Application Note #4-10-10-A

ConnectCore™ for i.MX51

U-Boot: How to boot an OS image from memory card

NOTE: The capabilities outlined in this document require U-Boot version *1.0-3P* (or greater). Please also visit www.digi.com/support for most recent software and documentation updates.

Refer to the U-Boot Reference Manual for detailed information on how to use U-Boot.

The U-Boot direct boot command `dboot` can be used to initiate the OS image boot process from an external SD/MMC memory card. The command has the following general syntax:

```
dboot <os> <media> [<dev>[:partition] <filesystem>] <bootfilename> [opt]
```

- Values for parameter **`<os>`** are **`linux`**, **`wce`**, or **`netos`**
- Values for parameter **`<media>`** are **`flash`**, **`tftp`**, **`nfs`**, or **`mmc`**
- **`dev[:partition]`** is the device index (starting at 0) and the corresponding partition number (starting at 1). The default values are device **0** and partition **1**.

  The device numbers associated with the MicroSD/SD connector on the development board are

  0 – MicroSD socket

  1 – SD/MMC socket
- Values for parameter **`<filesystem>`** are **`fat`**, **`vfat`**, **`ext2`**, or **`ext3`**, and must match the file system of the partition containing the boot image. The default value is **`FAT`**.
- **`<bootfilename>`** is the name of the boot image file

  If the parameter is not provided, the default file name is based on the value specified in the U-Boot environment variables `kimg` (Linux) or `wimg` (Windows Embedded CE).
- **`[opt]`** specifies boot options

  The currently available option is **`cleanhive`**, which applies to Windows Embedded CE and initiates a registry cleaning process after boot-up.

On the following pages you will find examples showing how to use the `dboot` command to boot your OS image from a memory card.

In order to access the U-Boot command prompt (`#`), you need to interrupt the module's boot process by pressing a key on the serial console connected to UART2 [X27] on the development board.

Example #1: Manual OS boot from SD card

This example shows how to *manually* boot an OS image from partition 0 on the memory card inserted in the SD/MMC card socket (device 1) of the development board.

It is assumed that the memory card's first partition is formatted with a FAT file system.

- Windows Embedded CE

  ```
  # dboot wce mmc 1:1 fat
  ```

or

- Linux

  ```
  # dboot linux mmc 1:1 fat
  ```

  Please note that in order to boot Linux from the memory card, it is also necessary to store a valid root file system in a separate partition of the card.

Example #2: Automatic OS boot from SD card

This example shows how to configure an *automatic boot* of an OS image from partition 0 on the memory card inserted in the MicroSD card socket (device 0) of the development board.

- Windows Embedded CE

  ```
  # setenv bootcmd dboot wce mmc 0:1 fat
  # saveenv
  ```

or

- Linux

  ```
  # setenv bootcmd dboot linux mmc 0:1 fat
  # saveenv
  ```

Example #3: User-selected OS boot from memory card

This example shows how to initiate booting an OS image from partition 0 on the memory card inserted in the SD/MMC card socket (device 1) *by pressing User Button 2* [S5] on the development board at the start of the boot process. If no button is pressed, the image stored in the on-module flash will be booted instead.

- Windows Embedded CE

  ```
  # setenv key2 dboot wce mmc 1:1 fat
  # saveenv
  ```

or

- Linux

  ```
  # setenv key2 dboot linux mmc 1:1 fat
  # saveenv
  ```

By using the U-Boot environment variables `key1` or `key2`, you can assign a specific U-Boot command to User Button 1 [S3] and User Button 2 [S5].

Please note that the user buttons are active during the start phase of the boot process. Restart the module by pushing the reset button once, while keeping the selected user button pressed until you have verified that the corresponding boot action has been initiated.

Tip: Slow memory cards

Certain slow memory cards may require special timing requirements to work properly. Without impacting the performance of faster devices, a U-Boot environment variable can be configured to slow down the interface and enable support for those memory cards.

Enter the following U-Boot commands to enable the `slowmmc` mode:

```
# setenv slowmmc yes
# saveenv
```

Document Change Log

Revision	Date	Comments
A	4-10-2010	Initial