

## НАЛАШТУВАННЯ ТАКТУВАННЯ RTC ТА РОЗРАХУНОК ДІЛЬНИКІВ (ASYNCHRONOUS/SYNCHRONOUS PRESCALERS)

**Мета:** Ознайомитися з принципом роботи RTC у мікроконтролері STM32F411 та роллю джерела тактування.

### Теоретичні відомості

RTC (Real-Time Clock) — це автономний апаратний модуль мікроконтролера, призначений для безперервного відліку календарного часу (години, хвилини, секунди, дата, місяць, рік) незалежно від стану центрального процесора. На відміну від звичайних таймерів, RTC функціонує у спеціальному backup domain, що дозволяє зберігати хід часу навіть при вимкненні основного живлення (за наявності живлення на виводі VBAT).

RTC застосовується для:

- підтримки системного часу;
- формування часових міток (timestamp);
- реалізації будильників (Alarm);
- пробудження мікроконтролера з режимів енергозбереження;
- календарного керування подіями;
- логування даних із часовою прив'язкою.

Робота RTC базується на підрахунку імпульсів від стабільного джерела частоти. У мікроконтролерах сімейства STM32 можливі такі джерела:

1. **LSE (Low-Speed External)** — зовнішній кварцовий резонатор 32.768 кГц
2. **LSI (Low-Speed Internal)** — внутрішній RC-генератор (~32–40 кГц)
3. **HSE/RTCPRE** — високочастотний кварц (наприклад, 8 МГц), поділений спеціальним дільником

Найбільш точним є LSE, оскільки частота 32 768 Гц дорівнює  $2^{15}$ , що дозволяє реалізувати точний поділ до 1 Гц без залишку.

Модуль RTC у мікроконтролерах STM32 формує секундний імпульс шляхом цифрового ділення вхідної частоти  $f_{RTC}$ , що надходить від обраного джерела тактування (LSE, LSI або HSE/RTCPRE). Секундний імпульс формується виключно апаратно, без участі центрального процесора, шляхом каскадного використання двох лічильників-подільників: асинхронного та синхронного.

Частота  $f_{RTC}$  визначається вибраним джерелом:

- LSE = 32 768 Гц
- LSI  $\approx$  32 000–40 000 Гц (нестабільна)
- HSE / RTCPRE

Асинхронний подільник зменшує частоту до рівня, прийнятного для синхронного лічильника. Він є 7-бітним або 15-бітним (залежно від серії), та зберігається у регістрі: RTC\_PRER[22:16]

Фактичне ділення виконується за формулою:

$$f_{async} = \frac{f_{RTC}}{PREDIV_A + 1}$$

Асинхронний дільник працює незалежно від шинної частоти APB та не синхронізується з нею.

Синхронний подільник отримує сигнал  $f_{async}$  і виконує остаточне ділення до 1 Гц. Зберігається у регістрі RTC\_PRER[14:0]

Частота після синхронного поділу:

$$f_{1Hz} = \frac{f_{async}}{PREDIV_S + 1}$$

Синхронний подільник є зворотним лічильником. Його поведінка:

