W.*

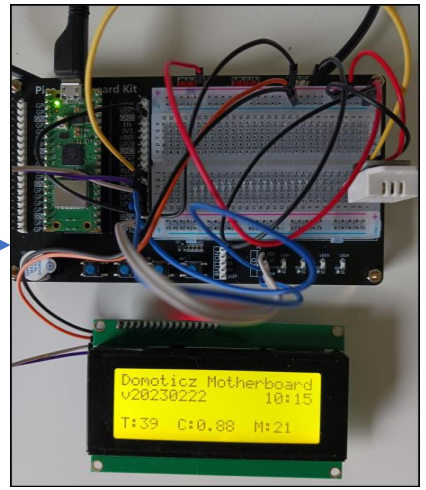
Domoticz Hardware

| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 5 | RPi Motherboard | Yes | Motherboard sensors | | | Disabled |

Domoticz Devices (selective)

| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 18 | RPi Motherboard | 0001 | 1 | Internal Temperature | Temp | LaCrosse TX3 | 38.9 C |
| | 26 | RPi Motherboard | 0000044D | 1 | CPU_Usage | General | Percentage | 0.41% |
| | 22 | RPi Motherboard | 0000044C | 1 | Memory Usage | General | Percentage | 18.35% |

Pico W with Breadboard Kit & LCD 20x4




On Off picow_lcdmotherboard

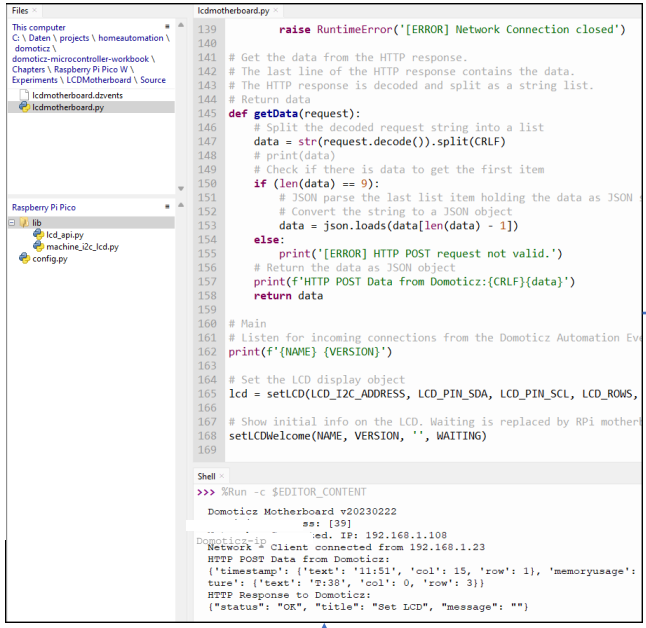
```
-- setSensor(3,5,'T:',round(domoticz.devices(IDX_INTERNALTEMPERATURE
local function setSensor(row,col,pre,data)
    return setText(row,col,string.format('%s%s', pre, tostring(data))
end

-- {["internaltemperature"]={["col"]=5, ["data"]="T:39.4", ["row"]=3

local function getMotherboardData(domoticz)
    local data = {}

    -- row=1, col=15, data=HH:MM len=5, range 15+5-1=19
    data['timestamp'] = setText(1, 15, string.sub(domoticz.time.rawT

    -- row=3, col=0, data=T:NN len=4, range = 0+4-1=3
    data['internaltemperature'] = setSensor(3, 0, 'T:', round(domotic
    -- row=3, col=6, data=C:NN.NN len=5, range = 6+5-1=10
    data['cpuusage']            = setSensor(3, 6, 'C:', round(domotic
    -- row=3, col=14, data=M:NN len=4, range = 14+4-1=17
    data['memoryusage']         = setSensor(3, 14, 'M:', round(domot

    --[[
    data['internaltemperature'] = round(domoticz.devices(IDX_INTERNA
    data['armclockspeed']       = round(domoticz.devices(IDX_ARMCLOCK
    data['v3dclockspeed']       = round(domoticz.devices(IDX_V3DCLOCK
    data['coreclockspeed']      = round(domoticz.devices(IDX_CORECLO
    data['memoryusage']         = round(domoticz.devices(IDX_MEMORYU
    data['processusage']        = round(domoticz.devices(IDX_PROCESSU
    data['hddboot']             = round(domoticz.devices(IDX_HDDBOOT
    data['hdd']                 = round(domoticz.devices(IDX_HDD).pe
    data['cpuusage']            = round(domoticz.devices(IDX_CPUUSAG
    ]]--

    -- Test embedding json: r=row 0-3, c=col 0-19, d=data
    --[[
    local x = {}
    x['r']=3
    x['c']=5
    x['d']=string.format('T:%s', tostring(data['internaltemperature'
    data['temp'] = x
    ]]--
    domoticz.log(data)
    return data
end
```

Domoticz Automation Event dzVents
Trigger: Timer
An event is reading in regular intervals the RPi motherboard data.

The data is converted to a JSON object and sent as HTTP POST request to the Pico W Web Server.

```
{
 ["internaltemperature"]=
  {["col"]=0,["text"]="T:38", ["row"]=3},
 ["timestamp"]=
  {["col"]=15,["text"]="16:19",["row"]=1},
 ["memoryusage"]=
  {["col"]=14,["text"]="M:21",["row"]=3},
 ["cpuusage"]=
  {["col"]=6,["text"]="C:0.35",["row"]=3}
}
```

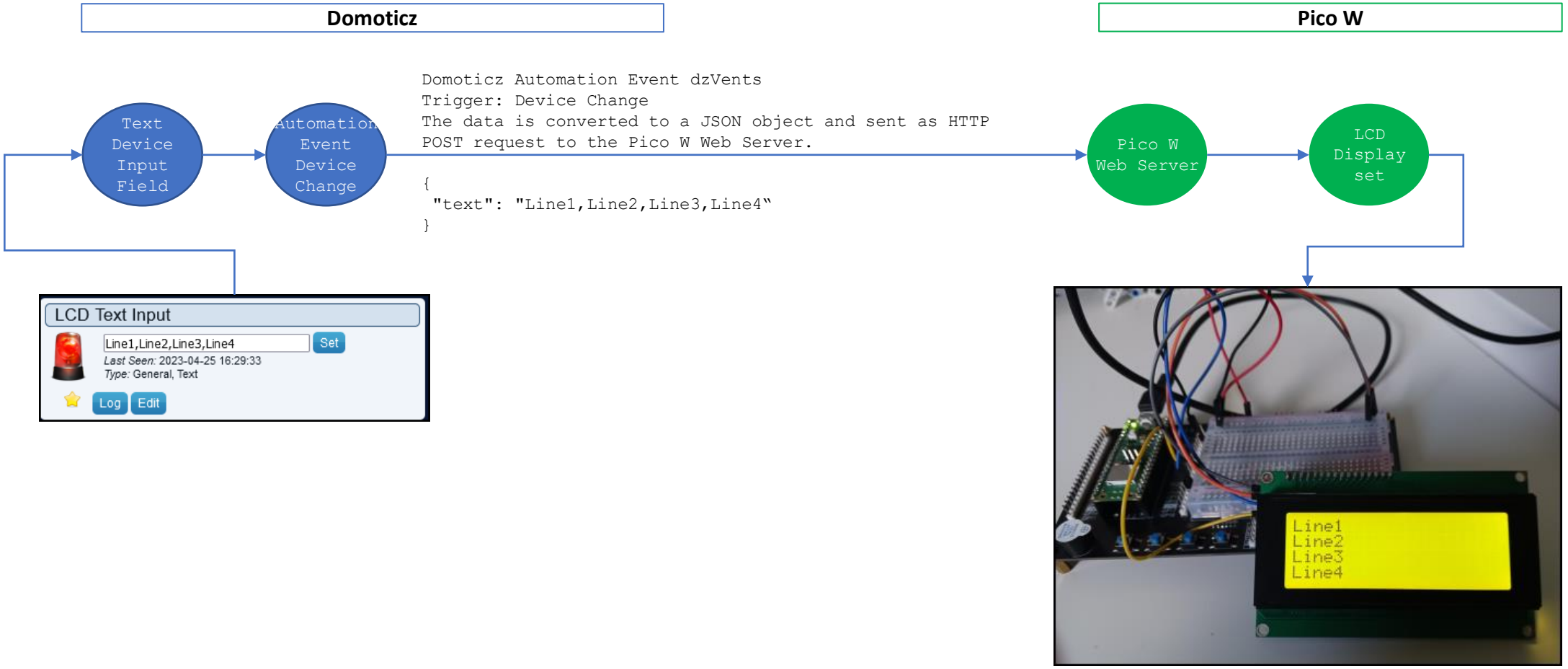
Thonny - Pico W Web Server Script & Log



```
        raise RuntimeError('[ERROR] Network Connection closed')

# Get the data from the HTTP response.
# The last line of the HTTP response contains the data.
# The HTTP response is decoded and split as a string list.
# Return data
def getData(request):
    # Split the decoded request string into a list
    data = str(request.decode()).split(CRLF)
    # print(data)
    # Check if there is data to get the first item
    if (len(data) == 9):
        # JSON parse the last list item holding the data as JSON
        # Convert the string to a JSON object
        data = json.loads(data[len(data) - 1])
    else:
        print('[ERROR] HTTP POST request not valid.')
    # Return the data as JSON object
    print(f'HTTP POST Data from Domoticz:{CRLF}{data}')
    return data

# Main
# Listen for incoming connections from the Domoticz Automation Ev
print(f'{NAME} {VERSION}')

# Set the LCD display object
lcd = setLCD(LCD_I2C_ADDRESS, LCD_PIN_SDA, LCD_PIN_SCL, LCD_ROWS,

# Show initial info on the LCD. Waiting is replaced by RPi mother
setLCDWelcome(NAME, VERSION, '', WAITING)

>>> %Run -c $EDITOR_CONTENT
  Domoticz Motherboard v20230222
                    ss: [39]
                    ed. IP: 192.168.1.108
  Network - Client connected from 192.168.1.23
  HTTP POST Data from Domoticz:
  {'timestamp': {'text': '11:51', 'col': 15, 'row': 1}, 'memoryusage':
  ture': {'text': 'T:38', 'col': 0, 'row': 3}}
  HTTP Response to Domoticz:
  {"status": "OK", "title": "Set LCD", "message": ""}
```

# Project LCD Text Input

*Domoticz sends text from a Text Device enabling Input to set the LCD 20x4 I2C display connected to the Pico W.*




Inspired by this Domoticz Forum Post (thanks for sharing).

# Project LED Display TM1637

*Domoticz sends RPi Motherboard Internal Temperature to set the 4-Digit 7-Segment display connected to the Pico W.*

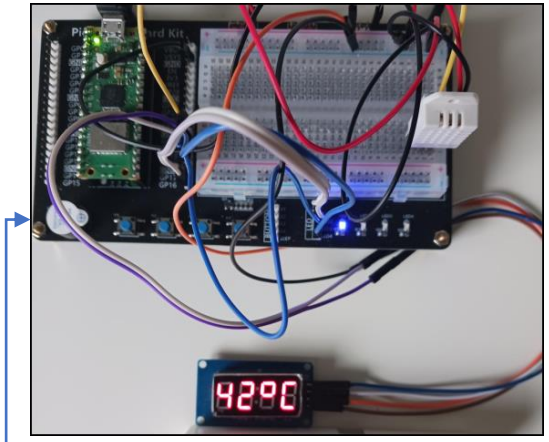
Domoticz Hardware

| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 5 | RPi Motherboard | Yes | Motherboard sensors | | | Disabled |

Domoticz Devices

| Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|
| 18 | RPi Motherboard | 0001 | 1 | Internal Temperature | Temp | LaCrosse TX3 | 37.0 C |

Pico W with Breadboard Kit & TM1637




picow_tm1637_motherboard_internal_temperature

On Off picow_tm1637_motherb

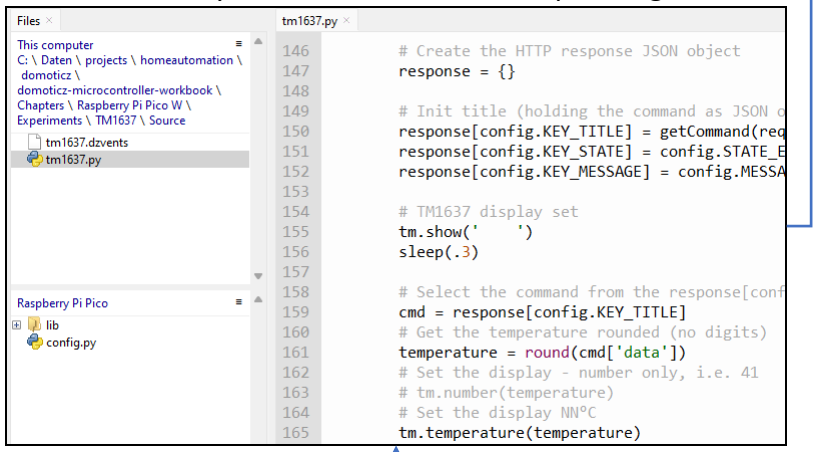
```
20 HTTP_RES         = 'RES_PICOW_LCDLEDCONTROL'
21
22 LOG_MARKER       = 'LOG_PICOW_LCDLEDCONTROL'
23
24 TIMER_RULE = "every minute"
25
26 -- Helpers
27 local function round(number, decimals)
28     local power = 10^decimals
29     return math.floor(number * power) / power
30 end
31
32 -- Post Data to the Pico W Webserver.
33 -- The data is a JSON object with the temperature roun
34 local function HTTPPost(domoticz)
35     local data = {}
36     data['data'] = domoticz.devices(IDX_INTERNAL_TEMPE
37     domoticz.log(string.format('device=%s, temperature
38     domoticz.log(data)
39     -- Submit remote HTTP POST request to set the tm16
40     domoticz.openURL({
41         url = HTTP_URL,
42         method = 'POST',
43         headers = { ['content-type'] = 'application/js
44         postData = data,
45         callback = HTTP_RES,
46        })
47 end
```

Domoticz Automation Event dzVents
Trigger: Timer or Switch
An event is reading in regular intervals the RPi motherboard internal temperature sensor.

