



STM32MP15 U-Boot



Contents



A quality version of this page, approved on 11 June 2021, was based off this revision.

This article briefly describes the STM32MP15x lines  support in U-Boot.

Contents

1 STM32MP15x lines support	4
2 Selecting targets : choose defconfig and Device Tree	5
3 Compilation	6
4 U-Boot integration in FIP	7
5 SPL compilation	8
6 Examples	9
6.1 STM32MP157F-EV1 Evaluation board	9
6.2 STM32MP157C-EV1 Evaluation board	9
6.3 STM32MP157D-DK1 Discovery kit	9
6.4 STM32MP157C-DK2 Discovery kit	9
6.5 Custom board with external device tree	9
6.6 SPL for STM32MP157F-EV1 Evaluation board	9




1 STM32MP15x lines support

For detailed information, please read the file in the delivered U-Boot code = `doc/board/st/stm32mp1.rst` or the `stm32mp1` documentation generated by `make htmldocs[1]`.

Also available in official U-Boot Git.

Code :

- `arch/arm/mach-stm32mp` : arch specific code for STM32 Arm[®]Cortex[®] MPUs 
- `board/st/common` : STMicroelectronics common code
- `board/st/stm32mp1` : generic STMicroelectronics board for STM32MP1 Series
- `drivers/*/*stm32*` : drivers

Configuration Files:









- defconfig file
 - `configs/stm32mp15_trusted_defconfig`
 - `configs/stm32mp15_basic_defconfig`
- config files
 - `include/configs/stm32mp1.h`
- `STM32MP15_device_tree` in `arch/arm/dts` : `stm32mp15*.dts*`
 - `<Device tree>.dts` : same as kernel
 - `<Device tree>-u-boot.dtsi` : addition for u-boot automatically included in build process



2 Selecting targets : choose defconfig and Device Tree

The STM32 MPU boot chain is supported by one U-Boot configuration with defconfig = `stm32mp15_trusted_defconfig`

Several boards are supported with the associated device-tree (same name as kernel):

Board part number	Device tree	Description
STM32MP157A-EV1 Evaluation board 	stm32mp157a-ev1	MB1262+MB1263
STM32MP157C-EV1 Evaluation board 	stm32mp157c-ev1	
STM32MP157D-EV1 Evaluation board 	stm32mp157d-ev1	
STM32MP157F-EV1 Evaluation board 	stm32mp157f-ev1	
STM32MP157A-DK1 Discovery kit 	stm32mp157a-dk1	MB1272
STM32MP157D-DK1 Discovery kit 	stm32mp157d-dk1	
STM32MP157C-DK2 Discovery kit 	stm32mp157c-dk2	MB1272+MB1407
STM32MP157F-DK2 Discovery kit 	stm32mp157f-dk2	

For information the 'Evaluation' daughter-board only (MB1263 without MB1262) is also supported by `stm32mp15*-ed1.dts` device tree files.



3 Compilation

see U-Boot_overview#U-Boot_build

With the defconfig file: `stm32mp15_trusted_defconfig`

```
PC $> make stm32mp15_trusted_defconfig
PC $> make DEVICE_TREE=<Device tree> all
```


The supported variables are:

- **DEVICE_TREE**: select in arch/arm/dts the device tree that is used
- **KBUILD_OUTPUT**: change the destination directory for the build
- **EXT_DTB**: select external device tree

The output files `u-boot.dtb` and `u-boot-nodtb.bin` are integrated in FIP.

Nota: All the compiled device tree are available in `$KBUILD_OUTPUT/arch/arm/dts/*.dtb`.

You can select them instead of `u-boot.dtb` without U-Boot recompilation.

The output file is `u-boot.stm32` for ecosystem release v2.1.0 or if you activate `CONFIG_STM32MP15x_STM32IMAGE` in your defconfig for ecosystem release v3.0.0  (this temporary option is only introduced to facilitate the FIP migration but it will be removed in the next EcosystemRelease).

