Indoor air quality meter demonstrator v.2

User's guide

GTC INTERNAL CODE:

DEPARTMENT : Sensors for Application and Processes

DOCUMENT NUMBER: v02

AUTHORS: Andrei Golovin

DISTRIBUTION LIST:	NAME	COMPANY (ELECTROLUX UNIT)

	DATE	NAME	POSITION
ISSUED		Andrei Golovin	Activity Leader
CHECKED		Michele Toppano	Line Manager

I. Sensor part

The list of sensors which are used:

Parameter	Sensor	Manufacturer	Description
Temperature	Si7013	Silicon Labs	Capacitive humidity sensor with I2C interface
Relative humidity	Si7013	Silicon Labs	
PM2.5/10	SDS021	Nova Fitness	Optical laser diode-based sensor with UART interface
VOCs	MiCS-5524 *1	SGX Sensortech	Analog MOX-based resistive sensor
CO2	Cozir Ambient	Gas sensing solutions	NDIR-based sensor with UART interface
Ambient light	TSL2561T *2	AMS	Light sensor with I2C interface
Noise	SPM0404HE5H-T *3	Knowles	MEMS microphone with electronics developed by GTC

Comments:

1. Sensor signal isn't calibrated. The readings in arbitrary units are proportional to the total level of VOCs.
2. Sensor signal isn't calibrated. The readings in arbitrary units are proportional to the lighting level on the sensor sensitive surface.
3. Sensor signal isn't calibrated. The readings in arbitrary units are proportional to an average noise level in space around the microphone.

II. Short description of the demonstrator

The iAQ meter demo combines hardware and firmware parts. Hardware is formed by a set of sensors and electronics to maintain power lines, interfaces and signal conditioning.

Firmware part running on the arduino Atmega MCU requests the readings from each sensor by turn. Each measurement cycle takes approx. 5 seconds. Data is collected during 12 cycles (appr. 1 minute). Then the readings are averaged for the following sensors: CO2, PM, VOCs, noise and light. The current cycle readings are taken for the temperature and humidity sensor. Those values are joined to one string.

Firmware part running on the arduino Atheros MCU makes cURL POST request over HTTP protocol with that string as content of the request. The target server for the POST request is a MS Azure IoT hub.

The iAQ meter's WiFi module works in wireless client mode. For successful connection with the server the WiFi module should be configured to connect to existing WLAN.

The structure of the string:

V3.25_T 28.7_H42.3_CO21200_PM255.2_PM106.3_L27_N0.12_F0

Where:

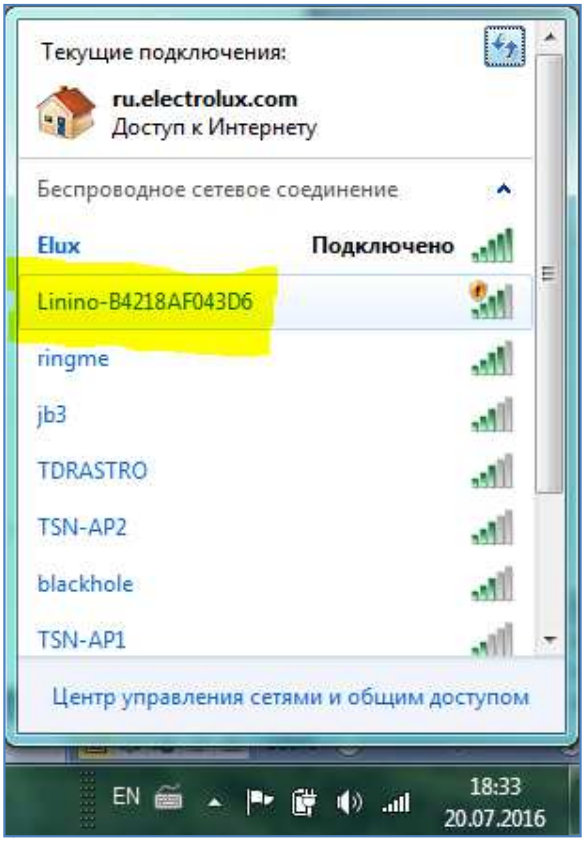
- VX.XX – data in arbitrary units from the VOCs sensor ranged from 1 to 5 a.u.
- TXX.X – data from the temperature sensor in °C
- HXX.X – data from the humidity sensor in %RH
- CO2XXXX – data from the carbon dioxide sensor in ppm
- PM25X.X – data about PM2.5 from the particulate matters sensor in $\mu g/m^3$
- PM10X.X – data about PM10 from the particulate matters sensor in $\mu g/m^3$
- LXX – data from the ambient light sensor in a.u. ranged from 0 to 5047
- NX.XX – data from the noise sensor in a.u. ranged from 0 to 2
- FX – the last symbol of the string is the fan state. 1 means the fan is working, 0 – powered off

