



## How to manage OpenSTLinux project in STM32CubeIDE



---

This article explains how to import OpenSTLinux projects into STM32CubeIDE from version 1.4.0 on Linux workstation.



### Contents

1 Overview .....	3
2 OpenSTLinux project structure .....	4
3 Linux project .....	5
4 Other Projects .....	8

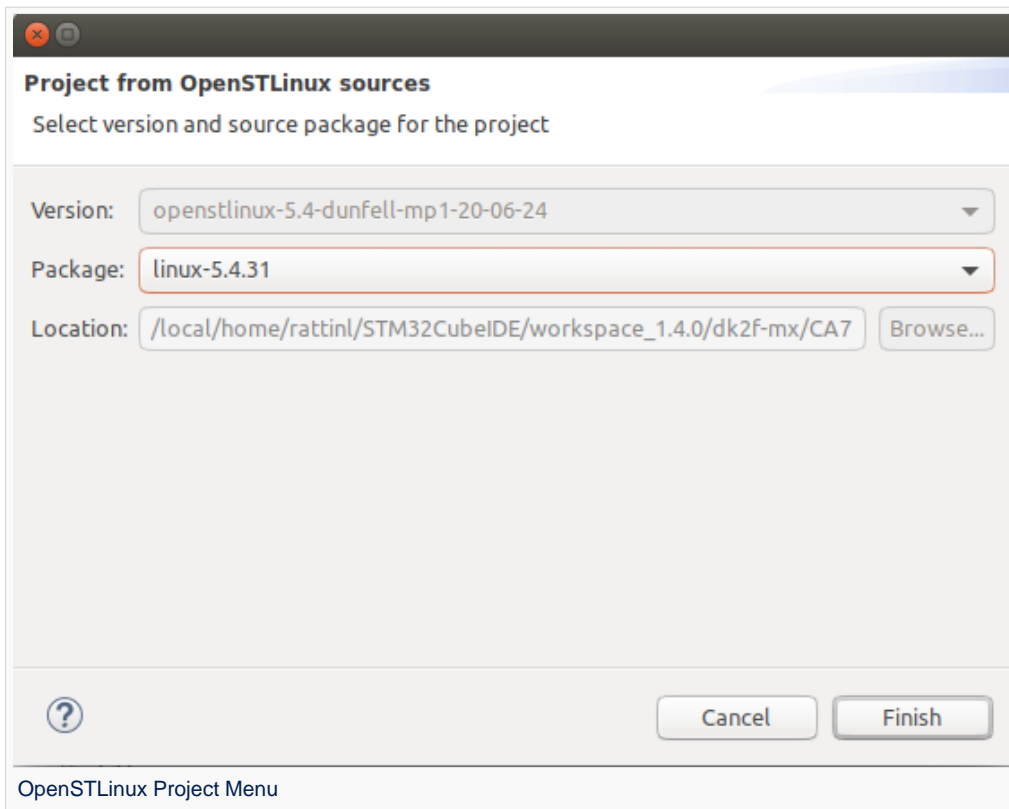


## 1 Overview

It is possible to import into an existing STM32MP15 Project some OpenSTLinux projects:

- *STM32CubeIDE 1.5.0 provides support for ecosystem release v2.1.0* 
- *STM32CubeIDE 1.4.0 provides support for ecosystem release v2.0.0* 

This is done thanks to **Import an OpenSTLinux project...** Cortex-A hierarchical project contextual menu.

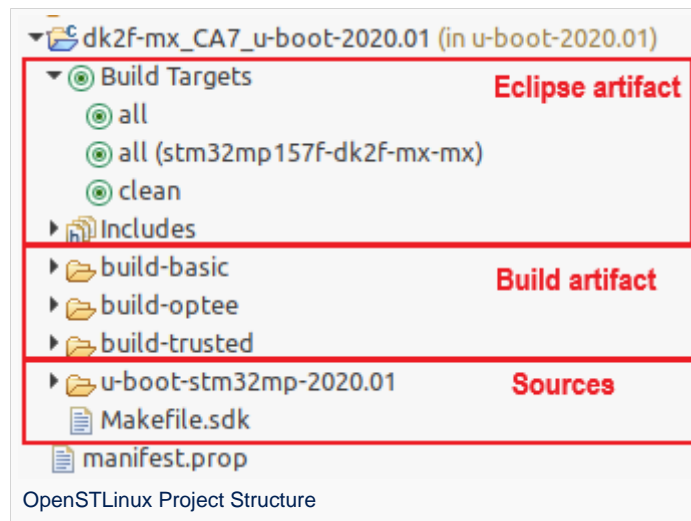




## 2 OpenSTLinux project structure

All the OpenSTLinux projects follow the same project structure approach with three main parts:

- Eclipse artifact, in order to ease project usage,
- build artifact, identifying clearly the project output,
- sources, as they appear in Yocto environment



Device trees generated by STM32CubeMX into *DeviceTree/* directory of Cortex-A hierarchical project, are automatically included inside the corresponding OpenSTLinux project with:

- source file links,
- dedicated *Build target* .

