# Contents

# 1 SCK 2.3 Test Plan

## 1.1 Introduction

**Four** binary files are used for testing and flashing the Smart Citizen Kit:

1. SAMD21 Bootloader
2. ESP12-F firmware
3. SAMD21 testing firmware (customized with the Wi-Fi credentials provided by you)
4. SAMD21 production firmware.

The SAMD21 *test firmware* is provided to you via email. It's customised with the Wi-Fi credentials of your facilities. The rest of the files can be build from the code on our Github repository.

The production firmware is provided in the Github Repository:

https://github.com/fablabbcn/smartcitizen-kit-2x/releases/latest

Please refer to the binaries named `sck23_air.*` for SAMD21 Chip:



Figure 1: imagen

And `ESP_firmware.bin` for the ESP8266 Chip:



Figure 2: imagen

## 1.2 Tools needed

There are two main possibilities to do this process: - by using the toolset provided (based on python) - by using atmel-studio or similar

Common needs:

- Computer running Linux or Windows OS (not tested, but should work with minor modifications)
- Micro USB cable
- Atmel-ICE programmer (https://digikey.es/en/product-highlight/a/atmel/atmel-ice-programmer-debugger)

**Depending on your toolset, choose one of the following:**

A) **Tools needed for python toolset:**

- Python 3.7 installed
- Git installed
- PlatformIO (https://platformio.org/platformio-ide) (only needed if you are compiling the firmware)
- OpenOCD (included with PlatformIO) or Atmel-Studio

B) **Tools needed for Atmel-Studio toolset:**

- Atmel-studio installed
- Python 3.7 installed

## 1.3 Steps

### 1.3.1 Cloning the firmware repository

If you are using Atmel-Studio to upload the bootloader or flash the firmware, you don't need to do the next step and we can provide you with the necessary binary files.

All the tools needed to flash the Smart Citizen Kit firmware are in our github repository. You can clone it by:

```
git clone --recursive git@github.com:fablabbcn/smartcitizen-kit-2x.git
```

Or you can download manually going to this link (https://github.com/fablabbcn/smartcitizen-kit-2x):
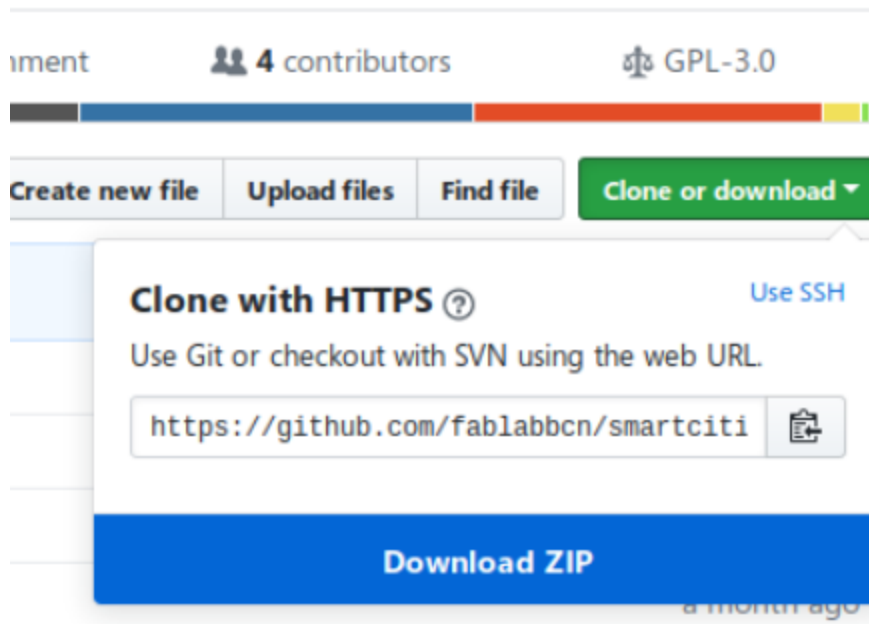
If you follow this method and download the ZIP file, you will also need to: 1. Download the bootloader repository from https://github.com/fablabbcn/uf2-samdx1 2. Uncompress the zip file in the firmware repository, inside `smartcitizen-kit-21/bootloader/` folder 3. Rename the uncompressed zip folder to `uf2-samdx1`

