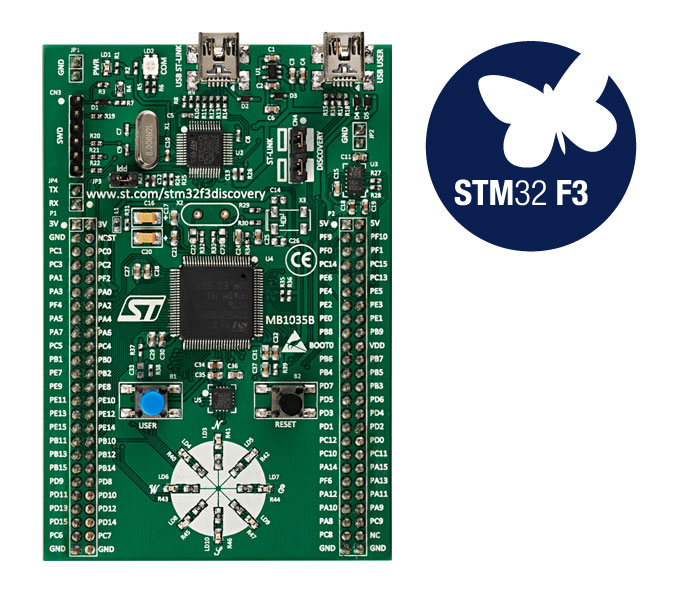
Основы работы с таймерами в STM32F3. Организация периодических прерываний

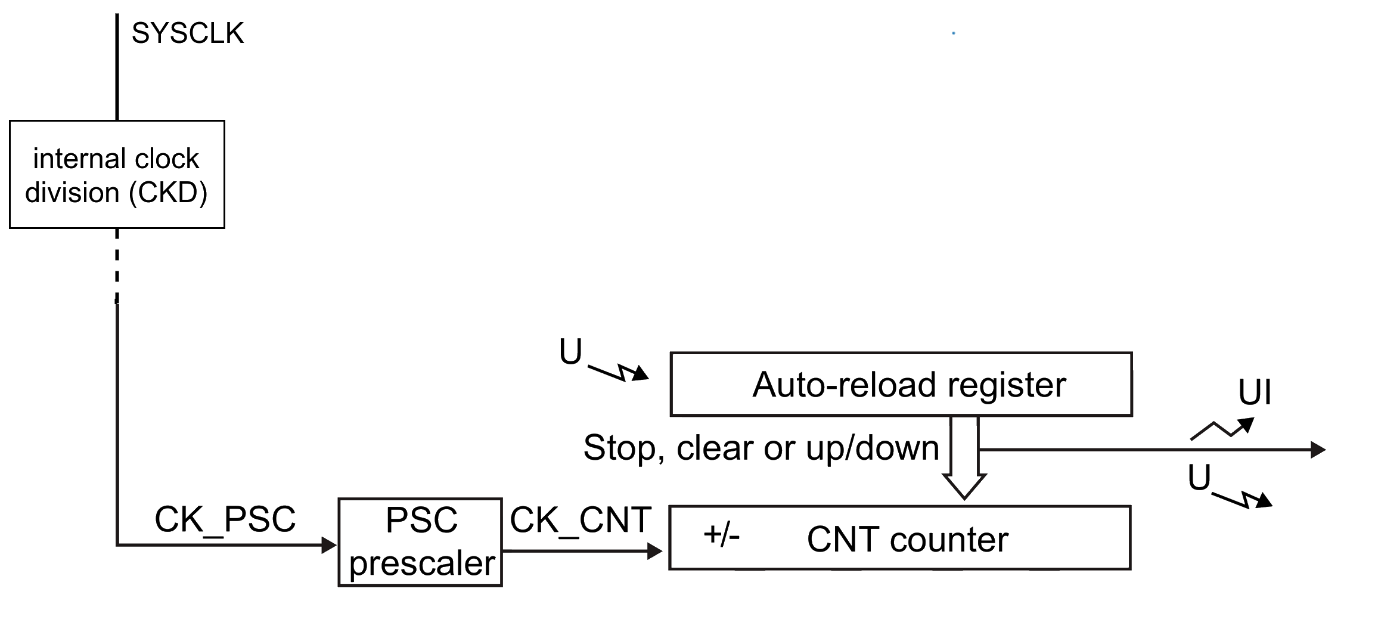


# Основные сведенье о таймерах

Одним из применений таймеров является периодический вызов прерываний через заданный промежуток времени. Это может использоваться, например, для опроса датчиков с нужной частотой, или проверку каких-либо интерфейсов.

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

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, таймер инициирует прерывание, сбрасывает значение счетчика (CNT) в 0, и продолжает его инкрементировать. Далее данный процесс повторяется.

Частота инкрементации  равна частоте процессора , уменьшенная пределителями internal clock division (CKD) и prescaler (PSC). Internal clock division может уменьшать частоту в 1, 2 или 4 раза. Prescaler – в диапазоне от 1 до . Поэтому итоговая частота вызова прерываний считается как:

 , где

 - частота процессора,

 - коэффициент CKD,

 - значение PSC регистра,

 - значение ARR регистра.

# Инициализация таймера в базовом режим с прерываниями

Настройка таймера состоит из 2-х шагов: настройка параметров таймера и его запуск.

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)
5. разрешить процессору обрабатывать прерывания от таймера. Для этого нужно установить нужный бит в 256-ти битном регистре NVIC ->ISER. Номер бита – это позиция прерывания в таблице прерываний (см. столбец position в таблице «STM32F303xB/C/D/E, STM32F358xC and STM32F398xE vector table» из «Reference manual»).

Запуск таймера:

1. разрешите самому таймеру вырабатывать прерывания (регистр TIM2->DIER, бит «Update interrupt»)
2. запустите таймер (регистр TIM2->CR1, бит «Counter enable»)

## Обработчик прерывания

Для обработки прерывания необходимо реализовать обработчик. Он представляет из себя обычную C функцию, которая ничего не принимает на вход и выход. Единственный способ общения из нее с о остальной частью программы – глобальные переменные.

Для того чтобы функция обрабатывала нужное прерывание, она должна иметь заранее определенное имя. Имена обработчиков можно найти в файле startup\_stm32f303xc.s, который есть у вас в проекте.

Внутри обработчика, кроме логики ваше программы, должен быть код, который отмечает, что вы обработали событие, вызвавшее прерывание. Для таймера в базовом режиме вы должны установить бит **«Update interrupt flag»** регистра **TIMx->SR** в **0**. Если этого не сделать, то после выхода из обработчика, он будет немедленно вызван снова.