FOUNDRIES.IO

FoundriesFactory®

The software platform that reinvents IoT

Quick Start Guide

Variscite

Variscite and Foundries.io deliver an end-to-end DevSecOps platform for embedded developers of IoT and Edge devices.

This Reference Guide provides step-by-step instructions, from creating a Factory, to flashing, booting and updating the Variscite platform.

Foundries.io offers a free 30-day trial subscription - no credit card required.

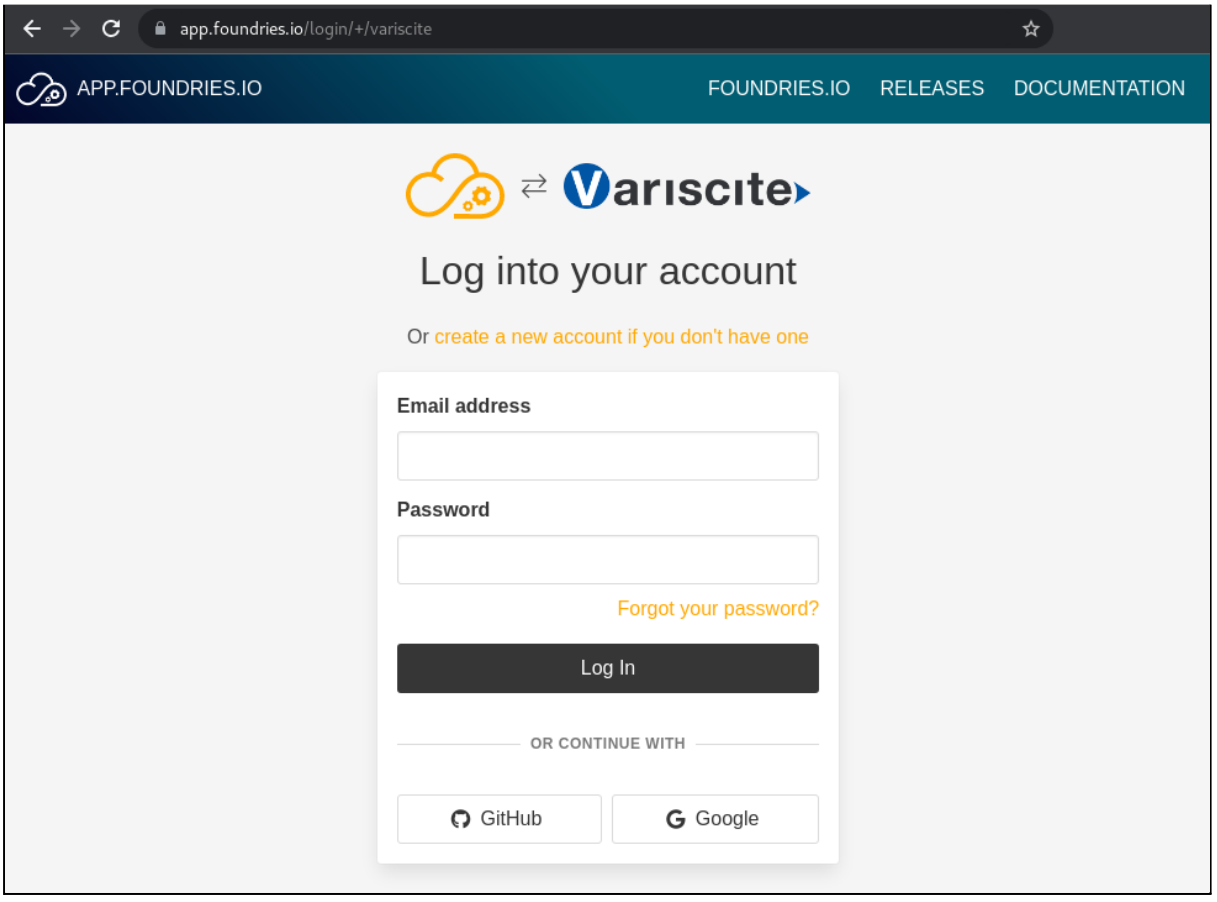
Table of Contents

Getting Started

Access the link below and follow the instructions to sign up and create your FoundriesFactory.
https://app.foundries.io/factories/+/variscite

