**Flashing Tool -Examples**

April 2, 2018

* **Purpose**

* **methods on how to flash the board**
  1. **Get Starter Package from wiki**
  2. **Flash with CubeProgrammer**
     1. **Flash layout overview**
     2. **Flash sdcard under windows**
     3. **CubeProgrammer installation on Linux**
     4. **Flash scard under Linux**

**2 Fast sdcard cloning from a “raw” image**

**2.1 Create raw image**

**2.2 Flash Raw image on linux**

**2.3 Flash Raw image on windows**

**1.0 Get the starter package SDcard binary**

**Get starter package from**

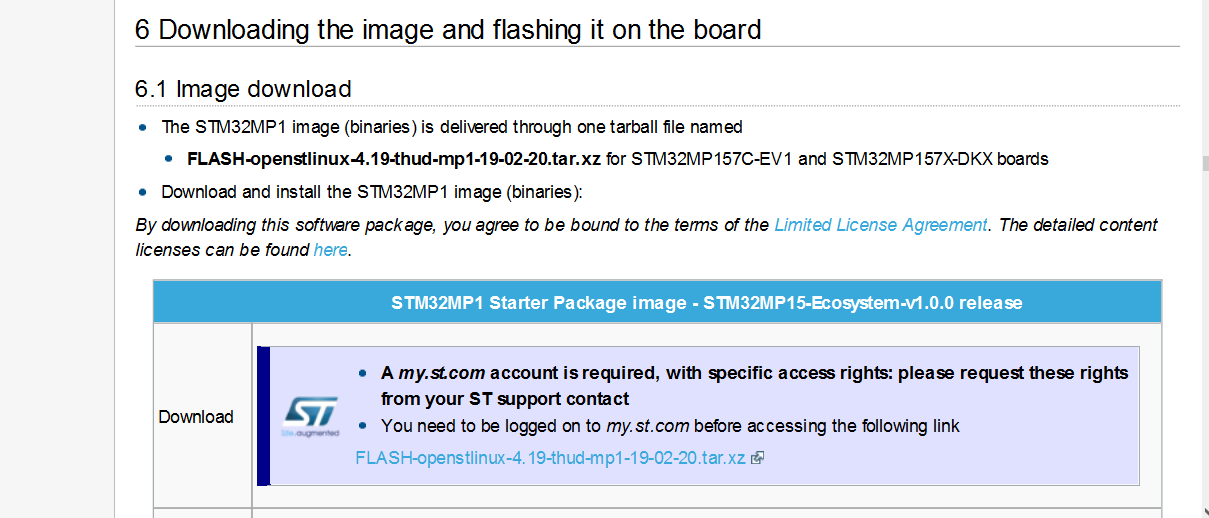
[**https://wiki.st.com/stm32mpu/index.php/STM32MP15\_Discovery\_kits\_-\_Starter\_Package**](http://intranet.lme.st.com:8000/php-bin/ug_mcdmpu/index.php/STM32MP15_Discovery_kits_-_Starter_Package)

**for board revision C only (otherwise go to archive images**

[**FLASH-stm32mp1-openstlinux-4.19-thud-mp1-19-02-20.tar.xz**](https://citools.st.com/artifacts/artifactory/MPUPUBDV-codex-st-com/openstlinux-4.19-thud-mp1-19-02-20/FLASH-stm32mp1-openstlinux-4.19-thud-mp1-19-02-20.tar.xz) **(temporary repository)**

[**https://wiki.st.com/stm32mpu/index.php/STM32MP1\_Starter\_Package\_-\_images**](http://intranet.lme.st.com:8000/php-bin/ug_mcdmpu/index.php/STM32MP1_Starter_Package_-_images)