The data is converted to a JSON object and sent as HTTP POST request to the Pico W Web Server.

```
{
 ["data"]=40.900001525879
}
```

Thonny - Pico W Web Server Script & Log



# Project Servo Control

*Domoticz Dimmer Device State Change triggers Automation Event to set the angle of a servo motor connected to the Pico W.*

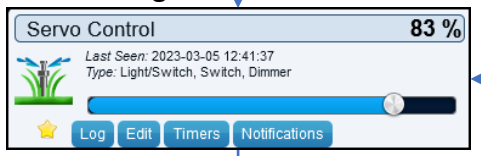
Domoticz Hardware

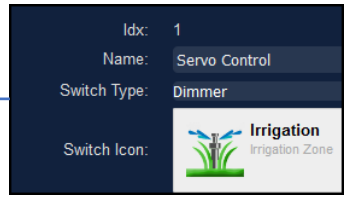
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

Domoticz Devices

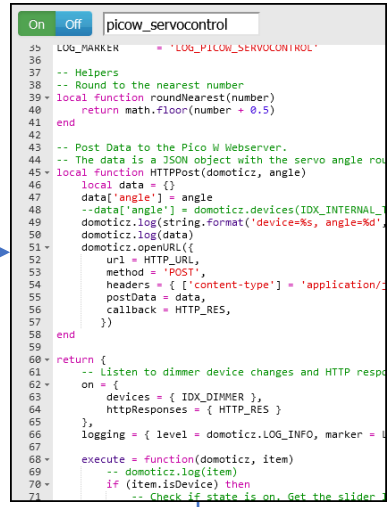
| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 1 | VirtualSensors | 00014051 | 1 | Servo Control | Light/Switch | Switch | Set Level: 83 % |

Device Widget




Edit Set Switch Type to Dimmer and Switch Icon to Irrigation

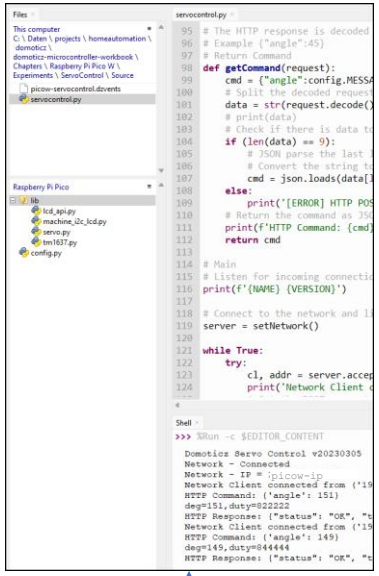



Domoticz Automation Event dzVents triggered by state change of the Domoticz Dimmer device (IDX=1).
The servo angle is converted from the dimmer level 0-100% to 0-180 deg.
83% = 149 deg.

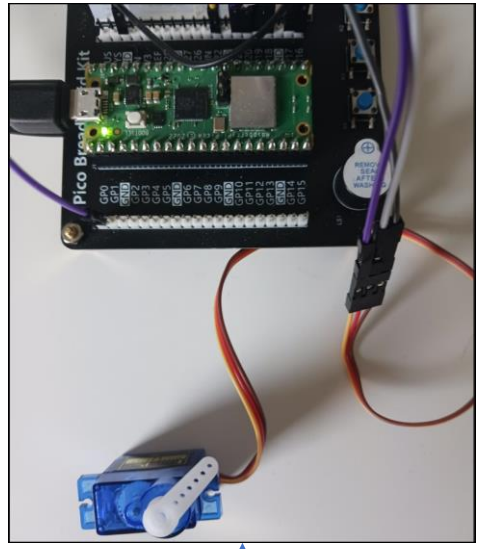
The event submits HTTP POST request to the Pico W Web Server to set the servo angle.
The post-data is a JSON Object:

```
{"angle": 149}
```

Thonny - Pico W Web Server MicroPython Script & Log



Pico W with Breadboard Kit & Servo Motor SG90

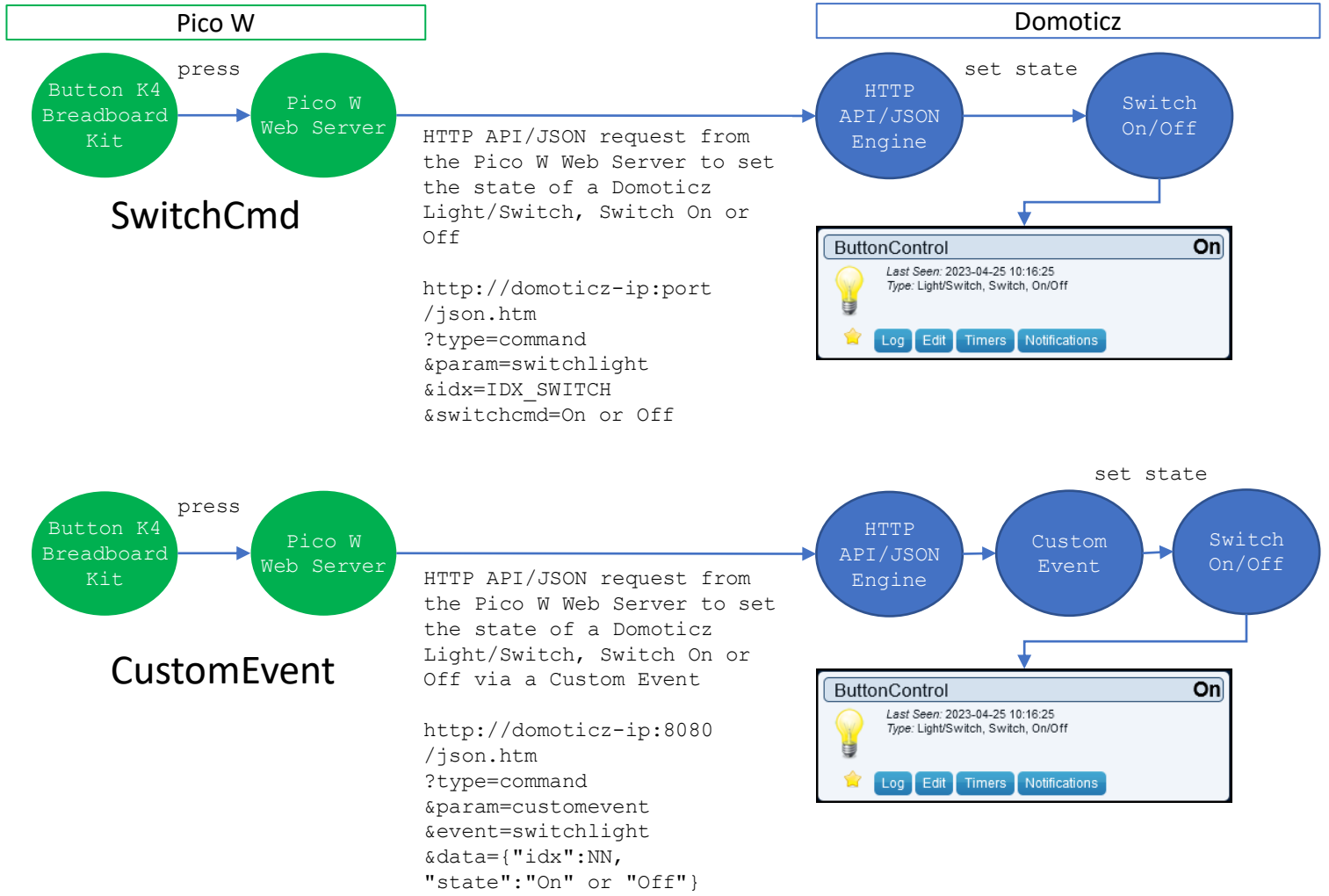


Servo Angle
149 deg

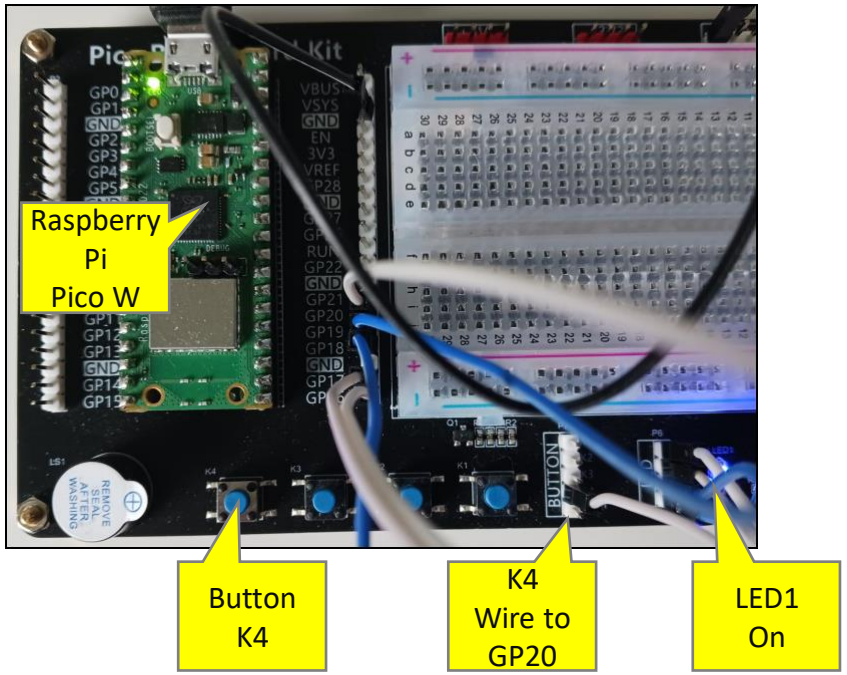
Device HTTP POST Request to Pico W Web Server