Create an Account

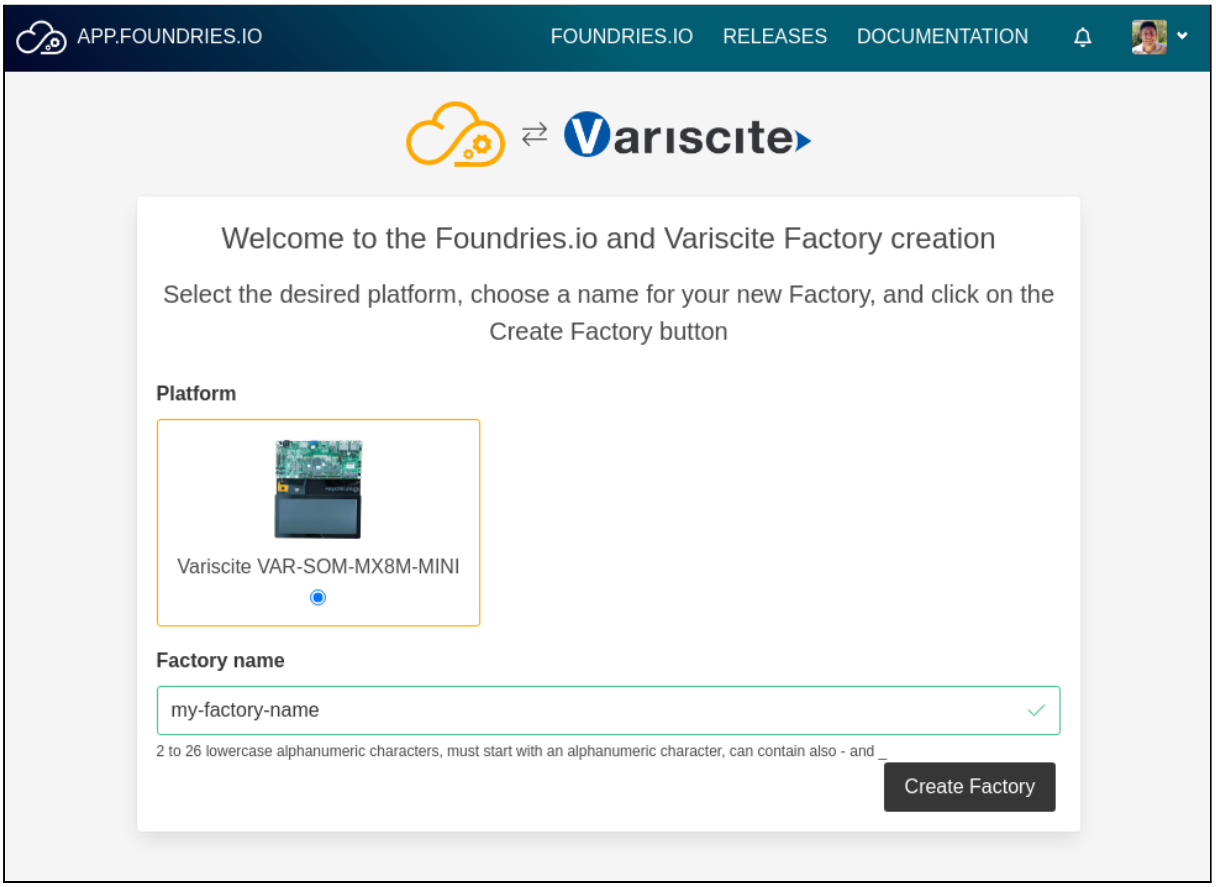
Create a new account if you do not have one, or continue with your existing Github or Google account.



