Approved version. Approved on: 12:18, 26 July 2023

How to boot the kernel via TFTP from U-Boot

Applicable for STM32MP13x lines, STM32MP15x lines

This page explains how to boot the Linux® kernel from U-Boot through a TFTP server installed on a host PC, based on the `pxe` U-Boot command.

Contents↑ [hide]

1. Documentation↑

The U-Boot documentation is available in:

- doc/README.pxe
- Generic Distro Configuration Concept page or doc/develop/distro.rst

2. TFTP server installation on host PC↑

The procedure, to install a TFTP server on your PC host, is explained here [1].

3. TFTP server configuration↑

U-Boot follows pxelinux's rules to download binaries from the TFTP server.

See more details here[2]

U-Boot `pxe` command loads and parses a PXE configuration file found in the pxelinux.cfg directory exported by this TFTP server (for example in /srv/tftp/pxelinux.cfg/). This

kernel.

3.1. PXE configuration file↑

Each PXE configuration file has same format as the **extlinux.conf** file, generated by a standard Yocto distribution.

The format of the PXE configuration text file is explained in this chapter. More explanations on the name of the file loaded by U-Boot are given in next chapter.

For each element of the menu :

- the **LABEL** is a string, for example the boot device such as an e•MMC, SD card, NAND, and so on.
- the kernel and the device tree loaded by U-Boot are defined via **KERNEL** and **FDT** options.
- the Linux® kernel boot arguments are defined via the **APPEND** option, including the rootfs to use;

For example with STM32MP157F-EV1 Evaluation board and with the rootfs on the **8th** partition of the SD card (**/dev/mmcblk0p8**), the PXE configuration file is:

```
menu title Select the boot mode
DEFAULT sdcard
LABEL sdcard
        KERNEL uImage
        FDT stm32mp157f-ev1.dtb
        APPEND root=/dev/mmcblk0p8 rootwait rw earlyprintk console=ttySTM0,115200
```

To fully boot on a network, the root filesystem (rootfs), the files required by the kernel image, can be defined :

- as initial RAM disk with **INITRD** command in configuration file (this initramfs file **uInitrd** need to be generated, it is a generic feature not explained here).
- or as NFS mounting point with Linux® kernel boot arguments **nfsroot** added in **APPEND** command (**root**=**/dev/nfs nfsroot**=...).

In the configuration file, the menu can have several options (see **LABEL**, **DEFAULT** and **TIMEOUT** commands in README.pxe for details).

See Kernel Documentation for details on:

- the initial RAM disk (initrd)
- the kernel's command-line parameters (bootargs)
- the root filesystem via NFS (nfsroot)

For example with STM32MP157C-EV1 Evaluation board ⚠:

```
menu title Select the boot mode
DEFAULT initramfs
TIMEOUT 20
LABEL initramfs
        KERNEL uImage
        FDT stm32mp157c-ev1.dtb
        INITRD uInitrd
        APPEND rootwait rw earlyprintk console=ttySTM0,115200
LABEL nfs
        KERNEL uImage
        FDT stm32mp157c-ev1.dtb
        APPEND root=/dev/nfs nfsroot=192.168.1.1:/nfsroot,nfsvers=4 nfsrootdebug ip=dhcp rootwait rw earlyprintk console=ttyS
LABEL sdcard
        KERNEL uImage
        FDT stm32mp157c-ev1.dtb
        APPEND root=/dev/mmcblk0p8 rootwait rw earlyprintk console=ttySTM0,115200
```

In this example, the label **nfs** expects that the the Linux kernel running on the device can found on the network:

- a DHCP server with bootargs `ip=dhcp`, to get a dynamic IP address
- a NFS server v4[3] at 192.168.1.1 (for example your PC) with bootargs "**nfsroot**=**192.168.1.1:/nfsroot**,nfsvers=4" and the roots is exported by this NFS server in /nfsroot in the exported directory (for example /srv/nfs/nfsroot).

3.2. pxelinux.cfg directory↑

Because more than one board may be booted from the same server, the "PXE configuration file" name depends on U-Boot parameters (hardware address of the board/IP ad

U-Boot searches for a PXE configuration file in the pxelinux.cfg directory in this order:

1. UUID comes from "pxeuuid" variable
2. hardware type and address: ARP type "1" for Ethernet and **MAC address**
3. **IP address** of the board and each subnet mask
4. at the end, the file named **"default-$CONFIG_SYS_ARCH-$CONFIG_SYS_SOC"**

You need to provide at least one configuration file on the TFTP server per board.

Each file of the pxelinux.cfg directory can have the same content (or can be a symbolic link to a default one).

For example with STM32MP1 series, when:

- UUID is not defined
- ethaddr=00:80:e1:01:2d:6f
- IP =10.48.0.141 (hexa coding=0A30008D)
- CONFIG_SYS_ARCH=arm and CONFIG_SYS_SOC="stm32mp"

U-Boot searches the configuration file is this order:

1. **"01-00-80-e1-01-2d-6f"**
2. **0A30008D** then subnet **0A30008**, **0A3000**, **0A300** ...

3. **default-arm-stm32mp**

With ethaddr=00:80:e1:01:2d:6f, the exported directory on the TFTP server (for example /tftpboot or /srv/tftp) may look like:

```
├── pxelinux.cfg                directory
│   ├── 01-00-80-e1-01-2d-6f    configuration file for MAC address 00:80:e1:01:2d:6f
├── stm32mp157c-ev1.dtb         device tree
├── uImage                      kernel
```

The content of the "PXE configuration" text file named "01-00-80-e1-01-2d-6f" is described in the previous chapter.