# Project Button Control

*Pico W with Pushbutton to set the state of a Domoticz Switch On/Off Device.*

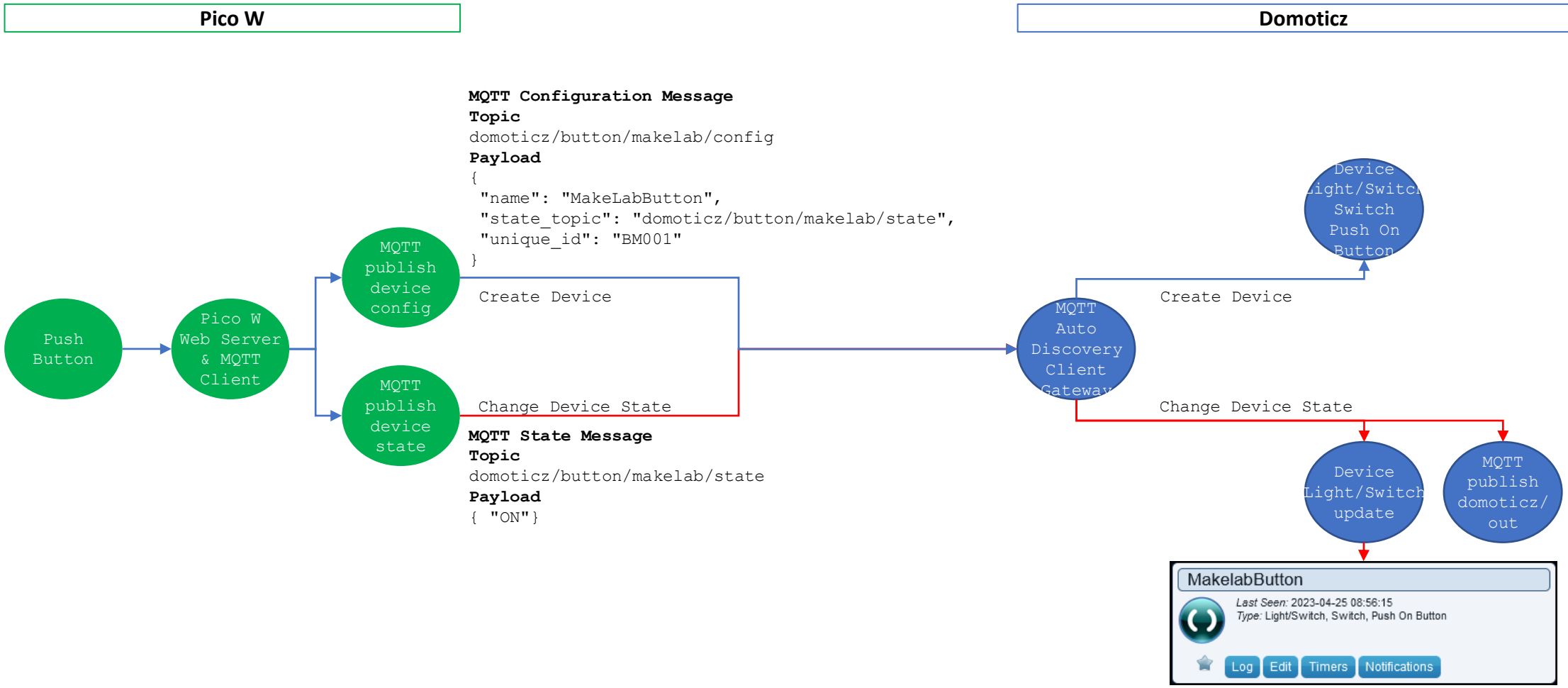



Pico W with Breadboard Kit Button K4 connected



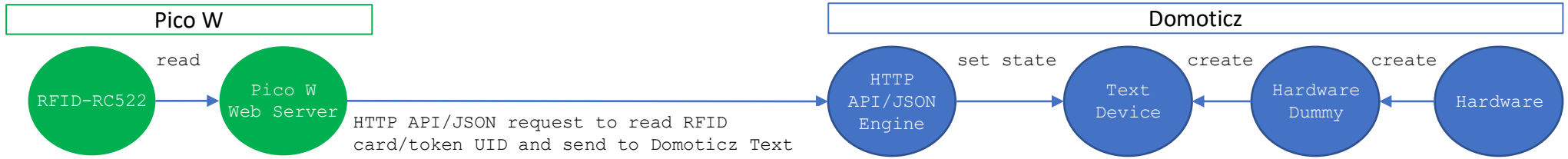
# Project Button Control - MQTT Autodiscover

*Pico W Push button set state of Domoticz Push On Button Device using Domoticz MQTT Auto Discovery Client Gateway.*



# Project RFID Reader

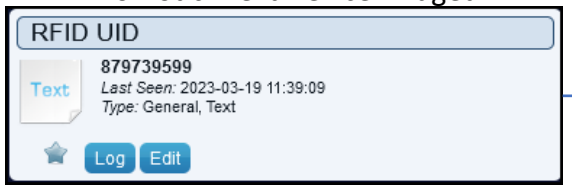
*Pico W to read RFID cards/tokens via RFID-RC522 module and send the UID to a Domoticz Text Device.*




HTTP API/JSON request to read RFID card/token UID and send to Domoticz Text Device

```
http://domoticz-ip:port
/json.htm
?type=command
&param=udevice
&idx=IDX_RFID
&nvalue=0
&svalue=879739599
```

Domoticz Text Device Widget

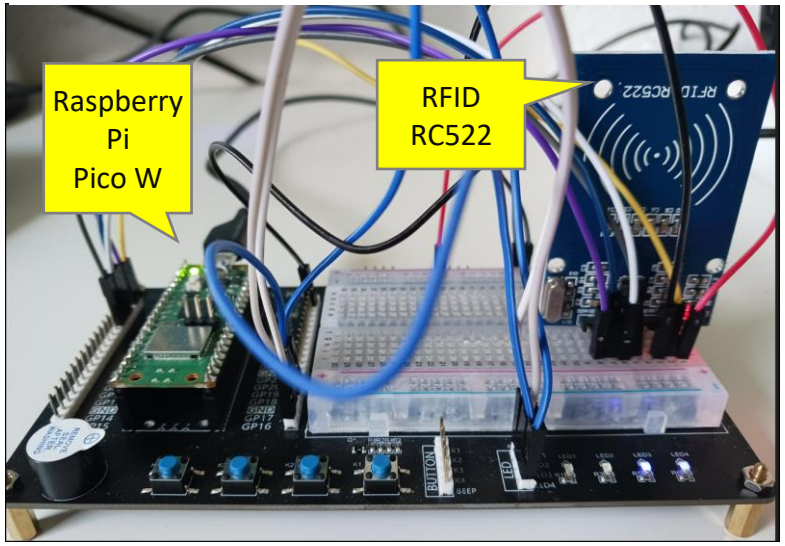


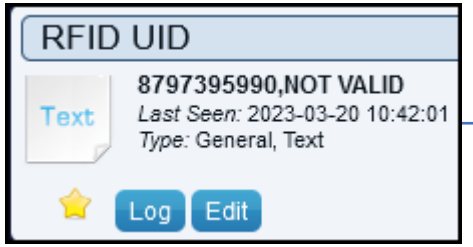

```
The RFID card/token UID is:
879739599 (HEX 346FC2CF).
The card type is 10.
```

Domoticz Text Device Log

| Date | |
|---|---|
| 2023-03-19 12:04:31 | 1446518016 |
| 2023-03-19 12:04:26 | 879739599 |
| 2023-03-19 12:04:19 | 249942437 |
| 2023-03-19 11:39:08 | 879739599 |
| 2023-03-19 11:39:04 | 1446518016 |
| 2023-03-19 11:37:42 | 879739599 |
| 2023-03-19 11:37:22 | 1446518016 |
| 2023-03-19 11:32:21 | 879739599 |

Pico W with Breadboard Kit & RFID-RC522 connected



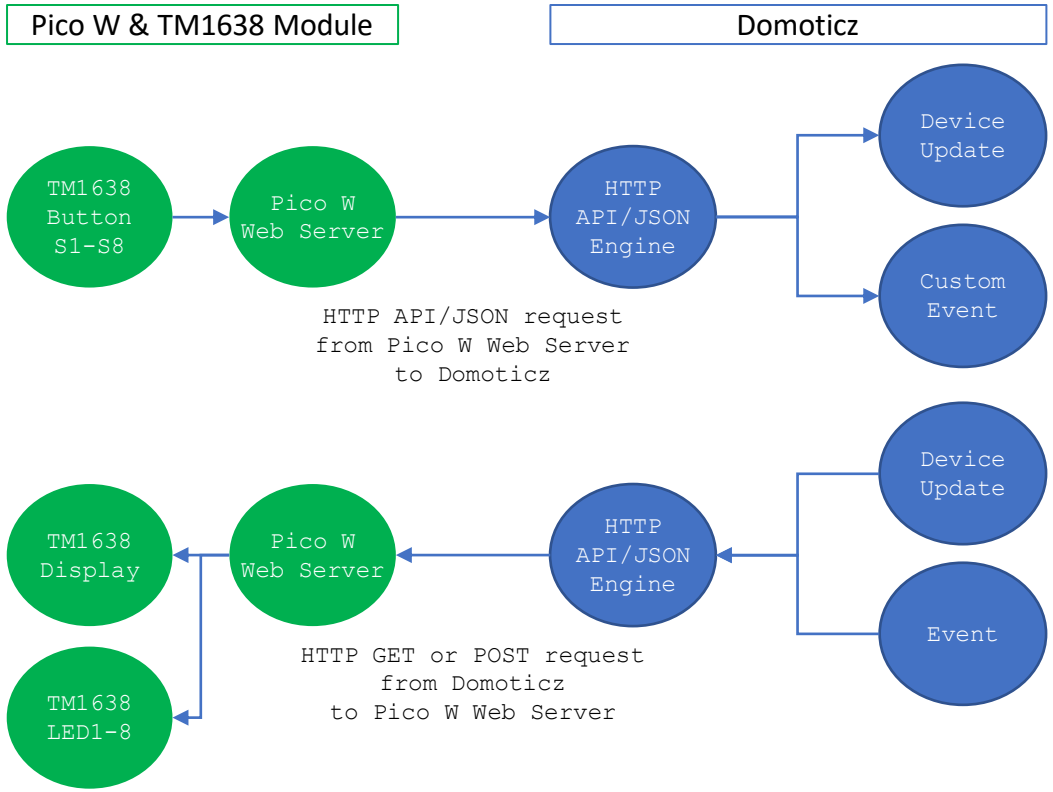




Domoticz Text Device Log with Flag

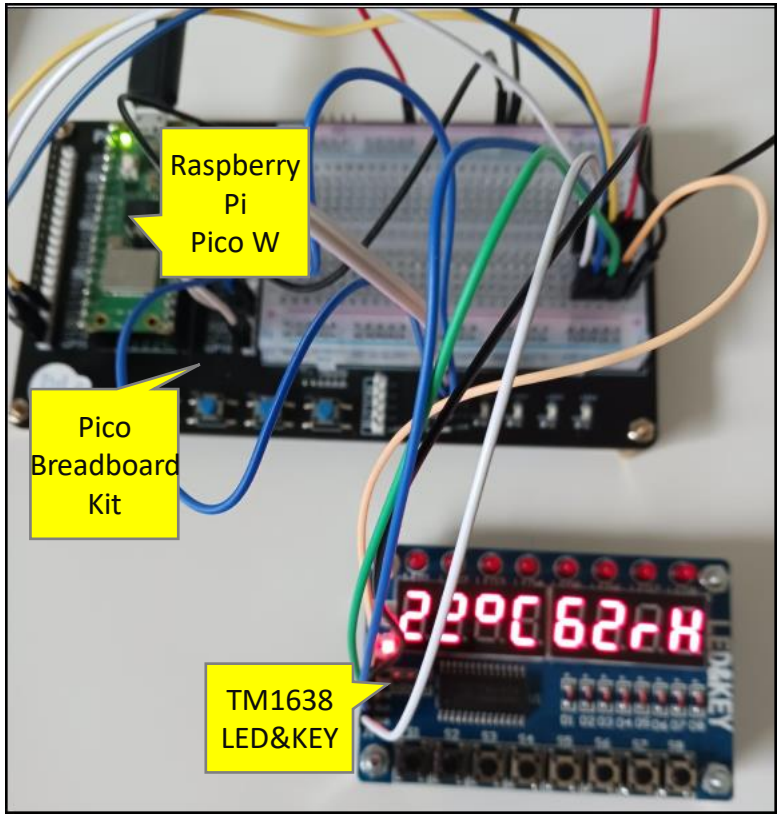
| Date | |
|---|---|
| 2023-03-20 10:42:01 | 8797395990,NOT VALID |
| 2023-03-20 10:42:01 | 8797395990 |
| 2023-03-20 10:41:49 | 879739599,VALID |
| 2023-03-20 10:41:49 | 879739599 |

# Project TM1638 LED&KEY

*Pico W with TM1638 to trigger Domoticz device action or Domoticz to set TM1638 LED or Display.*




Pico W with Breadboard Kit & TM1638 LED&KEY



# Project OLED Motherboard

*Domoticz sends RPi Motherboard sensor data to set the 0,96” I2C OLED display connected to the Pico W.*

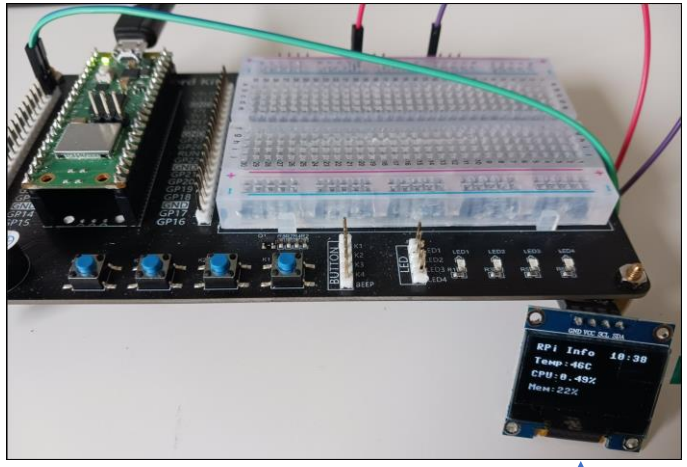
Domoticz Hardware

| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 5 | RPi Motherboard | Yes | Motherboard sensors | | | Disabled |

Domoticz Devices (selective)

| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 18 | RPi Motherboard | 0001 | 1 | Internal Temperature | Temp | LaCrosse TX3 | 38.9 C |
| | 26 | RPi Motherboard | 0000044D | 1 | CPU_Usage | General | Percentage | 0.41% |
| | 22 | RPi Motherboard | 0000044C | 1 | Memory Usage | General | Percentage | 18.35% |

Pico W with OLED Display




Automation Event dzVents

Domoticz Automation Event dzVents
Trigger: Timer
The event is reading in regular intervals the RPi motherboard data.
The data is converted to a JSON object and sent as HTTP POST request to the Pico W Web Server.