Create a Factory in 3 Steps

1. Select VAR-SOM-MX8M-MINI **platform***
2. **Name** for your new Factory
3. Click on the **Create Factory** button

**If you want to try FoundriesFactory on a different Variscite platform, create the Factory as suggested for VAR-SOM-MX8M-MINI and contact Foundries.io at contact@foundries.io.*

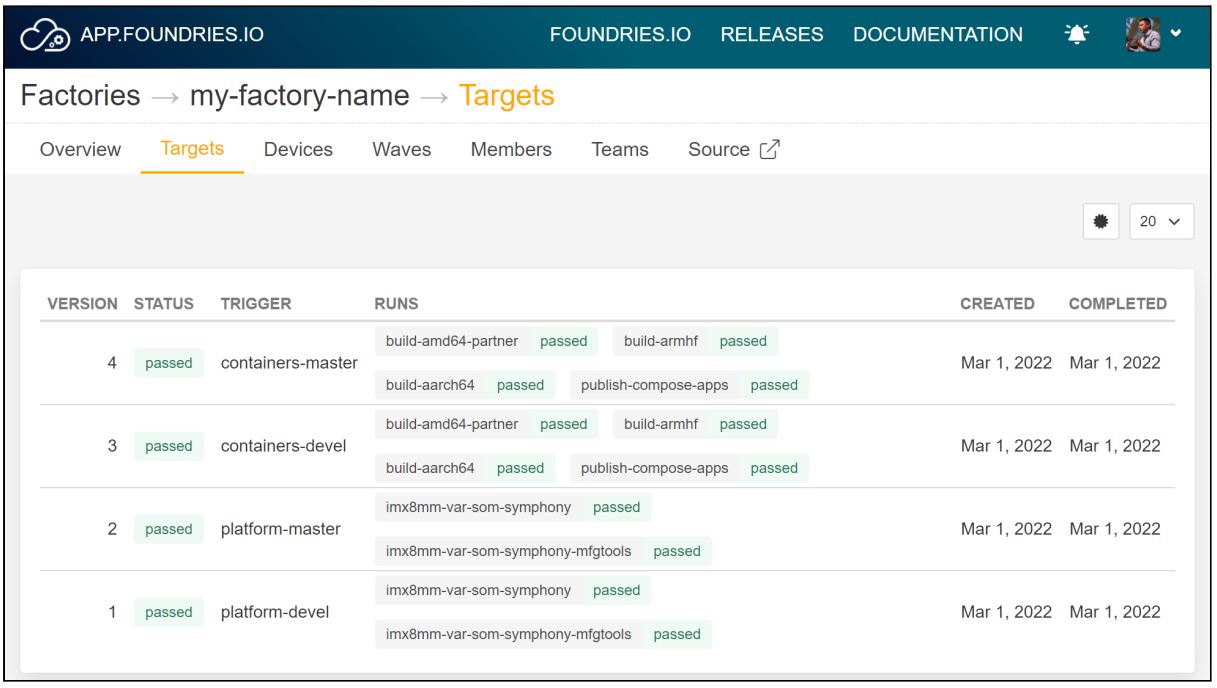


Watch Your Factory Build

An initial build of the Foundries.io Linux microPlatform™ (LmP) will be generated for you to build your product on top of. You can monitor the build progress in the **Targets** tab of your Factory after a few minutes. Additionally, you will receive an email once this initial build is complete.

Targets are a reference to a platform image and docker applications. When developers push code, the FoundriesFactory produces a new target. Registered devices update and install Targets.

The **Targets** tab of the Factory will become more useful as you begin to build your application and produce new Targets for the Factory to build.



Bootstrapping your Factory securely takes some time. Your first build can take 30 minutes or more to complete.