It should look like this: `smartcitizen-kit-21/bootloader/uf2-samdx1/`

### 1.3.2   Install requirements

This step is necessary for both toolset options

In the firmware folder you downloaded, do:

```
cd smartcitizen-kit-21
pip install -r requirements.txt
```

Depending on your platform, it may be that you need to type:

```
pip3 install -r requirements.txt
```

### 1.3.3   Bootloader

This step is necessary for both toolset options. Flashing the bootloader can be done with OpenOCD (https://openocd.org) or with Atmel-Studio, depending on your preferences.

You need an Atmel-ICE programmer connected to the SWD connector of the Smart Citizen Data Board. You will also need to provide power via the USB connector.
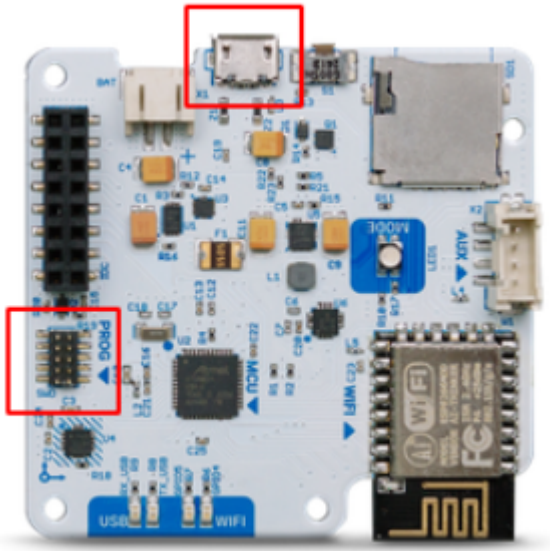
**Option A: With Python Scripts**

In the firmware folder, run:

```
./make.py boot -v
```

Or:

```
python make.py boot -v
```

Or:

```
python3 make.py boot -v
```

All the commands above are equivalent. It only changes depending on your OS setup.

If no errors pop up, the LED on the Smart Citizen Kit should turn **green**. You are ready for the next step.

**Option B: With Atmel-Studio**

For this, you will use the file `bootloader-sck2.3.bin` provided with the test firmware.

We do not provide specific instructions for flashing the bootloader with Atmel Studio. Please follow the application instructions.

If no errors pop up, the LED on the Smart Citizen Kit should turn **green**. You are ready for the next step.

### 1.3.4 SAMD21 Testing firmware

For this step, you have received a testing firmware with **hardcoded Wi-Fi credentials** called: `SAM_firmware_test`