HTTP API/JSON Engine

```
{
 ["timestamp"]={["row"]=0,["col"]=0,["text"]="RPi Info 10:52"},
 ["cpuusage"]={["row"]=2,["col"]=0,["text"]="CPU:0.4%"},
 ["memoryusage"]={["row"]=3,["col"]=0,["text"]="Mem:18%"},
 ["internaltemperature"]={["row"]=1,["col"]=0,["text"]="Temp:39C"}
}
```

Pico W Web Server

OLED Display

HTTP POST request from Domoticz to Pico W Web Server

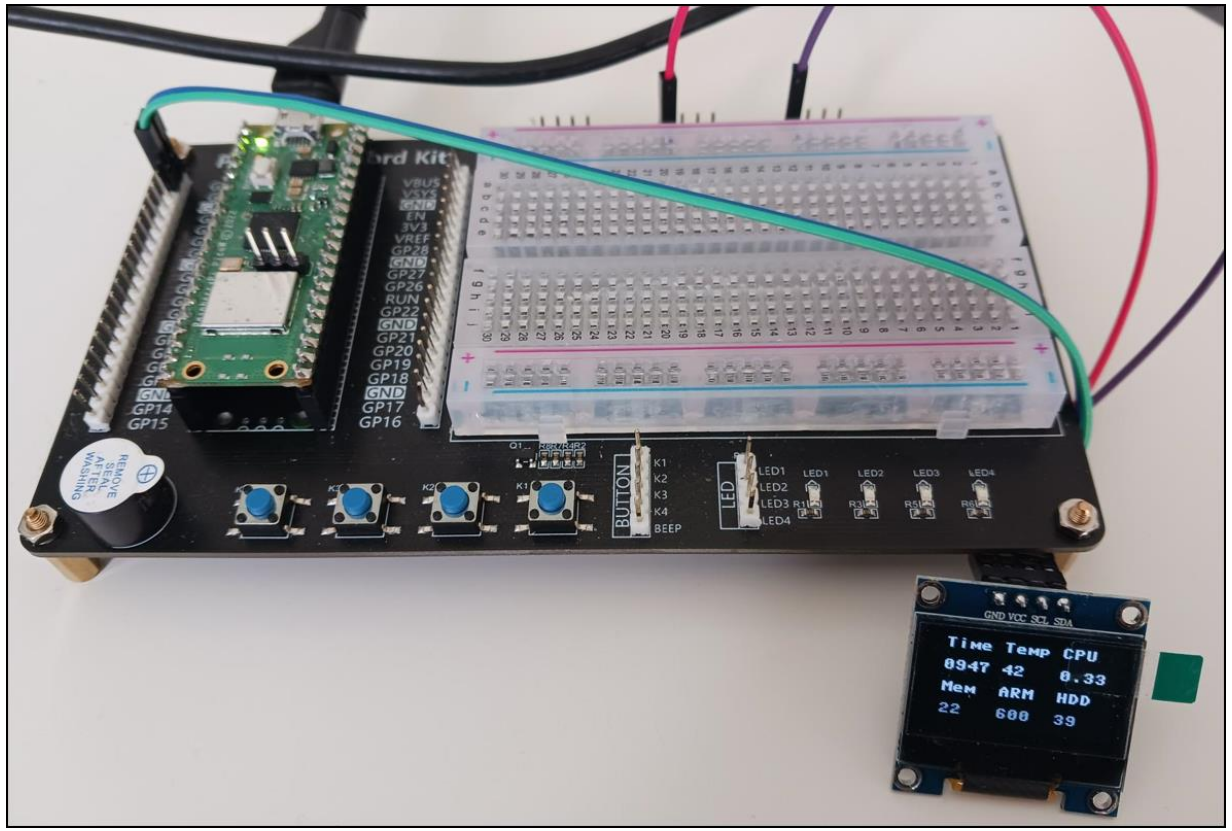
# Project OLED Motherboard /2

*Domoticz sends RPi Motherboard sensor data to set the 0,96” I2C OLED display connected to the Pico W.*
*The sensor data is displayed in up-to 6 blocks with title & value*

OLED Display 6 Blocks Setup
Each block 2 rows with 4 characters

| Col | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Row** | | | | | | | | | | | | | | | | |
| 0 | | Title | | | | | Title | | | | | Title | | | | |
| 1 | | Value | | | | | Value | | | | | Value | | | | |
| 2 | | Title | | | | | Title | | | | | Title | | | | |
| 3 | | Value | | | | | Value | | | | | Value | | | | |

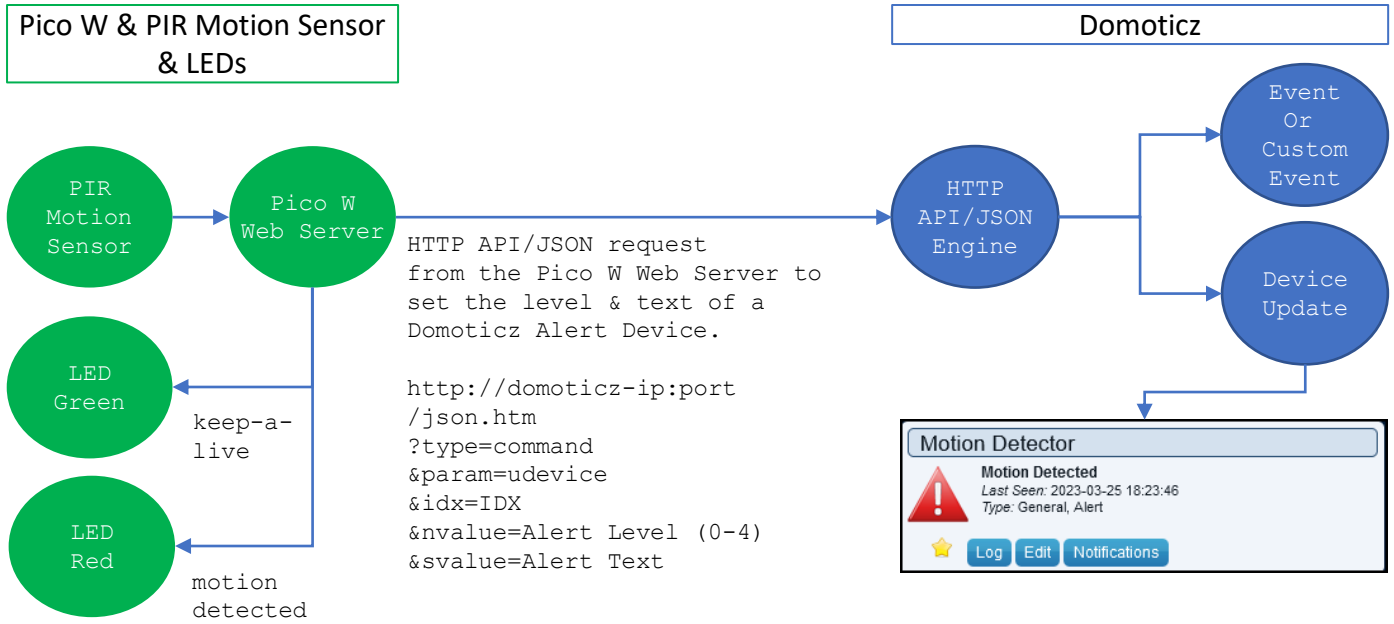
```
Block Data generated by Domoticz Automation Event.
Example JSON array with 6 blocks:
[
 {'block': 1, 'title': 'Time', 'value': '0947’},
 {'block': 2, 'title': 'Temp', 'value': 42},
 {'block': 3, 'title': 'CPU', 'value': 0.33},
 {'block': 4, 'title': 'Mem', 'value': 22},
 {'block': 5, 'title': 'ARM', 'value': 600},
 {'block': 6, 'title': 'HDD', 'value': 39}
]
```



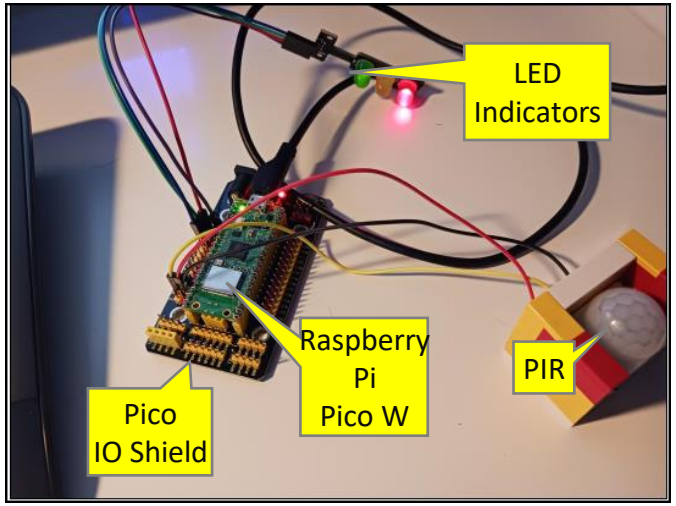

Pico W with OLED Display showing 6 blocks with RPi Motherboard sensor data

# Project PIR Motion Sensor

*PicoW to detect motion and sent message to a Domoticz Alert sensor.*



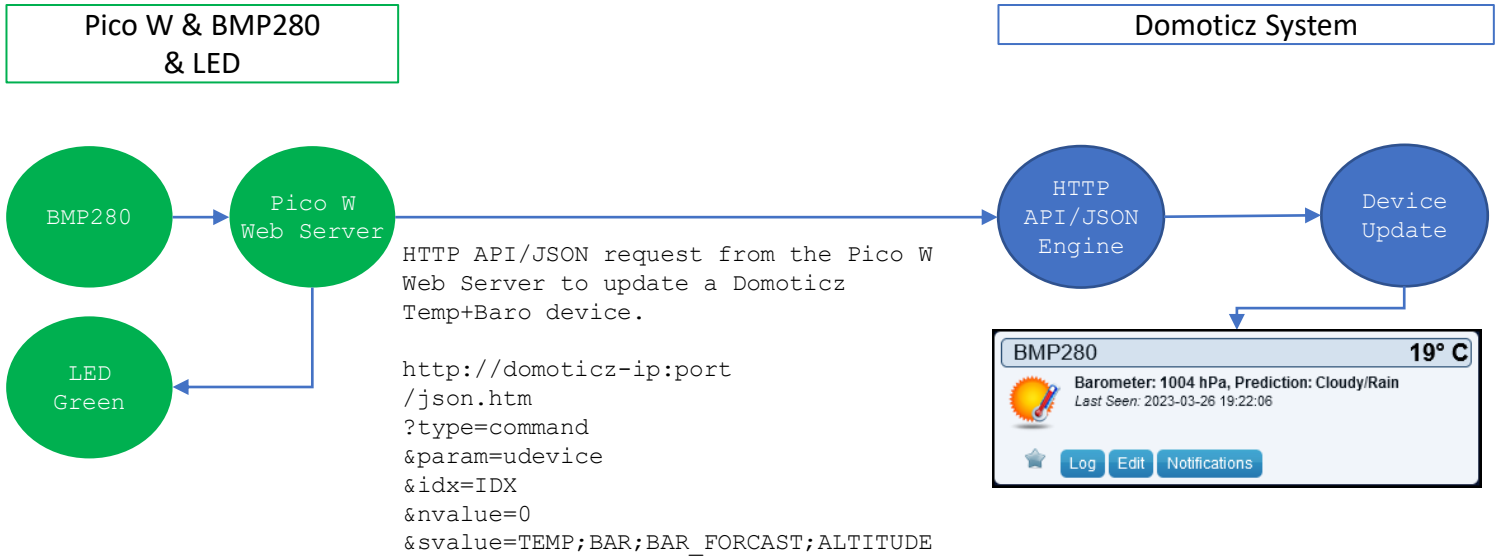

Pico W with PIR connected, and motion detected



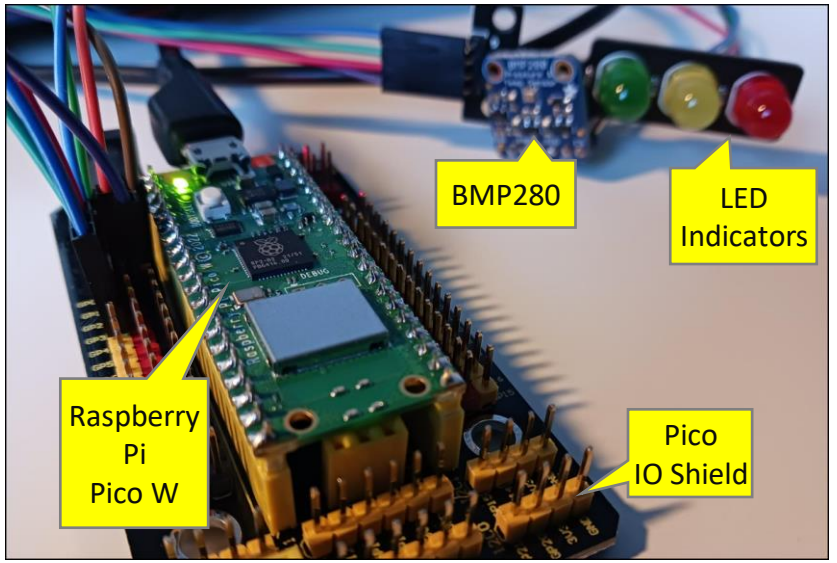

Note: The yellow LED from the LED traffic light is not used.