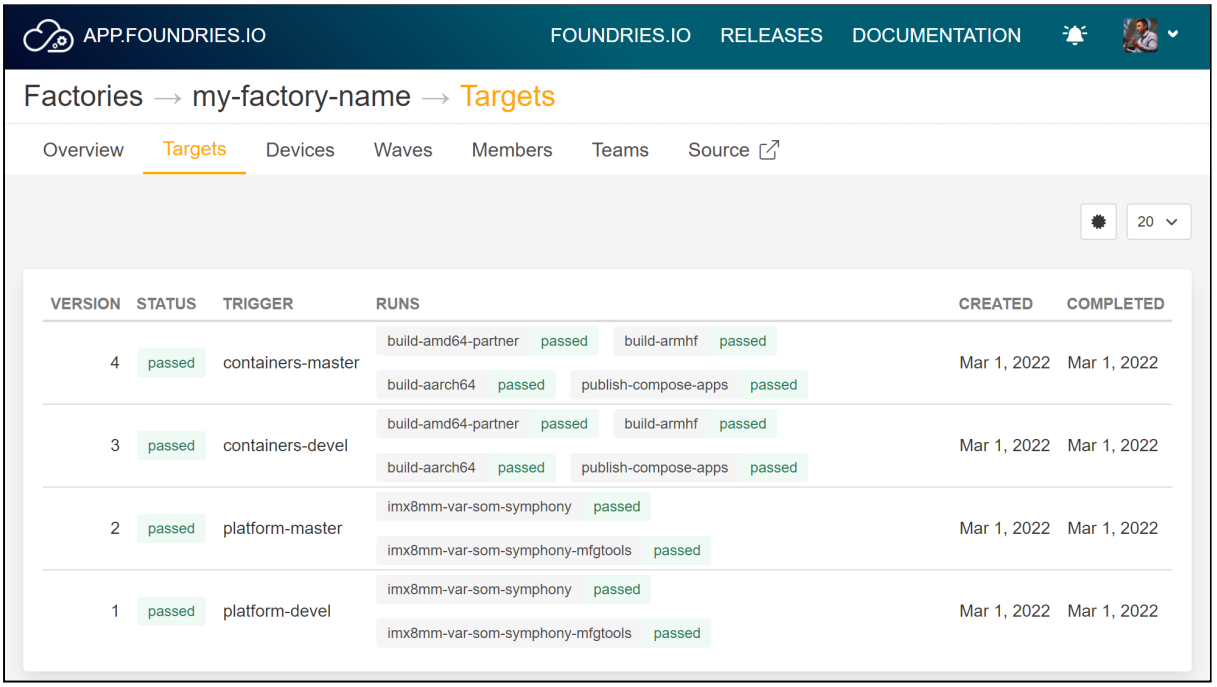
Use this time to set up your development environment and get started with docker commands. The next steps shown below do not require any hardware.

- *Configuring Git*
- *Fioctl CLI Installation*
- *Getting Started with Docker*

Download LmP artifacts

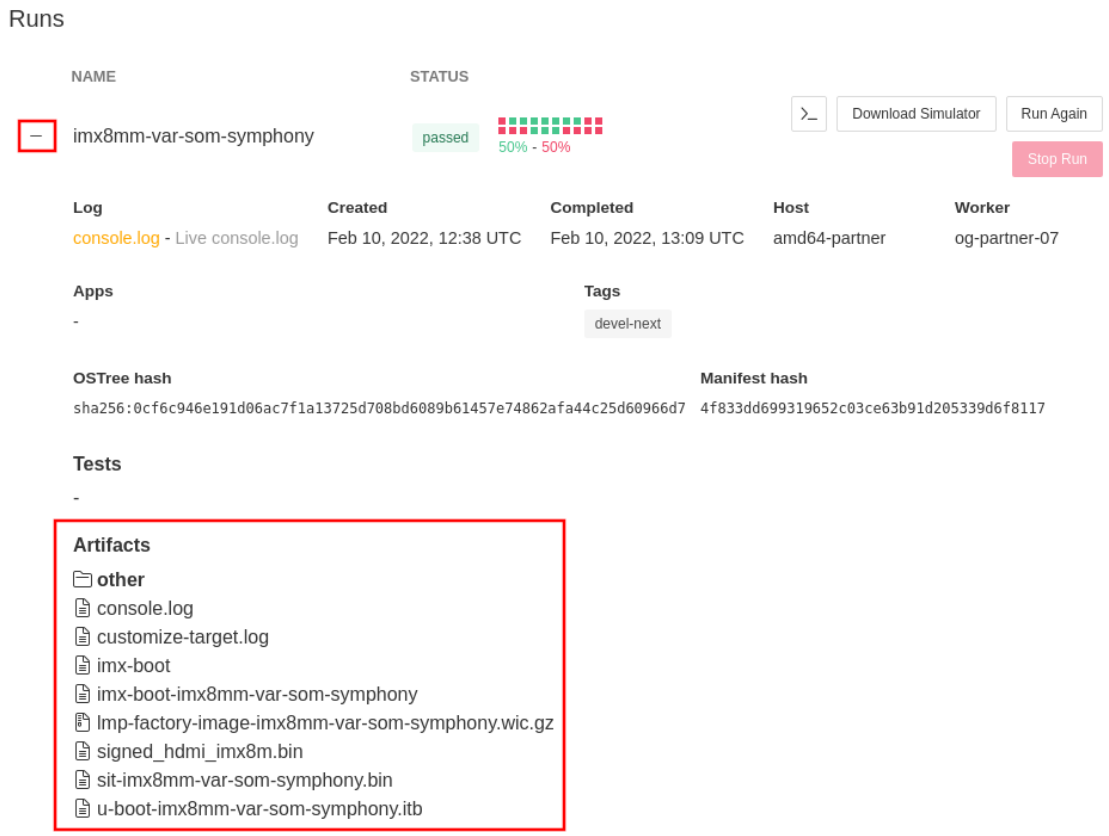
After your Factory setup completes, your device image and Factory tools will become available in the **Targets** tab of the Factory UI. These steps will walk you through downloading and installing the LmP image onto your device.