**(soon available)**



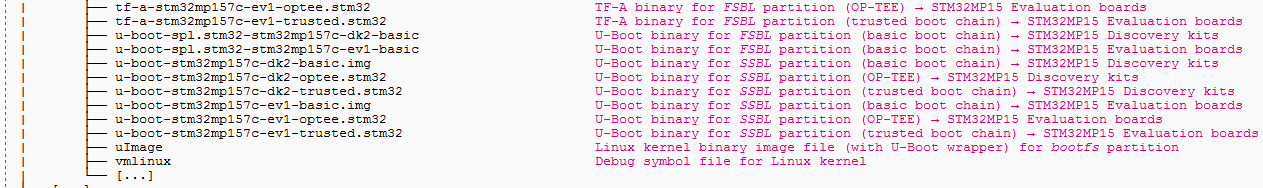
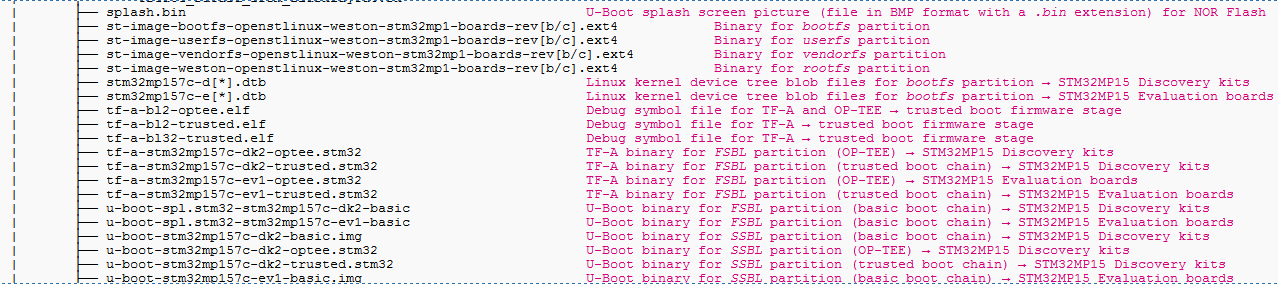
**1.1 STM32CubeProgrammer flash-layout overview**

 https://wiki.st.com/stm32mpu/index.php/STM32MP1\_Distribution\_Package

* In **Distribution package**, the st-image-weston binaries to flash are under

$DISTRI\_ROOT/$DISTRI\_RELEASE/build-openstlinuxweston-stm32mp1**/tmpglibc/deploy/images/stm32mp1**

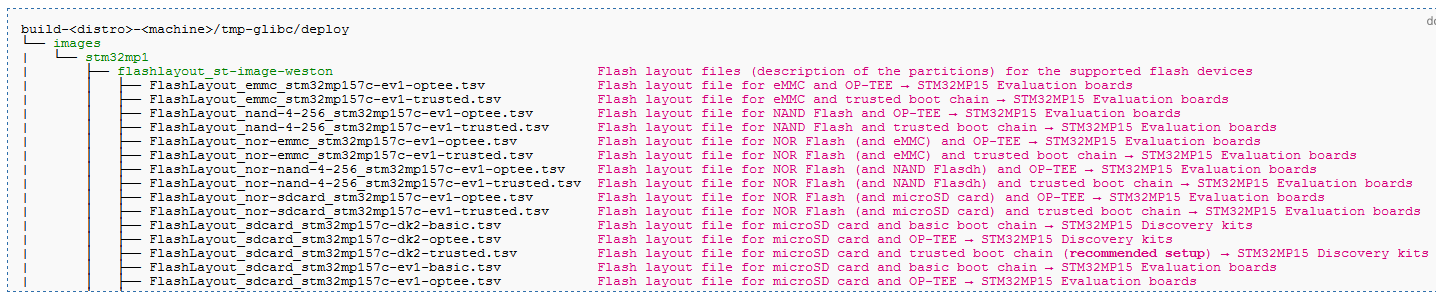
* Contains all binaries : partitions (bootfs,rootfs,…) + firmware (tf-a,Uboot,Uimage) + device tree (\*.dtb)



* Under sub-directory **flashlayout\_st-image-weston** are the Flash *layout descriptor* \*.tsv files

Different \*.tsv depending on

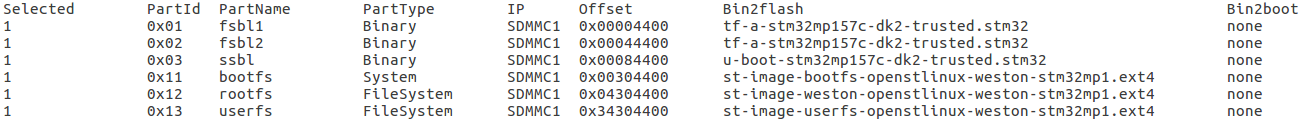
* + Board (discovery , eval,),
  + booting devices (eMMC,NAND,SDCard,…)
  + boot chains (basic or trusted, op-tee)



* Flash layout descriptor \*.tsv files list the partitions to flash with address and boot device.

