

Prelab for Lab #8: Pulse Width Modulation and Servo Motor Control

Week of 28 March 2022

Part A – Textbook Readings / Videos

1. Read Textbook Chapter 15.3 to review pulse-width modulation.
2. For background on how the timer works, the textbook recommends the following videos (note, I haven't extensively reviewed these).

(a) PWM: <http://www.youtube.com/watch?v=zkrVHICLGww>

Part B – Prelab assignment

In this lab we will set up the TIM1 timer and use it to generate Pulse-width modulated (PWM) signals. We will first use it to pulse the green LED on the STM board, and then we will use it to control a servo motor.

1. Settings needed to PWM the Green LED

We want to PWM the Green LED on the board. Remember it is connected to GPIOE8.

You will need to set the following fields. Write the values to mask/set. If no mask is needed you can let that blank. You can use pre-defined names for the bits rather than raw hex values.

- Set GPIOE->MODER for pin 8 to be “alternate” mode.
MASK MODER=_____
VALUE MODER=_____
- Set GPIOE->AFR[0] and GPIOE->AFR[1] for alternate function of Pin 8 to be TIM1_CH1N. You can look in Appendix I of the book to see which one this is. (Note: in theory these values should be in one of the datasheets but I've been unable to find it documented anywhere besides the textbook).
MASK AFR[0]=_____
VALUE AFR[0]=_____
MASK AFR[1]=_____
VALUE AFR[1]=_____
- Set GPIOE->OTYPER for Pin 8 to be push-pull
MASK OTYPER=_____
VALUE OTYPER=_____
- Set GPIOE->PUPDR for Pin 8 for no pull-up/pull-down
MASK PUPDR=_____
VALUE PUPDR=_____

2. Settings needed to use TIM1 for PWM mode

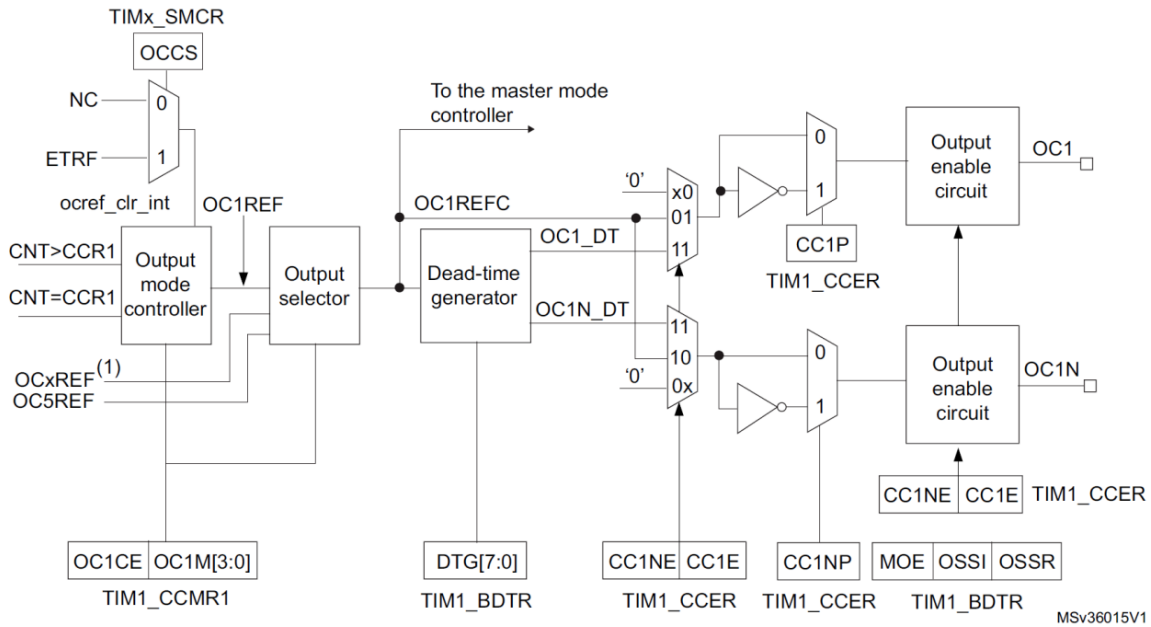


Figure 1: The TIM1 timer has many more configuration settings than the SysTick timer did.

Setting up the TIM1 timer has a lot more options than the SysTick timer, as you can see in Figure 1. You can find the settings in Chapter 30 of the STM manual, and also in Chapter 15.3 of the textbook (specifically example 15-4).

You will need to set the following fields. Write the values to mask/set. If no mask is needed you can let that blank. You can use pre-defined names for the bits rather than raw hex values.

- Set `TIM1->CR1` for the counting direction to be up.
 MASK CR1=_____

 VALUE CR1=_____
- We will use a 4MHz MSI clock for this lab. Set the prescaler `TIM1->PSC` to count at 100kHz.
 MASK PSC=_____

 VALUE PSC=_____
- Set the `TIM1->ARR` register to a value that will give a period of 0.01s
 MASK ARR=_____

 VALUE ARR=_____
- Clear the `OC1M` field in the `TIM1->CCMR1` register and select PWM Mode 1 (`OC1M = 110`):
 MASK CCMR1=_____

 VALUE CCMR1=_____

- Enable the Output 1 preload enable in TIM1->CCMR1:
 MASK CCMR1=_____

 VALUE CCMR1=_____

- Select the output polarity by clearing the CC1NP field in the TIM1->CCER register:
 MASK CCMR1=_____

 VALUE CCMR1=_____

- Enable complementary output of Channel 1 (CH1N) by setting the CC1NE bit in TIM1->CCER:
 MASK CCER=_____

 VALUE CCER=_____

- Set the main output enable (MOE) in TIM1->BDTR:
 MASK BDTR=_____

 VALUE BDTR=_____

- Set the output compare register for channel 1 TIM1->CCR1 to have a duty cycle of 50%:
 MASK CCR1=_____

 VALUE CCR1=_____