



TIM Linux driver



Contents

| | |
|--|----|
| 1. TIM Linux driver | 3 |
| 2. ADC internal peripheral | 10 |
| 3. DAC internal peripheral | 10 |
| 4. DFSDM internal peripheral | 10 |
| 5. DMA internal peripheral | 10 |
| 6. Debugfs | 10 |
| 7. Ftrace | 10 |
| 8. How to use the IIO user space interface | 11 |
| 9. IIO overview | 11 |
| 10. Menuconfig or how to configure kernel | 11 |
| 11. PWM overview | 11 |
| 12. TIM device tree configuration | 11 |
| 13. TIM internal peripheral | 11 |



Contents

| | |
|--------------------------------|----|
| 1 Article purpose | 4 |
| 2 Short description | 5 |
| 3 Configuration | 6 |
| 3.1 Kernel configuration | 6 |
| 3.2 Device tree | 6 |
| 4 How to use | 7 |
| 5 How to trace and debug | 8 |
| 6 Source code location | 9 |
| 7 References | 10 |



1 Article purpose

This article introduces the TIM Linux[®] driver for the TIM internal peripheral^[1]:

- Which TIM features are supported by the driver
- How to configure, use and debug the driver
- What is the driver structure, and where the source code can be found.



2 Short description

The *TIM*^[1] Linux driver (kernel space) is based on the PWM, IIO and counter frameworks. It provides several functionalities:

MFD driver:

- handles registers, clock and DMA^[2] resources
- detects the TIM counter resolution, e.g. 16 or 32 bits.

PWM driver:

- detects the number of TIM channels.
- handles **PWM output** channels.
- handles **PWM capture** channels (input). Note that the PWM capture relies on DMA, which is handled by the MFD core.

IIO driver:

- handles hardware **trigger sources** (synchronously with PWM) for other internal peripherals such as ADC^[3], DAC^[4], DFSDM^[5].

counter driver:

- handles the **quadrature encoder** interface^[6].



3 Configuration

3.1 Kernel configuration

Activate the TIM^[1]Linux driver in the kernel configuration using the Linux Menuconfig tool: [Menuconfig](#) or [how to configure kernel](#).

Enable the following configurations (and their dependencies):

- CONFIG_MFD_STM32_TIMERS
- CONFIG_PWM_STM32
- CONFIG_IIO_STM32_TIMER_TRIGGER
- CONFIG_STM32_TIMER_CNT

```
Device Drivers --->
-> Multifunction device drivers --->
  <*> Support for STM32 Timers
-> Pulse-width modulation (PWM) support --->
  <*> STMicroelectronics STM32 PWM
-> Industrial I/O support --->
  -> Triggers - standalone --->
    <*> STM32 timer trigger
-> Counter support --->
  <*> STM32 Timer encoder counter driver
```

3.2 Device tree

Refer to the [TIM device tree configuration](#) article when configuring the TIM Linux kernel driver.



4 How to use

How to use PWM with sysfs interface

How to set up a TIM or LPTIM trigger using the sysfs interface

How to use the quadrature encoder with the sysfs interface



5 How to trace and debug

The *TIM*^[1] Linux driver can access the timer registers through REGMAP.

It comes with debugfs^[7] entries, which allow dumping registers:

```
$ cd /sys/kernel/debug/regmap
$ ls
40004000.timer  44000000.timer

$ cd 44000000.timer
$ cat registers
000: 00000081
004: 00000000
008: 00000000
00c: 00000000
...
```

It also comes with tracepoints^[8]:

```
$ cd /sys/kernel/debug/tracing
$ cat available_events | grep regmap
...
regmap:regmap_reg_read
regmap:regmap_reg_write
```




6 Source code location

The TIM Linux driver source code is composed of:

- `stm32-timers.c` MFD driver to handle common resources: registers, clock, dmas.
- `pwm-stm32.c` PWM driver to handle PWM channel(s).
- `stm32-timer-trigger.c` IIO driver to handle trigger source for other internal peripherals.
- `stm32-timer-cnt.c` counter driver to handle the quadrature encoder interface.
- `include/linux/mfd/stm32-timers.h` and `include/linux/iio/timer/stm32-timer-trigger.h` header files



7 References

- 1.01.11.21.3 TIM internal peripheral
- DMA_internal_peripheral
- ADC internal peripheral
- DAC internal peripheral
- DFSDM internal peripheral
- Incremental encoder Incremental encoder overview
- Debugfs
- Ftrace

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

Multifunction device

Direct Memory Access

Pulse Width Modulation

Industrial I/O Linux[®] subsystem

Analog-to-digital converter. The process of converting a sampled analog signal to a digital code that represents the amplitude of the original signal sample.

Digital-to-analog converter (Electronic circuit that converts a binary number into a continuously varying value.)

Digital Filter for Sigma-Delta Modulator

Register map (Linux[®] registers map abstraction API)

Debug File System (See <https://en.wikipedia.org/wiki/Debugfs> for more details)

Stable: 25.09.2020 - 08:43 / Revision: 25.09.2020 - 08:41

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

[Return to ADC internal peripheral](#)

Stable: 12.02.2020 - 16:42 / Revision: 12.02.2020 - 16:41

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

[Return to DAC internal peripheral](#)

Stable: 12.02.2020 - 16:42 / Revision: 12.02.2020 - 16:41

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

[Return to DFSDM internal peripheral](#)

Stable: 13.10.2020 - 08:29 / Revision: 13.10.2020 - 08:29

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

[Return to DMA internal peripheral](#)

Stable: 04.02.2020 - 07:47 / Revision: 04.02.2020 - 07:34

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

[Return to Debugfs](#)

Stable: 30.03.2021 10:18 / Revision: 16.02.2021 - 15:49



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

[Return to Ftrace.](#)

Stable: 30.03.2021 - 13:19 / Revision: 29.03.2021 - 09:31

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

[Return to How to use the IIO user space interface.](#)

Stable: 17.02.2021 - 16:24 / Revision: 17.02.2021 - 16:22

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

[Return to IIO overview](#)

Stable: 31.03.2021 - 08:47 / Revision: 26.03.2021 - 08:44

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

[Return to Menuconfig or how to configure kernel.](#)

Stable: 18.02.2021 - 13:27 / Revision: 18.02.2021 - 10:37

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

[Return to PWM overview.](#)

Stable: 26.03.2021 - 16:11 / Revision: 26.03.2021 - 16:01

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

[Return to TIM device tree configuration.](#)

Stable: 30.03.2021 - 13:21 / Revision: 29.03.2021 - 09:46

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

[Return to TIM internal peripheral.](#)