\*.tsv file is an Input of the STM32CubeProgrammer

Example **FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.tsv**



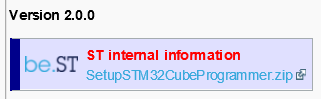
**1.2 Flash image on SDcard with STM32CubeProgrammer from windows**

* Install STM32CubeProgrammer windows version

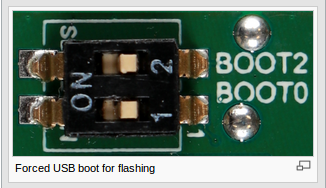
Open in web browser:

[https://wiki.st.com/stm32mpu/index.php/STM32CubeProgrammer](http://intranet.lme.st.com:8000/php-bin/ug_mcdmpu/index.php/STM32CubeProgrammer)

**download from internal repository and run installer (.exe)**



* + Discovery switch to activate USB DFU bootloader

When programming the discovery board boot pins set to ‘00’ (off) 

Connect the second USB type C cable to the board

USB ST-link cable for hyper terminal



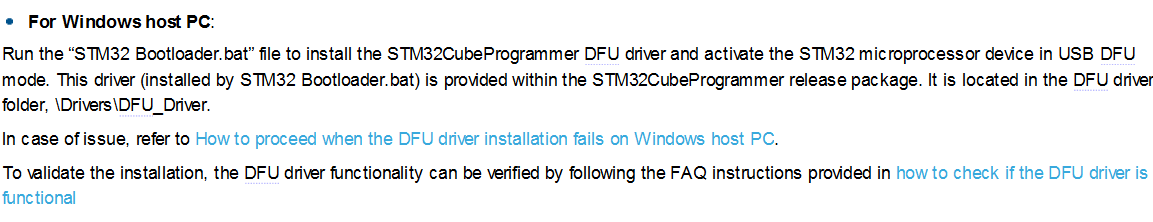
Boot pins switch “near” “12”

* bootloader is in USB DFU mode

USB OTG cable for CubeProgrammer

USB Power cable

Check dfu driver is installed on windows



* + - If you are using Win10 you may need

Download Zadig : http://zadig.akeo.ie/downloads/zadig-2.3.exe

Launch Zadig.

Plug your board or just Reset it.

in Zadig : options-> List All Devices.

Choose your board from the device list (DFU ...)

Press Install Driver Button

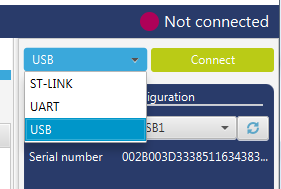
* Run CubeProgammer under windows

Select **USB**

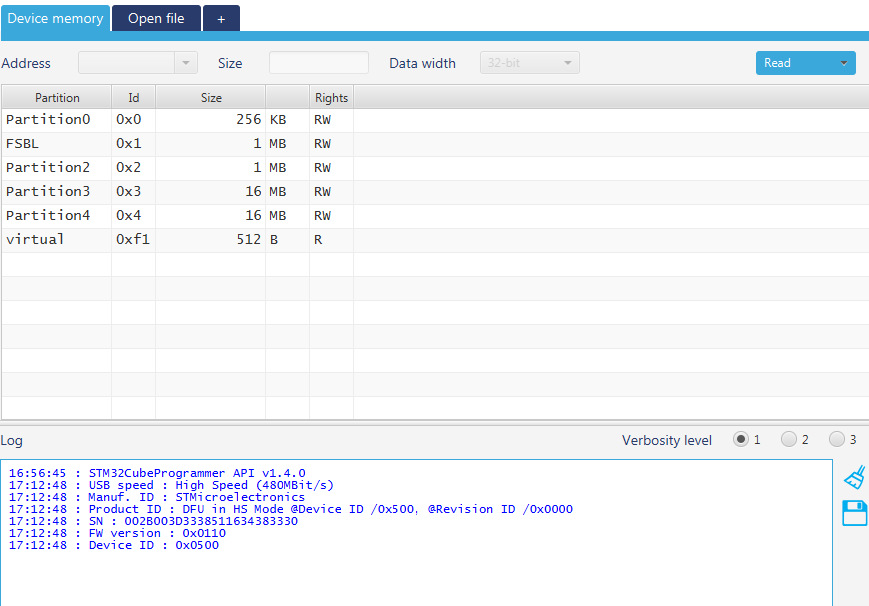
and the configuration shall be detected (SB1, serial number)

