

Национальный исследовательский университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №4
по дисциплине «Встроенные системы»

Вариант 2

Выполнил: Ларочкин Г.
Шуст И.
Тарасенко Д.
Группа: Р3400
Преподаватель: Ключев А.О.

Санкт-Петербург
2020 г.

Описание задачи

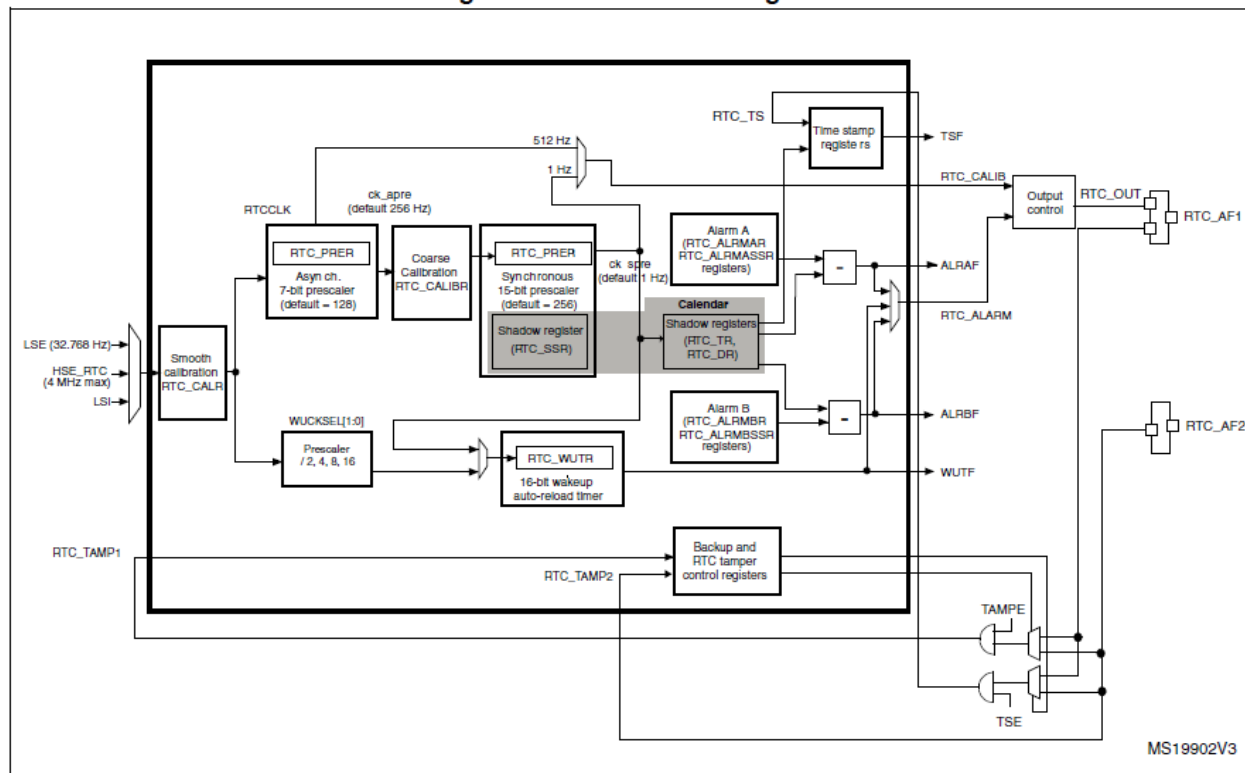
Реализовать считывание времени (часы, минуты, секунды) с часов реального времени и вывод в трассировочный буфер.

Выполнение

Real-time clock

Часы реального времени предназначены для того, чтобы вести отсчет текущего времени. Можно сказать, что модуль RTC представляет из себя таймер, но есть одно важное отличие. Часы реального времени могут продолжать работу от резервного источника питания, подключенной к выводу VBAT микроконтроллера. Также есть возможность сохранять текущее время в памяти, чтобы при следующем включении время началось с того же самого места. На рисунке 1 представлена схема Real-time clock.

Figure 237. RTC block diagram



1. On STM32F4xx devices, the RTC_AF1 and RTC_AF2 alternate functions are connected to PC13 and PI8, respectively.

Рисунок 1 – схема real-time clock

Для реализации вывода даты в трассировочный буфер был добавлен модуль RTC в среде разработки STM32CubeIDE. На рисунке 2 представлены параметры модуля RTC, используемые в лабораторной работе.