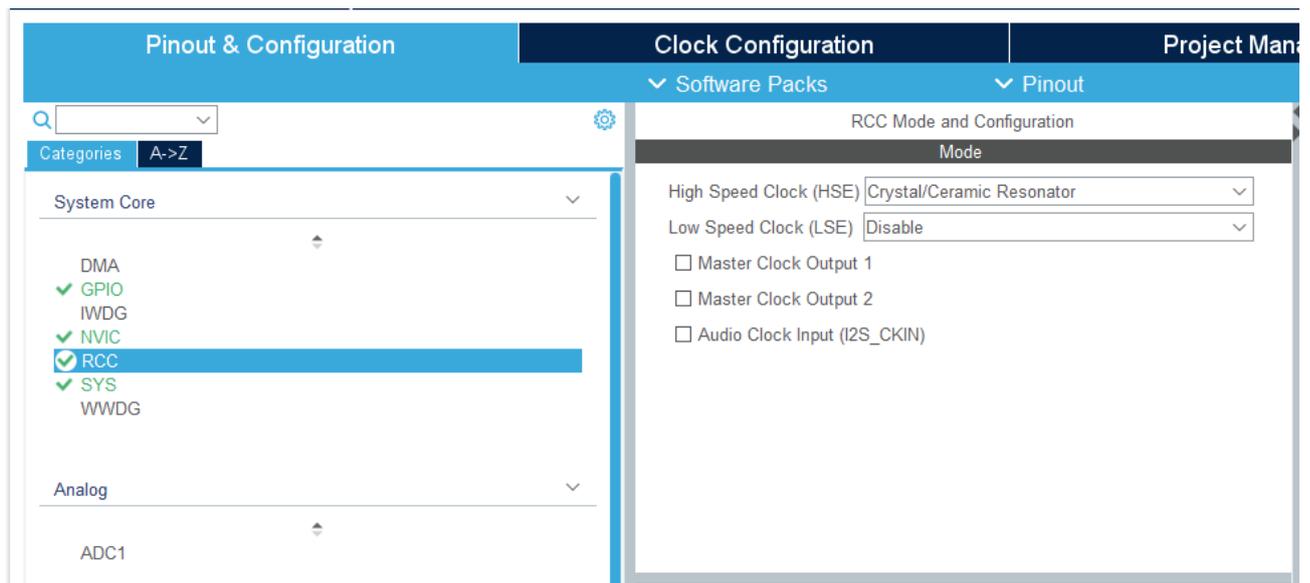
1. Початкове значення = PREDIVSPREDIV\_SPREDIVS
2. Кожен імпульс  $f_{async}$  зменшує лічильник на 1
3. Коли значення досягає 0:
  - формується імпульс 1 Гц
  - відбувається інкремент регістра секунд
  - лічильник перезавантажується значенням PREDIVSPREDIV\_SPREDIVS

Таким чином, регістр секунд (RTC\_TR[Seconds]) інкрементується виключно апаратно після кожного завершення повного циклу поділу.