1. Navigate to the **Targets** section of your Factory.
   1.1. Click the latest **Targets** with the `platform-devel` **Trigger**.



1.2. Expand the run in the **Runs** section which corresponds with the name of the board and **download the Factory image for that machine**.

```
lmp-factory-image-imx8mm-var-som-symphony.wic.gz
u-boot-imx8mm-var-som-symphony.itb
sit-imx8mm-var-som-symphony.bin
imx-boot-imx8mm-var-som-symphony
```



1.3. Extract the file lmp-factory-image-imx8mm-var-som-symphony.wic.gz:

```
gunzip lmp-factory-image-imx8mm-var-som-symphony.wic.gz
```

1.4. Expand the run in the **Runs** section which corresponds with the name of the board mfgtool-files and **download the tools for that machine**.

`mfgtool-files-imx8mm-var-som-symphony.tar.gz`

1.5. Extract the file mfgtool-files-imx8mm-var-som-symphony.tar.gz:

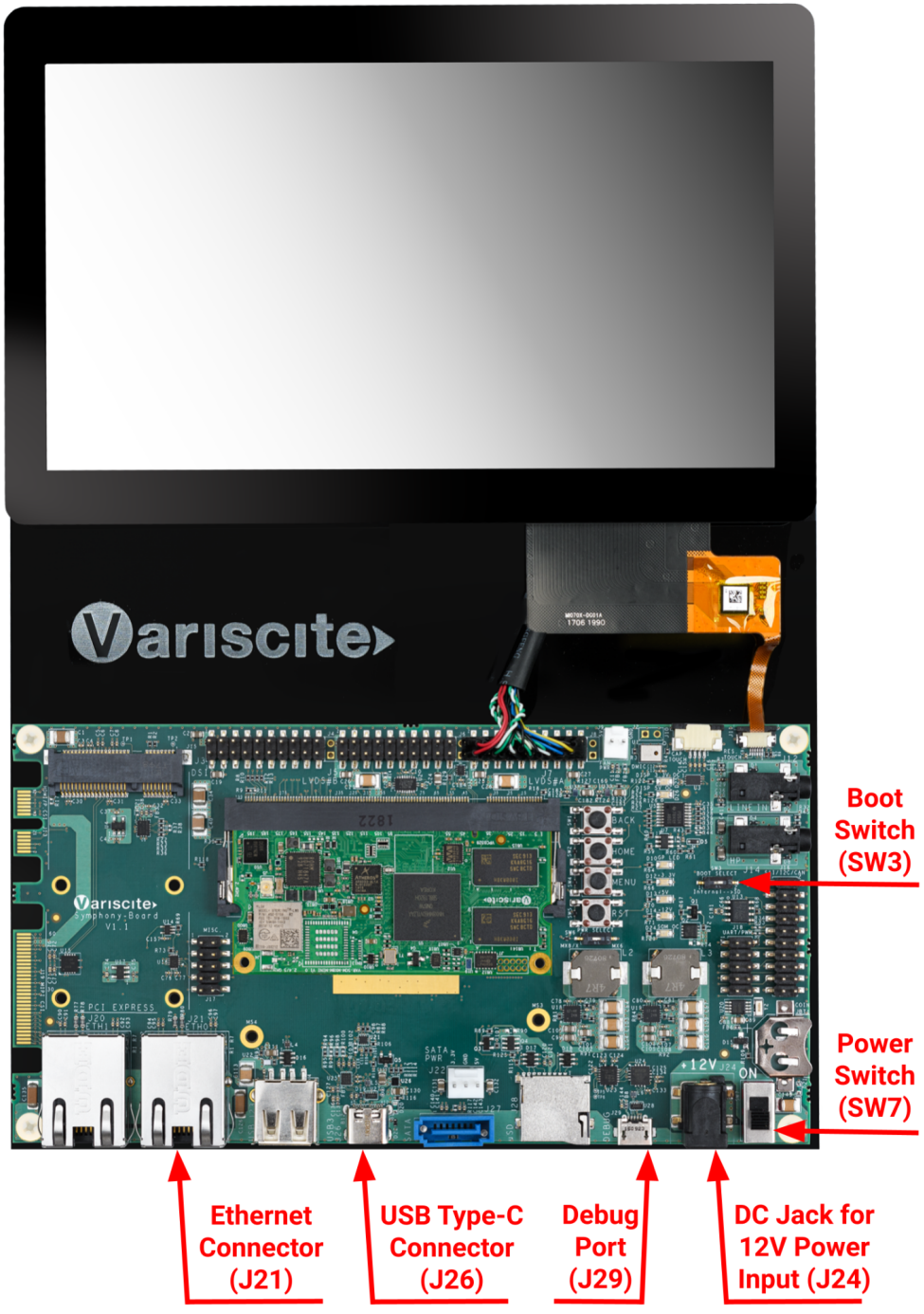
```
tar -zxvf mfgtool-files-imx8mm-var-som-symphony.tar.gz
```

