

Automated Humidifier Supporting Device Prototype

“The best supporting tool for your humidifier.”

COSC 417 - Yuan Hu AKA. YooHoo 

Overview

- Inspiration
 - The origin problem
- Solution
 - Design and Implementation
- Obstacles
 - “Just a few outrages, it feels good to smash some keyboards”
- Improvements
 - What could be done better and more thoughts for the future

Inspirations?

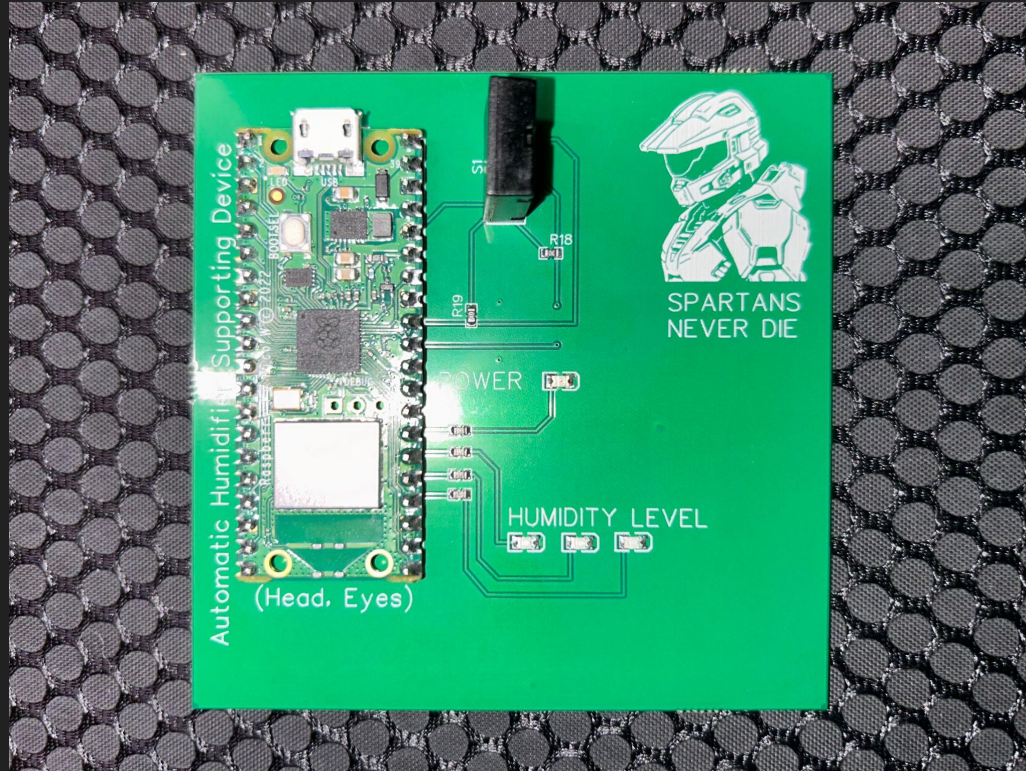
DRY!!