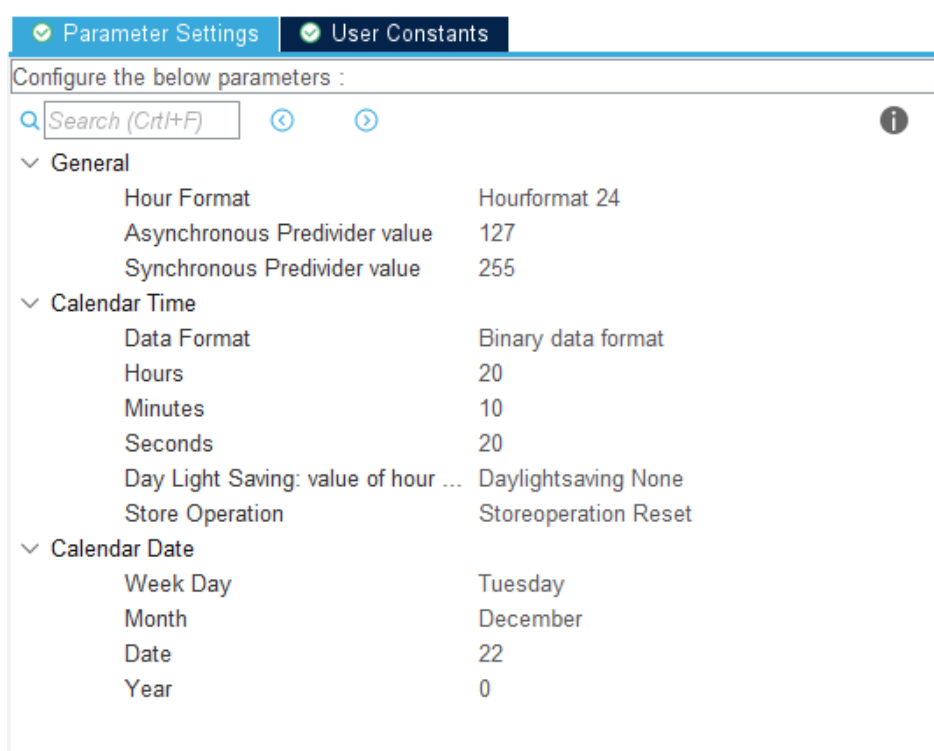


Рисунок 2 – параметры модуля RTC

Часы реального времени могут быть подключены к источнику питания тремя вариантами:

1. Внешний низкочастотный кварцевый резонатор (LSE)
2. Внутренний низкочастотный генератор (LSI)
3. Внешний высокочастотный генератор

Для данной лабораторной был выбран второй (LSI). На рисунке 3 представлена схема конфигурации.

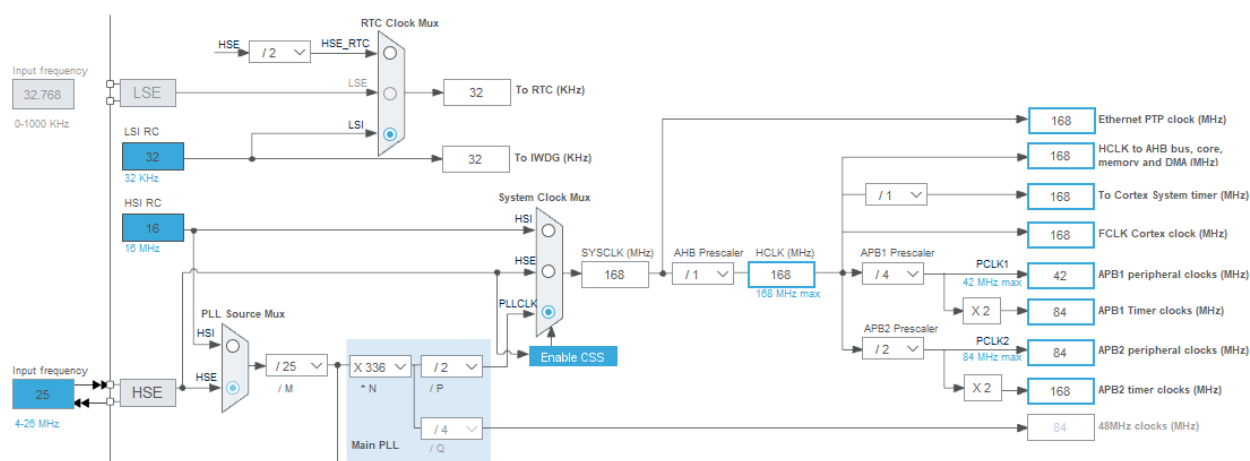


Рисунок 3 – схема конфигурации RTC

Сама программа состоит из 2-х основных частей:

1. Инициализация текущего времени RTC
2. Циклическое получение текущего времени в формате hh:mm:ss и ожиданием в 1 секунду

Ниже представлен пример кода функции *main*.

```
RTC_TimeTypeDef time;
RTC_DateTypeDef date;
int main(void)
{
    HAL_Init();

    SystemClock_Config();
    MX_GPIO_Init();
    MX_IWDG_Init();
    MX_RTC_Init();

    time.Hours = 20;
    time.Minutes = 13;
    time.Seconds = 30;
    HAL_RTC_SetTime(&hrtc, &time, RTC_FORMAT_BIN);

    MX_TRACE_Init();
    SDK_TRACE_Start();

    for (int i = 0; i < 5; i++)
    {
        SDK_TRACE_Timestamp(PRINT, 1);

        HAL_RTC_GetDate(&hrtc, &date, RTC_FORMAT_BIN);
        HAL_RTC_GetTime(&hrtc, &time, RTC_FORMAT_BIN);

        SDK_TRACE_Print("Current time: %02d:%02d:%02d", time.Hours, time.Minutes, time.Seconds);

        SDK_TRACE_Timestamp(PRINT, 0);
        HAL_Delay(1000);
    }
    SDK_TRACE_Stop();
}
```

Результат

На рисунке 4 представлен результат выполнения программы в виртуальной лаборатории cLab. Как можно видеть в окне событий, время замеряется каждую секунду. В трассировочном буфере выводится текущее время в формате *hh:mm:ss*. К сожалению, виртуальная лаборатория не всегда выдает правильный результат текущего времени RTC. Это может быть связано с тем, что функционал виртуальной лаборатории отличается от реальной платы. Например, каждому пользователю отводится определенное время для работы, далее при смене пользователя его контекст исполнения сохраняется и восстанавливается при продолжении работы. Для этого могут быть переопределены стандартные функции *HAL_Delay* и использоваться регистры RTC. В виртуальной лаборатории довольно много факторов, которые могут влиять на работу RTC.

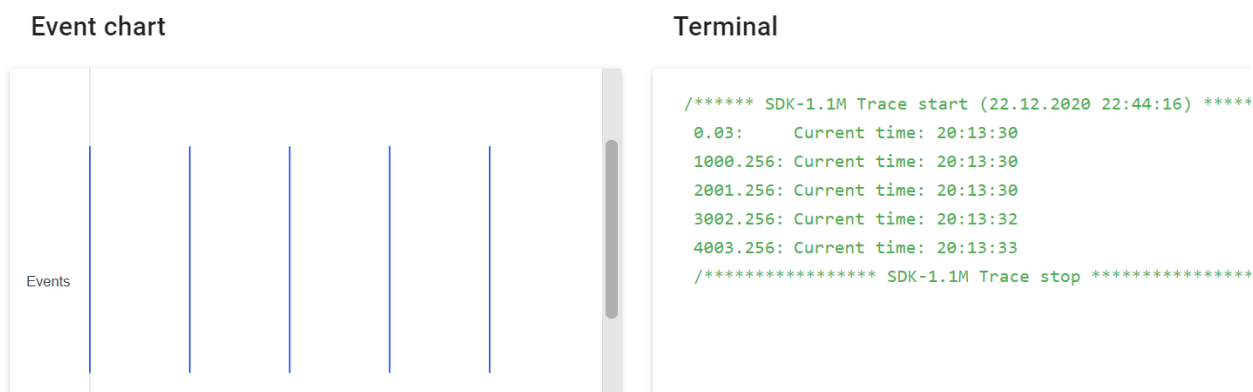


Рисунок 4 – результат выполнения программы в виртуальной лаборатории

Вывод

В ходе данной лабораторной работы был изучен модуль real-time clock. Были изучены возможные настройки времени и даты RTC, изучены способы получения текущей даты/времени. Также была разработана программа с использованием данного модуля.