

# Installation of dev-Environment for Sensormodule

---

Download and install git from <https://git-scm.com/downloads>

Download and install visual studio code from <https://code.visualstudio.com/download>

Start visual studio code

Go to extensions (Ctrl-Shift-X)

Enter "platformio" in search field

Install "PlatformIO IDE" extension

Wait until installation is finished, do the necessary reload window afterwards (may take some time)

Click on the new PlatformIO-Icon on the left



In "Quick Access", choose open

In the new "PIO Home" tab, click on "New Project..."

In the upcoming dialog, provide the name "Test", Board "Sparkfun SAMD21 Dev Breakout", Framework "Arduino" and Location "Use default location"

Click "Finish" and wait until finished. Visuals Studio Code will open the newly created project afterwards.

On the left side (Activity Bar) there is a PlatformIO Icon, click on it

Again "Quick Access" appears, click "Miscellaneous->PlatformIO Core CLI"

A new terminal (within Visual Studio Code) appears, the path is home of the new test project. We don't need the test project, it was just used to create all necessary path for development. From now on we work in this terminal window:

```
cd ..
```

You should be now in a directory ending with ...\\Documents\\PlatformIO\\Projects

```
pio lib -g install 805
pio lib -g install 166
pio lib -g install 31
pio lib -g install 5449@1.0.3
```

These commands should install following libraries:

"ClosedCube\_HDC1080" Library

"Adafruit\_BME280" Library

"Adafruit\_Sensor.h" Library

"SparkFun\_SCD30\_Arduino\_Library" Library

```
git clone https://github.com/mumpf/knx.git
git clone https://github.com/mumpf/knx-common.git
git clone https://github.com/mumpf/knx-logic.git
git clone https://github.com/mumpf/knx-sensor.git
cd knx-sensor
code Sensormodul.code-workspace
```

Go to SENSORMODUL (WORKSPACE), open the "src" folder and click on Sensormodul.cpp file

Press Ctrl-Shift-P, enter "platformio build" and select the appearing Task "PlatformIO: Build"

Now the compiler starts, this may take a while, there will be many yellow warnings, they can be ignored.

At the end, there should be a message like

```
Checking size .pio\build\build\firmware.elf
Advanced Memory Usage is available via "PlatformIO Home > Project Inspect"
RAM:   [==          ]  21.7% (used 7096 bytes from 32768 bytes)
Flash: [=====     ]  61.8% (used 161904 bytes from 262144 bytes)
===== [SUCCESS] Took 62.08 seconds =====
```

Now you successfully build the Firmware for the Sensormodule, containing up to 50 logic blocks.

## How to upload the Firmware to your Hardware:

Connect your device via USB to your PC

Open (again) the file Sensormodul/src/Sensormodul.cpp

Press Ctrl-Shift-P, enter platformio upload and click on the appearing "PlatformIO: Upload"

Wait until file is uploaded.

## How to build a knxprod for this firmware:

Open <https://github.com/mumpf/multiply-channels/releases>

Download the newest release of multiply-channels, currently it is the fourth beta

The executable is MultiplyChannels.exe

Ideally you save it to C:\Users\<username>\bin (usually you have to create bin directory)

If this is not your ETS-PC, install ETS5 on this PC (ETS5.7.x demo is sufficient, even any 5.6.x should do)

Go to the Visual Studio Code instance, which is containing the knx-sensor project

Open file Sensormodul/src/Sensormodul.cpp

Press Ctrl-Shift-P, enter "run test task" and click the appearing "Tasks: Run Test Task"

In the following dropdown select "MultiplyChannels knx-sensor"

Wait for the success message in the terminal window

The freshly build Sensormodul.knxprod you will find in the src directory

You can import this knxprod in your ETS (minimum 5.6) like any other knxprod.