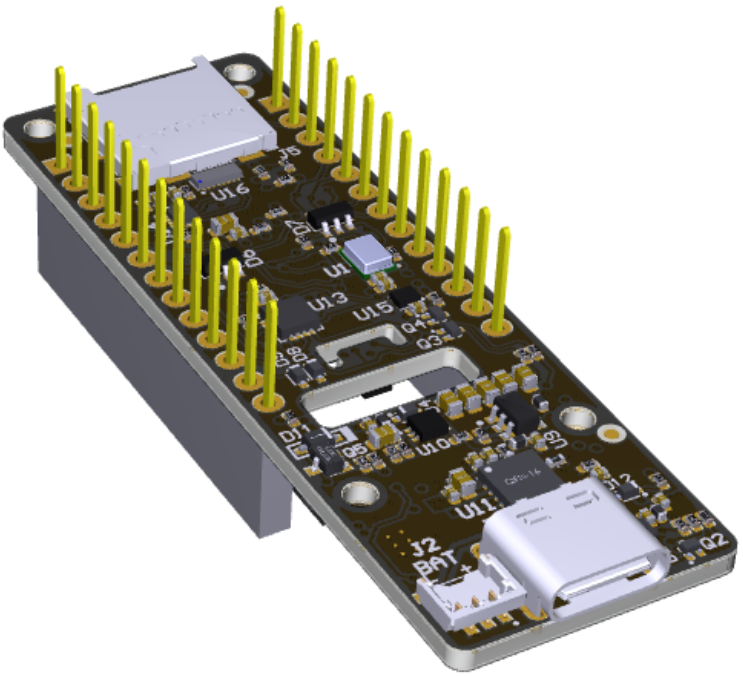
SensiEDGE
Simplify & Accelerate

CommonSense

Sony's Spresense Sensor Board

Getting Started

by SensiEDGE and Edge Impulse
V 1.0
Date Release 4.1.23



List of Content

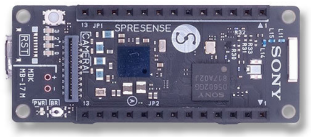
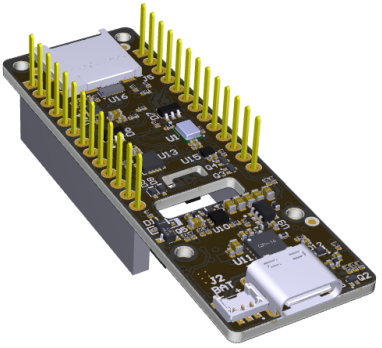
1 Overview

This document is intended to help get started with Sony's Spresense Sensor Board. In this document can be found a list of materials, software and how to install.

2 List of materials and software

The project consists of software (SW) and hardware (HW).

Before starting, be sure to have on hand all 6 following items:

1. Sony's Spresense Sensor Board 	2. CommonSense Board 	3. USB - micro USB cable 
4. PC x64 with Ubuntu 20.04.2 LTS 	5. CommonSense BSP 	6. Installing tools 

3 Getting, Installing, and Running SW

To program the main MCU with Sony's Spresense Sensor Board SW package, necessary to follow 3 next steps:

3.1 Get a project.
3.2 Installing tools.
3.3 Run the project.

After successful completion of the steps, as result is programming of Sony's MCU, it is possible to move to the stage of changing the project for any needs.

3.1 Get a project - Sony's Spresense Sensor Board SW

The open source code can be downloaded from SensiEDGE CommonSense Github. The Git command is present in 3.2.1 below.
For more questions about CommonSense Sony's Spresense Sensor Board SW can be found at www.sensiedge.com/commonsense.

3.2 Installing tooling

To install tools please follow the following steps in 3.2.1 below. This steps will install some tools on the PC like:

1. Install Python 3 on your host computer.
2. Install Node.js v14 or higher on your host computer.
   Alternatively, run the following commands:
   curl -sL https://deb.nodesource.com/setup_14.x | sudo -E bash -
   sudo apt-get install -y nodejs
   node -v
   The last command should return the node version, v14 or above.
   Let's verify the node installation directory:
   npm config get prefix
   If it returns */usr/local/*, run the following commands to change npm's default directory:
   mkdir ~/.npm-global
   npm config set prefix '~/.npm-global'
   echo 'export PATH=~/.npm-global/bin:$PATH' >> ~/.profile

3. Install the CLI tools via:
   npm install -g edge-impulse-cli

   Installation process.

3.2.1 Linux console command sequence

Cloning Git project:

```
git clone https://github.com/SensiEDGE/CommonSense
```

GCC ARM Install:

```
sudo tar xjf
gcc-arm-none-eabi-9-2019-q4-major-x86_64-linux.tar.bz2 -C
/usr/share/

sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-g
cc /usr/bin/arm-none-eabi-gcc
sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-g
++ /usr/bin/arm-none-eabi-g++
sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-g
db /usr/bin/arm-none-eabi-gdb
sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-s
ize /usr/bin/arm-none-eabi-size
sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-o
bjcopy /usr/bin/arm-none-eabi-objcopy
sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-a
r /usr/bin/arm-none-eabi-ar
sudo ln -s
/usr/share/gcc-arm-none-eabi-9-2019-q4-major/bin/arm-none-eabi-l
d /usr/bin/arm-none-eabi-ld

sudo apt install libncurses-dev
sudo ln -s /usr/lib/x86_64-linux-gnu/libncurses.so.6
/usr/lib/x86_64-linux-gnu/libncurses.so.5
sudo ln -s /usr/lib/x86_64-linux-gnu/libtinfo.so.6
/usr/lib/x86_64-linux-gnu/libtinfo.so.5
```