- Connect the Smart Citizen Kit with a micro USB cable to the computer
- Double click the reset button of the Smart Citizen Kit. The SCK should be green and a new flash-drive named `SCK-23` should appear on your computer
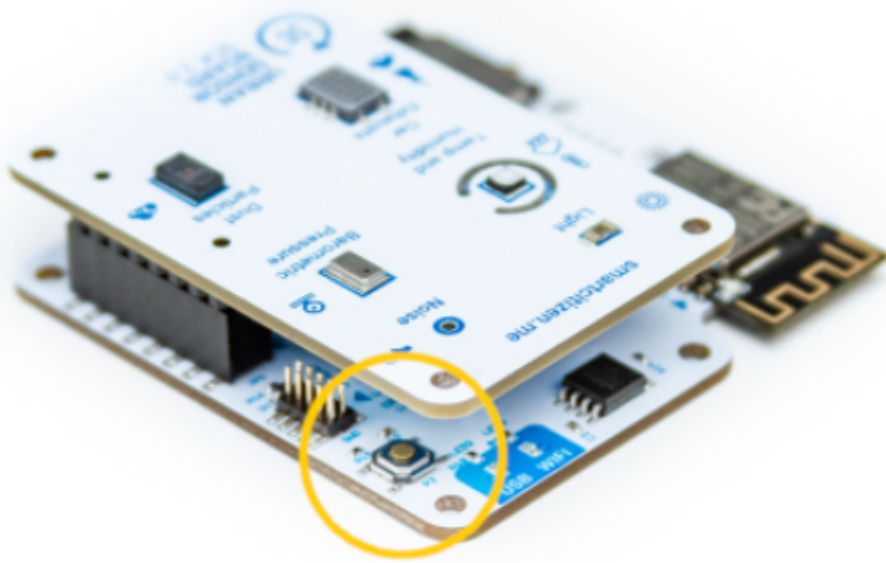
- Drag the firmware file over to the `SCK-23` drive. The LED should blink in **green** and, after some seconds, the device will reset and start the testing firmware
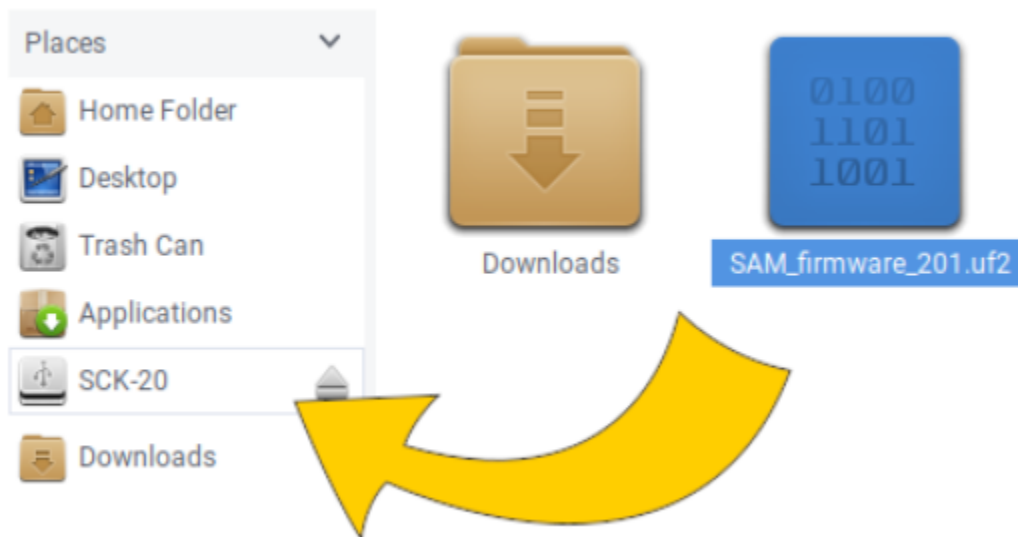


**Before proceeding with the Test process, you need to upload the ESP firmware in the next step.**

### 1.3.5 ESP Firmware

**For running this script, you need to have *python 3* installed in your computer.**

- Go to the folder where you downloaded the Smart Citizen Kit firmware:

  ```
  cd smartcitizen-kit-2x
  ```

- Get the firmware:

  **OPTION A:** If you are compiling the firmware:

  - Only the first time, build the ESP firmware (**only once**):

    ```
    python make.py build esp -v
    ```

  **OPTION B:** If you are not compiling the ESP firmware, and you are using the one provided via email, or from the firmware repository releases page, proceed as follows:

  1. Copy the file to the `smartcitizen-kit-2x/bin/ESP_firmware.bin`
  2. Proceed with the upload explained on the next step

- Flash the firmware

  To upload the firmware to the microcontroller, connect the SCK to the computer via USB and run:

  ```
  ./make.py flash esp -v
  ```

  Or:

  ```
  python make.py flash esp -v
  ```

  Or:

  ```
  python3 make.py flash esp -v
  ```

**If everything went OK, you are ready to start the test procedure on the SCK boards.**

### 1.3.6 Test Procedure

#### 1.3.6.1 Requirements  For the test you will need:

1. SCK Urban Board 2.3 connected to the SCK Data Board
2. Sensirion SEN5X sensor connected to the SCK Urban Board 2.3
3. SHT35 grove module connected to the SCK Data board via Grove Auxiliary Port
4. Writable micro sd-card inserted in the SCK Data Board slot.
5. LiPo Battery **NOT fully charged, or NOT fully dead** connected to the SCK Data Board
6. USB cable connected to the SCK Data Board
7. A serial port to check debugging information
8. A Wi-Fi network active