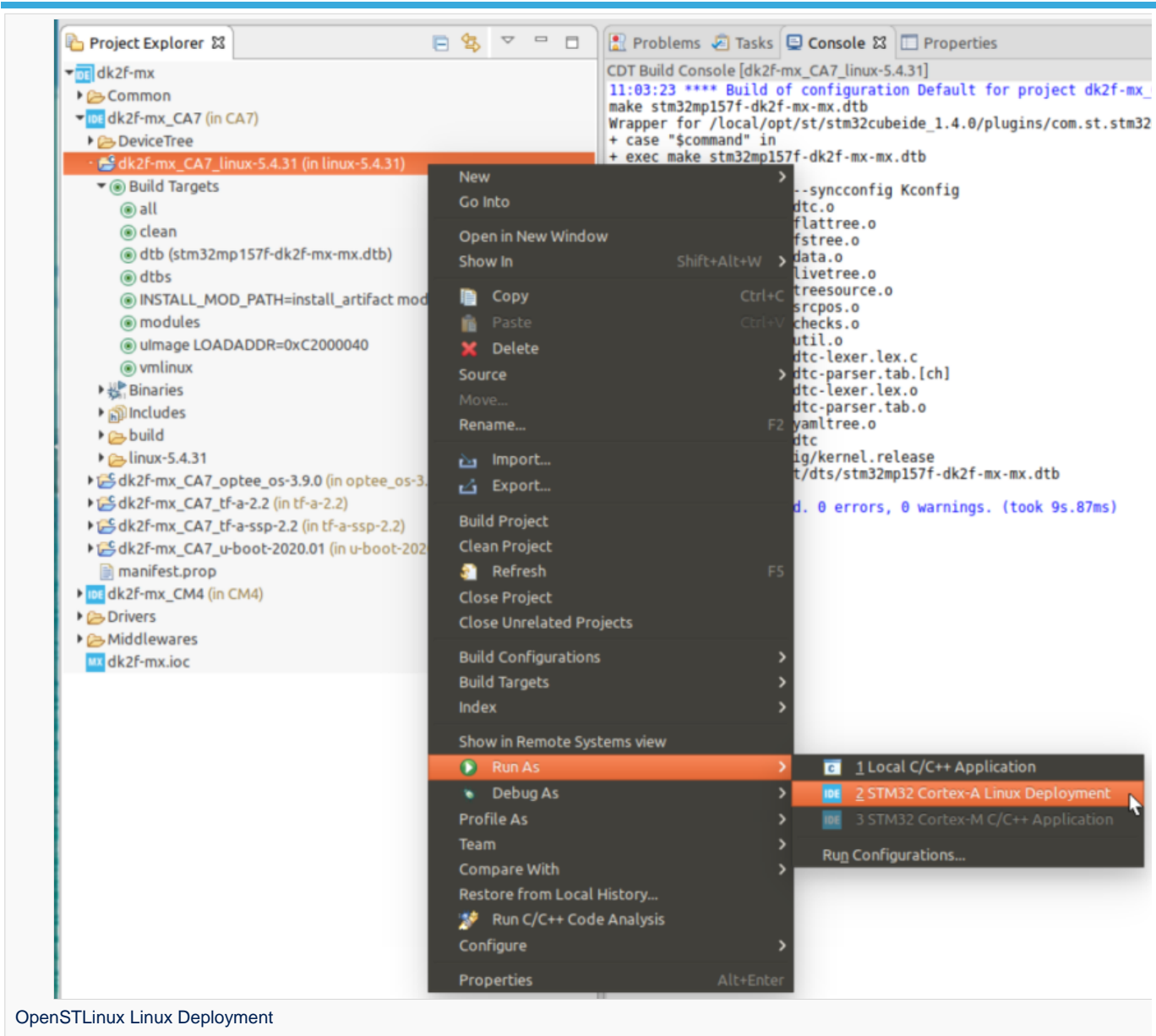


### 3 Linux project

Several *Build targets* are proposed for Linux project addressing device trees, module and kernel. Double click on *Build target* launches the corresponding compilation with associated Yocto SDK.