An other example with

- different boards (identified by a MAC address)
  each pxe file selects a different device tree (ed1, ev1, dk1, dk2) or kernel file
- fallback with the IP address submask (symbolic link to default-arm-stm32mp)
- default-arm-stm32mp (final fallback file, name based on CONFIG_SYS_SOC)

```
├── pxelinux.cfg
│   ├── 01-00-80-e1-01-2d-6f            configuration file for MAC address 00:80:e1:01:2d:6f
│   ├── 01-00-80-e1-42-42-8c            configuration file for MAC address 00:80:e1:42:42:8c
│   ├── 01-00-80-e1-42-42-cd            configuration file for MAC address 00:80:e1:42:42:cd
│   ├── 01-00-80-e1-42-46-76            configuration file for MAC address 00:80:e1:42:46:6f
│   ├── 0A300 -> default-arm-stm32mp    configuration file for IP address 10.48.0*.**
│   ├── 0A30001 -> default-arm-stm32mp  configuration file for IP address 10.48.00.1*
│   ├── 0A30004 -> default-arm-stm32mp  configuration file for IP address 10.48.00.4*
│   ├── 0A30008 -> default-arm-stm32mp  configuration file for IP address 10.48.00.8*
│   └── default-arm-stm32mp             default configuration file
├── stm32mp157c-ed1.dtb
├── stm32mp157c-ev1.dtb
├── stm32mp157a-dk1.dtb
├── stm32mp157c-dk2.dtb
├── uImage
├── uInitrd
```

4. Board command in U-Boot console↑

You can execute the DISTRO bootcmd for PXE:

```
STM32MP>  env set serverip <the PC address>
STM32MP>  run bootcmd_pxe
```

5. Step by step example↑

5.1. TFTP boot, roofs on SD-Card↑

1. STM32MP157C-EV1 board with MAC address: ethaddr=00:80:e1:01:60:da
2. DHCP server available on network
3. **rootfs** available on SD-Card (/dev/mmcblk0p6)
4. Install and configure a TFTP server:
   - TFTP server address is 10.201.21.107
   - the content of directory exported by this TFTP server (for example /srv/tftp) is:

```
├── pxelinux.cfg
│   ├── 01-00-80-e1-01-60-da
├── stm32mp157c-ev1.dtb
├── uImage
```

With the text file **01-00-80-e1-01-60-da** (#PXE configuration file in #pxelinux.cfg directory) :

```
menu title Select the boot mode
DEFAULT sdcard
LABEL sdcard
        KERNEL uImage
        FDT stm32mp157c-ev1.dtb
        APPEND root=/dev/mmcblk0p6 rootwait rw earlyprintk console=ttySTM0,115200
```

In the U-Boot console:

```
 STM32MP>  env set serverip 10.201.21.107                          set the server address
 STM32MP>  run bootcmd_pxe                                          execute the DISTRO command
 BOOTP broadcast 1
 DHCP client bound to address 10.48.1.229 (19 ms)             IP address provided by DHCP server
 missing environment variable: pxeuuid
 missing environment variable: bootfile
 Retrieving file: pxelinux.cfg/01-00-80-e1-01-60-da
 Using ethernet@5800a000 device
 TFTP from server 10.201.21.107; our IP address is 10.48.1.229; sending through gateway 10.48.3.254
 Filename 'pxelinux.cfg/01-00-80-e1-01-60-da'.                     load configuration file
 Load address: 0xc4200000
 Loading: #
         105.5 KiB/s
 done
 Bytes transferred = 435 (1b3 hex)
 Config file found
```

5.2. TFTP boot, roofs on NFS, no DHCP↑

1. STM32MP157C-EV1 board with MAC address: ethaddr=00:80:e1:01:60:da
2. local network without DHCP server with
   - your device = IP address 192.168.7.1
   - your PC = IP address 192.168.7.2, with 2 server correctly configured:
     - NFS server which export **rootfs**, for example st-image-weston-openstlinux-weston-stm32mp1.tar is extracted in /srv/nfs/nfsroot.
     - TFTP server, same directory than previous example
       with the text file **01-00-80-e1-01-60-da** (#PXE configuration file in #pxelinux.cfg directory) :

```
menu title Select the boot mode
DEFAULT nfs
LABEL nfs
        KERNEL uImage
        FDT stm32mp157c-ev1.dtb
        APPEND root=/dev/nfs  nfsroot=192.168.7.2:/srv/nfs/nfsroot,nfsvers=4 nfsrootdebug ip=192.168.7.1:192.168.7.2::::eth0:
```

In the U-Boot console, use the `pxe` directly, because `bootcmd_pxe` use `dhcp` :

```
 STM32MP>  setenv ipaddr  192.168.7.1
 STM32MP>  env set serverip 192.168.7.2
 STM32MP>  pxe get
 STM32MP>  pxe boot
```

6. References↑

1. ↑ https://importgeek.wordpress.com/2013/09/04/install-configure-and-test-tftp-server-in-ubuntu/
2. ↑ https://wiki.syslinux.org/wiki/index.php?title=PXELINUX
3. ↑ https://elinux.org/TFTP_Boot_and_NFS_Root_Filesystems

How to run use cases STM32MP15x STM32MP13x