Then

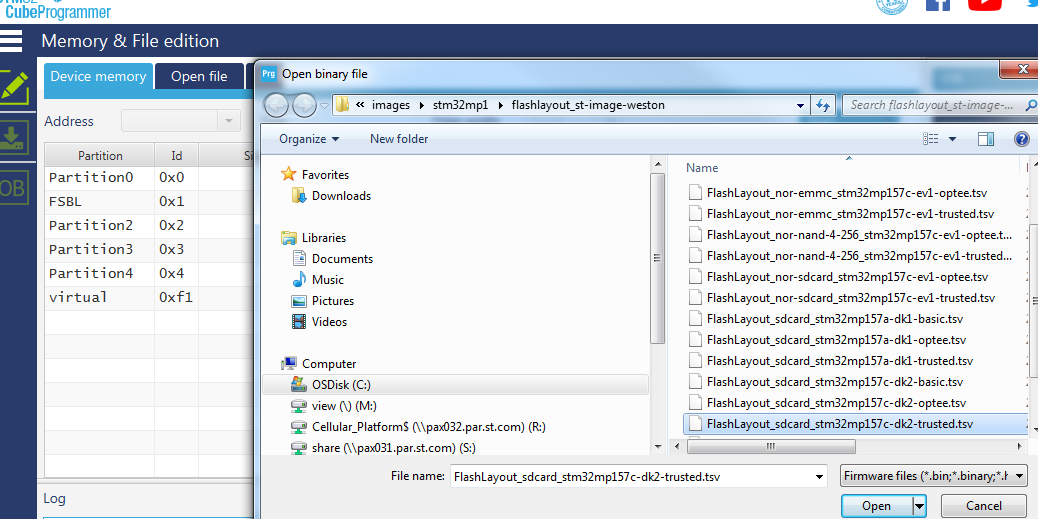
**Connect**



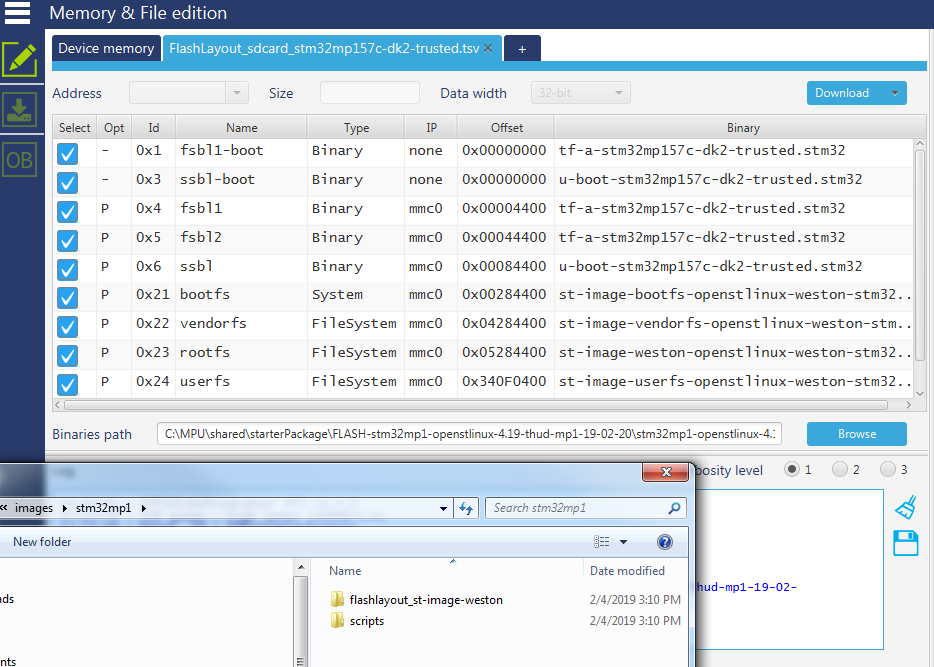
You see the SDCard partitions



Open the FlashLayout\_sdcard\_stmp32mp157c-dk2-trusted.tsv file   
in “Flashlayout\_st-image-weston” folder



