

# Getting Started With OpenSmartMonitor

---

## Setup

---

To clone the repository, execute the following command in your desired directory.

```
git clone --recursive git.devtank.co.uk:/git/osm\_firmware.git
git pull
git submodule init
git submodule update
```

There are several dependencies required for OSM to compile successfully, install these on your machine with the following command:

```
sudo apt install
  build-essential \
  git \
  pkg-config \
  libjson-c-dev \
  picolibc-arm-none-eabi \
  stm32flash \
  valgrind \
  minicom \
  idle-python3.10 \
  python3-influxdb \
  python3-pil \
  python3-pymodbus \
  python3-scipy \
  python3-yaml \
  nodejs
```

Note that if you are running Debian Stable, you may need to install picolibc from source as the version installed on the Debian package manager is too old. You will need version 1.7.4-1 at least.

## Build

---

In order to build OSM, run the command:

```
make -j8
```

You can also run *make* but the above command will compile much faster.

## Test

---

Now that `osm_firmware` has compiled, we can begin running tests and communicating with the fake osm. To run a test for the virtual OSM, enter the following command in the top level directory.

```
make penguin_test
```

This will spawn the virtual OSM test, connect to the virtual OSM and test values for each measurement, ensure you wait for the measurement loop to finish.

## Run Virtual OSM

---

To spawn the Virtual OSM, use the following command:

```
./build/penguin/firmware.elf
```

Once this is running, you can use `minicom` to open up communications with the fake sensor. The device that you will want to supply to `minicom` is created when starting the Virtual OSM and is called `UART_DEBUG_slave`.

```
minicom -b 115200 -D /tmp/osm/UART_DEBUG_slave
```

You can now communicate with the Virtual OSM through serial.

## Config GUI

---

To use the Configuration GUI, you will to connect your OSM to your machine. If you don't have an OSM, use the virtual OSM.

To start the GUI, run the following commands:

```
cd osm_firmware/tools/config_gui  
./config_gui.py
```

Select your device from the dropdown menu and press connect.