# Project BMP280

*Pico W samples BMP280 sensor data and triggers updating the Domoticz Temp+Baro device.*



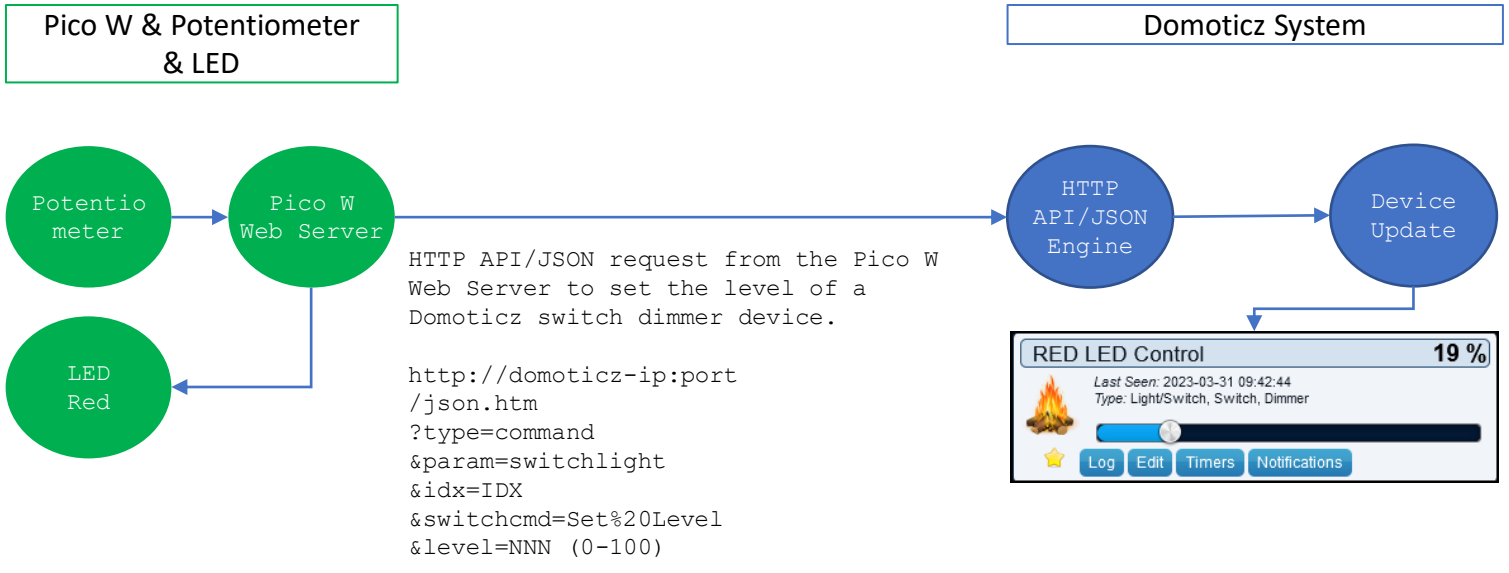

Pico W with Pico IO Shield & BMP280 & LEDs



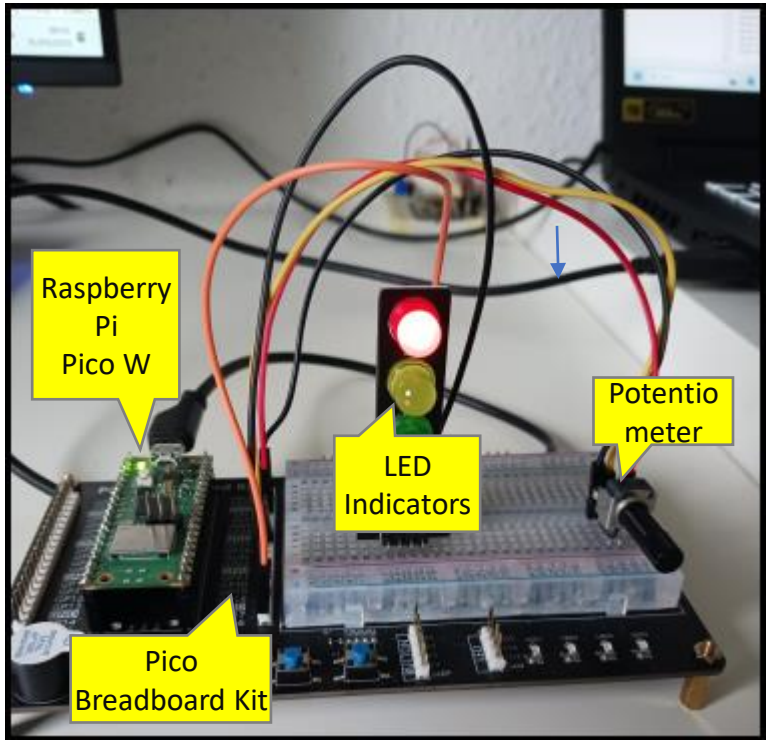

Note: The green LED is used only from the LED traffic light.

# Project Potentiometer Dimmer

*Pico W to set the level of a Domoticz Switch Dimmer device via Potentiometer.*



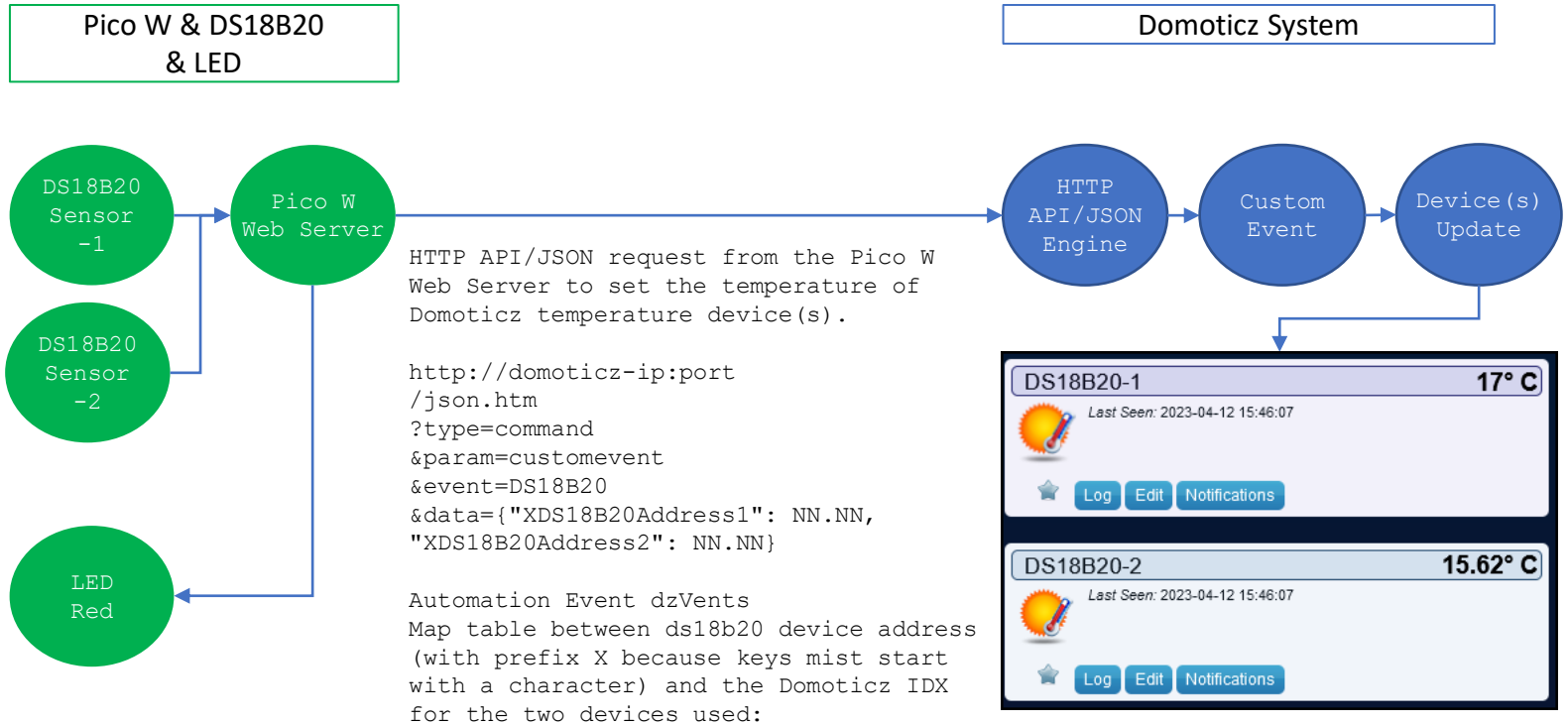

Pico W with Breadboard Kit & Potentiometer & LED




Note: The red LED is used only from the LED traffic light.

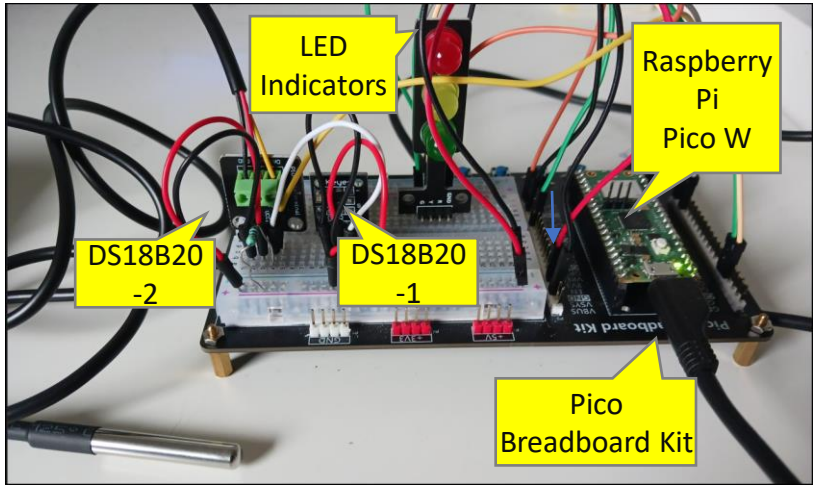
# Project DS18B20 - Custom Event

*Pico W samples DS18B20 sensor data and triggers Automation Event to update the Domoticz Temperature devices.*




```
HTTP API/JSON request from the Pico W
Web Server to set the temperature of
Domoticz temperature device(s).

http://domoticz-ip:port
/json.htm
?type=command
&param=customevent
&event=DS18B20
&data={"XDS18B20Address1": NN.NN,
"XDS18B20Address2": NN.NN}

Automation Event dzVents
Map table between ds18b20 device address
(with prefix X because keys mist start
with a character) and the Domoticz IDX
for the two devices used:

local devices = {
  X28FF5E1804150334 = 31, -- DS18B20-1
  X28330A9497040373 = 32  -- DS18B20-2
}
```

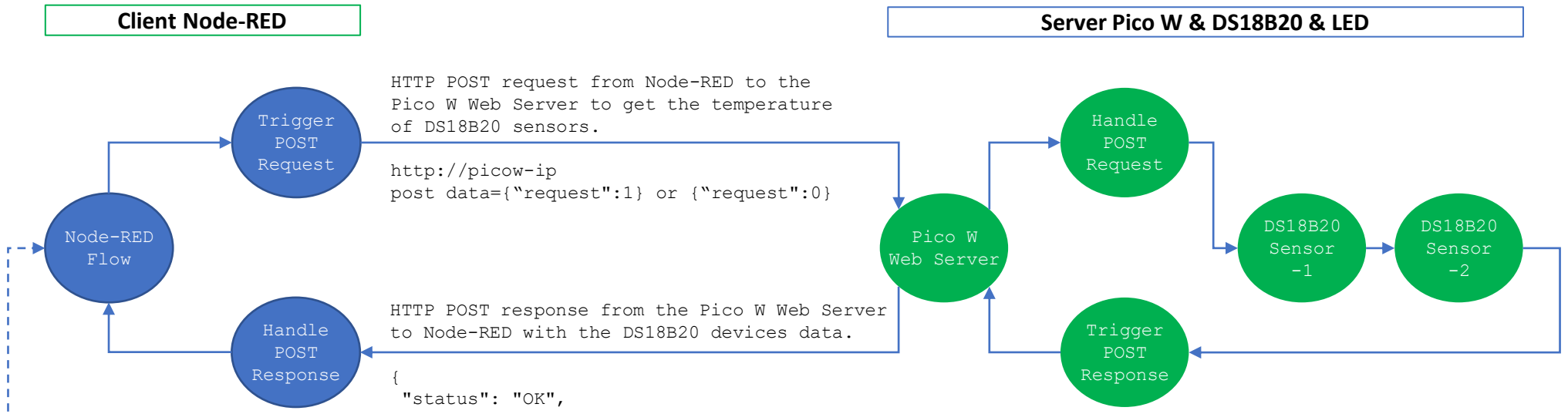
Pico W with Breadboard Kit & DS18B20 & LED

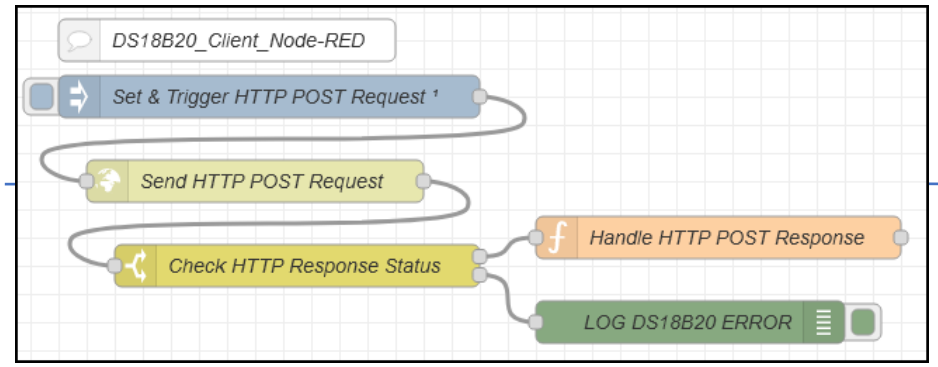



Note: There are two DS18B20 connected (DS18B20-1, DS18B20-2).
From the LED traffic light, the red LED is used only.