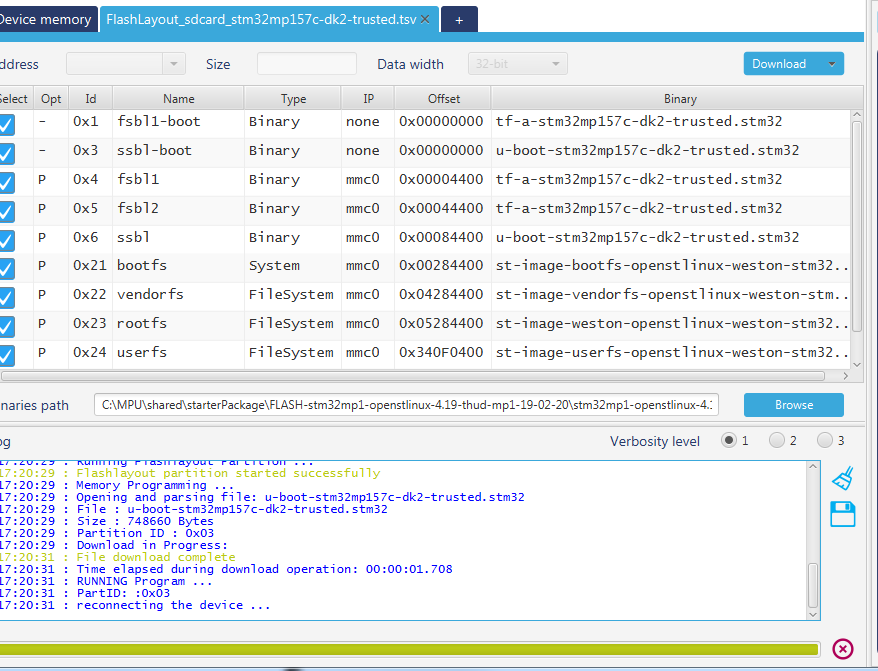
Browse to set the binary path



binaries are in same folder level

as “Flashlayout\_st-image-weston”

Then download



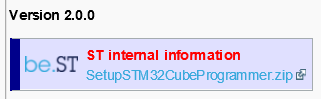
**1.3 STM32CubeProgrammer installation on linux host**

* Get STM32CubeProgrammer

Open in Linux web browser:

[https://wiki.st.com/stm32mpu/index.php/STM32CubeProgrammer](http://intranet.lme.st.com:8000/php-bin/ug_mcdmpu/index.php/STM32CubeProgrammer)

For the moment use the internal repository:



On Linux host download, unzip SetupSTM32CubeProgrammer.zip on under ~/Desktop

Note: On VMware Linux host You can download from windows then drag&drop under the Desktop

Summary installation of the sections 2.1 and 2.2 of [https://wiki.st.com/stm32mpu/index.php/STM32CubeProgrammer](http://intranet.lme.st.com:8000/php-bin/ug_mcdmpu/index.php/STM32CubeProgrammer)

* Install STM32CubeProgrammer

|  |
| --- |
| cd ~/Desktop  unzip "SetupSTM32CubeProgrammer v2.0.0.zip" -d "./SetupSTM32CubeProgrammer v2.0.0"  cd ~/Desktop/SetupSTM32CubeProgrammer\ v2.0.0/  ./SetupSTM32CubeProgrammer-xxx.linux |

Wizard installer is then started, let the default path/settings

|  |
| --- |
| sudo ln -s HOME/STMicroelectronics/STM32Cube/STM32CubeProgrammer/bin/STM32\_Programmer\_CLI $HOME/bin/STM32\_Programmer\_CLI $ |

