LSM6DSO-ESP8266 IMU

data collection prototype

User manual[1]

1 To avoid the risk of accidents or damage to the device, it is essential to read these instructions before it is used for the first time.

Table of Contents

1. Introduction

This device is used to collect data from the LSM6DSO sensor that is built-in. It sends the data through a MQTT broker via Wemos D1 Mini that has a ESP8266 module. The sensor and the module can be programmed by using the base code [1]. These steps will be showed in this manual, so it is important to read these instructions before the device is used to collect data.

Data cannot be collected without the proper configuration of the device!

The device can be opened and has replacebale parts. The parts must be the same parts as the device came with, otherwise the software will be incompatible. The parts are Wemos D1 Mini [2] and STEVAL-MKI196V1 [3]. The device is powered on by a 9V replaceable battery. A voltage regulator drops the input voltage to the required operating voltage of 5V.

The device has arrows indicating the sensors axes. The sensor is LSM6DSO[5], giving 6-degrees of freedom. For more details about the sensor check the datasheet[6].

Note: There can be small delays in the incoming data. This means that the timestamps between readings might not be consistent when these delays occur!

The data collected form this device is raw data collected using the recommended settings from the manufacturer for best results. It returns acceleration in milligravities (mg) and millidegrees per second (mdps). The output rate is one sample each 15ms, giving around 66,6 samples per second with the default configuration specified above.

Table 1: Device technical specs

Operating Voltage	3.3V to 5V
Clock Speed	80/160MHz
Flash	4MBytes
Size	70*45*60mm
Temperature range [°C]	-40 to +85

