

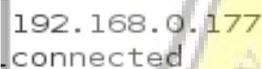
## CHAPTER 5

### IMPLEMENTATION AND TESTING

#### 5.1 Implementation

ARDUINO IDE application is used in this project, where the application is to program on arduino microcontroller, here are the implementation of the program.

First arduino ask IP, then after getting IP, arduino will link to the server MQTT to connect it to manager node-red. If connected on the server then the message will shows "192.168.0.101 Connected"



```
192.168.0.177
connected
```

Illustration 5.1.1: connected server

Then the program will check the message from server, when there is a message it will be saved first, then if the topic are match with the data in arduino then it will run the message order :

```
1. if(c == "box")
2. {
3.   Serial.print("Changing RoomBox lamp to ");
4.   if((char)message[0] == '1'){
5.     digitalWrite(LampuNeonBox, LOW);
6.     Serial.println("on");
7.     client.publish("box", "ada pesan on");
8.
9.     sensorTaman();
10.    client.publish("Status LDRB", LDR2);
11.   }
12.   else if((char)message[0] == '0'){
13.     digitalWrite(LampuNeonBox, HIGH);
14.     Serial.println("off");
15.     client.publish("box", "ada pesan off");
16.
17.     sensorTaman();
18.     client.publish("Status LDRB", LDR2);
```

```

19.     }
20.   }
21.   else if(c == "taman")
22.   {
23.     Serial.print("Changing taman lamp to ");
24.     if((char)message[0] == '1'){
25.       digitalWrite(LampuTaman, LOW);
26.       Serial.println("on");
27.
28.       client.publish("taman", "ada pesan on");
29.       sensorTaman();
30.       client.publish("Status LDR", LDR1);
31.     }
32.     else if((char)message[0] == '0'){
33.       digitalWrite(LampuTaman, HIGH);
34.       Serial.println("off");
35.       client.publish("taman", "ada pesan off");
36.
37.       sensorTaman();
38.       client.publish("Status LDR", LDR1);
39.     }
40.   }

```

topic “box” with the message “1” will turn the lights on the neon box, the program will publish the values that the LDR get. Then the manager will show the state of neon box, if the received message is to light up the neon box then it will turn on. But if the neon box received message is not turn them on, then the LED is broken or damaged. So also with the garden lamps.

For the mq-2 and mq-7 sensor program will publish data to mqtt, and subscribe by node-red as below :

```

41.
42.   void Gas(){
43.     RuangGas = analogRead(GasLPG);
44.     static char Gas[50];
45.     dtostrf(RuangGas, 6, 2, Gas);
46.     int state=0;
47.
48.     if(RuangGas > 130 && state==0)
49.     {
50.       client.publish("Status Gas LPG", Gas);
51.       state =1;
52.     }
53.     else if (RuangGas <130 && state==1)

```

```

54.     {
55.
56.         state = 0;
57.         client.publish("Status Gas LPG", Gas);
58.     }
59. }
60.
61. void Rokok(){
62.     RuangRokok = analogRead(GasMonoksida);
63.     static char rokok[50];
64.     dtostrf(RuangRokok, 6, 2, rokok);
65.     int state = 0;
66.
67.     if(RuangRokok > 125 && state == 0){
68.
69.         client.publish("Status Rokok", rokok);
70.         state = 1;
71.     }
72.     else if(RuangRokok < 125 && state == 1){
73.         state = 0;
74.         client.publish("Status Rokok", rokok);
75.
76.     }
77.
78. }

```

in this section the program will take the values on the mq-2 sensor, if the value is more than 130 and state 0, then the data is sent to the server and subscribe by node-red, and the state will change to one. So that the program will not send value continuously, that could make the system paralyzed. It also have the same system with MQ-7.

```

79.     if (client.connect("arduinoClient")) {
80.         //client.publish("a","hello world");
81.         client.subscribe("box");
82.         client.subscribe("taman");
83.         client.subscribe("dalam rumah");
84.         client.subscribe("Status LDR");

```

```

85.     client.subscribe("Status LDRB");
86.     client.subscribe("Status Gas LPG");
87.     client.subscribe("fan_besar");
88.     client.subscribe("fan_kecil");
89.     client.subscribe("Status Rokok");// topik dari nodered
90.     Serial.println("connected");
91.     }

```

In this section there is the void setup, where the program will subscribe or receive the published message by the node-red manager. The data received by the program will be sent to each topic, and run according to the received message.