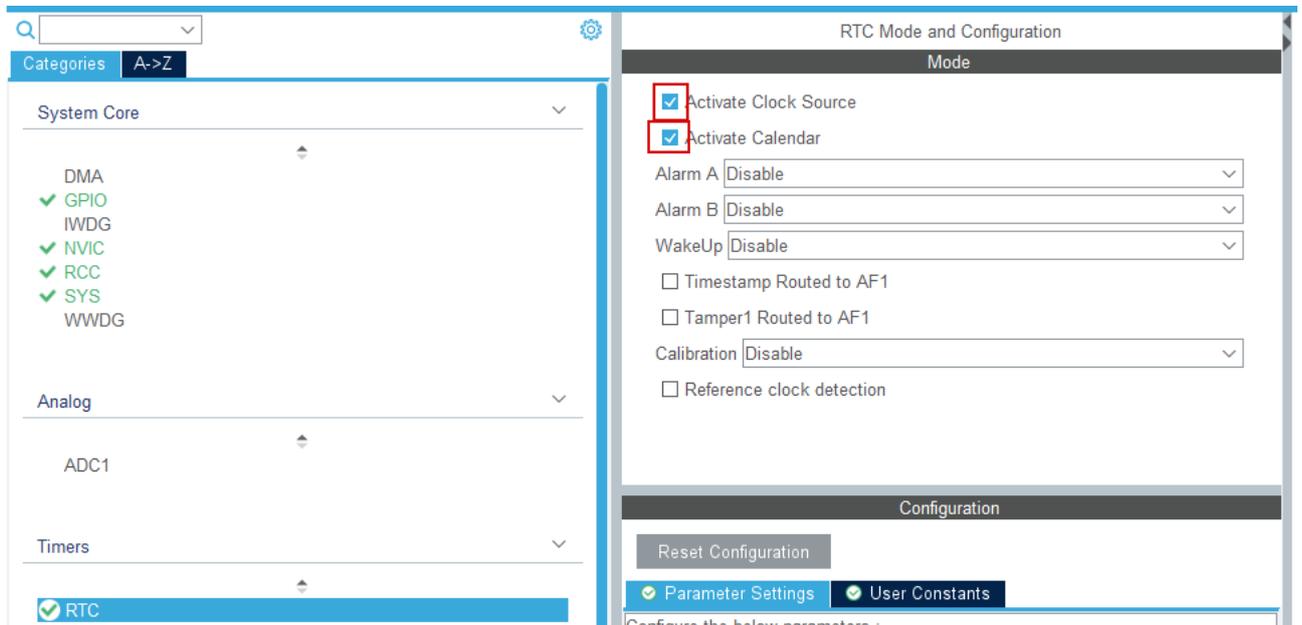
### Хід виконання роботи

1. Реалізувати секундомір, результат спостерігати в Debug -> Live Expressions;

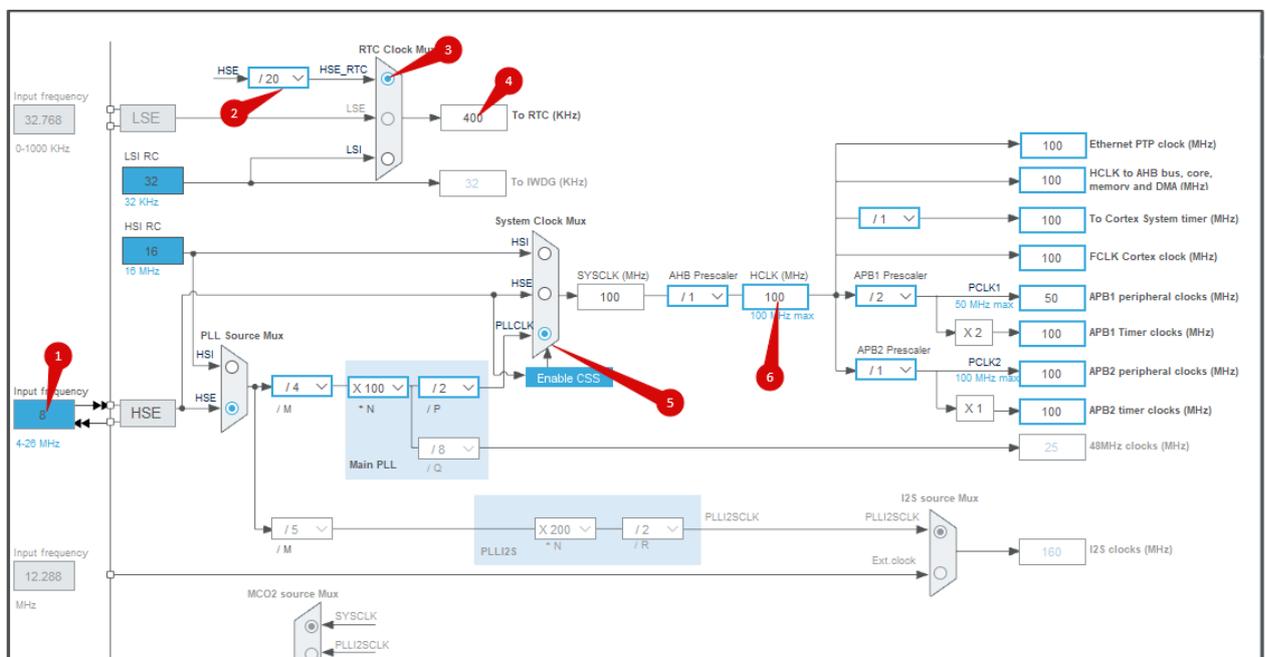
1.1 Увімкнути High Speed Clock(HSE)-> Crystal/Ceramic Resonator



1.2 В розділі Timers увімкнути RTC



1.3 Перейти в Clock Configuration та виставити наступні параметри:



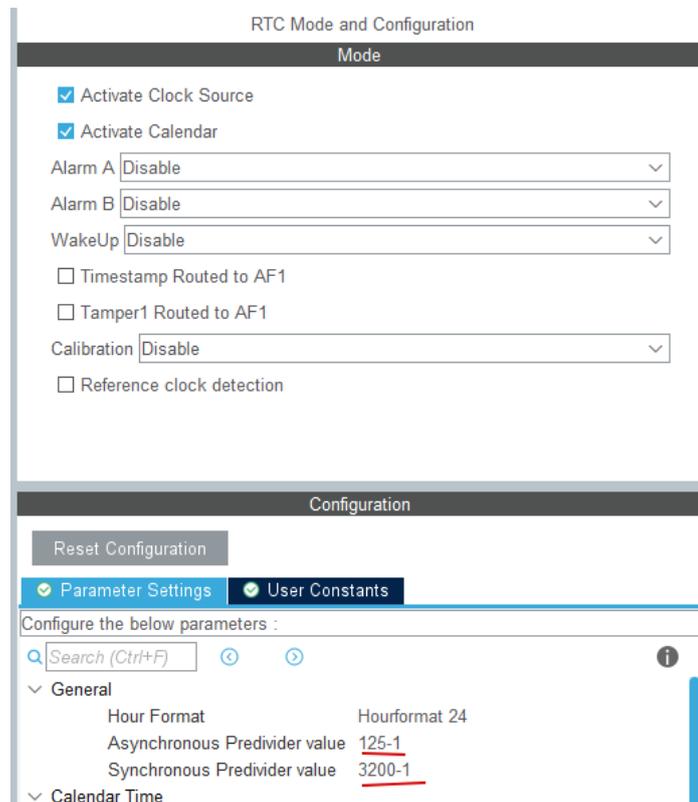
HSE = 8 MHz - фізична частота генератора на платі;

Дільник /20 ->  $8\,000\,000/20 = 400\,000$  Гц;

RTC clock mux вибрано на: HSE\_RTC тобто RTC живиться не від LSE і не від LSI, а від: HSE / 20 Тобто вхідна частота RTC:  $f_{RTC}=400\,000$  Гц.

SYSCCLK = 100 MHz - основний такт системи.

1.4 В налаштуваннях розрахувати дільники:



Розрахунок дільників розраховується за допомогою розкладання числа на 2 добутки

$$(PREDIVA + 1) \cdot (PREDIVS + 1) = f_{RTC}$$

$$124 + 1 * 3.199 + 1 = 400.000 \text{ Гц}$$

Згенерувати код

1.5 Додати після `RTC_HandleTypeDef hrtc;` наступний фрагмент коду

```
RTC_TimeTypeDef dbgTime;
RTC_DateTypeDef dbgDate;
```

Ці структури даних HAL, які описують формат часу і дати для модуля RTC в STM32.

Та в основний цикл

```
while (1)
{
    /* USER CODE END WHILE */
    HAL_RTC_GetTime(&hrtc, &dbgTime, RTC_FORMAT_BIN);
    HAL_RTC_GetDate(&hrtc, &dbgDate, RTC_FORMAT_BIN);
    /* USER CODE BEGIN 3 */
}
}
```

1.6 Перевірити роботу RTC через Debug сформувавши змінні для відслідковування та запустити виконання:

Expression	Type	Value	Address
dbgTime.Hours	uint8_t	0 '\0'	0x20000048
dbgTime.Minutes	uint8_t	0 '\0'	0x20000049
dbgTime.Seconds	uint8_t	0 '\0'	0x2000004a
+ Add new expression			

При правильному налаштуванні час зміни секунд на мікроконтролері має так само змінюватись як і реальний. Зробити висновки.

### **Індивідуальне завдання**

1. Виконати завдання в ході виконання роботи;
2. Обрахувати Synchronous Predivider value при значеннях Asynchronous Predivider value:64 та 32 та перевірити коректність роботи;
2. Написати код який би змінював стан світлодіода кожні 10 секунд від початку роботи.

### **Зміст звіту**

1. Тема та мета;
2. Скріншоти конфігурації RTC для Asynchronous Predivider value:64 та 32;
3. Код зміни стану світлодіода кожні 10 секунд;
4. Висновки.

### **Контрольні запитання**

1. Що таке RTC і чим він відрізняється від звичайних таймерів TIM?
2. Чому частота 32.768 кГц є стандартною для годинників?
3. Яку роль виконує асинхронний дільник?
4. Яку роль виконує синхронний дільник?