2. Device overview

Figure 1: Figure 1: Back view

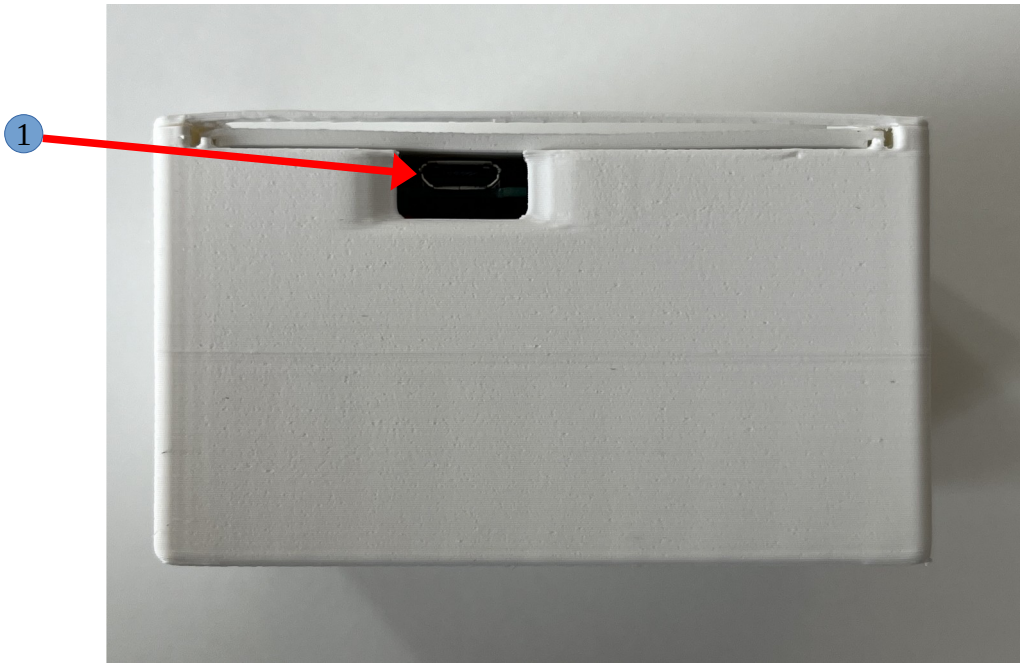


Figure 2: Figure 2: Front view

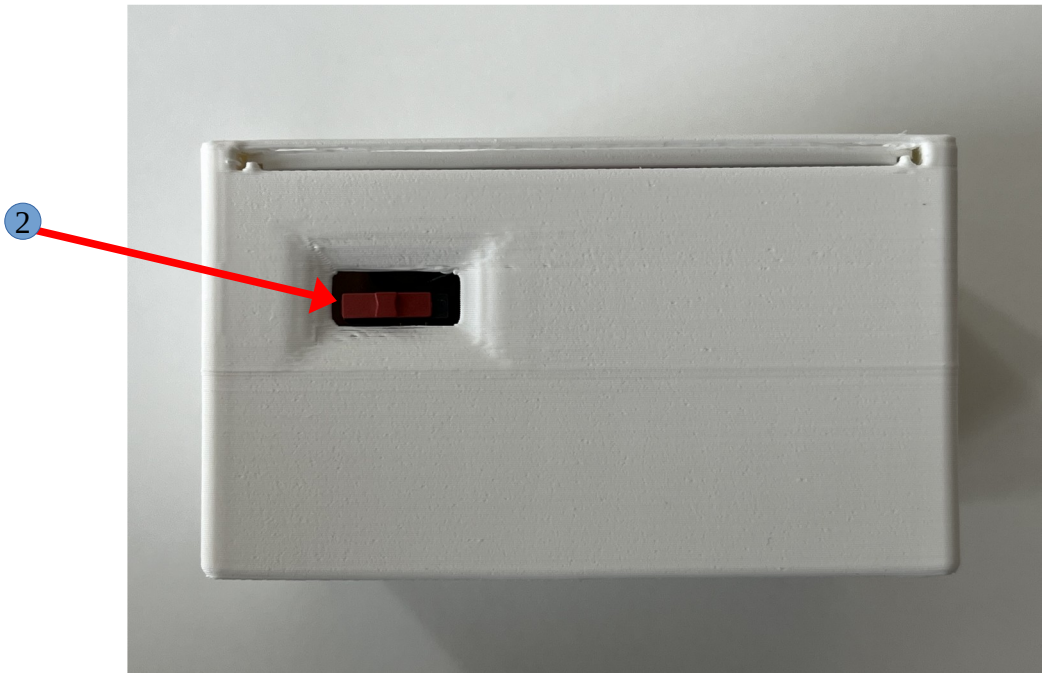
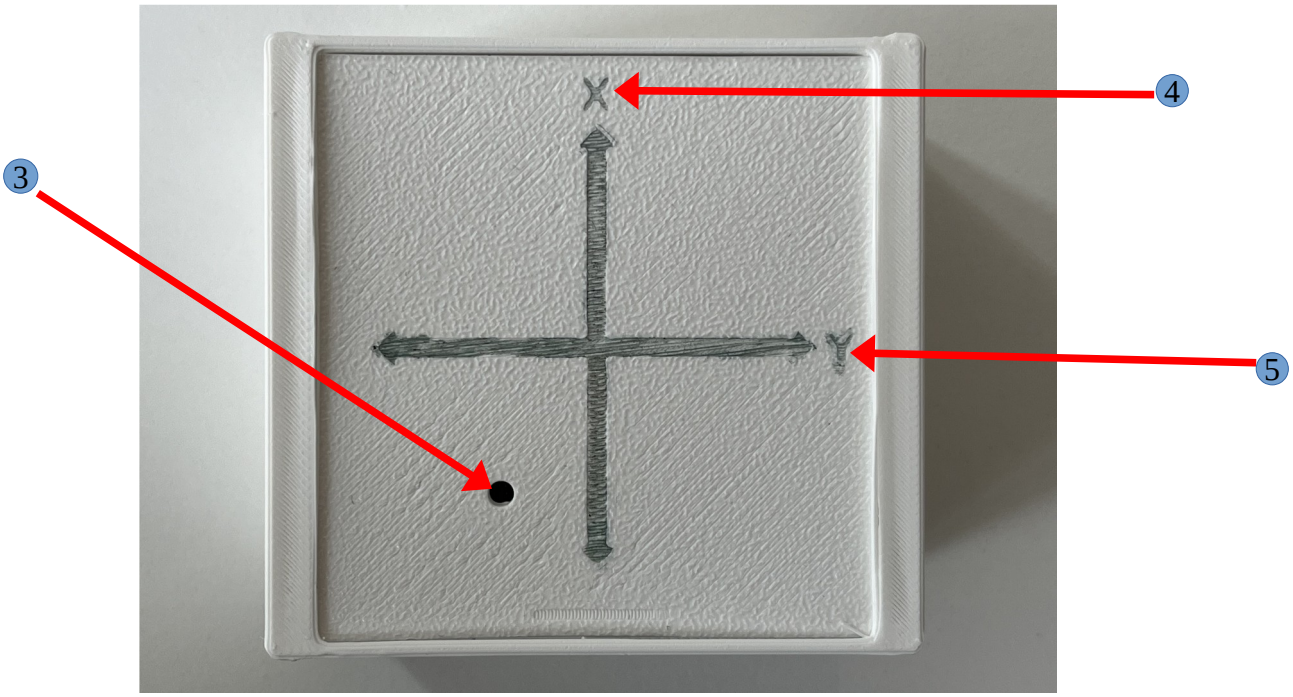