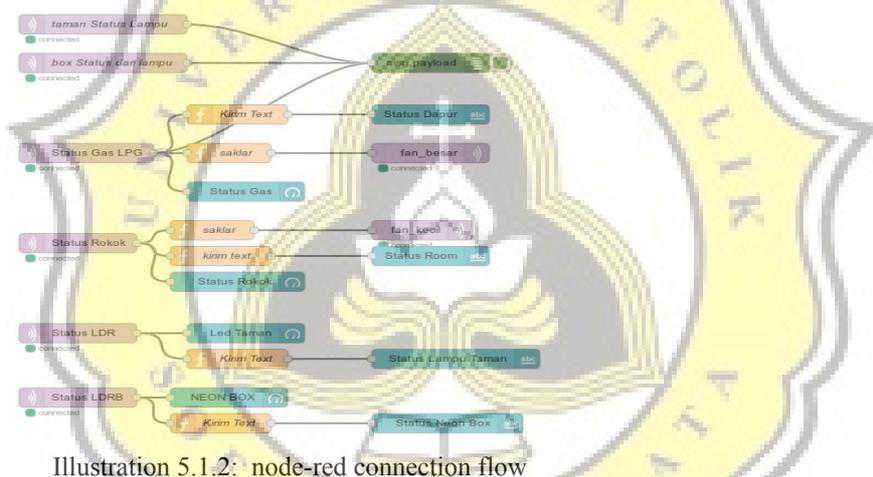


Illustration 5.1.2: node-red connection flow

In the last section, as seen above is a configuration where each part is connected between manager (node-red) with mqtt, and also mqtt with arduino. to monitor and control every parts are set in the manager node-red. The things that controlled are include turning on the garden lamps and neon box, that the turn off and turn on timing have been set. If the neon box ligths turn on, then the display will shows the sign. Beside that, the node red also monitor whether there is a gas leak or cigarette smoke in the cafe, if detected then the node-red will give notification and also turn the fan on to remove the gas.

## 5.2 Testing

In this project 2 testing is done, that is connection testing and dashboard node-red testing. Testing can be seen as follows:

### 5.2.1 Arduino Connection

Firstly, the program reuests IP from DHCP server taht is running in Router. Then it will connect to Mqtt server at 192.168.0.101 port 1883. After this the program will loop to subscribe messages from Mqtt server. Depend on the messages these are action those will be done:

The message consists of topic and value, the topic is a text data type and the value is an interger data type.

If the topic is “box” and the value is 1 then it turns on the neon box lamp by `digitalWrite(LampuNeonBox, HIGH)`. If the topic is “box” and the value is 0 then it turns off the neon box lamp by `digitalWrite(LampuNeonBox, LOW)`.

If the topic is “taman” and the value is 1 then it turns on the garden lighting by `digitalWrite(LampuTaman, HIGH)`. If the topic is “taman” and the value is 0 then it turns off the garden lighting by `digitalWrite(LampuTaman, LOW)`.

If the topic is “fan\_besar” and the value is 1 then it turns on the fan in kichen by `digitalWrite(fan_besar, HIGH)`. If the topic is “fan\_besar” and the value is 0 then it turns off the fan in kichen by `digitalWrite(fan_besar, LOW)`.

If the topic is “fan\_kecil” and the value is 1 then it turns on the fan in vestibule by `digitalWrite(fan_kecil, HIGH)`. If the topic is “fan\_kecil” and the value is 0 then it turns off the fan in vestibule by `digitalWrite(fan_kecil, LOW)`.

If the topic is “dalam rumah” and the value is 1 then it turns on the vestibule lamps by `digitalWrite(LampuDalam1, HIGH)`. If the topic is

“fan\_kecil” and the value is 0 then it turns off the vestibule lamps by `digitalWrite(LampuDalam1, LOW)`.