# Project DS18B20 - Node-RED (PULL)

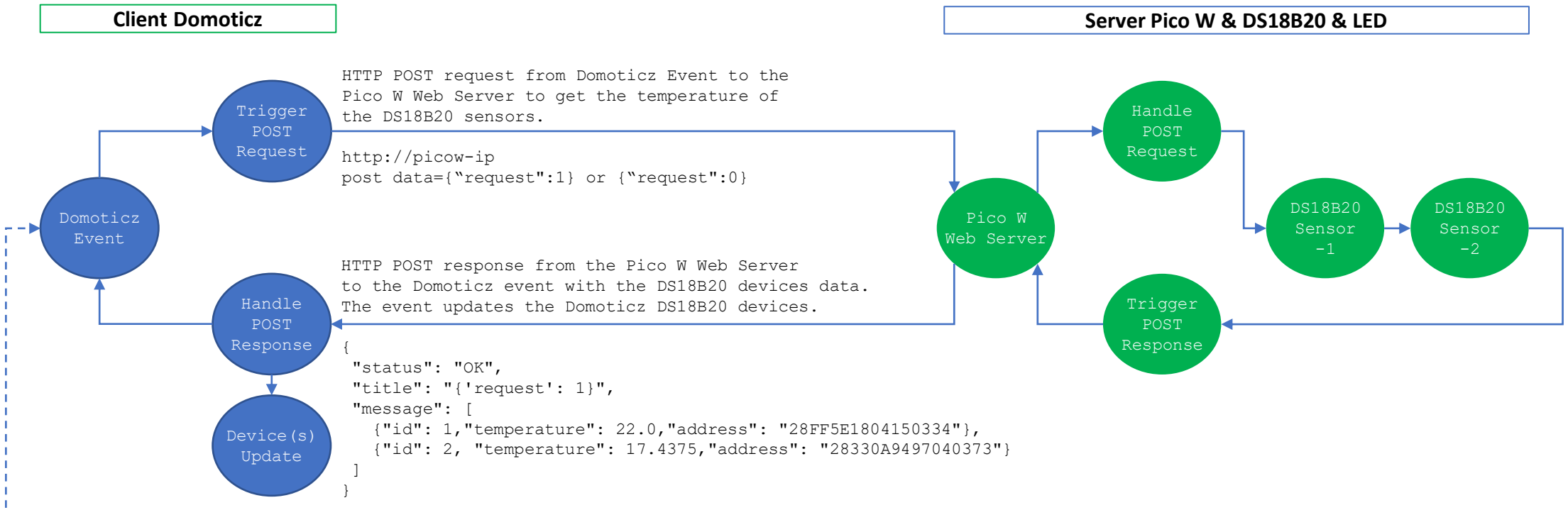
*Pico W Web Server listens to HTTP POST request from Node-RED and sends response with DS18B20 sensor data.*



# Project DS18B20 - Domoticz (PULL)

*Pico W Web Server listens to HTTP POST request from Domoticz and sends response with DS18B20 sensor data.*




```
dzVents Logic
return {
    on = {
        timer = { TIMER_RULE },
        devices = { IDX_SWITCH },
        httpResponses = { RES_HTTP }
    },
    execute = function(domoticz, triggeredItem)
        if (triggeredItem.isDevice or triggeredItem.isTimer) then
            requestData(domoticz)
        end
        if (triggeredItem.isHTTPResponse) then
            local data = triggeredItem.json
            domoticz.log(data)
            updateDevices(domoticz, data.message)
        end
    end
}
```

```
Start internal script: ds18b20_client_domoticz: Device: "PicoW Switch Control (Virtu
Finished ds18b20_client_domoticz
Start internal script: ds18b20_client_domoticz: HTTPResponse: "RES_DS18B20_REQUEST"
{["title"]="{'request': 1}", ["message"]={{["temperature"]=20.75, ["address"]="28FF
Updated: idx=31, name=DS18B20-1, a=28FF5E1804150334, t=20.75
Updated: idx=32, name=DS18B20-2, a=28330A9497040373, t=16.38
```

# Project Stepper Motor – Selector Switch Angle Move

*Domoticz Selector Switch Device State Change triggers Automation Event to move the Stepper Motor by Angle.*

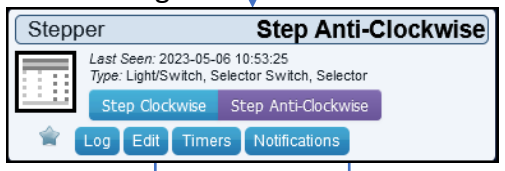
Domoticz Hardware

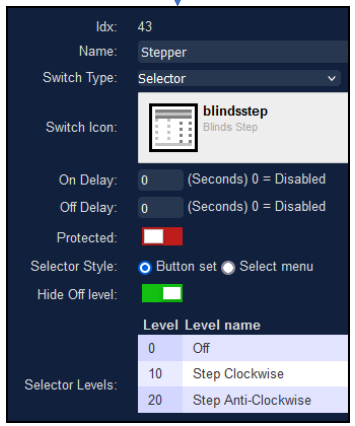
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

Domoticz Devices

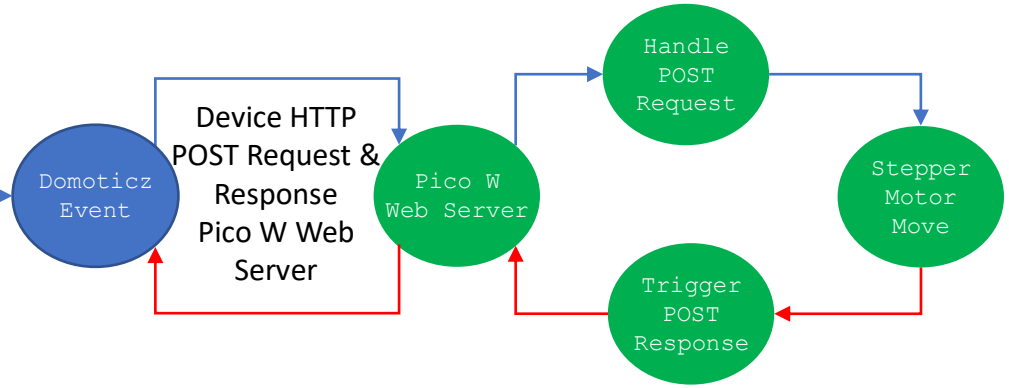
| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 43 | VirtualSensors | 0001407B | 1 | Stepper | Light/Switch | Selector Switch | Set Level: 20 % |

Device Widget



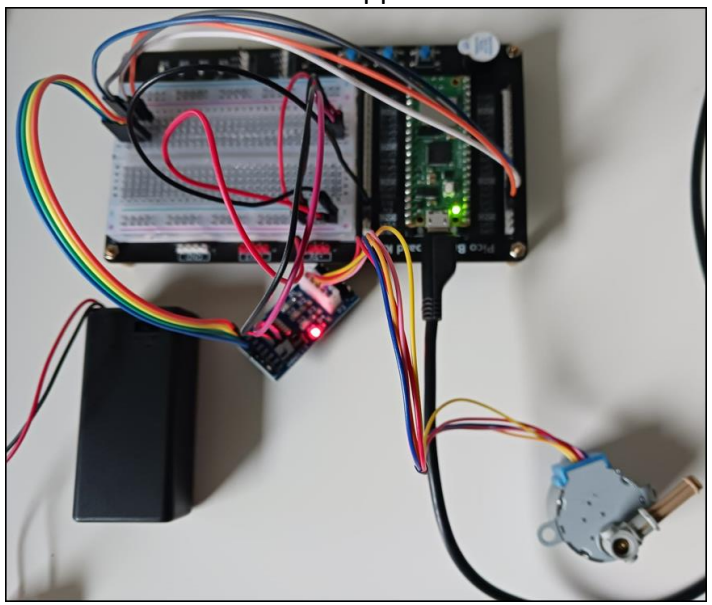

Device properties with
- custom icon
- 3 levels; Off level hidden.




Domoticz Automation Event dzVents triggered by state change of the Domoticz Selector Switch device (IDX=43).
The stepper angle moves 5° (hardcoded in the Automation event script).
The event submits HTTP POST request to the Pico W Web Server to move the stepper motor by the angle specified.

The post-data is a JSON Object:
{"angle": 5} or {"angle": -5}

Pico W with Breadboard Kit & UNL2003 driver & 28BYJ-48 stepper motor



# Project Stepper Motor - Blind Simulation

*Simulate Domoticz Blind Device State Change triggers Automation Event to move the Angle of a Stepper Motor.*

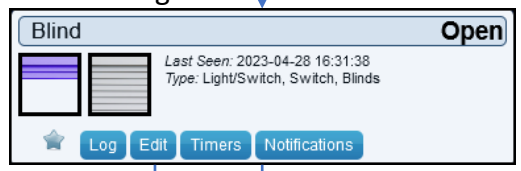
Domoticz Hardware

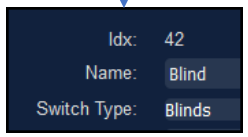
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

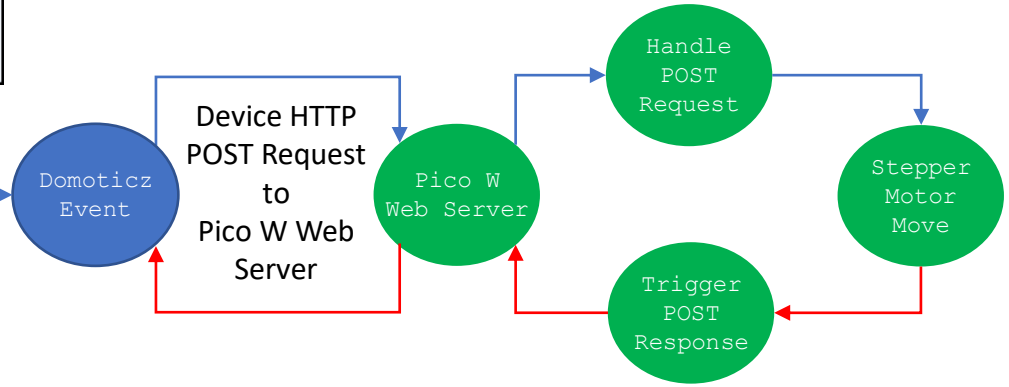
Domoticz Devices

| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 42 | VirtualSensors | 0001407A | 1 | Blind | Light/Switch | Switch | Open |

Device Widget

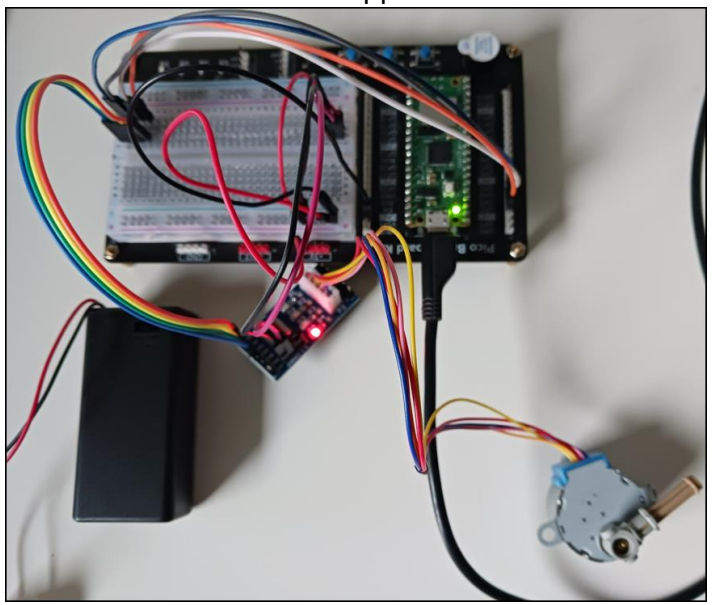



```
Domoticz Automation Event dzVents triggered by state change of the
Domoticz Blinds device (IDX=42).
The stepper angle to move is for state Open 180° and Closed -180°.
The event submits HTTP POST request to the Pico W Web Server to move
the stepper motor by the angle specified.

The post-data is a JSON Object:
{"angle": 180} or {"angle": -180}

Note: Instead of using the command "angle", the command "rotate" N
times is another possibility.
```