OpenSTLinux Linux Device Tree MX Build

It is then possible to update the running STM32MP15 target with the new build artifacts using the *STM32 Cortex-A Linux Deployment Run Configuration* menu.



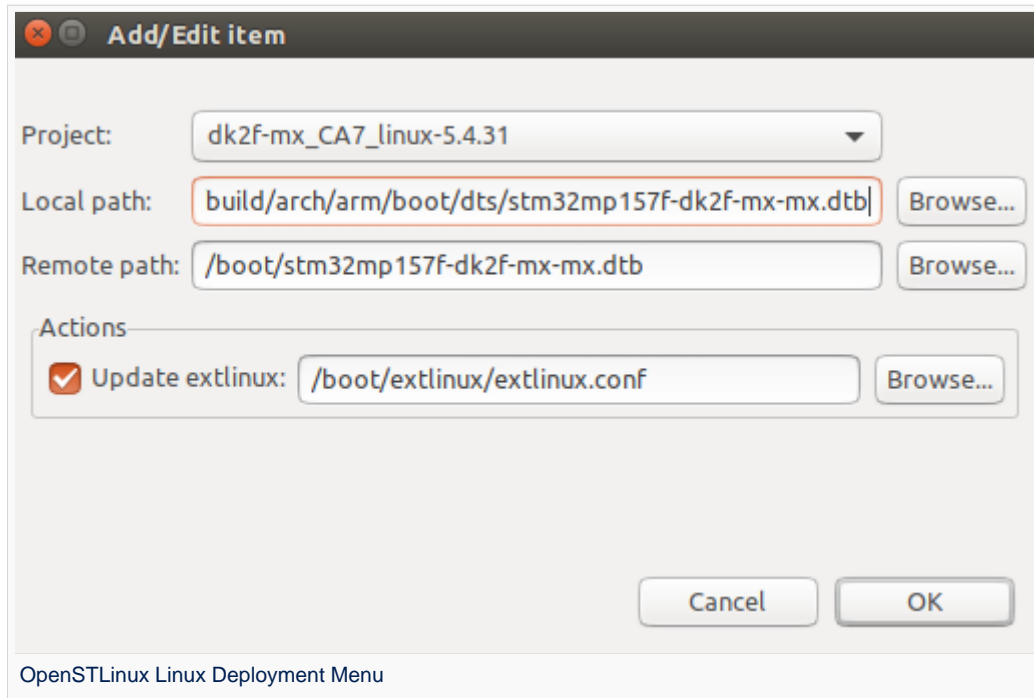
In this dedicated *Run Configuration* menu, build artifact (device tree binary, module, kernel..) is added using *Add...* button and giving:

- Local path, on the Linux workstation (en.FLASH-stm32mp1-openstlinux-5-4-dunfell-mp1-20-11-12.tar.xz ; here build/arch/arm/boot/dts/stm32mp157-dk2f-mx-mx.dtb)
- Remote path, browsing the remote target file system

For a new device tree, it is possible to update *extlinux.conf* target file in order to take it into account on the next target reboot, see *Actions* sub-window.



**To get correct support please select the *extlinux.conf* file corresponding to your board browsing onto the target!**



After download of build artifacts to the STM32MP15 board:

- a script can be executed (optional)
- STM32MP15 board is rebooted (default)



