

# Task 5. PWM (Pulse-Width Modulation)

AMM embedded course

# Links

- stm32\_reference\_manual.pdf - 17.3.10 PWM mode (Advanced-Control Timers)
- [https://en.wikipedia.org/wiki/Pulse-width\\_modulation](https://en.wikipedia.org/wiki/Pulse-width_modulation)
- <http://microtechnics.ru/stm32-uchebnyj-kurs-tajmery-chast-2/>
- <http://visualgdb.com/tutorials/arm/stm32/pwm/>
- [http://chipSPACE.ru/httpchipSPACE-rustm32-general-purpose-timers-3-pwm-stdperiph\\_lib/](http://chipSPACE.ru/httpchipSPACE-rustm32-general-purpose-timers-3-pwm-stdperiph_lib/)

# Timer with PWM mode

- Pulse Width Modulation mode allows you to generate a signal with a frequency determined by the value of the TIMx\_ARR register and a duty cycle determined by the value of the TIMx\_CCRx register:
- In the PWM mode the timer controls the output of 1 or more output channels. When the counter value reaches 0, maximum or a **compare value** defined for each channel, the output value of the channel can be changed. Various configuration options define which events change the value and how it is changed.

# Task 5.1. Different LED brightness without PWM

- Use usual scheme of LED output
- Use delay cycle to control brightness (turn LED on – delay – turn LED off)
- Decrease delay by pressing the button

# Task 5.2. Different LED brightness with PWM

- Find appropriate pin for Alternate Function output – Timex X Channel Y (check STM32F4 datasheet)
- Init the pin with AF mode (GPIO\_PinAFConfig())
- Init timer for this pin+channel with OCmode == TIM\_OCMode\_PWM1 (TIM\_OC1Init(), TIM\_OC1PreloadConfig(), TIM\_ARRPreloadConfig(), TIM\_Cmd())
- Regulate the brightness via TIMx->CCRx value