Note: The format of the string has been changed compared to the previous version of the iAQ meter. There is no space symbol in the string in order to ensure correct work of the POST request.

The demonstrator is equipped with a small fan. By default the fan is turned off. The fan can be turned on by using a software tool.

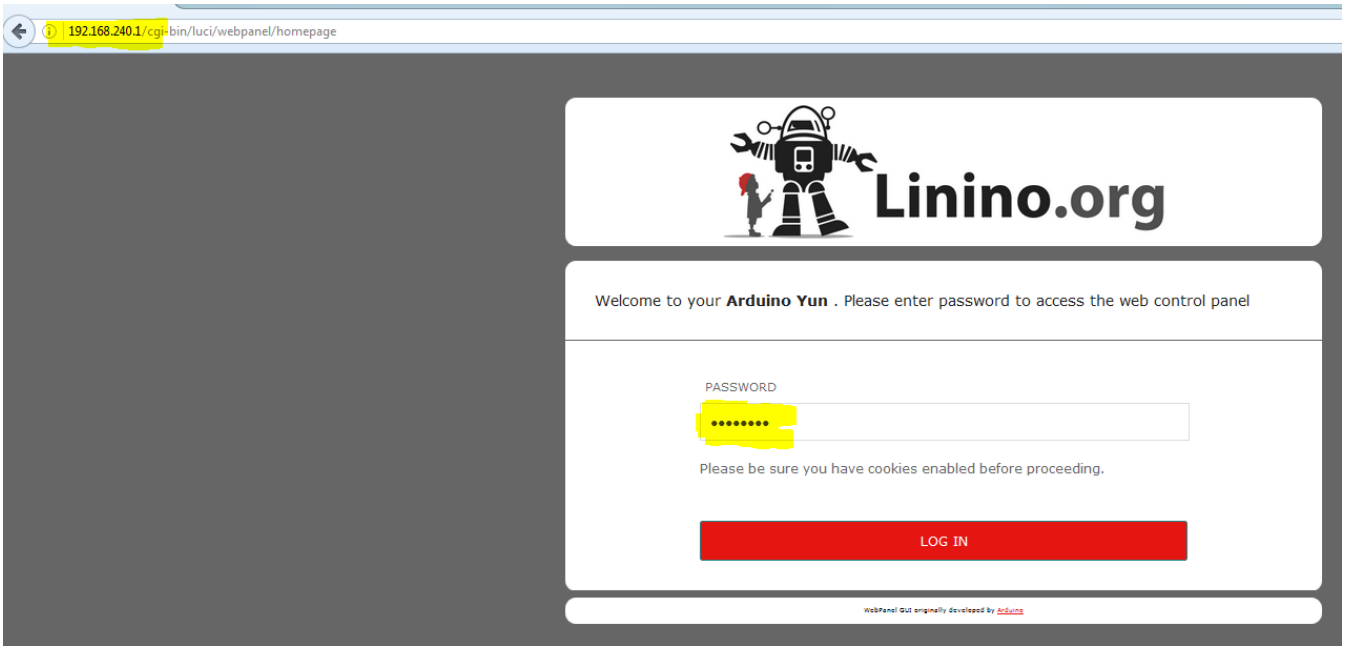
III. Getting started

1. Connect the iAQ meter to power
2. Wait for the Arduino board starts WLAN. It may take a minute. Then in the list of available connections you will see a new wireless connection with SSID like Linino-XXXXXXXXX. Connect your PC to that WLAN. There is no password protection.