---

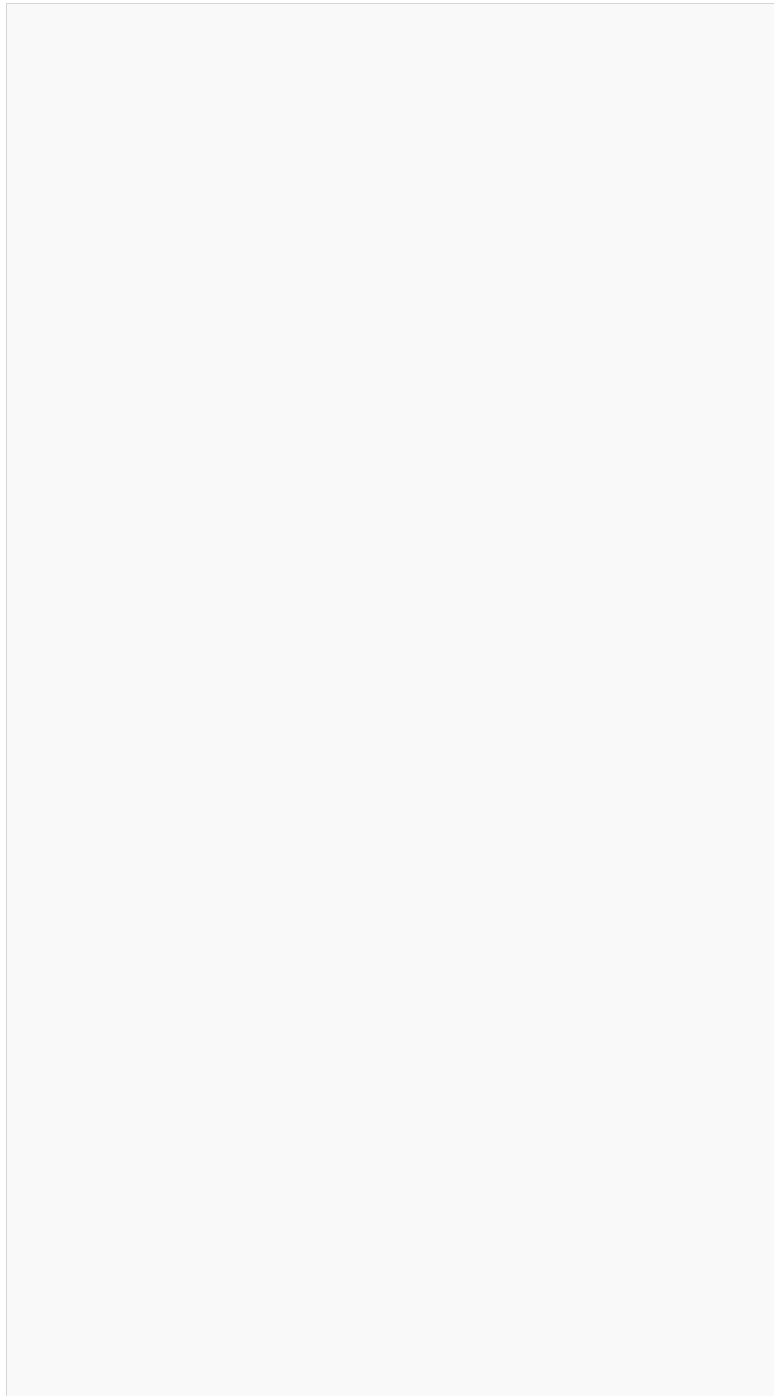
## 4 Other Projects

---

TF-A, U-boot and OP-TEE projects are proposing as Build targets:

- all
- clean
- all, including Device Tree generated by STM32CubeMX

They generate flash partition that has to be updated on the STM32MP15 board typically using STM2CubeProgrammer.







dk2f-mx\_CA7\_u-boot-2020.01 (in u-boot-2020.01)

- ▼ Build Targets
  - all
  - all (stm32mp157f-dk2f-mx-mx)
  - clean
- ▶ Includes
- ▶ build-basic
- ▼ build-optee
  - ▶ arch
  - ▶ board
  - ▶ cmd
  - ▶ common
  - ▶ disk
  - ▶ drivers
  - ▶ dts
  - ▶ env
  - ▶ examples
  - ▶ fs
  - ▶ include
  - ▶ lib
  - ▶ net
  - ▶ scripts
  - ▶ source
  - ▶ tools
  - Makefile
  - System.map
  - u-boot
  - u-boot.bin
  - u-boot.cfg
  - u-boot.cfg.configs
  - u-boot.dtb
  - u-boot.lds
  - u-boot.map
  - u-boot.srec
  - u-boot.stm32
  - u-boot.stm32.log
  - u-boot.sym
  - u-boot-dtb.bin
  - u-boot-nodtb.bin
  - u-boot-stm32mp157f-dk2f-mx-mx-optee.elf
  - u-boot-stm32mp157f-dk2f-mx-mx-optee.stm32**
- ▶ build-trusted
- ▶ u-boot-stm32mp-2020.01
  - Makefile.sdk
- manifest.prop

OpenSTLinux TFA Project



---

Software development kit (A programming package that enables a programmer to develop applications for a specific platform.)

Trusted Firmware for Arm Cortex-A

Open Portable Trusted Execution Environment