IWDG device tree configuration

Contents

Stable: 11.06.2020 - 09:36 / Revision: 11.06.2020 - 07:00

Stable: 11.06.2020 - 00.00 / Revision: 11.06.2020 - 01.00

Contents

1 Article purpose

This article explains how to configure the **IWDG** internal peripheral when it is assigned to the Linux®OS. In that case, it is controlled by the IWDG Framework (see watchdog overview).

The configuration is performed using the device tree mechanism that provides a hardware description of the IWDG peripheral.

2 DT bindings documentation

The **IWDG** internal peripheral is a watchdog device. Refer to st,stm32-iwdg.txt [1] for the corresponding binding document.

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)

The IWDG peripheral node is located in *stm32mp151.dtsi* [2] file.

```
iwdg2: iwdg@5a002000 {
        compatible = "st,stm32mp1-iwdg";
        reg = <0x5a002000 0x400>;                          /* Registers location and length
*/
        clocks = <&rcc IWDG2>, <&rcc CK_LSI>;                  /* phandles on clocks needed
*/
        clock-names = "pclk", "lsi";
        status = "disabled";
};
```

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 IWDG hardware on a board, and define a custom timeout (timeout-sec) in seconds:

```
&iwdg2 {
        timeout-sec = <32>;        /* Watchdog timeout value in seconds */
        status = "okay";
};
```

3.3 DT configuration examples

Only the timeout-sec parameter can be modified, as illustrated above.

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

- Documentation/devicetree/bindings/watchdog/st,stm32-iwdg.txt
- arch/arm/boot/dts/stm32mp151.dtsi

Linux® is a registered trademark of Linus Torvalds.

Operating System

Independent Watchdog

Device Tree

Low Speed Internal oscillator (STM32 clock source)

Stable: 19.03.2021 - 08:52 / Revision: 19.03.2021 - 08:49

Invalid target: no reviewed revision corresponds to the given ID.

Return to Device tree.

Stable: 25.09.2020 - 09:42 / Revision: 25.09.2020 - 09:37

Invalid target: no reviewed revision corresponds to the given ID.

Return to IWDG internal peripheral.

Stable: 23.09.2020 - 13:22 / Revision: 12.06.2020 - 13:25

Invalid target: no reviewed revision corresponds to the given ID.

Return to STM32CubeMX.

Stable: 31.01.2020 - 13:22 / Revision: 31.01.2020 - 13:13

Invalid target: no reviewed revision corresponds to the given ID.

Return to Watchdog overview.