Pico W with Breadboard Kit & UNL2003 driver & 28BYJ-48 stepper motor



# Project Stepper Motor – Timer Run Stop

*Domoticz Selector Switch State Change triggers Automation Event to run or stop a Stepper Motor.*

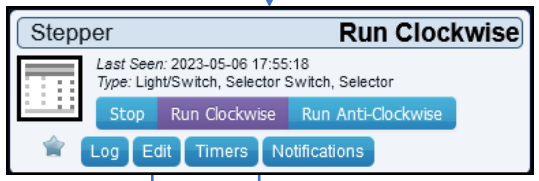
Domoticz Hardware

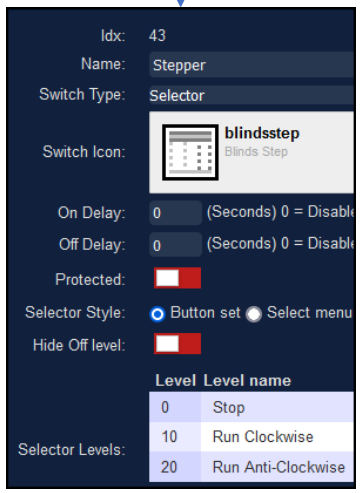
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

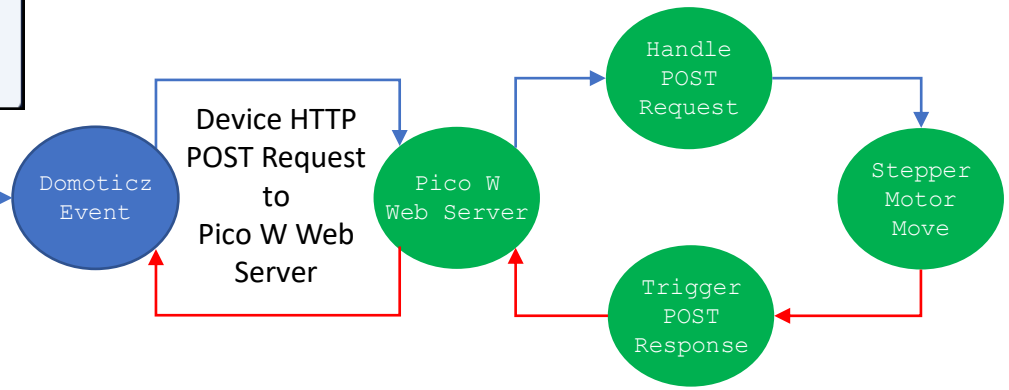
Domoticz Devices

| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 43 | VirtualSensors | 0001407B | 1 | Stepper | Light/Switch | Selector Switch | Off |

Device Widget



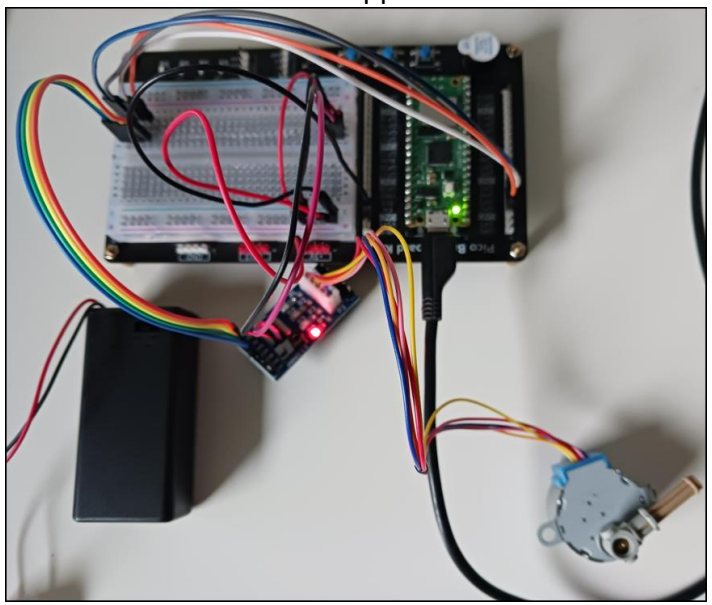




Domoticz Automation Event dzVents triggered by state change of the Domoticz Selector Switch device (IDX=43).

The event submits HTTP POST request to the Pico W Web Server to run or stop the stepper motor.
The post-data is a JSON Object depending action:

- Stop = {"command"="stop"}
- Run Clockwise = {"command"="run", "direction"=1}
- Run Anti-Clockwise = {"command"="run", "direction"=-1}

Pico W with Breadboard Kit & UNL2003 driver & 28BYJ-48 stepper motor



# Project Distance Sensor - HTTP API/JSON

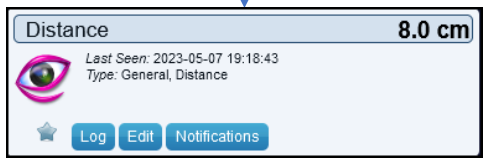
*Pico W samples Distance sensor HC-SR04 data and triggers updating the Domoticz Distance device.*

Domoticz Hardware

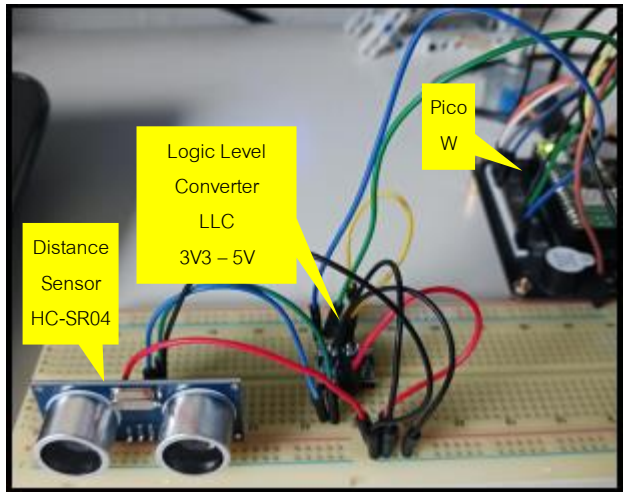
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

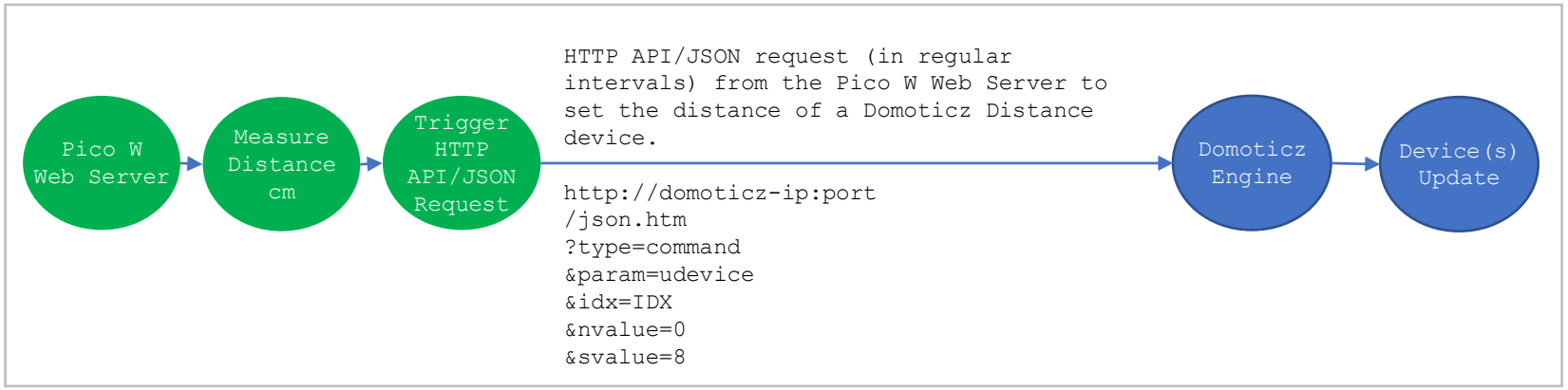
Domoticz Devices

| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 44 | VirtualSensors | 00082044 | 1 | Distance | General | Distance | 8.0 cm |




Pico W with Distance Sensor HC-SR04 and LLC



# Project Distance Sensor - MQTT Autodiscover

*Pico W publishes MQTT Config & State Messages to Domoticz MQTT Auto Discovery Client Gateway.*

**Pico W**

**MQTT Configuration Message**
**Distance Sensor**
**Topic**
domoticz/sensor/distance/config
**Payload**

```
{
 "name": "Distance",
 "device_class": "distance",
 "state_topic": "domoticz/sensor/distance/state",
 "value_template": "{{value_json.distance}}",
 "unit_of_measurement": "cm",
 "unique_id": "DIST001"
}
```

HC-SR04 Sensor

Pico W Web Server & MQTT Client

MQTT publish device config

Create Device

MQTT publish device state

Change Device State

**MQTT State Message**
**Topic**
domoticz/sensor/distance/state
**Payload**

```
{
 "distance":16.2
}
```

**Domoticz**

| Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|
| 45 | MQTTADGateway | DIST001 | 1 | Distance | General | Distance | 16.2 cm |

MQTT Auto Discovery Client Gateway

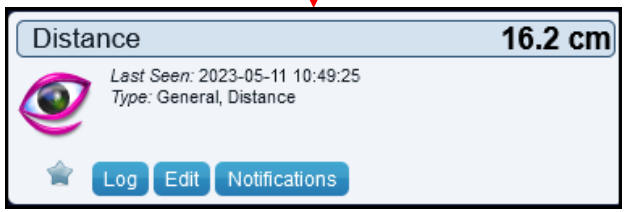
Create Device

Device Distance create

Change Device State

Device Distance update

MQTT publish domoticz/ out



# Project IKEA VINDRIKTNING Air Quality Sensor – HTTP API/JSON

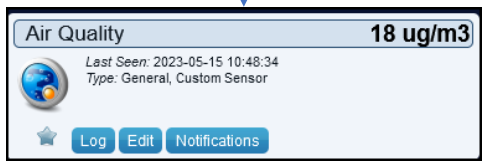
*IKEA VINDRIKTNING Air Quality sensor sends PM 2.5 ug/m3 values to the Pico W to update Domoticz Custom Sensor.*

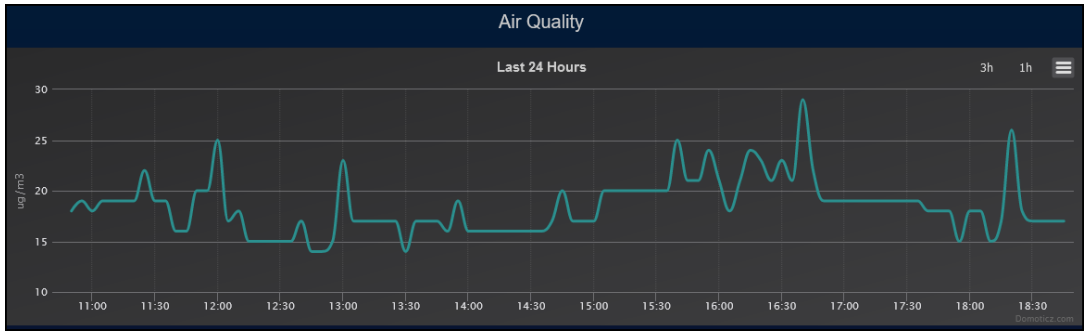
Domoticz Hardware

| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 3 | VirtualSensors | Yes | Dummy (Does nothing, use for virtual switches only) Create Virtual Sensors | | | Disabled |