* + Install DFU USB driver for USB OTG cable

|  |
| --- |
| sudo apt-get install libusb-1.0-0  sudo cp $HOME/STMicroelectronics/STM32Cube/STM32CubeProgrammer/Drivers/rules/\*.\* /etc/udev/rules.d/ |

* + Install Java if not already done (you will get a warning at gfx launch)   
    before launching graphical interface or command line

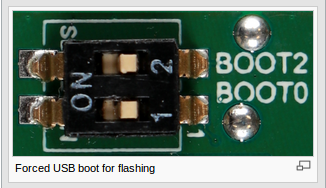
(/home/osboxes/STMicroelectronics/STM32Cube/STM32CubeProgrammer/bin/STM32CubeProgrammer)

|  |
| --- |
| sudo apt-get install openjfx |

**1.4 Flash image on SDcard with STM32CubeProgrammer from Linux host**

* + Flash & Run

* + Discovery switch to activate USB DFU bootloader

When programming the discovery board boot pins set to ‘00’ (off) 

* Connect second USB type C OTG port is to Linux Host
* on VM before running commands, activate the USB DFU driver use **VM Player menu "removable devices"** to connect to STMicroelectronics DFU in HS mode

Flashing the “st-image-weston“ image with SDcard partitions on a disco SDK2 with the trusted boot chain

|  |
| --- |
| **for distribution package**  **export DISTRI\_BUILD\_PATH=$DISTRI\_ROOT/$DISTRI\_RELEASE/build-openstlinuxweston-stm32mp1**  **export FLASH\_IMAGE\_DIR=$DISTRI\_BUILD\_PATH/tmp-glibc/deploy/images/stm32mp1**  **for starter package**  **export FLASH\_IMAGE\_DIR=/<starterPackage>/../images/stm32mp1**  **export FLASH\_IMAGE\_TRUSTED=flashlayout\_st-image-weston/FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.tsv**  **cd $FLASH\_IMAGE\_DIR**    STM32\_Programmer\_CLI -c port=usb1 -w $**FLASH\_IMAGE\_TRUSTED** |
|  |

* + Set switch to boot from SDcard
    - When booting the discovery board boot pins set to ‘11’ (on) Machine generated alternative text:
      BOOTO 
      Boot from microso card on STM32MP15 kit 

1. **Flash raw image**
   1. **Create raw image for sdcard**

**Install starter package on Linux machine**

**Create directory /local/STM32MP15-Ecosystem-v1.0.0/Starter-Package**

**mkdir /local**

**chmod 7777 /local**

**mkdir /local/STM32MP15-Ecosystem-v1.0.0**

**mkdir /local/STM32MP15-Ecosystem-v1.0.0/Starter-Package**

**starter\_package\_path = /local/STM32MP15-Ecosystem-v1.0.0/Starter-Package**/**stm32mp1-openstlinux-4.19-thud-mp1-19-02-20/**

**Unzip starter package images**

**cd /local/STM32MP15-Ecosystem-v1.0.0/Starter-Package**

**tar -xvf FLASH-stm32mp1-openstlinux-4.19-thud-mp1-19-02-20.tar.xz**

**from starter package**

**<starter\_package\_path>/images/stm32mp1/scripts**

**/local/STM32MP15-Ecosystem-v1.0.0/Starter-Package**/**stm32mp1-openstlinux-4.19-thud-mp1-19-02-20/images/stm32mp1/scripts**

**or in distribution build**

**<distri\_package\_path>/build-openstlinuxweston-stm32mp1-boards-revb/tmp-glibc/deploy/images/stm32mp1-boards-revb/script**

**Run the script create\_sdcard\_from\_flashlayout.sh**

|  |
| --- |
| **sudo ./create\_sdcard\_from\_flashlayout.sh ../flashlayout\_st-image-weston/FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.tsv** |
|  |

**The .raw file image is created under <starter\_package\_path>/images/stm32mp1/**

**See also**

[**https://wiki.st.com/stm32mpu/index.php/How\_to\_populate\_a\_microSD\_card\_with\_a\_script**](http://intranet.lme.st.com:8000/php-bin/ug_mcdmpu/index.php/How_to_populate_a_microSD_card_with_a_script)

**Mass market raw image for disco revC is shared under: \\PRGCWD0579.prg.st.com\TOMAS\_Share\MPU\STM32MP1\_STFAE\_Part2\stm32mp1-session2-Linux introduction & tool set-up\** **FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.raw.zip**

**2.1 Flash raw image on sdcard with windows tools (fastest method)**

The raw image contains all the partitions for SD card boot device.