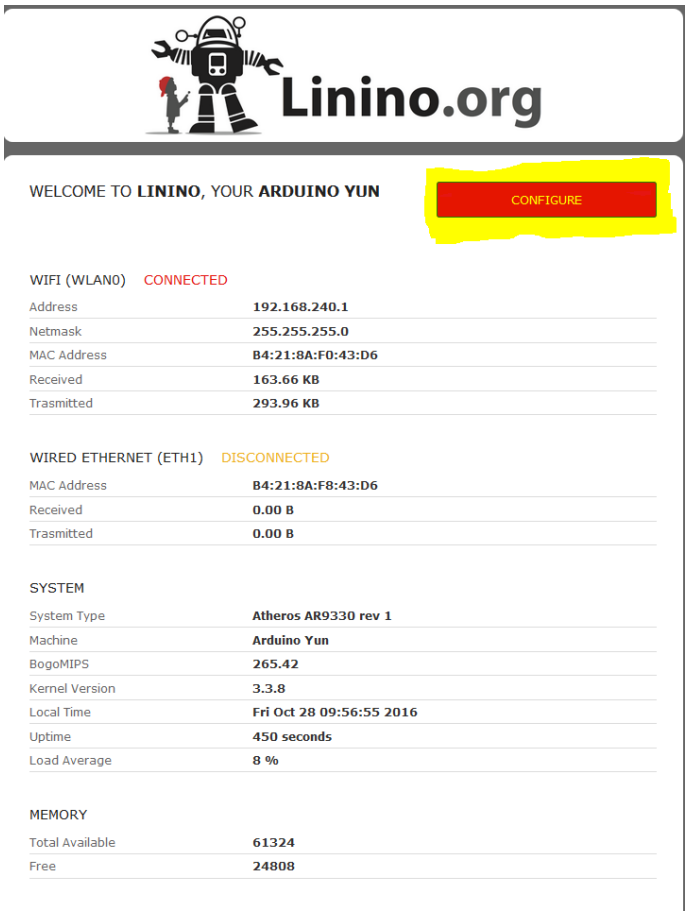


3. Go to Arduino's OpenWRT web panel by using browser with the address 192.168.240.1

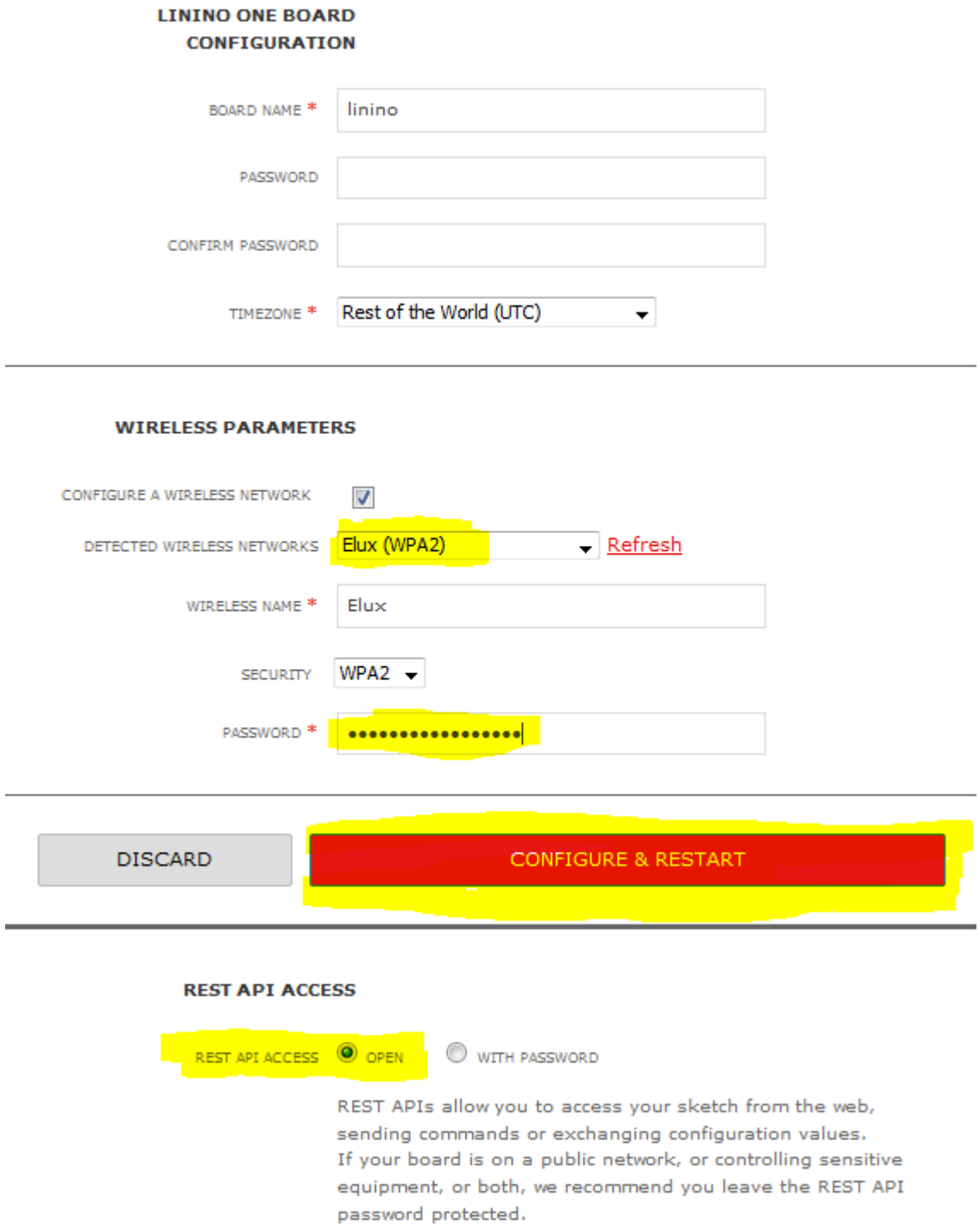
   Type the password and click **log in**. The password: **arduino**