Inspirations

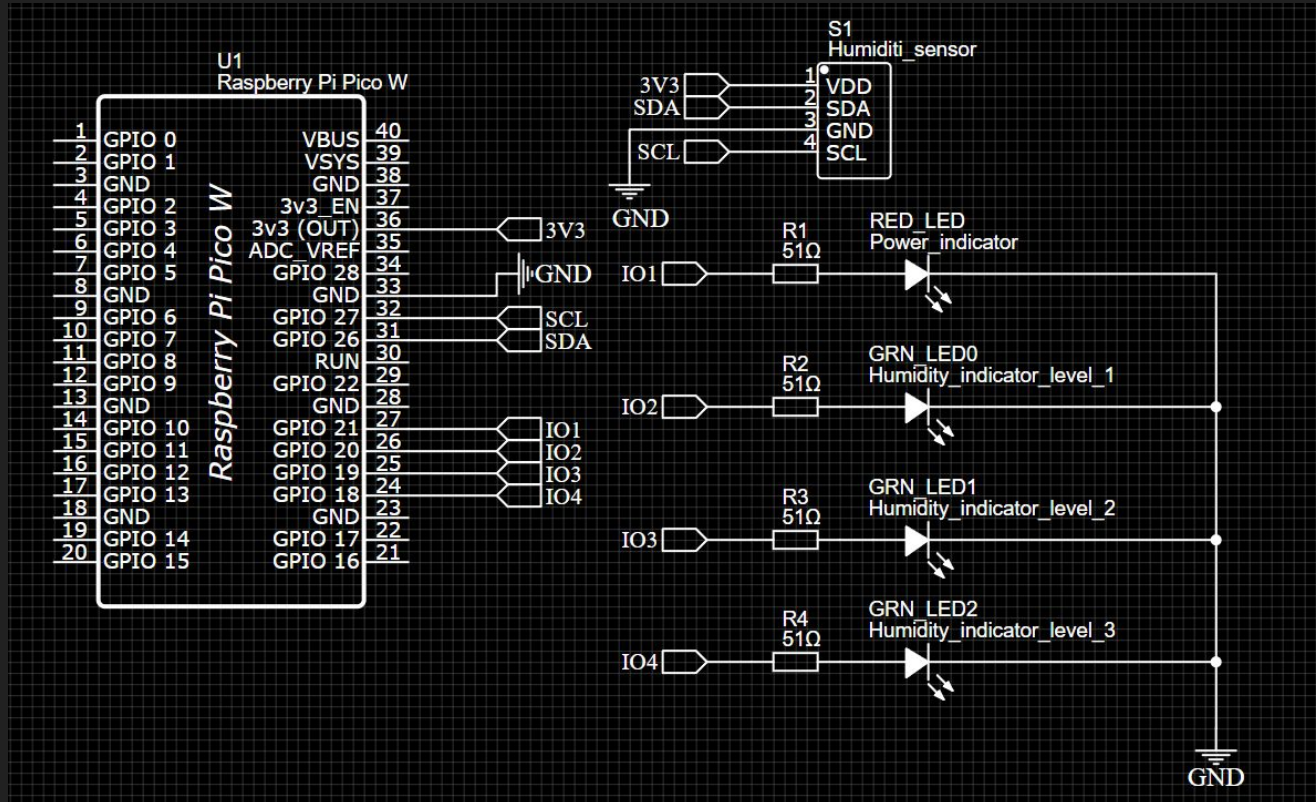
- 01 Arid summer and indoor air being dry during winter
- 02 Keep turning on and off the humidifier is too much of a hustle
- 03 Because Fritter told us to and it's worth 60%!

Solutions?

All mighty AHSD! (*Automated Humidifier Supporting Device*)



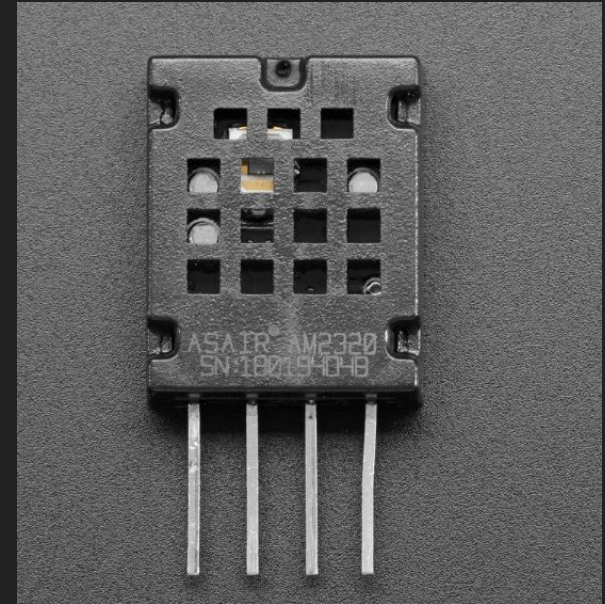
Solutions - Designing with the EDA



Solutions - AM2320 humidity & temperature sensor

- Capacitance Humidity Measurement Technology
- I2C Communication protocol

Source: <https://mallimages.ofweek.com/Upload/guige/2021/01/04/2690/5ff271536d392.pdf>



Solutions - Relationships

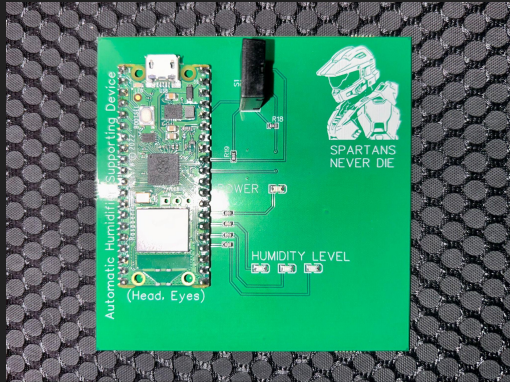
Mighty AHSD

----->

Smart plug

----->

Dumb Humidifier



Solutions - How to send love letters to the smart plug?