Run `sudo apt-get install -y nodejs` to install Node.js 14.x and npm
You may also need development tools to build native addons:

```
  sudo apt-get install gcc g++ make
```

To install the Yarn package manager, run:

```
  curl -sL https://dl.yarnpkg.com/debian/pubkey.gpg | gpg
--dearmor | sudo tee /usr/share/keyrings/yarnkey.gpg >/dev/null
  echo "deb [signed-by=/usr/share/keyrings/yarnkey.gpg]
https://dl.yarnpkg.com/debian stable main" | sudo tee
/etc/apt/sources.list.d/yarn.list
     sudo apt-get update && sudo apt-get install yarn
```

VNC:

```
sudo apt-get install dconf-editor
```

Open dconf-editor >>org>>gnome>>desktop>>remote session >> require-encryption

In dconf-editor we need to disable encryption

Python:

```
pip install pyserial
pip install pyinstaller
```

COM, viewer:
if you have some issues with port connection you can try: `sudo chmod -R 777 /dev/ttyUSB0`

```
sudo apt install minicom
minicom -D /dev/ttyUSB0
```

Also may be used CuteCom terminal or other.

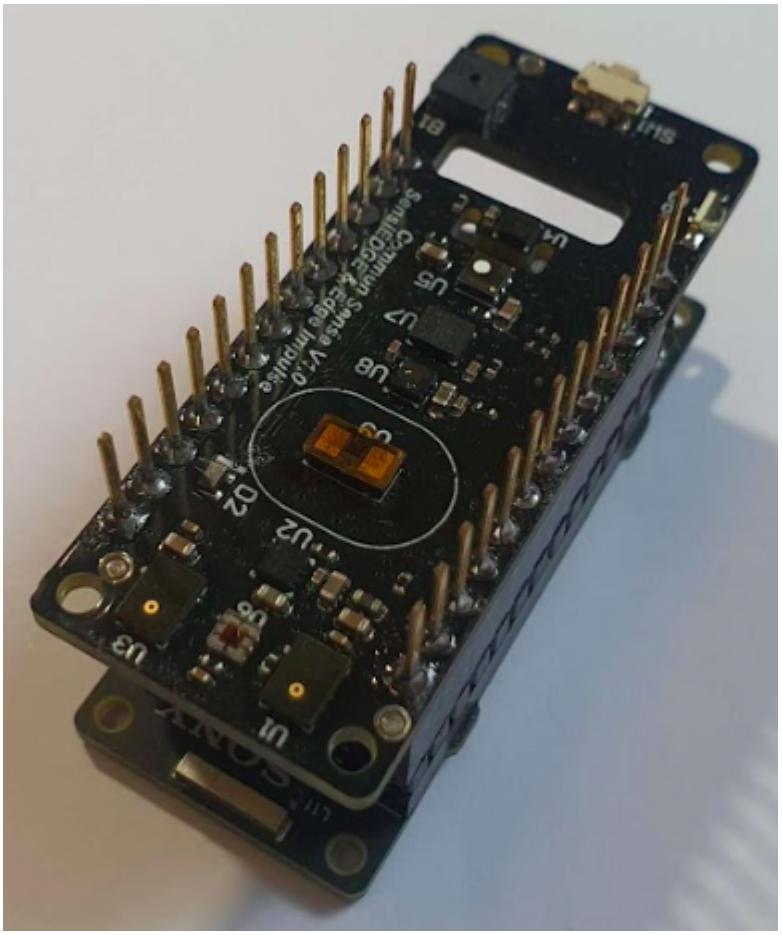
Visual Studio Code (VS):

```
sudo snap install --classic code
```

3.3 Compile and Run

The compilation of Sony's Spresense Sensor Board SW is from console commands. While commands may be run from the console in the project folder, or use VS code console, or using VS scripts - we used console commands.

1. Before running the project please connect Spresense Sensor Board with CommonSense Board in the following order.



2. Connect MicroUSB cable to Sony's Spresense Sensor Board

   There are list of console commands:

   - `make clean` – clean the project;
   - `make all` – build the project;

```
Memory region         Used Size  Region Size  %age Used
             ram:      243268 B      1536 KB     15.47%
mkspk/mkspk -c 2 build/firmware.elf nuttx build/firmware.spk
File build/firmware.spk is successfully created.
vladimir@vladimir:~/SensiEdge-EdgeShield$
```

- `make flash` – build the project if necessary and download Sony's Spresense Sensor Board firmware to the board.

```
>>> Install files ...
install -b 115200
Install out/Project.nuttx.spk
|0%----------------------------50%----------------------------100%|
################################################################################

181168 bytes loaded.
```

Congrats! You succeeded in running a demo project.

(*) For programming and examples of use of specific sensors, more information can be found in the CommonSense datasheet Paragraph 5, page 44.