4. Click **Configure** in order to set WiFi connection to the existing WLAN



5. On the next page choose your WiFi network from the list of detected WLANs and type password for that network. Check that REST API access is set as Open and click **Configure&Restart**.

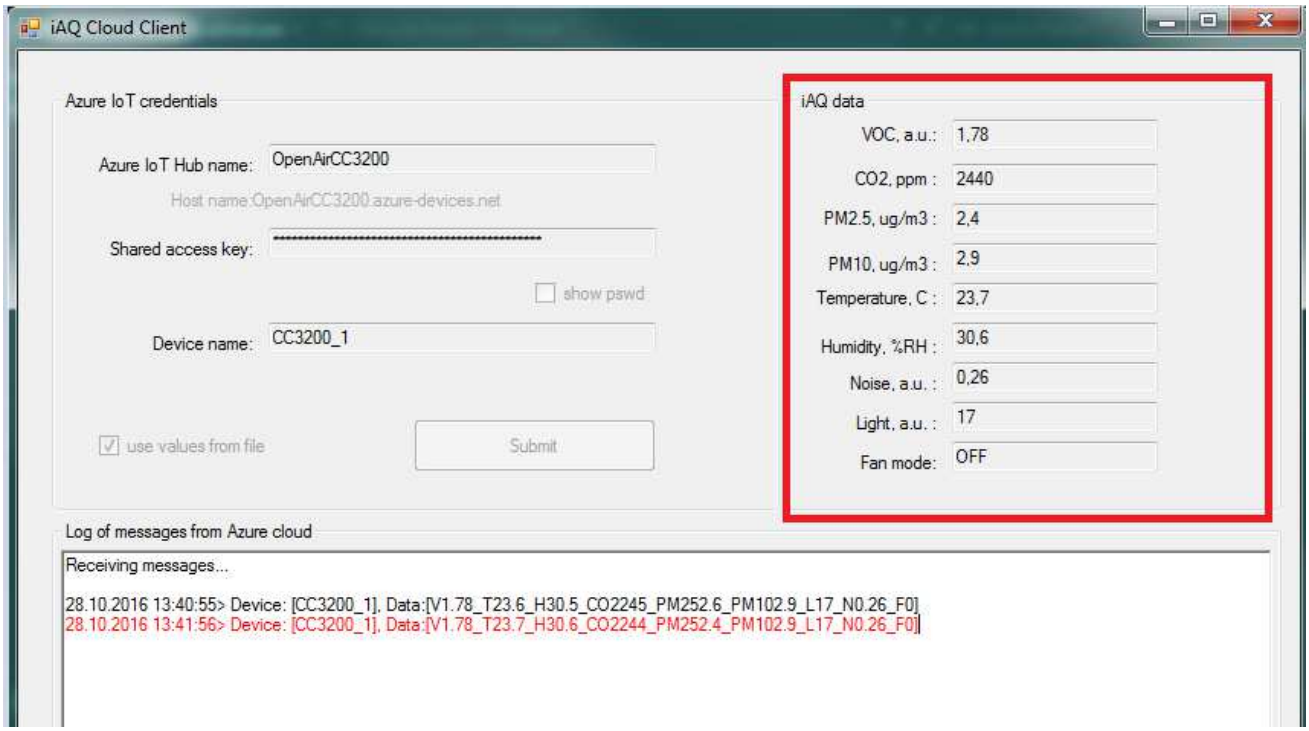


6. Wait for completing of configuration.

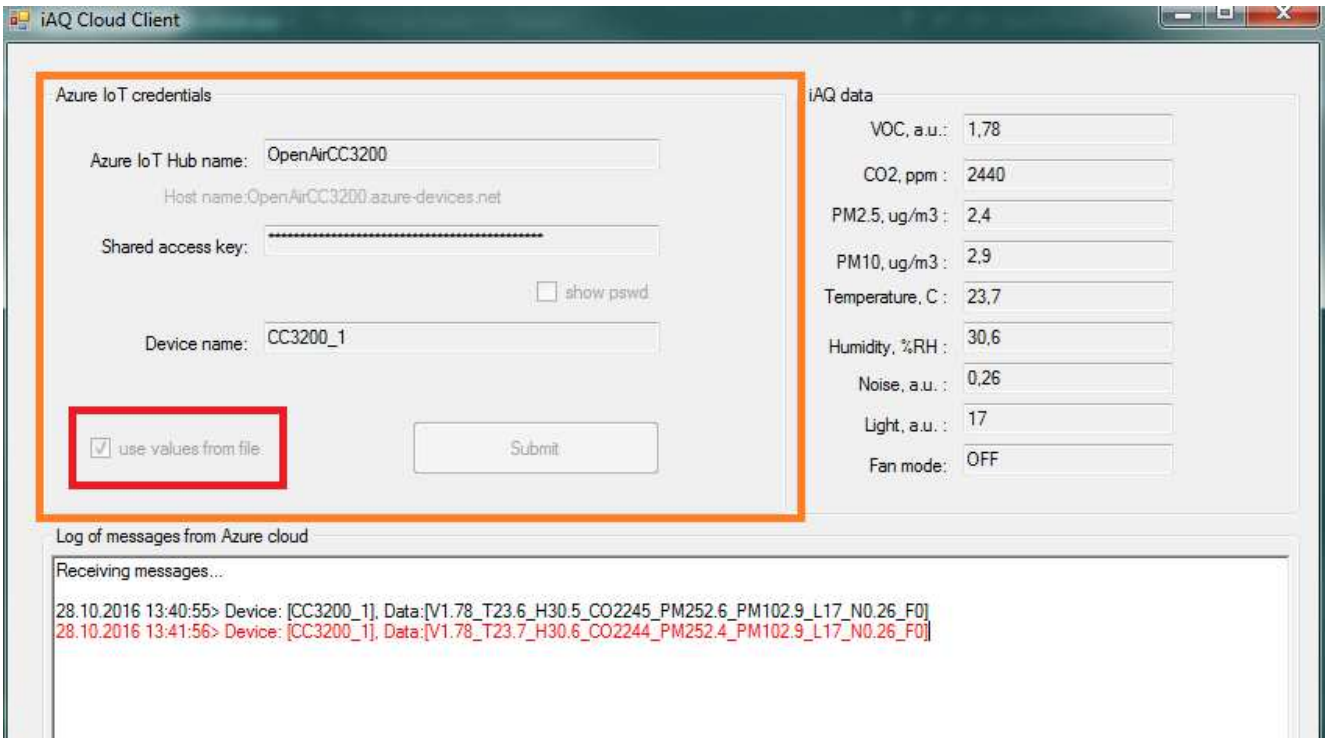
7. **Disconnect the iAQ meter from power and reconnect it again.** After a couple of minutes the iAQ meter will start sending messages to the IoT hub called OpenAirCC3200.