1.6. Organize all the files like the tree below:

```
├── lmp-factory-image-imx8mm-var-som-symphony.wic
├── u-boot-imx8mm-var-som-symphony.itb
├── sit-imx8mm-var-som-symphony.bin
├── imx-boot-imx8mm-var-som-symphony
└── mfgtool-files-imx8mm-var-som-symphony
      ├── bootloader.uuu
      ├── full_image.uuu
      ├── imx-boot-mfgtool
      ├── uuu
      └── uuu.exe
```

Hardware Preparation

Set up the board for updating using the manufacturing tools:



1. OPTIONAL - Only required if you want to see the boot console output.

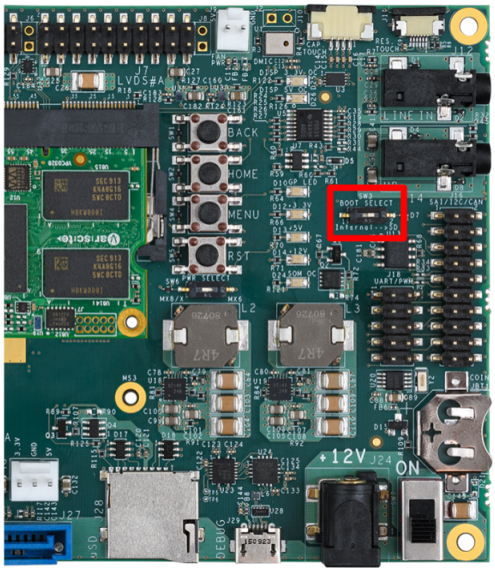
Connect the micro-B end of the USB cable into debug port J29. Connect the other end of the cable to a PC acting as a host terminal. Two UART connections will appear on the PC. On a Linux host for example:

```
$ ls -l /dev/serial/by-id/
total 0
lrwxrwxrwx 1 root root 13 Feb 24 01:30
usb-FTDI_FT230X_Basic_UART_DM02RUWP-if00-port0 -> ../../ttyUSB0
```

Using a serial terminal program like minicom, connect to the port with if00 in the name (in this example ttyUSB0) and apply the following configuration

- Baud rate: 115200
- Data bits: 8
- Stop bit: 1
- Parity: None
- Flow control: None

2. Ensure that the power is off (SW7)
3. Put the VAR-SOM-MX8M-MINI into programing mode:
   Switch SW3 to SD as shown below.



