



MDMA device tree configuration



Contents

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



1 Article purpose

This article explains how to configure the MDMA internal peripheral when it is assigned to the Linux[®]OS. In that case, it is controlled by the [Dmaengine overview](#).

The configuration is performed using the [Device tree](#) mechanism that provides a hardware description of the MDMA internal peripheral, used by the STM32 MDMA Linux driver and by the [DMA framework](#).

Hardware description is a combination of:

- STM32 MDMA peripheral
- and STM32 MDMA client



2 DT bindings documentation

Complete device tree bindings can be found at this location: ^[1].



3 DT configuration

This hardware description is a combination of the **STM32 microprocessor** device tree files (*.dtsi* extension) and **board** device tree files (*.dts* extension). See the [Device tree](#) for an explanation of the device tree file split.

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)

At device level, MDMA is declared as follows:

```
mdma1: dma@58000000 {
    compatible = "st,stm32h7-mdma";
    reg = <0x58000000 0x1000>;
    interrupts = <GIC_SPI 122 IRQ_TYPE_LEVEL_HIGH>;
    clocks = <&rcc MDMA>;
    resets = <&scmi0_reset RST_SCMI0_MDMA>;
    #dma-cells = <6>;
    dma-channels = <32>;
    dma-requests = <48>;
};
```

The DTS file is located under `arch/arm/boot/dts/stm32mp151.dtsi`.



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

3.2 DT configuration (board level)

No board device tree configuration is required.

The whole configuration remains at STM32 level.



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:

- [Documentation/devicetree/bindings/dma/stm32-mdma.txt](#)

Operating System

Device Tree

Generic Interrupt Controller

Serial Peripheral Interface

Device Tree Source (in software context) or Digital Temperature Sensor (in peripheral context)