8. Device Explorer could be used to monitor messages sent by the iAQ meter. The device ID is CC3200_1.

9. Another option is an app **iAQ Cloud Client** developed by GTC. The app is based on MS Azure SDK as well as Device Explorer but it can also display sensor readings by splitting the incoming string. The 4.6 Microsoft.NET framework is needed.
   Note that Electrolux corporate firewall settings may block connection between MS Azure cloud and app (for both Device Explorer and iAQ Cloud Client).



10. In case of using of iAQ Cloud Client app copy the app folder to your PC and run .exe file. The app may be used with any Azure IoT hub by typing manually the hub credentials. For the OpenAirCC3200 hub the settings are stored in the service file. Choose "use values from file" option and click Submit button. The app will start displaying the readings.

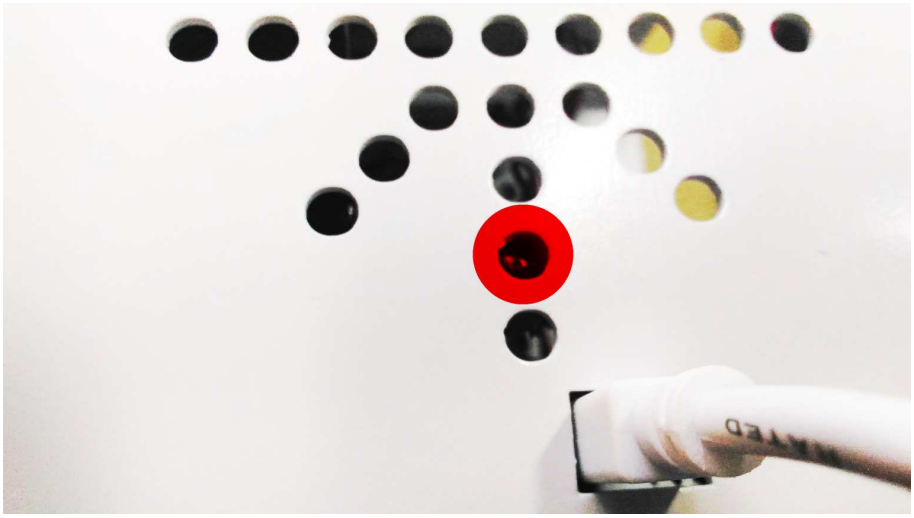


IV. WiFi network changing

In case of need to change WiFi network follow the steps:

- Connect the demo to power and wait for 2-3 min while OpenWRT booting
- Hold the reset button (located as in the picture below) for longer than 5 sec but less than 30 sec. The WiFi module will be switched to the access point mode.
- In a few minutes the WLAN Linino-XXXXXXXXX will appear in the list of available networks on PC
- Follow the steps 2-7 of the section "Getting started" of this document and select a new WLAN

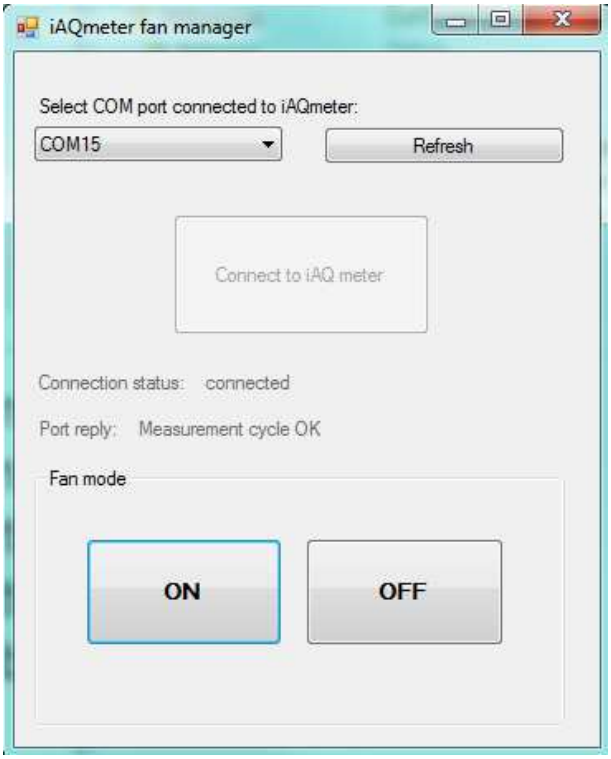
The reset button location:



V. Fan mode switching

The iAQ meter has a small fan inside. By default the fan is turned off. The mode can be switched by using a tool iAQmeterFanManager. To switch the fan mode:

- Connect the iAQ meter to PC via USB cable
- Run iAQmeterFanManager.exe
- Choose COM port that the iAQ connected
- Click Connect button
- Switch the mode by clicking a desired button

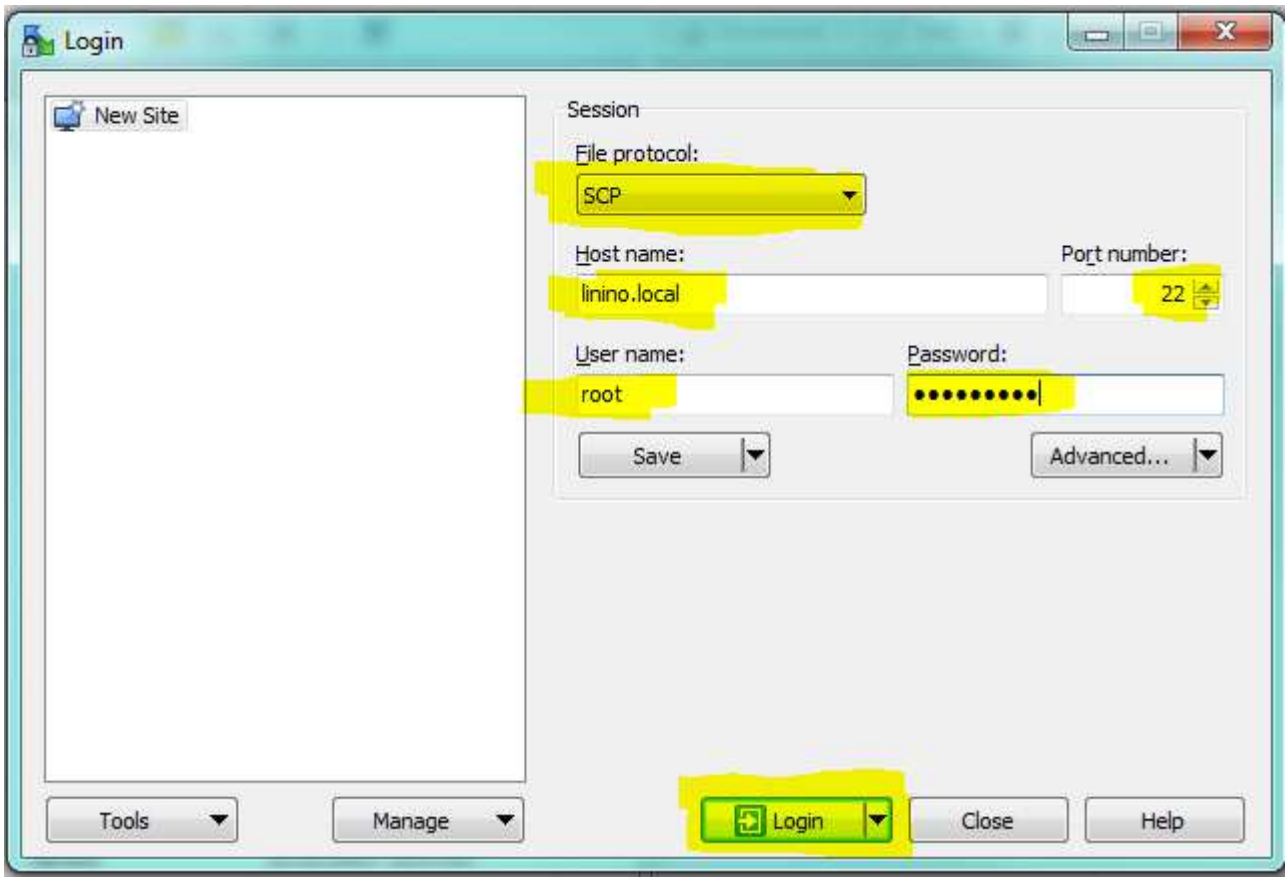


Note that info about fan mode is stored in non-volatile memory. That means after powering on the iAQ meter the fan will be in the same mode as it was before turning off.

