

Boot From an SD Card / USB Stick

Introduction

Booting your system from an external storage device such as an SD card or a USB stick might be useful for some use cases.

1. For offline and/or isolated systems that have no other means of installing or upgrading their system.
2. For resource-scarce systems or systems that do not provide internal storage.
3. For development and testing environments which lack a more efficient setup (see [Boot from a TFTP/NFS Server](#)).

For (1), Toradex provides the [Toradex Easy Installer](#) as well as certain [Recovery Modes](#), whereas (2) is commonly addressed with the NAND-based inter all modules are equipped with. Note that removable storage devices may still be used to provide images to be installed into an already running system ([Toradex Easy Installer](#)) and as additional storage. This is not considered *booting* from an external device. Nonetheless, booting from these media might be desirable in some development and testing environments or in exceptional cases. This article describes how to proceed.

CAUTION

It is recommended to use SD booting only during development also because normal SD cards aren't reliable for long-term usage. For a simple installation of bootable images please use our [Toradex Easy Installer](#) where available.

Requirements

In order to boot from an SD card (hereafter also valid for a USB stick) you need:

- U-Boot running on your module. If this is not the case, please use the [Toradex Easy Installer](#) to install a running image first. Alternatively, if not (yet) possible on your particular module revert to using the Legacy Update Procedure.
- An SD card with two partitions. The first holding a FAT filesystem to host the kernel image and device trees, the second holding an ext4 (ext3 on older modules) or ext4 (ext3 on older modules) filesystem to hold the rootfs.

Procedure

There are two ways to prepare the SD card:

- [Using the images provided by Toradex.](#)
- [Using the wic files as generated by an Yocto Project build.](#)

The second option is faster, but it requires that you build the images first using the OpenEmbedded build system. Use it only if you are actually modifying the images.

Using Toradex Images

Use your preferred partitioning tool to create the mentioned partitions on your SD card or follow the steps below.

INFO

The size will depend on your module and images, but 100-500MB for the boot partition, and 1-2GB for the rootfs should be sufficient in most cases. The partitions is important since U-boot will look for the kernel and device trees in the first, and for the rootfs in the second one.

1. Create the boot partition as a primary partition formatted as `fat32`, with 100MB of space, and the rootfs partition formatted as `ext4`, with 2GB of space. If you are connecting an external drive, it could be an SD card or an MMC block device. So keep in mind to partition the right device, for example, `/dev/sda` or `/dev/mmcblk(n)` (where `(n)` stands for the number of your mmc device), if using mmc ones. You can check by looking at the output of `lsblk` after plugging the SD to your computer or using the `lsblk` command.

```
$ sudo parted -s /dev/sd(x) mklabel gpt mkpart primary fat32 1MiB 100MiB mkpart ext4 100MiB 2000MiB name 1 boot name 2 rootfs
$ sudo mkfs.fat -n boot /dev/sd(x)1 && sudo mkfs.ext4 -L rootfs /dev/sd(x)2
$ sudo lsblk /dev/sd(x) -o NAME,FSTYPE,LABEL,PARTLABEL #(to verify)
```

We labeled the partitions *boot* and *rootfs*, but you can name them at will.

2. Mount the partitions afterwards to be able to access them

```
$ sudo mount -L boot /mnt/boot; sudo mount -L rootfs /mnt/rootfs
```

3. **Download** an image for the Toradex Easy Installer corresponding to your module and extract the kernel image (`zImage`) and the device trees (`*.dtb`) into the first partition, and the rootfs into the second. This example complies the Colibri iMX6, the procedure is similar for other Toradex SoMs.

```
$ sudo tar xf Reference-Minimal-Image-upstream-colibri-imx6.bootfs.tar.xz --no-same-owner -C /mnt/boot
$ sudo tar xf Reference-Minimal-Image-upstream-colibri-imx6.tar.xz -C /mnt/rootfs
```

4. Unmount the partitions

```
$ sudo umount /mnt/{boot,rootfs}
```

5. Insert the SD card into (one of) the carrier board slot(s) and boot your module entering U-boot (by hitting any key in the early boot screen).
6. (Optional) At the U-boot prompt you can check that your partitions are recognized. You should see the names you gave them during the partitioning example, for mmc devices or usb sticks:

```
> mmc dev 1; mmc part
> usb dev 0; usb part
```

7. You can now boot using **one of the following commands**, depending on the media you are using:

```
> run bootcmd_mmc0 # For eMMC
> run bootcmd_mmc1 # SD card
> run bootcmd_usb0 # USB Flash Drive
> run bootcmd_dhcp # TFTP/NFS
```

! INFO

For more information about boot commands on U-Boot, please refer to [Distro Boot](#).

Using wic Files

The OpenEmbedded build system supports the generation of **wic images** that can be directly burned to an SD card. By default, this feature is enabled in since release 2.8b4 (on older releases only NXP i.MX modules support it). Thus, if you are building your own images from the Toradex repositories, you


corresponding wic image after a build in the `deploy/images/name-of-the-board/` folder of your OpenEmbedded tree. The image is compressed a `xxx.wic.gz` (e.g. `Console-image-colibri-imx6.wic.gz`). Decompress it first to get the original wic image and burn it directly to your SD card using e.g. t

```
$ gzip -d xxx.wic.gz  
  
$ sudo dd if=xxx.wic of=/dev/sd-card-device bs=4M
```

Or in one go:

```
$ zcat xxx.wic.gz | sudo dd of=/dev/sd-card-device bs=4M
```

where `sd-card-device` is the name of the Linux device exposing the SD card (you can find it using `lsblk` or looking at the output of `dmesg` after card). Move the SD card to your carrier board slot, turn on your module and quickly trigger the U-boot console (by pressing any key within the first 2 seconds there, you can boot from the external media by executing in console `run bootcmd_mmc1` or `run bootcmd_usb0` depending on the media you are using).

 [Edit this page](#)