#### 1.3.6.2 Steps

1. Connect the **battery** and the **USB cable** to the SCK Data Board. The USB cable is only needed if you want to see debug logs
2. LED color should change from green to white and finally blue when SCK is ready to start the test
3. Click the button 3 times: LED should cycle red-green-blue colors
4. Test is running. The LED will blink in blue during the tests.
5. When the test is finished, **LED will be green OK**, **red if there is a problem**

In case of problems, you can open a serial terminal while the SCK is running the test to see what's failing (**WARNING:** this image does not show the full test):

Figure 3: image

We have detected that **Flash Memory** test fails sometimes. Repeating the test normally fixes the issue.

### 1.3.7 Production firmware

If the test succeeded, you need to upload the SAMD21 production firmware. To do so, follow the next steps:

1. Go to the folder where you downloaded the Smart Citizen Kit firmware:

   ```
   cd smartcitizen-kit-2x
   ```

2. Get the firmware:

   **OPTION A:** If you are compiling the firmware:

   - Only the first time, build the SAMD21 firmware (**only once**):

     ```
     python make.py build sam -v --env sck23_air
     ```

   **OPTION B:** If you are not compiling the SAM firmware, and you are using the one provided via email, or from the firmware repository releases page, proceed as follows:

   1. Copy the file to the `smartcitizen-kit-2x/bin/SAM_firmware_sck23_air.uf2`
   2. Proceed with the upload explained on the next step

3. Upload the SAM firmware:

   **OPTION A:**

   If you want to use the command line:

   ```
   python make.py flash sam -v --env sck23_air
   ```
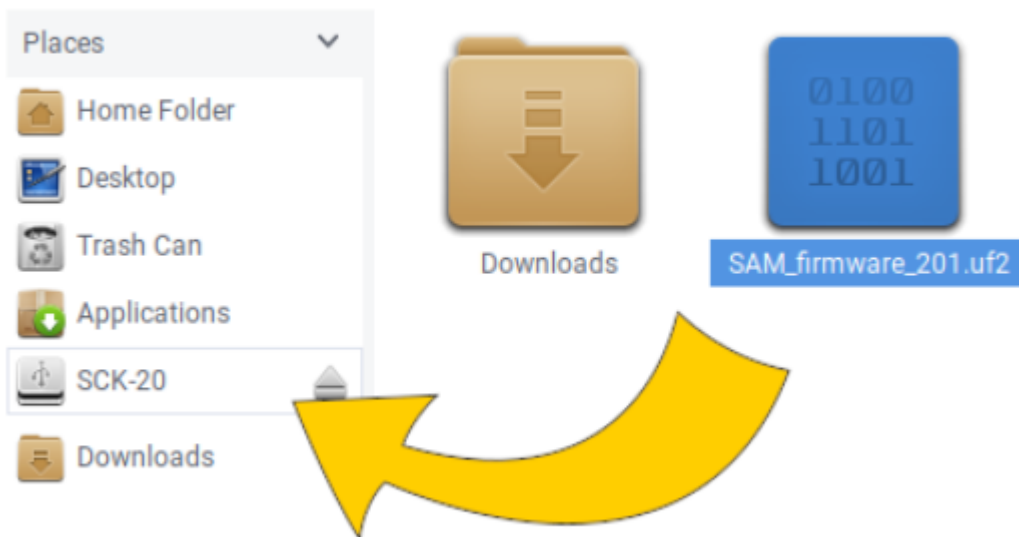
   **OPTION B:**

   If not:

   - Double click the reset button of the Smart Citizen Kit. The SCK should be green and a new flash-drive named `SCK-23` should appear on your computer:

- Drag the production firmware file over to the `SCK-23` drive. The LED should blink in **green** and, after some seconds, the device will reset and start the testing firmware



**Done!**