When arduino run, arduino will request Ip from roter and will connect to the server and the terminal monitor will display as bellow :

```
192.168.0.177
connected
```

Illustration 5.2.1: Arduino ip connect

Next to open the node-red, and know that node-red is connected with MQTT, the computer terminal will show up as the following

```
karuna@karuna:~$ node-red
25 Jul 00:35:40 - [info]
Welcome to Node-RED
=====
25 Jul 00:35:40 - [info] Node-RED version: v0.18.4
25 Jul 00:35:40 - [info] Node.js version: v4.2.6
25 Jul 00:35:40 - [info] Linux 4.13.0-45-generic x64 LE
25 Jul 00:35:42 - [info] Loading palette nodes
25 Jul 00:35:50 - [info] Dashboard version 2.8.3-beta started at /ui
25 Jul 00:35:53 - [warn]
-----
25 Jul 00:35:53 - [warn] [node-red/rpi-gpio] Info : Ignoring Raspberry Pi specific node
25 Jul 00:35:53 - [warn]
-----
25 Jul 00:35:53 - [info] Settings file : /home/karuna/.node-red/settings.js
25 Jul 00:35:53 - [info] User directory : /home/karuna/.node-red
25 Jul 00:35:53 - [warn] Projects disabled : set editorTheme.projects.enabled=true to enable
25 Jul 00:35:53 - [info] Flows file : /home/karuna/.node-red/flows_karuna.json
25 Jul 00:35:53 - [info] Server now running at http://127.0.0.1:1880/
25 Jul 00:35:53 - [info] Starting flows
25 Jul 00:35:53 - [info] Started flows
25 Jul 00:35:53 - [info] [mqtt-broker:9087a8fc.3950a] Connected to broker: mqtt://192.168.0.101:1883
```

Illustration 5.2.2: Node-red connect mqtt

display above shows that node-red is connected to the server.

## 5.2.2 Node-red Dashboard Testing

First on this test to test whether the flow on the node-red function properly or not.

If node-red is not connected with arduino, then dashboard node-red first display will show as follows:

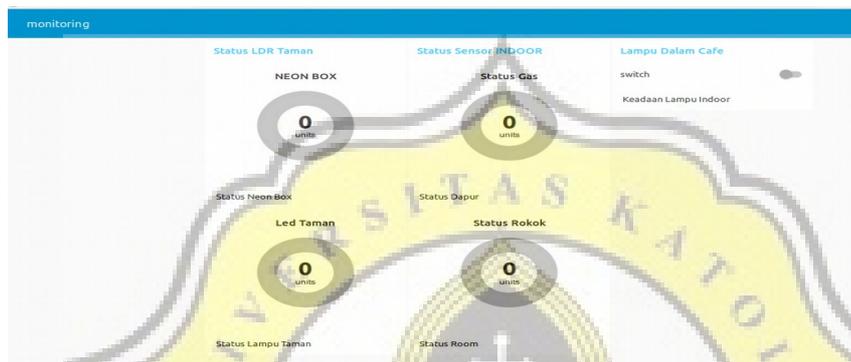


Illustration 5.2.3: Dashboard node-red not connect to server

This indicates that there is a connection failure, where arduino and node-red are not connected.

When all connected correctly dashboard display will show as follows :



Illustration 5.2.4: Dashboard node-red normally

When the display has a dashboard like this, the connection between node-red and arduino has reconnected. The next node-red can control the devices that are enclosed on the arduino via the mqtt server.

In the picture below is a view on the prototype. The prototype is made up of a garden section, a bar room, and a kitchen. Controlled equipment is in the garden garden lights and neon box. In the bar room installed sensors to detect cigarette smoke and also to control the lights of the room blades. In the kitchen in the pairs of sensors to detect lpg gas.