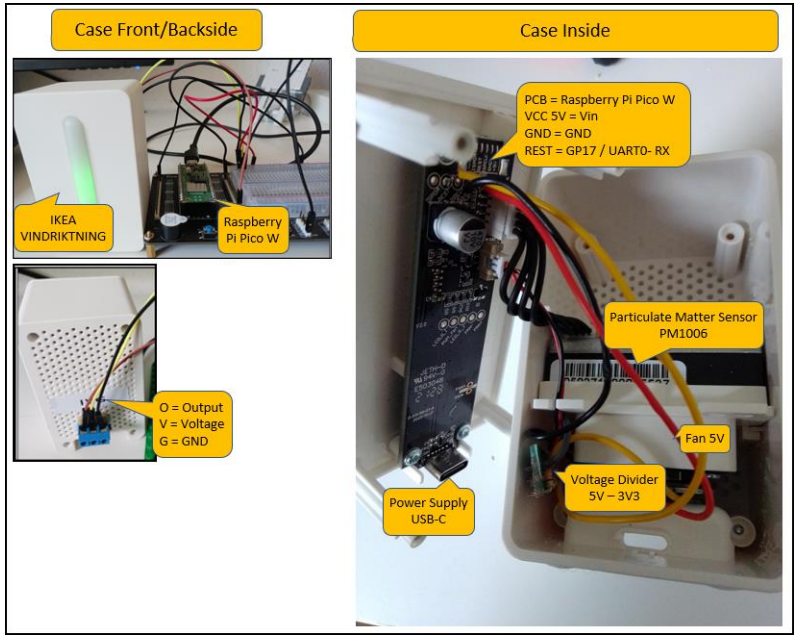
Domoticz Devices

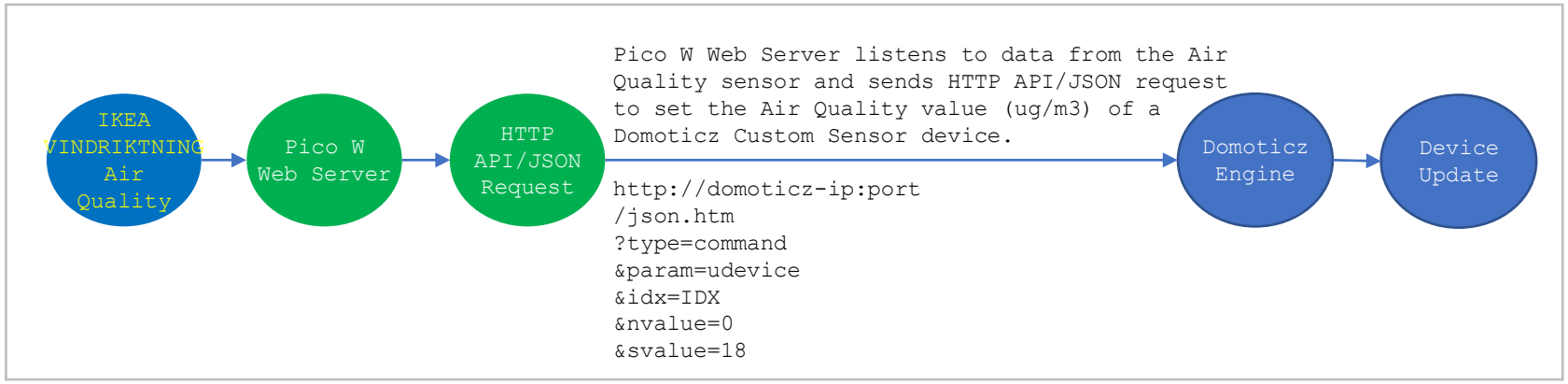
| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 46 | VirtualSensors | 00082046 | 1 | Air Quality | General | Custom Sensor | 18 ug/m3 |




Pico W with IKEA VINDRIKTNING Air Quality sensor




*IKEA®* is a *trademark* of Inter-*IKEA* Systems B.V.

# Project Bluetooth Low Energy - Environment Sensor

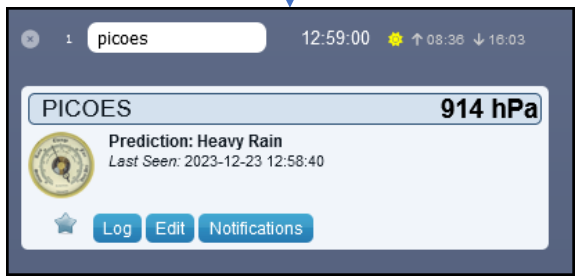
*Pico W advertises simulated environment data > OpenMQTTGateway > Node-RED > Domoticz Auto Discovery Client Gateway > Device Temp+Humidity+Baro.*

Domoticz Hardware

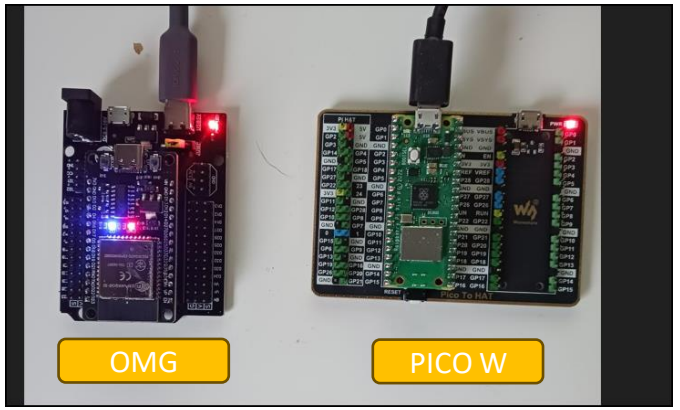
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 13 | OMGGW | Yes | MQTT Auto Discovery Client Gateway with LAN interface Setup | localhost | 1883 | Disabled |

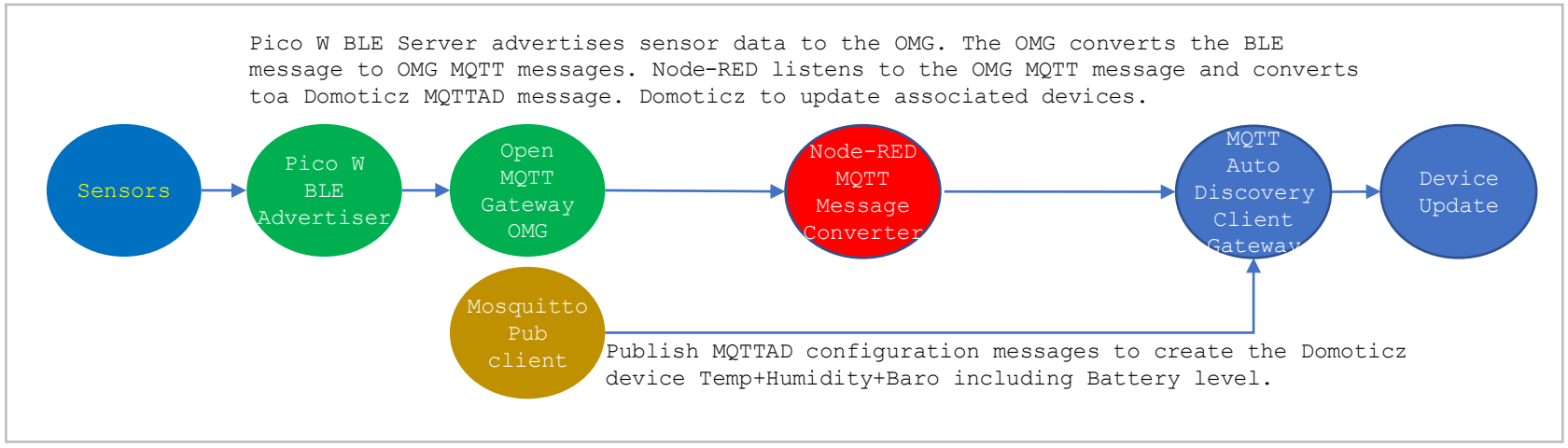
Domoticz Devices

| Idx | Hardware | ID | Unit | Name | Type | SubType | Data |
|---|---|---|---|---|---|---|---|
| 38 | OMGGW | 0001 | 1 | PICOES | Temp + Humidity + Baro | Weather Station | 26.3 C, 56 %, 909.0 hPa |




Pico W with OMG



# Project Bluetooth Low Energy - Actuator

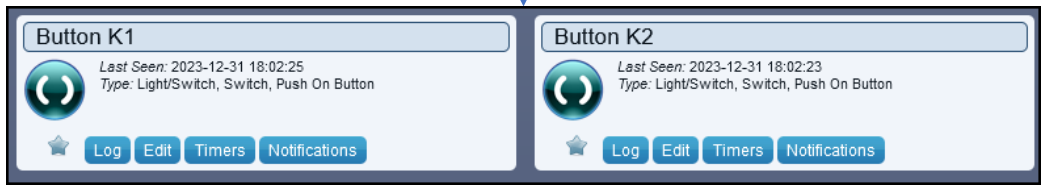
*Pico W advertises actuator data via OMG > Node-RED > Domoticz Auto Discovery Client Gateway > Device Push On Button.*

Domoticz Hardware

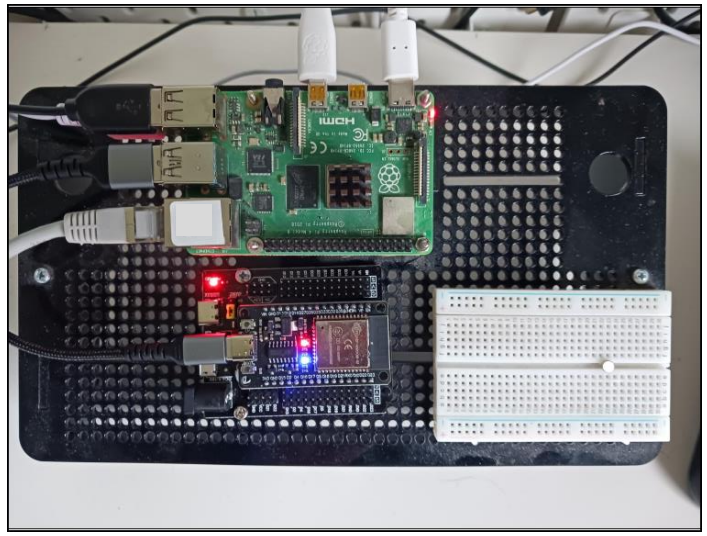
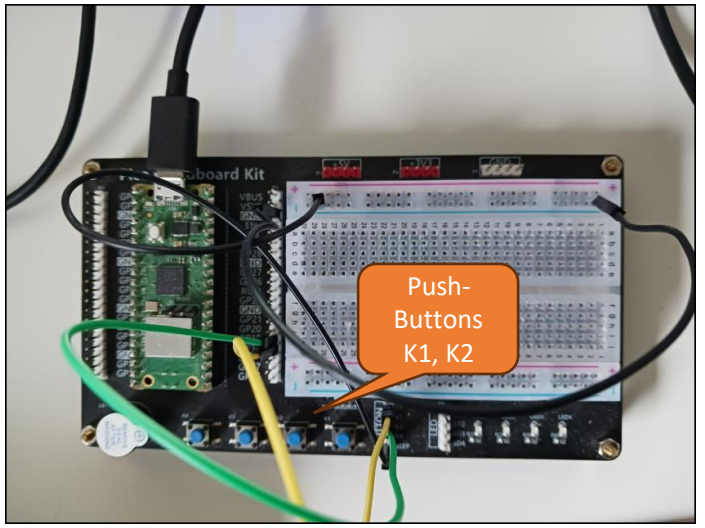
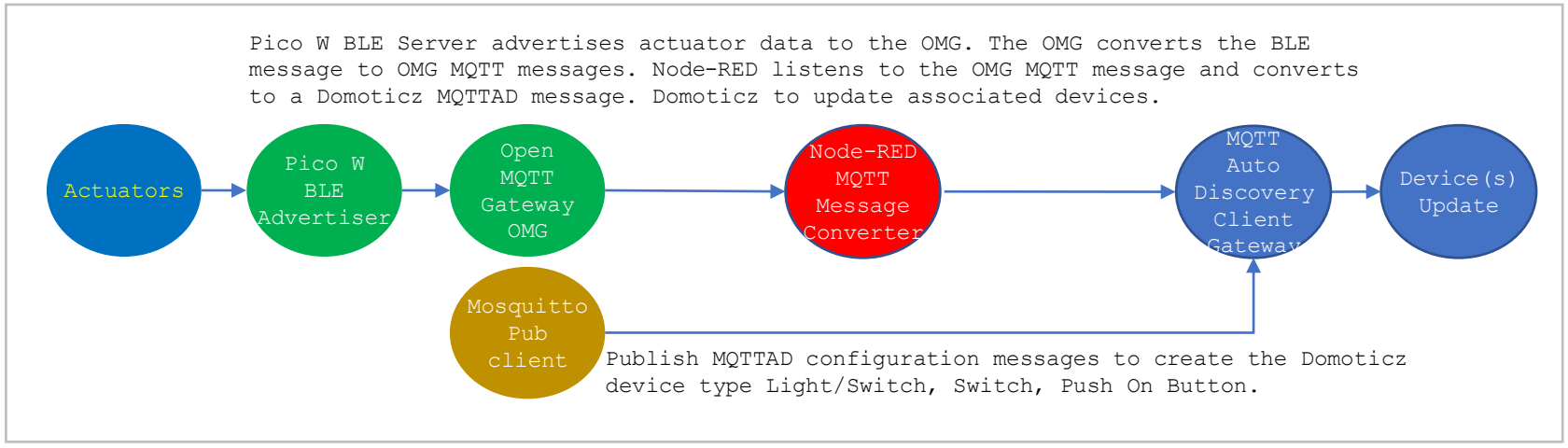
| Idx | Name | Enabled | Type | Address | Port | Data Timeout |
|---|---|---|---|---|---|---|
| 13 | OMGGW | Yes | MQTT Auto Discovery Client Gateway with LAN interface Setup | localhost | 1883 | Disabled |

Domoticz Devices

| | Idx | Hardware | ID | Unit | Name | Type | SubType | |
|---|---|---|---|---|---|---|---|---|
| | 39 | OMGGW | K1FFFE | 1 | Button K1 | Light/Switch | Switch | On |
| | 40 | OMGGW | K2FFFE | 1 | Button K2 | Light/Switch | Switch | On |




Raspberry Pi 4 with ESP32-WROOM-32 OMG



Pico W with Pico Breadboard Kit