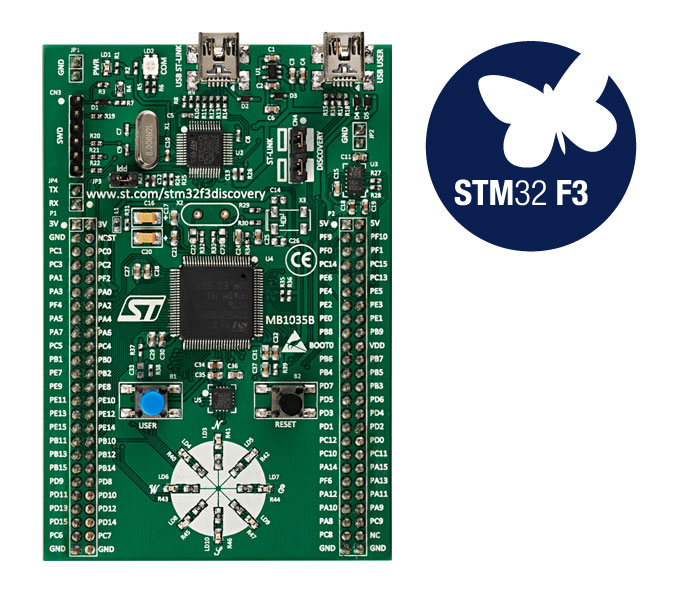
Основы работы с таймерами в STM32F3. Формирование ШИМ сигнала