Figure 3: Figure 3: Top view



1. Micro USB
2. On/Off Switch
3. Status LED
4. X-axis – it points the sensors X-axis
5. Y-axis – it points the sensors Y-axis

3. Device configuration

To start using this device do the following configuration.

Clone the git repository[1] and configure it with the following:

```
const char *ssid = "Enter your SSID";
const char *pwd = "Enter your SSID password";

const char* mqtt_server = "Enter the ip address of the device the MQTT broker is running on";

const int mqtt_port = 1883;
const char* topicOut = "Enter chosen topic for outgoing data";
const char* topicIn = "Enter chosen topic for incoming data";
const char* mqttUser = "Enter the chosen MQTT user";
const char* mqttPass = "Enter the chosen MQTT password";
```

To collect the data it is required a MQTT broker that runs on another device. The broker can be any MQTT broker. Use the details from the broker[4] configuration while configuring the base code above.

NOTE: **The MQTT broker and the device must be connected to the same network!**

Flash the device by connecting a micro USB to the connetion port. Upload the code with VSCode and PlatformIO or any other suitable IDE.

NOTE: **The device must be turned OFF from the switch before uploading the code!**

4. Device Start

After the device is succesfully configured, turn the switch ON.

Wait for the LED to turn blue. This indicates that the device successfully connected to the network and to the MQTT broker. If the LED is OFF, there must be an issue with either the network connection or the broker. Check again that the configuration is corect.

5. Data collection

Once the LED is ON the device is ready for collecting data. To start the data collection, send 1 to the device through the topicIn. To stop the data collection send 0 to the device through the topicIn.

Data consists of timestamps that are in epoch format, acceleration (ax, ay, az), and gyroscope (gx, gy, gz).

6. Troubleshooting

1. Data errors

In case of errors in the incoming data, turn the device OFF and ON.

2. LED Off

If the device is powered ON but the LED is OFF, this indicates a problem in either connecting to the network or to the MQTT broker. Check again the configuration for both network connection and broker configuration.

3. Battery replacement

To replace the battery, open the battery cover and take out the old battery. Connect the new battery to the connector and place the battery back into the designated place. Close the cover and turn ON the device.

4. Replacing components

To replace a component, open the casing of the device. Gently remove the component (Note: Only the Wemos D1 Mini and the STEVAL can be replaced!) and gently place the new component. Be sure that the component is placed correctly.

To place the STEVAL correctly, the VDD Pin must be towards the switch.

To place the Wemos correctly, the USB must be towards the edge of the case.

Note: **Missplacing the components can burn the components!**

5. Failed flashing the device

If the flash fails, check that the device is turned OFF from the switch and try again.

Check if there are any compiling errors and address them before flashing again.

References

[1] https://github.com/GbrlBln/LSM6-ESP--Git-.git

[2] https://www.wemos.cc/en/latest/d1/d1_mini.html

[3] https://www.st.com/en/evaluation-tools/steval-mki196v1.html

[4] https://mosquitto.org/

[5] https://www.st.com/en/mems-and-sensors/lsm6dso.html

[6] https://www.st.com/resource/en/datasheet/lsm6dso.pdf