- Reverse engineer the KASA home APP

```
POST https://eu-wap.tplinkcloud.com/?token=74adcc7e-64f7-47c1-a751-dece6d2f4704
&appName=Kasa_Android&termID=c69d10e5-5307-4602-b2c8-eee8f3761238&appVer=1.4.4.607&ospf=Android+6.0.1
&netType=wifi&locale=en_US HTTP/1.1
Content-Type: application/json
User-Agent: Dalvik/2.1.0 (Linux; U; Android 6.0.1; XXPhoneModelXX)
Connection: Keep-Alive
Content-Length: 160
Host: eu-wap.tplinkcloud.com

{
  "method": "passthrough",
  "params": {
    "deviceId": "80067AC4FDBD41C54C55896BFA28EAD38A87A5A4",
    "requestData": "{\"system\": {\"set_relay_state\": {\"state\": 1}}}"
  }
}
```

Solutions - How to send love letters to the smart plug?

- Acquire a authenticated token

Generate the UUID

Online UUID Generator

Your Version 4 UUID:

2cc908cf-ad84-4bfb-8776-314b83fbe44c

Copy

[Refresh](#) page to generate another.

Destination	POST ▼	https://wap.tplinkcloud.com
Follow redirects: Off		
Authentication	+ Add Authentication	
Headers	Content-Type	application/json
remove all		
+ Add another header		
Parameters	<pre>{ "method": "login", "params": { "appType": "Kasa_Android", "cloudPassword": "xxxxxxx", "cloudUserName": "xxxxxxx@gmail.com", "terminalUUID": "e933737a-9226-43db-a07d-123456789123" } }</pre>	
remove all		

Solutions - How to send love letters to the smart plug?

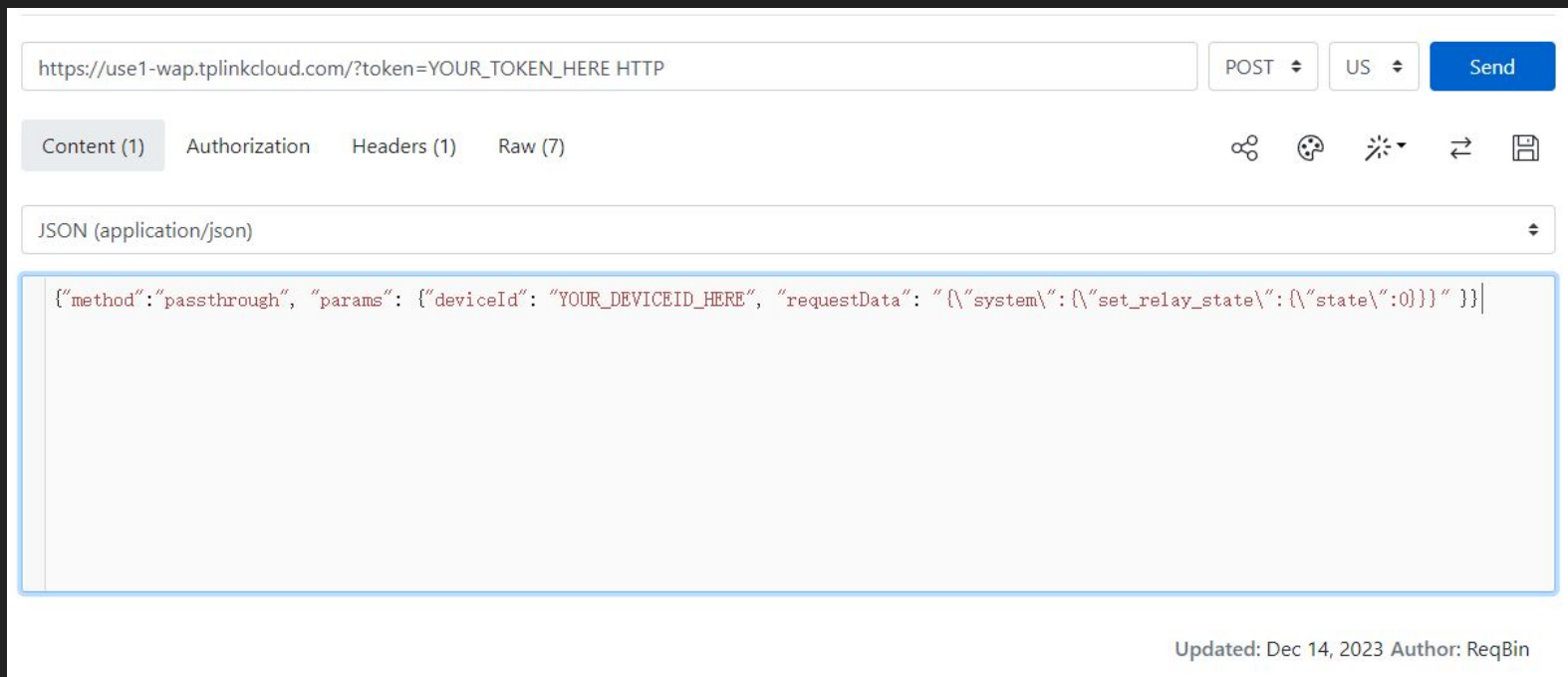
- Acquire the device ID

```
curl -s --request POST "https://wap.tplinkcloud.com?token=YOUR_TOKEN_HERE HTTP/1.1" \
--data '{"method":"getDeviceList"}' \
--header "Content-Type: application/json"
```

```
"deviceList" : [
{
  "appServerUrl" : "https://eu-wap.tplinkcloud.com",
  "isSameRegion" : true,
  "deviceMac" : "",
  "status" : 1,
  "hwId" : "XXXX...",
  "deviceId" : "XXXX...",
  "oemId" : "XXXX...",
  "fwVer" : "1.0.8 Build 151101 Rel.24452",
  "deviceType" : "IOT.SMARTPLUGSWITCH",
  "alias" : "My Smart Plug",
  "fwId" : "BFF24826FBC561803E49379DBE74FD71",
  "deviceName" : "Wi-Fi Smart Plug",
  "deviceHwVer" : "1.0",
```