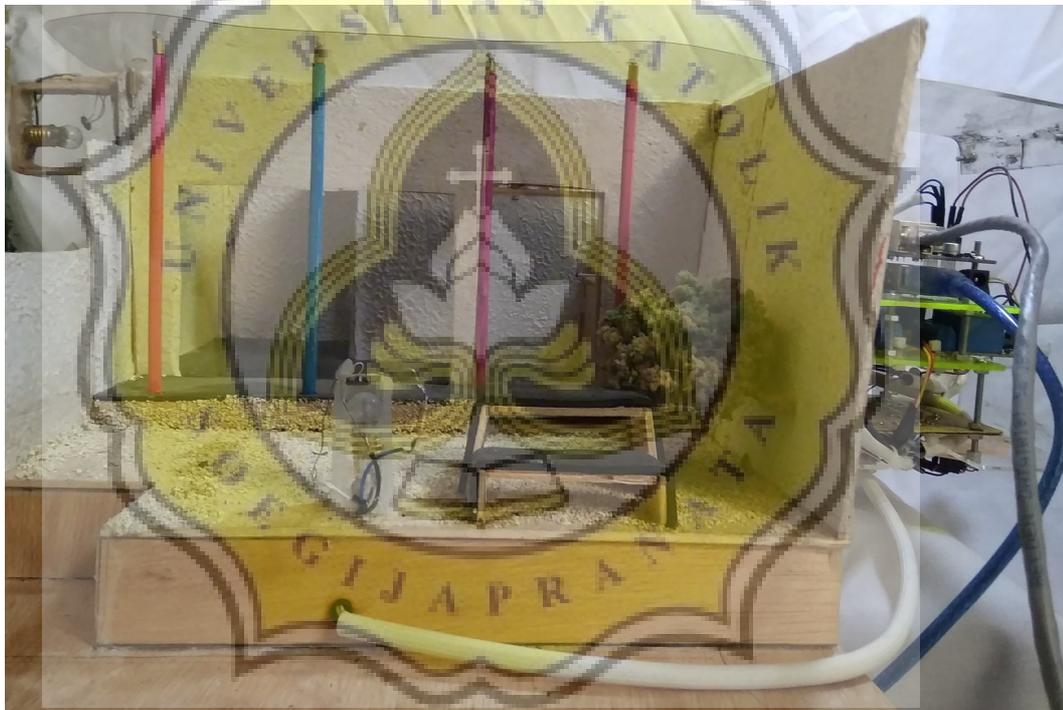


Illustration 5.2.5: Prototype view

if we turn the light on in a cafe and dashboard display will look like as follow.

When we hit the switch on the node-red, the node-red sends a message to mqtt with a value of 1. then arduino do subscribe to mqtt, from mqtt arduino get message 1, then arduino do the action turn on the light on the prototype. Below is the look of the experiment.