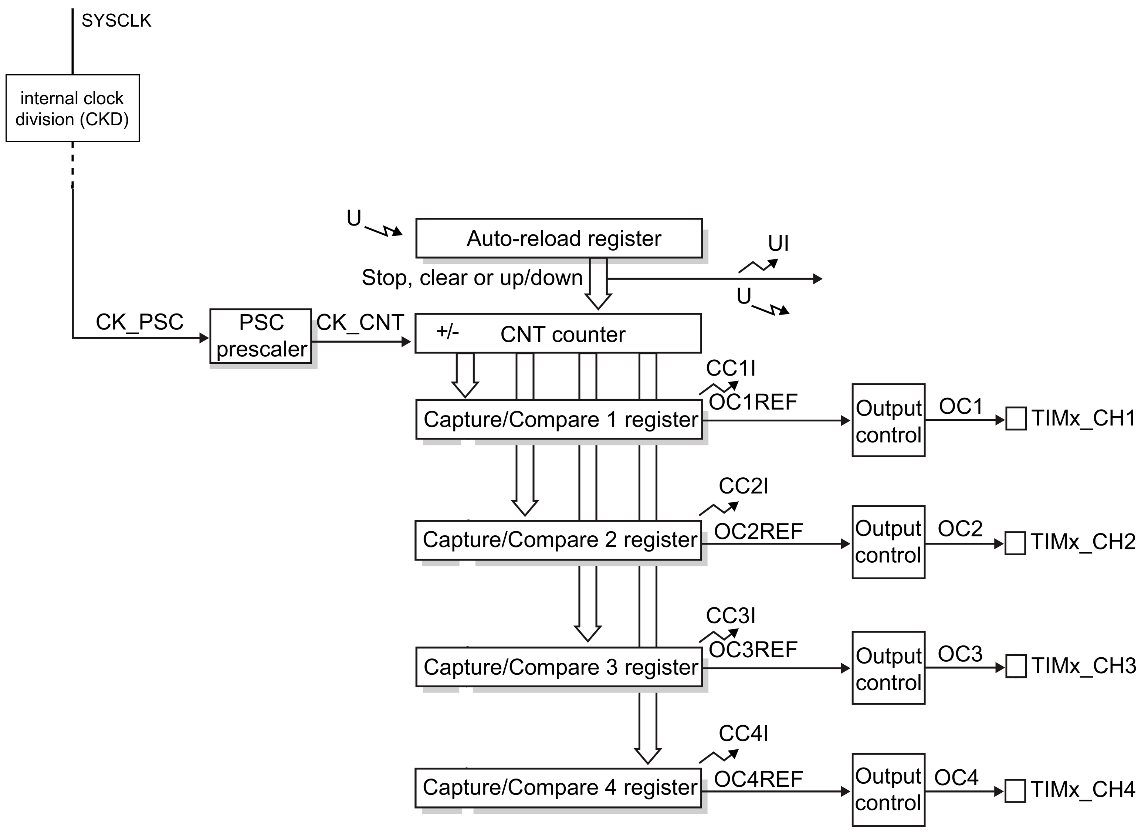


# Основные сведенье о таймерах

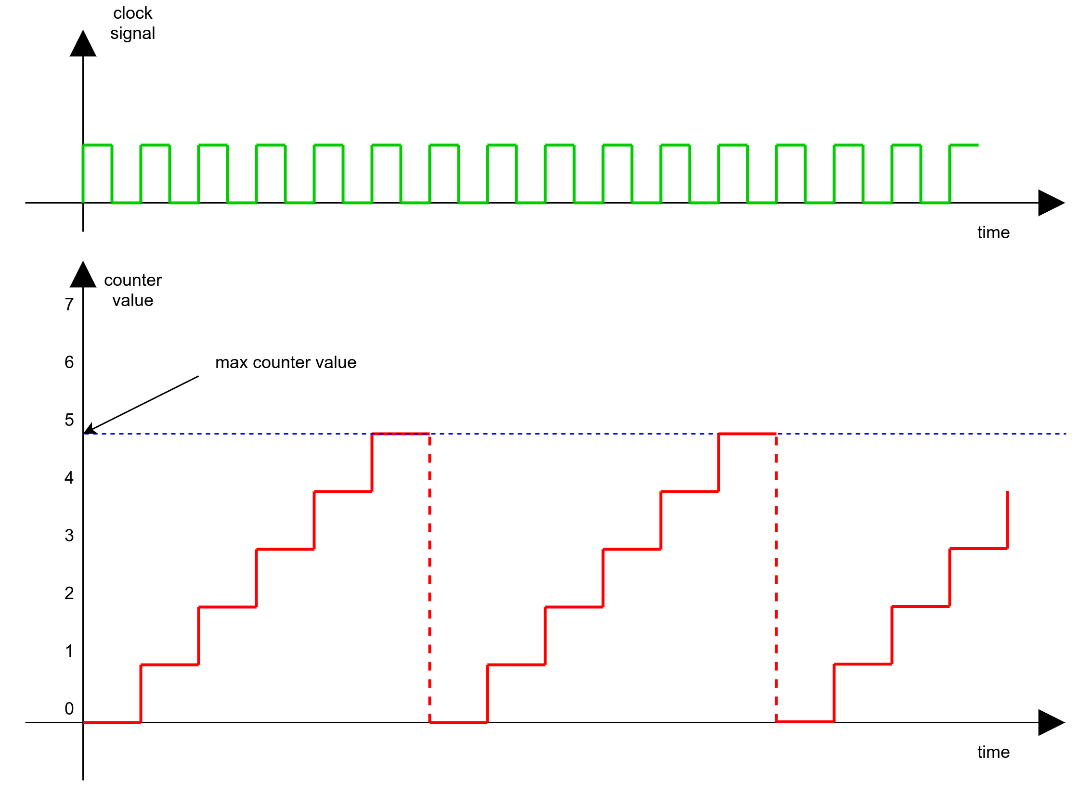
Одним из применений таймеров является формирование сигнала широтно-импульсная модуляции (ШИМ или PWM). Такой сигнала обычно используется для регулировки мощности различный устройств, управлением сервоприводами и т.д.

Микроконтроллеры STM32F3 имеют довольно большое число таймеров (около 10), с обширной функциональностью. Но в данном разделе мы остановимся только на части связной с формированием ШИМ сигнала.

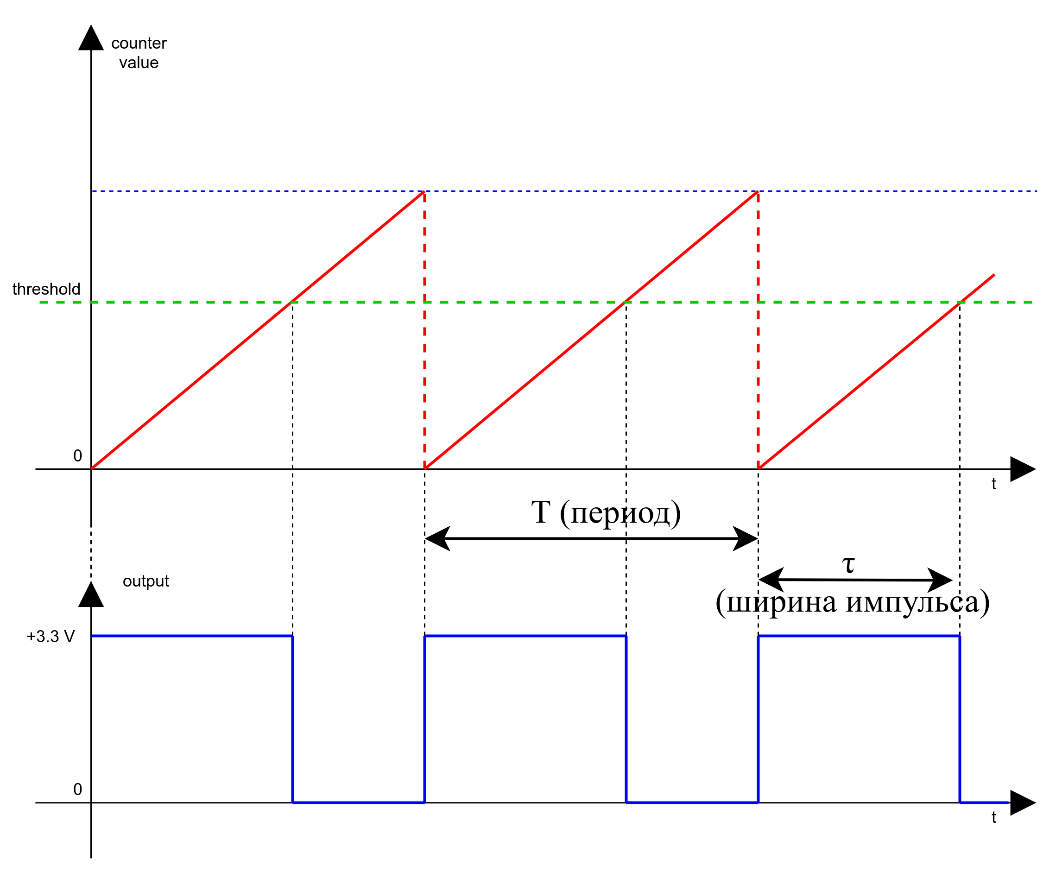
Схема таймера, связная с данной функциональностью, представлена на следующем рисунке:



В данном случае мы имеем счетчик CNT counter, который инкрементируется с частотой . Его значения изменяются в диапазоне от 0 до n включительно, где n – значение в Auto-reload register. После достижения значения n, значение счетчика (CNT) сбрасывается в 0, и снова продолжает инкрементироваться:



В тоже время у таймера есть несколько каналов, которые имеют capture / compare регистры (CCR), значение которых постоянно сравнивается со значением счетчика. Если значение CCR меньше или равно значению счетчика, на входе получаем канала получаем логическую 1, иначе логический 0:



В результате на входе канала имеет повторяющиеся прямоугольные импульсы с периодом:

 , где

 - частота процессора,

 - коэффициент CKD (1, 2 или 4),

 - значение PSC регистра,

 - значение ARR регистра;

и длительностью импульса:

 , где

 - частота процессора,

 - коэффициент CKD (1, 2 или ;4),

 - значение PSC регистра,

 - значение CRR регистра.

# Инициализация таймера для ШИМ

Настройка таймера состоит из 4-х шагов: настройка пинов, настройка параметров таймера, настройка каналов таймера и его запуск.

Настройка выходных пинов:

1. после выбора таймера (например TIM2) и его каналов, найдите пины, которые подключены к выходам каналов таймера;
2. включите тактирующий сигнал для пинов портов (регистр RCC -> AHBENR);
3. выключите подтягивающие резисторы (регистр GPIOx->PUPDR);
4. установите режим pull/push (регистр GPIOx-> OTYPER);
5. выберете альтернативную функцию на работу с таймером (регистр GPIOx->AFR);
6. переключите пин на альтернативную функцию (регистр GPIOx->MODER).

Настройка параметров таймера:

1. включите тактирующий сигнал таймера (регистр RCC->APB1ENR или RCC->APB2ENR в зависимости от таймера);
2. установите значение internal clock division (CKD) (регистр TIMx->CR1). Значение по умолчанию , поэтому если этого достаточно, то этот регистр можно не изменять;
3. установите значение prescaler (PSC) (регистр TIMx->PSC);
4. установите значение ARR регистра (регистр TIMx->ARR).

Настройка каналов таймера (выполнить для каждого канала):

1. установить режим канала в «pwm mode 1» (регистр TIMx->CCMR1 или TIMx->CCMR2 в зависимости от номера канала, бит «Output Compare <channel\_no> mode»);
2. установить начальное значение CCR (регистр TIMx->CCRx). В ходе выполнения программа будет изменять этот регистр для изменения длительности импульсов.

Запуск таймера:

1. включите выход канала (регистр TIMx->CCER, бит «Capture/Compare <channel\_no> output enable»); повторить данный шаг для всех каналов, с которыми вы работаете;
2. если таймер имеет регистр BDTR, то установите в нем бит main output enable (**MOE**) в **1**;
3. запустить сам таймер (регистр TIMx->CR1, бит «Counter enable»)

После этого таймер начнет формировать ШИМ сигнал. Параметры сигнала можно изменять через ARR и ССRx регистры без остановки таймера.

## Управление сервоприводами



Типичные сервоприводы управляются с помощью PWM сигнала со следующими параметрами:

1. период: 20000 мкс (50 Гц)
2. длительность импульса: примерно от 500 мкс до 2500 мкс.

Длительность импульса задает угол поворота сервопривода.