Look on the wiki to build a raw image from the partitions.

FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.raw for the discovery board revC

**On window10 method**

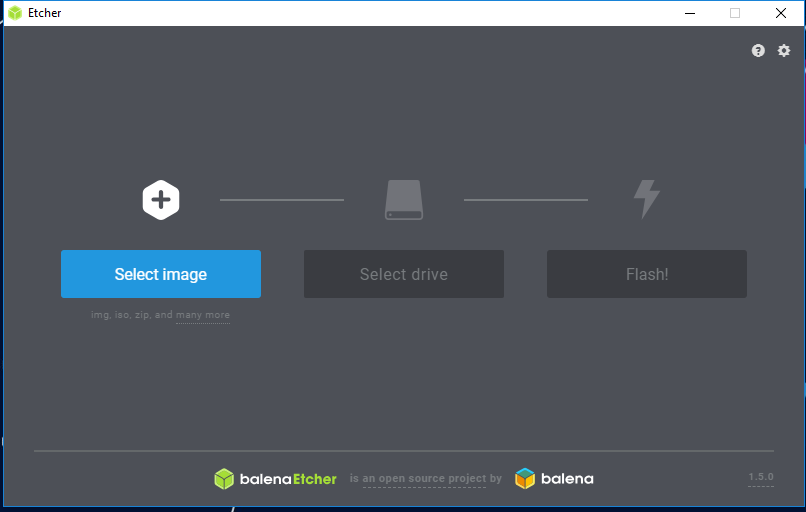
**Use Etcher** (for raspberry pie, download it from web)

1/ Plug your sdcard **directly to your PC**

2/ Get the raw image: FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.raw

(shared on \\PRGCWD0579.prg.st.com\TOMAS\_Share\MPU\STM32MP1\_STFAE\_Part2\stm32mp1-session2-Linux introduction & tool set-up)

3/Use Etcher to write the .raw image file on the scdcard, it will ask to format the scd card in fat32, no issue.