VI. Switching to a new Azure IoT hub

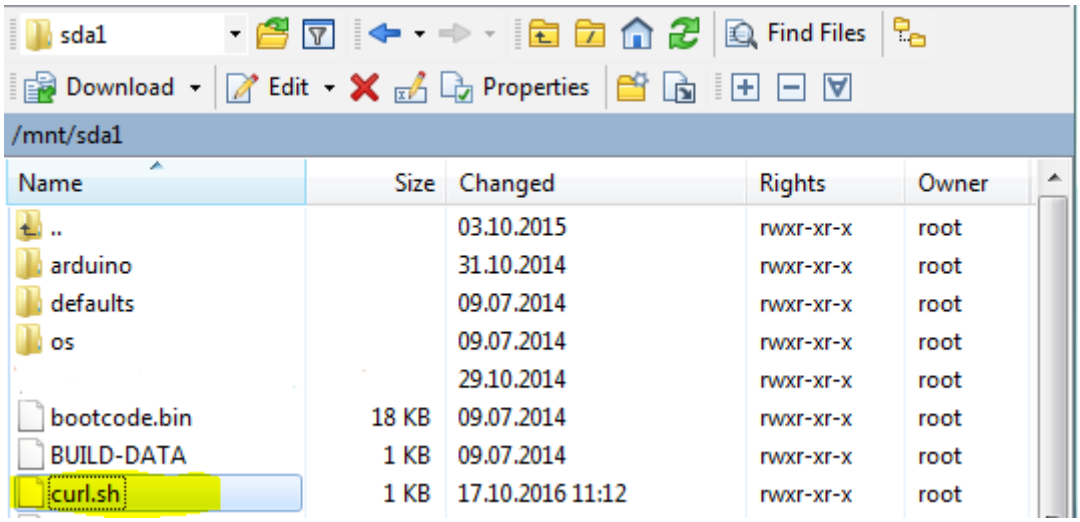
In case of need to send data from the iAQ meter to another Azure IoT hub the following should be performed. Data is sent to the hub by using cURL on the Arduino's Linux side. The cURL script is stored on SD card of the Arduino board. In order to change the IoT hub the script should be modified. Since physical access to SD card is limited inside the box a software tool can be used to replace the file on SD. WinSCP tool could be recommended: https://winscp.net/

To get access to SD card run WinSCP, choose SCP protocol, type linino.local as host name, root as user name, arduino as password and click login:



Note the iAQ meter has to be connecting to the same wlan as PC.

Go to /mnt/sda1/ directory and replace the curl.sh file with the new one that should contain settings for the new IoT hub.



The structure of curl.sh script:

```
#!/bin/ash
/usr/bin/curl -k -X POST -H 'Content-Type: application/json' -H "Content-Length: $1" https://[host name] -H 'Authorization: SharedAccessSignature sr=[SAS token]' -w "%{http_code}" -d "$2"
```

In order to change the IoT hub two parameters of the script should be modified: host name and SAS token.

An example of host name structure:

```
https://OpenAirCC3200.azure-devices.net/devices/CC3200_1/messages/events?api-version=2016-02-03
```

Here the hub name **OpenAirCC3200** should be replaced with a new one and Device ID **CC3200_1** should be replaced with a new device ID in case of need.

An example of SAS token structure:

```
-H 'Authorization: SharedAccessSignature sr=OpenAirCC3200.azure-devices.net&sig=uUOOu0wXirYEa5UoQT%2F1nVgsmYGXuY7jt5WhBTOizd8%3D&se=1509015994&skn=iothubowner'
```

Here the shared access signature should be replaced with a new one.

VII. Appendix

iAQ meter schematics