4. Connect your computer to the VAR-SOM-MX8M-MINI board via the USB Type-C connector J26 jack.
5. Connect the Power Supply plug to the DC J24 jack.
6. Power on the VAR-SOM-MX8M-MINI board by sliding power switch SW7.

Flashing

Once in serial downloader mode and connected to your PC the evaluation board should show up as an NXP USB device.

1. Verify target is present:

```
$ lsusb | grep NXP
Bus 001 Device 013: ID 1fc9:0134 NXP Semiconductors SE
Blank M845S
```

   In this mode you will use the uuu tools to program the images to the eMMC.

2. Run the command below to program the LmP to the EMMC:

```
$ sudo mfgtool-files-imx8mm-var-som-symphony/uuu -pp 1
mfgtool-files-imx8mm-var-som-symphony/full_image.uuu
uuu (Universal Update Utility) for nxp imx chips --
libuuu_1.4.43-0-ga9c099a

Success 1    Failure 0


1:31     3/ 3 [================100%================]
SDPV: jump
2:31     8/ 8 [Done                                 ]
FB: done
```

3. Turn off the power.
4. Put the board into run mode

   To put the VAR-SOM-MX8M-MINI into run mode, switch SW3 to BOOT setting.

   Power on the EVK board by sliding power switch SW7 to ON.

Booting your device

Once your new image has booted, you can access the device in three ways.

- Serial Console
- WiFi
    - `sudo nmcli dev wifi connect “network-ssid” password "network-password"`
- Ethernet

If you have established a network connection, login over ssh with the command using the device name or the IP address assigned via DHCP:

```
ssh fio@imx8mm-var-som-symphony.local
ssh fio@<IP>

The password is: fio
```

Register your device

Your Linux microPlatform image includes the `lmp-device-register` tool, that registers your device(s) via the Foundries.io REST API. It does require an active internet connection for registration. Follow the instructions in the link below to register your device:

If you prefer to have the demo application installed automatically after registration use the following command on the device:

```
sudo lmp-device-register -n <device-name> -a x-kiosk-imx8-fishtank
```

Otherwise:

```
sudo lmp-device-register -n <device-name>
```

Now, you will be prompted by `lmp-device-register` to complete a challenge with our API.

After completing the challenge, the device is registered and should be visible by navigating to the web interface at https://app.foundries.io/factories/, clicking your Factory and selecting the Devices tab.

Or by using *fioctl* on your host:

```
fioctl devices list
```

On the device you can follow aktualizr-lite logs and monitor the status of the update agent. This is where you can find information about update events happening on the system.

```
sudo journalctl -f -u aktualizr-lite
```

Creating Your First Over-The-Air (OTA) Update

You have now registered a device either with or without a default application. That state can be changed by our management tool *fioctl*, using some examples given below.

If you registered your device **with** the default application, an OTA update installed a Docker Compose application demonstrating a Chromium-base WebGL example (x-kiosk-imx8-fishtank).

To add or remove the application remotely,

1. Locate your device name and copy it:

```
fioctl devices list
```

2. Install the demo application:

```
fioctl devices config updates <device-name> -f <my-factory-name> --tags
devel --apps x-kiosk-imx8-fishtank
```

3. Or, remove the demo application:

```
fioctl devices config updates <device-name> -f <my-factory-name> --tags
devel --apps ,
```

This configuration update may take up to five minutes (this interval is configurable) to be noticed.

Next Steps

If you would like to explore creating platform (firmware/OS) updates, you can read our documentation about customizing the platform.

Support

Foundries.io wants you to know that we are here to assist you. We provide several ways to help get you going, including our documentation, which covers most aspects of your FoundriesFactory experience, including device customization. Feel free to contact our support team via email at support@foundries.io, or directly through our public Slack support channel. However you choose to communicate, we look forward to working with you.
For hardware support you use the Variscite Portal or email sales@variscite.com.

Subscriptions

Thank you for starting a 30-day free evaluation. You can transition to a paid subscription within your evaluation Factory using a credit card. You can contact us through email at contact@foundries.io for other payment options.