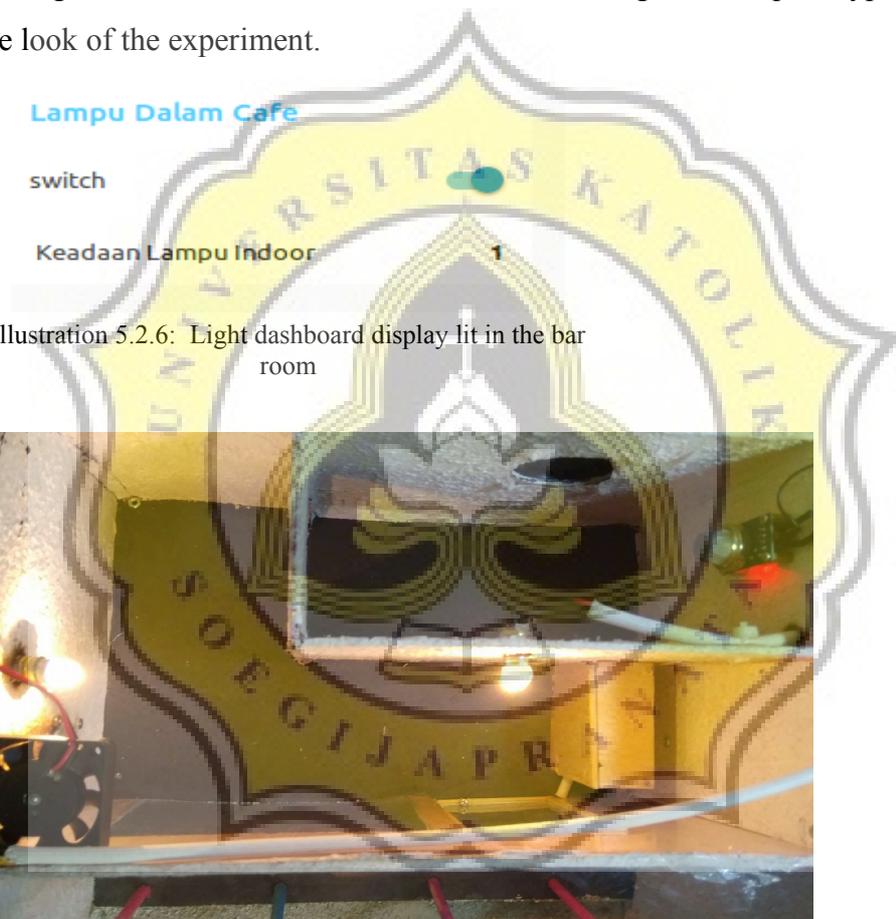


Illustration 5.2.6: Light dashboard display lit in the bar room

Illustration 5.2.7: Light on in a cafe

Next test is turn the neon box light on and garden lamp with the display. In the garden lights and also the neon box, to turn on the light is already set alarm. When the time indicates at 6 pm the park lights and the neon box lights will light up. How to tell if the light is actually on, node-red will display graphics and messages on the dashboard. This graph is obtained from the sensor value ldr, this value is then published to node-red by arduino. Display whether the program is running normally we can see in the picture below.

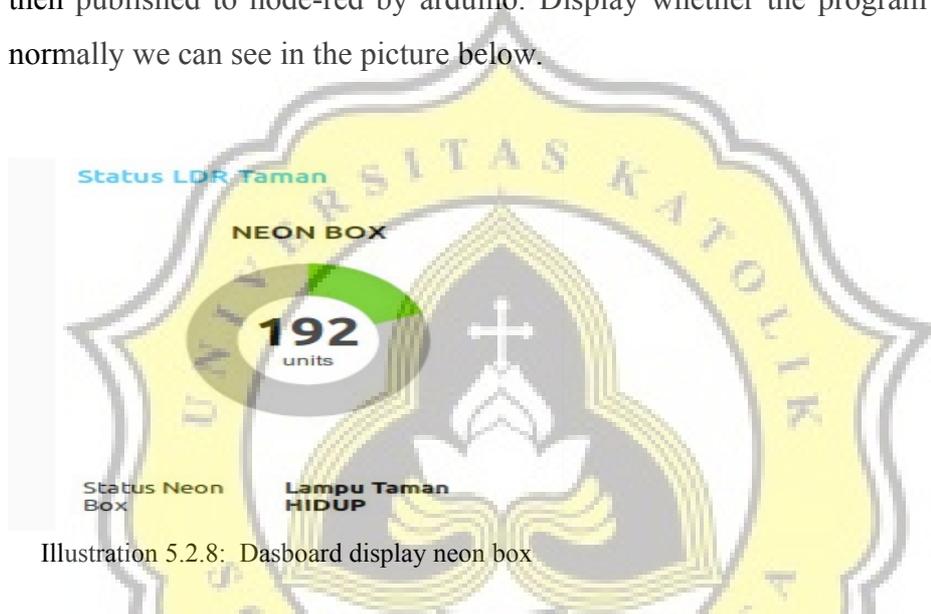


Illustration 5.2.8: Dashboard display neon box



Illustration 5.2.9: Neon box light on



Illustration 5.2.10: Garden lamp on

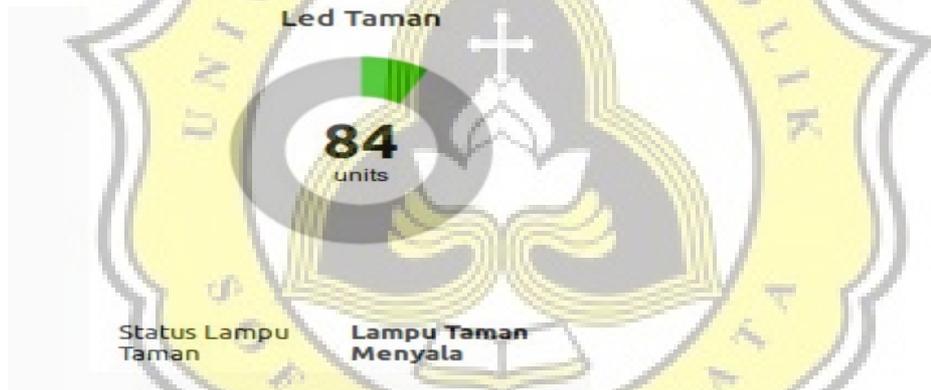


Illustration 5.2.11: This is a display for garden lamp

Last is to test the sensor mq-2 and mq-7. In the test used matches to find out whether sensors are sensitive to methane gas. When the sensor detects a leak of methane gas and the presence of cigarette smoke, arduino will send a value that can be from the sensor and in publish to mqtt. When the node-red subscribes mqtt and gets a large value gas, node-red will display graphics and messages. The displayed message contains a sentence “terdeteksi adanya asap rokok”. The value that can be by node-red in the process. Node-red then publishes back to mqtt. When arduino subscribe mqtt and get the message, the arduino then performs the

action of turning on the fan. Display on node-red and prototype can be seen as picture below.