With FIP support for ecosystem release v3.0.0 , the offset on U-Boot environment in NOR are defined in the defconfig with:

```
CONFIG_ENV_OFFSET=0x480000
CONFIG_ENV_OFFSET_REDUND=0x4C0000
```

Without FIP support these MTD partition offsets in NOR for default flashlayout change and to reuse the ecosystem release v2.1.0 flashlayout the defconfig options become:

```
CONFIG_STM32MP15x_STM32IMAGE=y
CONFIG_ENV_OFFSET=0x280000
CONFIG_ENV_OFFSET_REDUND=0x2C0000
```



4 U-Boot integration in FIP

U-Boot binary and its associated device tree are part of the FIP binary, created with TF-A Makefile option :

- **BL33_CFG** = **u-boot.dtb**
- **BL33** = **u-boot-nodtb.bin**

or updated with fiptools:

- **--hw-config** **u-boot.dtb**
- **--nt-fw** **u-boot-nodtb.bin**

Example for fip update of STM32MP157C-EV1 Evaluation board :

```
PC $> fiptool update --verbose --nt-fw u-boot-nodtb.bin \  
    --hw-config u-boot.dtb \  
    tf-a-fip-stm32mp157c-ev1.bin
```

```
DEBUG: Replacing nt-fw with.../u-boot-nodtb.bin  
DEBUG: Replacing hw-config with .../u-boot.dtb  
DEBUG: Metadata size: ... bytes  
DEBUG: Payload size: ... bytes
```



5 SPL compilation

The alternate boot chain with SPL is also supported by "*stm32mp15_basic_defconfig*" but only for U-Boot_SPL:
_DDR_interactive_mode

With the defconfig file: *stm32mp15_basic_defconfig*

```
PC $> make stm32mp15_basic_defconfig
PC $> make DEVICE_TREE=<Device tree> all
```

The resulting U-Boot files are located in your build directory:

- **u-boot-spl.stm32** : FSBL = SPL binary with STM32 image header, loaded by ROM code
- **u-boot.img** : SSBL = U-Boot binary with U-Boot image header (uImage), loaded by SPL



This alternate boot chain with SPL is not supported/promoted by STMicroelectronics to make product.



6 Examples

6.1 STM32MP157F-EV1 Evaluation board i

```
PC $> make stm32mp15_trusted_defconfig
PC $> make DEVICE_TREE=stm32mp157f-ev1 all
```

6.2 STM32MP157C-EV1 Evaluation board i

```
PC $> make stm32mp15_trusted_defconfig
PC $> make DEVICE_TREE=stm32mp157c-ev1 all
```

6.3 STM32MP157D-DK1 Discovery kit i

```
PC $> make stm32mp15_trusted_defconfig
PC $> make DEVICE_TREE=stm32mp157d-dk1 all
```

6.4 STM32MP157C-DK2 Discovery kit i

Using export to select the device tree

```
PC $> export KBUILD_OUTPUT=./build/stm32mp15_trusted
PC $> export DEVICE_TREE=stm32mp157c-dk2
PC $> make stm32mp15_trusted_defconfig
PC $> make all
```

6.5 Custom board with external device tree

```
PC $> make stm32mp15_trusted_defconfig
PC $> make EXT_DTB=stm32mp151a-myboard.dtb all
```

6.6 SPL for STM32MP157F-EV1 Evaluation board i

```
PC $> make stm32mp15_basic_defconfig
PC $> make DEVICE_TREE=stm32mp157f-ev1 all
```

- <https://u-boot.readthedocs.io/en/stable/board/st/stm32mp1.html>

Das U-Boot -- the Universal Boot Loader (see [U-Boot_overview](#))



Arm[®] is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere. **arm**

Cortex[®]

Device Tree Binary (or Blob)

Firmware Image Package is a packaging format used by TF-A

Trusted Firmware for Arm[®] Cortex[®]-A

Boot Loader stage 3-3

Secondary Program Loader, *Also known as **U-Boot SPL***

First Stage Boot Loader

Read Only Memory

Second Stage Boot Loader