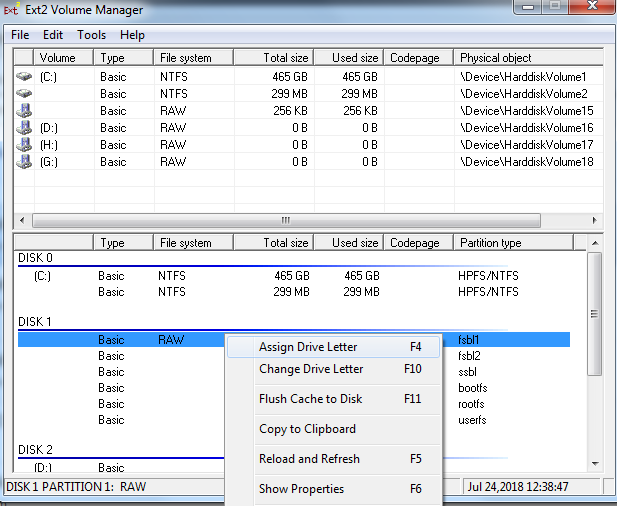


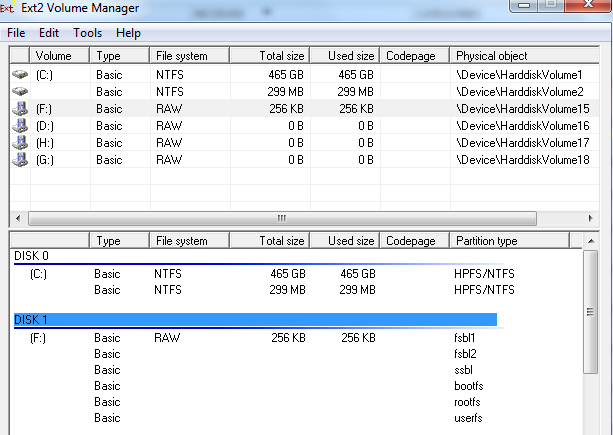
**Window 7 method**

Plug a external USB sdcard reader in windows PC

Install Ext2Fsd-0.69.exe

Then with Ext2Fsdand assign the sdcard drive to a letter for instance f:\



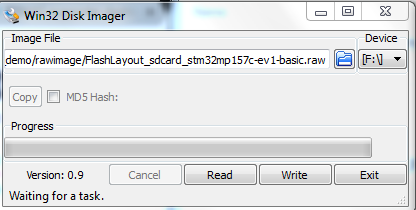


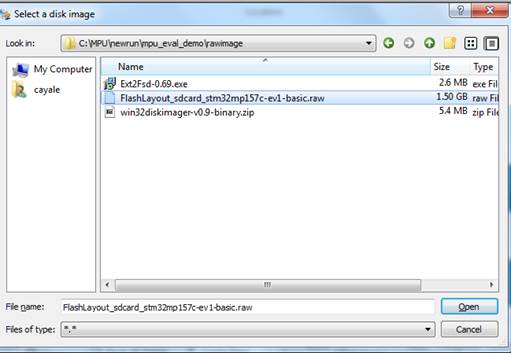
Win32diskmanager tool

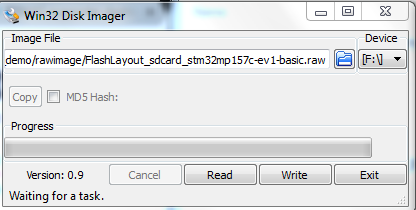
Unzip win32diskimager-v0.9-binary.zip

Launch Win32DiskImager.exe and select sdcard drive (for instance f:\) as device

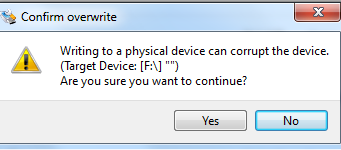
And select the raw image file



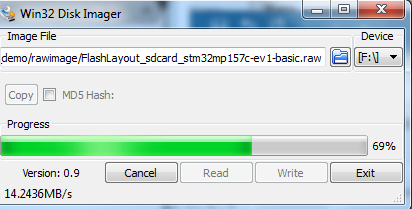


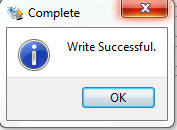


Then Write



Then Yes





Then sdcard is programmed with the image

**2.2Flash raw image on SDcard from linux host (fast flashing method)**

Look for article in wiki search: [How to populate a microSD card with a script]

The raw image contains all the partitions for SD card boot device.

Look on the wiki to build a raw image from the partitions.

**1/ Plug on Linux host a USB SDcard reader (with the SDcard inserted)**

**2/** **follow:**

|  |
| --- |
| export **FLASH\_IMAGE\_DIR**= **/local/STM32MP15-Ecosystem-v1.0.0/Starter-Package**/**stm32mp1-openstlinux-4.19-thud-mp1-19-02-20/images/stm32mp1**  cd $**FLASH\_IMAGE\_DIR**    **export RAW\_IMAGE=**flashlayout\_st-image-weston\_FlashLayout\_sdcard\_stm32mp157c-dk2-trusted.raw    lsblk    Example in sdc    Machine generated alternative text: )sboxes@osboxes : —S  Il:o  8:34  8:37  8:35  8:33  8:38  8:36  MAJ:MIN RM  SIZE  49K  1024M  7.5G  256K  750 .4M  2.5M  256K  718 . 6M  64M  200G  93. IG  4. 76  4. 76  97.6G  TYPE  disk  O rom  disk  part  part  part  part  part  part  disk  part  part  part  part  MOUNTPOINT  /media/osboxes/DIS MB1272  /media/osboxes/rootfs  /media/osboxes/userfs  /media/osboxes/bootfsl  / home  / boot  L SWAP J  •AME  sdc2  sdc5  sdc3  sdcl  sdc6  sdc4  sda4  sda2  sda3  sdal  8:16  8:32  1  1  1  1  1  1  1  1  1    umount /media/osboxes/rootfs  umount /media/osboxes/userfs  umount /media/osboxes/bootfs1    sudo dd if=$**RAW\_IMAGE** of=/dev/sdc bs=1M conv=fsync status=progress |