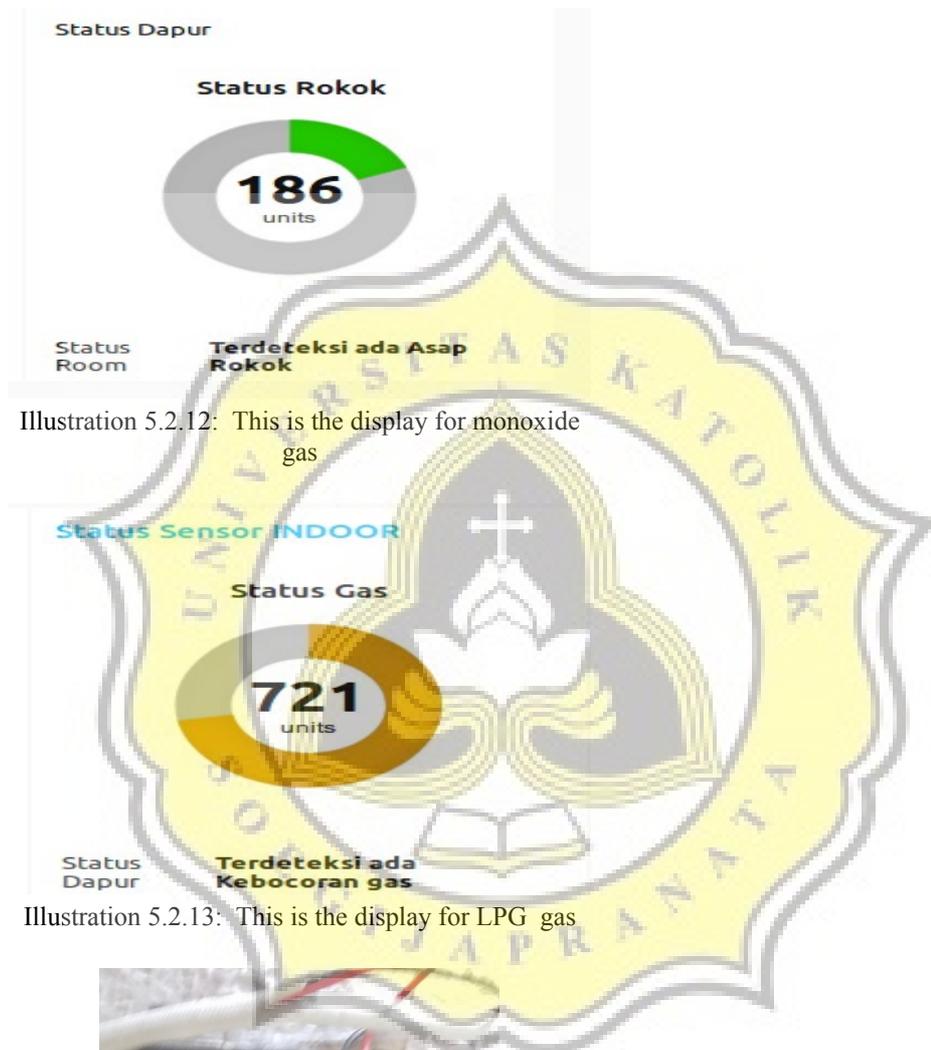


Illustration 5.2.12: This is the display for monoxide gas

Illustration 5.2.13: This is the display for LPG gas

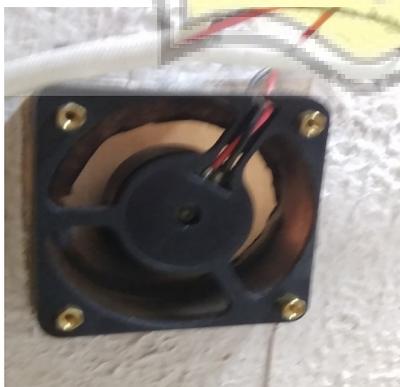


Illustration 5.2.14: This is the fan view