



CRYP device tree configuration



Contents



A quality version of this page, approved on 13 May 2020, was based off this revision.

Contents

1 Article purpose	4
2 DT bindings documentation	5
3 DT configuration	6
3.1 DT configuration (STM32 level)	6
3.2 DT configuration (board level)	6
3.3 DT configuration examples	6
4 How to configure the DT using STM32CubeMX	7
5 References	8



1 Article purpose

The purpose of this article is to explain how to configure the *CRYP*^[1] **when the peripheral is assigned to Linux**[®]OS.

The configuration is performed using the **device tree mechanism**^[2].

The *Device tree* provides a hardware description of the *CRYP*^[1], used by *STM32 CRYP Linux driver*.

If the peripheral is assigned to another execution context, refer to *How to assign an internal peripheral to a runtime context* article for guidelines on peripheral assignment and configuration.



2 DT bindings documentation

The *CRYP*^[1] is represented by the *STM32 CRYP device tree bindings*^[3].



3 DT configuration

This hardware description is a combination of STM32 microprocessor and board device tree files. See [Device tree](#) for explanations on device tree file split.

The **STM32CubeMX** can be used to generate the board device tree. Refer to [#How_to_configure_the_DT_using_STM32CubeMX](#) for more details.

3.1 DT configuration (STM32 level)

The CRYP node is declared in [stm32mp15xc.dtsi](#)^[4] and [stm32mp15xf.dtsi](#)^[5]. It provides the hardware registers base address, clock, interrupt and reset.

```

cryp1: cryp@54001000 {
    compatible = "st,stm32mp1-cryp";
    reg = <0x54001000 0x400>;
    interrupts = <GIC_SPI 79 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&scmi0_clk CK_SCMI0_CRYP1>;
    resets = <&scmi0_reset RST_SCMI0_CRYP1>;
    status = "disabled";
};

```

Warning

This device tree part is related to STM32 microprocessors. It should be kept as is, without being modified by the end-user.

3.2 DT configuration (board level)

This part is used to enable the CRYP used on a board. This is done by setting the **status** property to **okay**.

3.3 DT configuration examples

```

&cryp1 {
    status = "okay";
};

```



4 How to configure the DT using STM32CubeMX

The STM32CubeMX tool can be used to configure the STM32MPU device and get the corresponding platform configuration device tree files.

The STM32CubeMX may not support all the properties described in the above DT bindings documentation paragraph. If so, the tool inserts **user sections** in the generated device tree. These sections can then be edited to add some properties and they are preserved from one generation to another. Refer to STM32CubeMX user manual for further information.



5 References

Please refer to the following links for additional information:

- [1.01.11.2 CRYP internal peripheral](#)
- [Device tree](#)
- [Documentation/devicetree/bindings/crypto/st,stm32-cryp.txt](#)
- [STM32MP157C device tree file](#)
- [STM32MP157F device tree file](#)

Cryptographic processor

Linux[®] is a registered trademark of Linus Torvalds.

Operating System

Device Tree

Generic Interrupt Controller

Serial Peripheral Interface