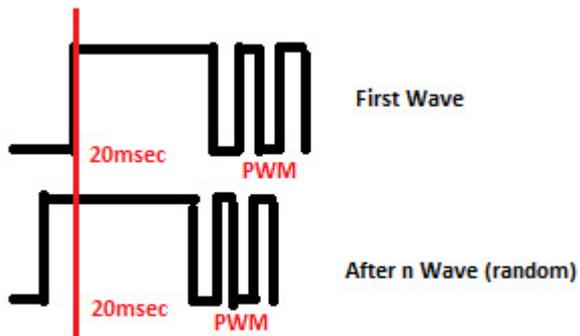


Hi,

I'm working with STM3220G. I have to create a project that: configure GPIOA(pin1) in output for 20msec. After 20msec, starts a PWM for 5msec and so on... My problem is: when I see in my oscilloscope, the waveform are not aligned. The situation is:



this is my code:

```
#include <stdio.h>
#include "stm32f2xx.h"
#include <RTL.h>
#include "stm32f2xx_rcc.h"
#include "stm32f2xx_tim.h"
#include "stm32f2xx_gpio.h"
#include "gpioInit.h"
#include "timInit.h"

#define STARTPWM 1
#define STOPPWM 0

#define CONFSTART 1
#define CONFPWM 2

int start = 0;

unsigned int counterTimeStart = 0;
unsigned int counterTimePwm = 0;
unsigned int counterEndPeriod = 0;

int number_of_steps = 5;
int current_step = 0;
int count = 0;
short int doPwm = 0;

OS_TID t_clock;
OS_TID t_start;
OS_TID t_pwm;

/*****

*****/

void RCC_Configuration(void)
{
    /* ----- System Clocks Configuration -----*/
    /* TIM2 clock enable */
    RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM2, ENABLE);

    /* GPIOA clock enable */
    RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE);
}
```

```

/*****/
void NVIC_Configuration(void)
{
    NVIC_InitTypeDef NVIC_InitStructure;

    NVIC_ClearPendingIRQ(TIM2_IRQn);
    /* Enable the TIM1 global Interrupt */
    NVIC_InitStructure.NVIC_IRQChannel = TIM2_IRQn;
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority = 0;
    NVIC_InitStructure.NVIC_IRQChannelSubPriority = 0;
    NVIC_InitStructure.NVIC_IRQChannelCmd = ENABLE;
    NVIC_Init(&NVIC_InitStructure);

    //NVIC_SetPriority(SysTick_IRQn,          2);
    //NVIC_SetPriority(TIM2_IRQn,          4);
}

/*****/

/*****/

void TIM2_Configuration(void)
{
    TIM_OCInitTypeDef TIM_OCInitStructure;
    TIM_TimeBaseInitTypeDef TIM_TimeBaseStructure;

    /* Time base configuration - SystemCoreClock = 12000000 for 120 MHz board */
    TIM_TimeBaseStructure.TIM_Prescaler = (uint16_t) (((SystemCoreClock / 1000000) / 2) - 1); //
    Shooting for 1 MHz, (1us)
    TIM_TimeBaseStructure.TIM_Period = 1000 - 1; // 1 MHz / 20000 = 50 Hz (20ms)
    TIM_TimeBaseStructure.TIM_ClockDivision = TIM_CKD_DIV1;
    TIM_TimeBaseStructure.TIM_CounterMode = TIM_CounterMode_Up;
    TIM_TimeBaseInit(TIM2, &TIM_TimeBaseStructure);

    /* Enable TIM2 Preload register on ARR */
    TIM_ARRPreloadConfig(TIM2, ENABLE);

    /* TIM PWM1 Mode configuration: Channel */
    TIM_OCInitStructure.TIM_OCMode = TIM_OCMode_PWM1;
    TIM_OCInitStructure.TIM_OutputState = TIM_OutputState_Enable;
    TIM_OCInitStructure.TIM_Pulse = 500; // Servo Top-Center
    TIM_OCInitStructure.TIM_OCPolarity = TIM_OCPolarity_High;

    /* Output Compare PWM1 Mode configuration: Channel2 PA.1 */
    TIM_OC2Init(TIM2, &TIM_OCInitStructure);
    TIM_OC2PreloadConfig(TIM2, TIM_OCPreload_Enable);

    /* TIM2 Interrupts enable */
    TIM_ITConfig(TIM2, TIM_IT_Update, ENABLE);

    /* TIM2 enable counter */
    TIM_Cmd(TIM2, ENABLE);
}

/*****/

void GPIO_Configuration(int mode)
{
    GPIO_InitTypeDef GPIO_InitStructure;
    if(mode == 1){
        GPIO_InitStructure.GPIO_Pin = GPIO_Pin_1;
        GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT;
        GPIO_InitStructure.GPIO_Speed = GPIO_Speed_100MHz;
        GPIO_Init(GPIOA, &GPIO_InitStructure);
    }
}

```

```

        GPIOA->ODR = 0x0002;
    }else{
        GPIO_InitStructure.GPIO_Pin = GPIO_Pin_1;
        GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF;
        GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
        GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL;
        GPIO_InitStructure.GPIO_Speed = GPIO_Speed_100MHz;
        GPIO_Init(GPIOA, &GPIO_InitStructure);
        GPIO_PinAFConfig(GPIOA, GPIO_PinSource1, GPIO_AF_TIM2); // PA1 TIM2_CH2
    }
}

void TIM2_IRQHandler(void)
{
    TIM_ClearITPendingBit(TIM2, TIM_IT_Update);
    //START
    if((current_step == 20) && (doPwm == 0)){
        //doPwm = 1;
        isr_evt_set(0x0001, t_start);
    }
    //PWM
    if((current_step == number_of_steps) && (doPwm == 2)){
        //doPwm = 3;
        isr_evt_set(0x0002, t_pwm);
    }
    current_step++;
}

__task void starts(void){
    for(;;){
        os_evt_wait_and(0x0001, 0xffff);
        TIM_Cmd(TIM2, DISABLE);
        GPIO_Configuration(CONFPWM);
        current_step = 0;
        doPwm = 2;
        TIM_Cmd(TIM2, ENABLE);
    }
}

__task void pwm(void){
    for(;;){
        os_evt_wait_and(0x0002, 0xffff);
        GPIO_Configuration(CONFSTART);
        TIM_Cmd(TIM2, DISABLE);

        doPwm = 0;
        current_step = 0;
        TIM_Cmd(TIM2, ENABLE);
    }
}

__task void init(void){
    t_start = os_tsk_create(starts,1);
    t_pwm = os_tsk_create(pwm,1);
    os_tsk_delete_self ();
}

int main(void)
{
    RCC_Configuration();
    NVIC_Configuration();
    TIM2_Configuration();
    GPIO_Configuration(CONFSTART);
    os_sys_init(init);
}

```

```
while(1)
{
}
}
```

OS_CLOCK 120000000 and OS_TICK 1000.

THANKS!!!