

Микроконтроллер  
STM32F100RBT6  
фирмы STMicroelectronics

# Базовые таймеры TIM6 и TIM7

- 16-битовый суммирующий счётчик с автоматической перезагрузкой;
- 16-битовый предварительный делитель частоты (прескалер) с программируемым коэффициентом деления от 1 до 65536 включительно;
- прерывания по переполнению.

# Настройка TIM6 и TIM7

TIM7 Mode and Configuration

Mode

Activated

One Pulse Mode

$$T = \frac{ARR * (PSC + 1)}{f_{МК}}$$

Configuration

Reset Configuration

Parameter Settings

User Constants

NVIC Settings

DMA Settings

Configure the below parameters :

Search (Ctrl+F)



Counter Settings

Prescaler (PSC - 16 bits value)	Коэф. Деления фмк	7999
Counter Mode		Up
Counter Period (AutoReload Register - 16 bits value )		1000      ARR
auto-reload preload		Disable

Trigger Output (TRGO) Parameters

Trigger Event Selection	Reset (UG bit from TIMx_
-------------------------	--------------------------

# Регистры таймеров TIM6 и TIM7

- TIMx\_CNT (Counter Register) – счетный регистр;
- TIMx\_PSC (Prescaler Register) - регистр предделителя; определяет коэффициент деления  $k=TIMx\_PSC+1$ ;
- TIMx\_ARR (Auto-Reload Register) - регистр авто-перезагрузки; когда значение в счётчике достигает величины, записанной в этом регистре, следующий импульс сбрасывает счётчик в 0, при этом генерируется сигнал переполнения счётчика, который используется как update event - событие обновления. Если в TIMx\_ARR записано значение 0, то счётчик таймера останавливается.

Содержимое счетного регистра, регистра автоматической перезагрузки и регистр предделителя программно доступны для чтения и записи в любой момент.

# Регистры таймеров TIM6 и TIM7

- TIMx\_CR1 (TIM6&TIM7 control register 1) - TIM\_CR1\_CEN – запуск таймера
- TIMx\_DIER (TIM6&TIM7 DMA/Interrupt enable register) – UIE (Update interrupt enable) - Бит разрешения генерации прерывания от таймера по событию обновления.
- TIMx\_SR (TIM6&TIM7 status register) - TIM\_SR\_UIF – флаг прерывания при переполнении таймера.

# Работа с таймерами TIM6 и TIM7

- main.c

```
LL_TIM_EnableIT_UPDATE(TIM7); // разрешить прерывания у  
TIM7
```

```
LL_TIM_EnableCounter(TIM7); // включить TIM6
```

- stm32f3xx\_it.c

```
297 void TIM7_IRQHandler(void)
298 {
299     /* USER CODE BEGIN TIM7_IRQn 0 */
300     if (LL_TIM_IsActiveFlag_UPDATE(TIM7) != RESET) // проверка флага
301     {
302         LL_TIM_ClearFlag_UPDATE(TIM7); // очистка флага прерывания
303
304
305     }
306     /* USER CODE END TIM7_IRQn 0 */
307     /* USER CODE BEGIN TIM7_IRQn 1 */
308
309     /* USER CODE END TIM7_IRQn 1 */
310 }
```