Solutions - How to send love letters to the smart plug?

- All dressed up and it's time to send



The screenshot shows a REST client interface with the following components:

- URL Bar:** `https://use1-wap.tplinkcloud.com/?token=YOUR_TOKEN_HERE HTTP`
- Method:** POST
- Environment:** US
- Buttons:** Send
- Tabs:** Content (1), Authorization, Headers (1), Raw (7)
- Content Type:** JSON (application/json)
- Request Body:**

```
{ "method": "passthrough", "params": { "deviceId": "YOUR_DEVICEID_HERE", "requestData": "{ \"system\": { \"set_relay_state\": { \"state\": 0 } } }" } }
```
- Footer:** Updated: Dec 14, 2023 Author: ReqBin

Demonstration



Obstacles - Lesson No.1, don't use crappy libraries

- Where is the problem?

```
Chip = AM2320
Chip = AM2320
  Humidity = 33.30%
  Temperature = 22.20*C
```

```
Chip = AM2320
Chip = AM2320
  Humidity = 33.20%
  Temperature = 22.10*C
```

```
Chip = AM2320
Chip = AM2320
  Humidity = 33.20%
  Temperature = 22.10*C
```

```
HTTP response code: 200
HTTP String: {"error_code":0,"result":{"responseData":{"system":{"set_relay_state":{"err_code":0}}}}}
  Humidity = 32.00%
  Temperature = 0.00*C
```

```
Connected to WiFi network with IP Address: 10.0.0.215
HTTP Response code: 200
HTTP String: {"error_code":0,"result":{"responseData":{"system":{"set_relay_state":{"err_code":0}}}}}
  Humidity = 32.00%
  Temperature = 0.00*C
```

```
Connected to WiFi network with IP Address: 10.0.0.215
HTTP Response code: 200
HTTP String: {"error_code":0,"result":{"responseData":{"system":{"set_relay_state":{"err_code":0}}}}}
  Humidity = 32.00%
  Temperature = 0.00*C
```

```
Connected to WiFi network with IP Address: 10.0.0.215
```


Obstacles - Lesson No.1, don't use crappy libraries

- Where is the problem?

```
State:
0
Chip = AM2320
State:
5
  Humidity = 33.50%
  Temperature = 22.10*C

Chip = AM2320
State:
0
Chip = AM2320
State:
5
  Humidity = 33.50%
  Temperature = 22.10*C
```

```
// ===== Start conversion =====
uint8_t AM2320::startConvert()
{
  State = 1;
  // Wake
  communication->beginTransaction(AM2320_address);
  communication->endTransmission();
  // Send request for read 4 bytes
  communication->beginTransaction(AM2320_address);
  communication->write(0x03); // request
  communication->write(0x00); // from 0 address
  communication->write(0x04); // 4 bytes for read
  State = communication->endTransmission(1);
  return State;
}
```

Improvements - What if I'm using induction stove on the socket

- Need to double check the state of the smart plug before it's too late
- Authenticated token may expire, it would be better for acquiring the token to be automated as well

Thank you!
And Merry Christmas! 🎅 YooHoo~

