

ГУАП

КАФЕДРА № 41

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

доц., канд. техн. наук		О.А. Кононов
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №12

по курсу: ОСНОВЫ МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	4711		Хасанов Б.Р.
		подпись, дата	инициалы, фамилия

Санкт-Петербург 2020

Цель работы

Изучить работу генератора случайных чисел RNG на примере процессора STM32F407VG.

1 Листинги программы, написанной на языке программирования C

Во время выполнения работы использовалась программа STM32CubeMX, позволяющая сгенерировать проект, а затем экспортировать его в Keil для дальнейшего написания программы.

Процесс написания программы можно условно разбить на два этапа: настройка проекта в среде «CubeMX» и написание программы в среде разработки «Keil». На рисунках 1 – 4 представлены этапы настройки проекта в CubeMX.

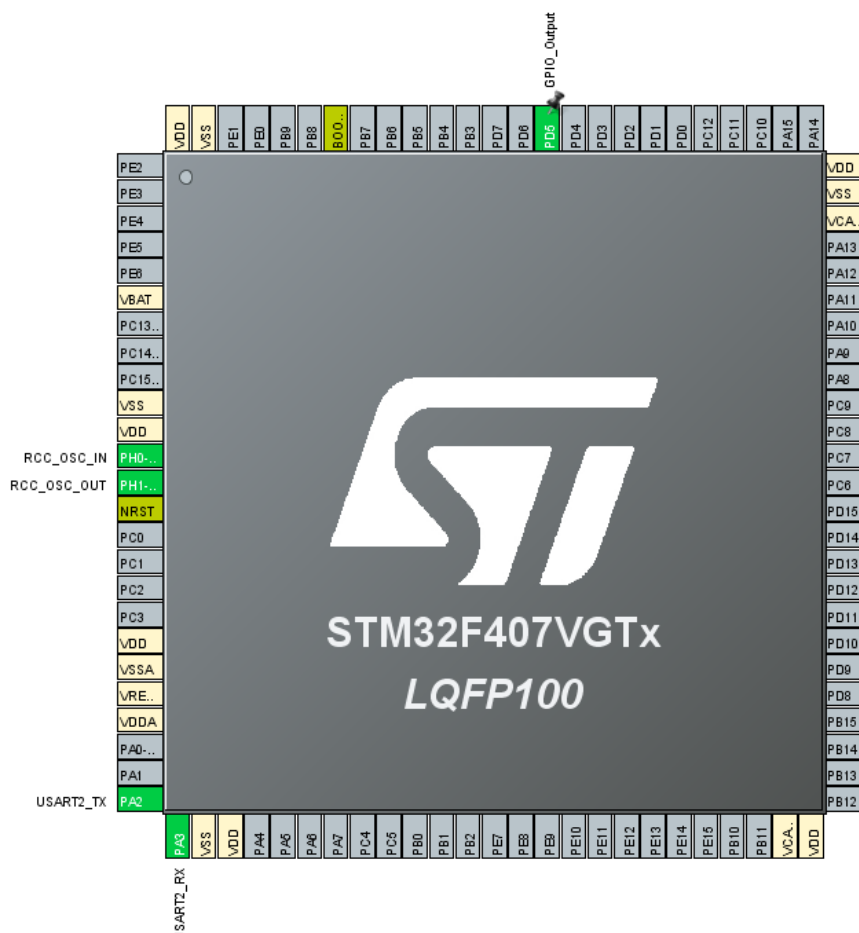


Рисунок 1 - Настройка STM32F407VG, Pinout View

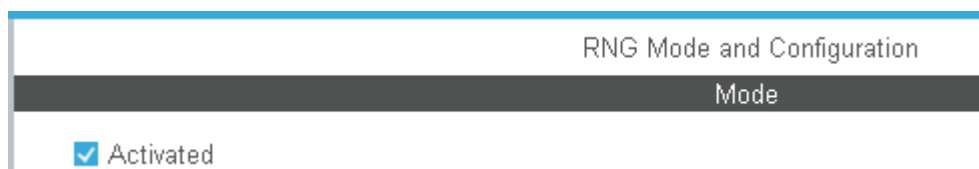


Рисунок 2 – Включение генератора случайных чисел

Далее мы должны настроить тактирование. Задаем входную частоту 8MHz. При настройке ориентируемся на документацию – часть диаграммы из нее представлена на рисунке 3 – частота, подаваемая на генератор случайных чисел, должна быть равна 168 МГц [1].

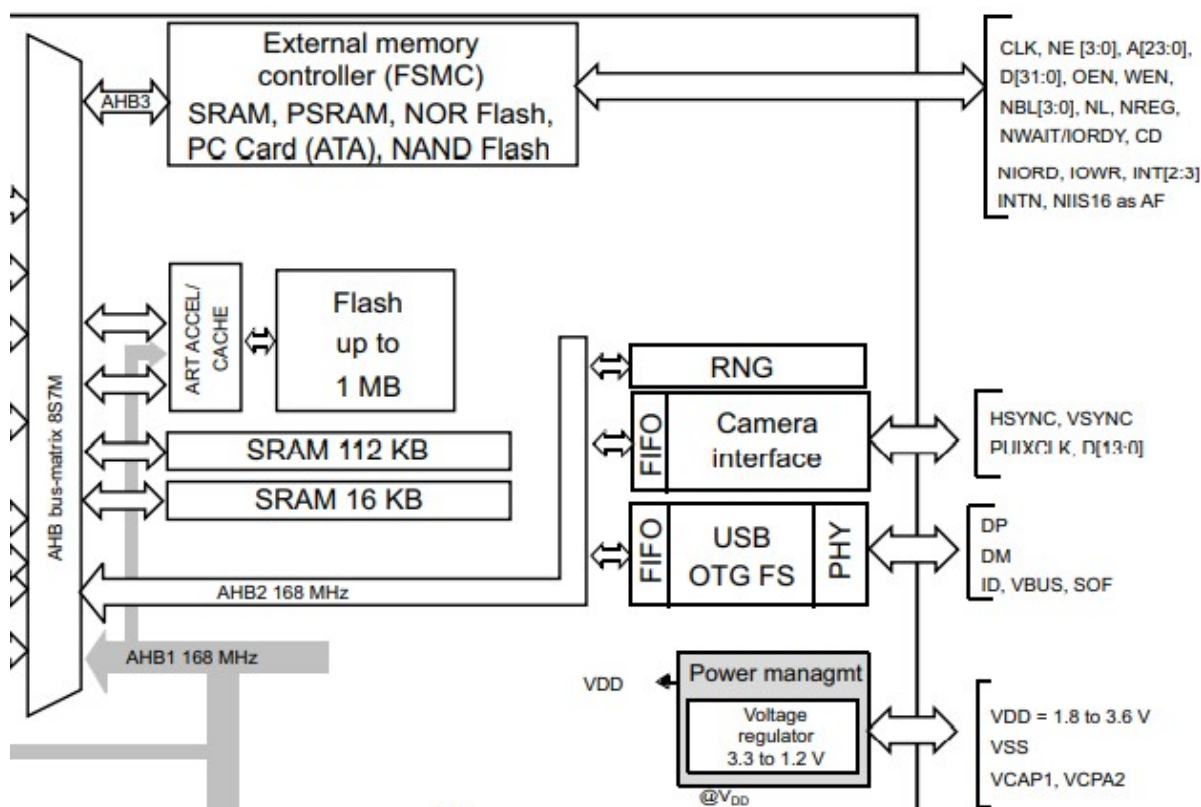


Рисунок 3 – Часть блок-диаграммы STM32F407VG [1]

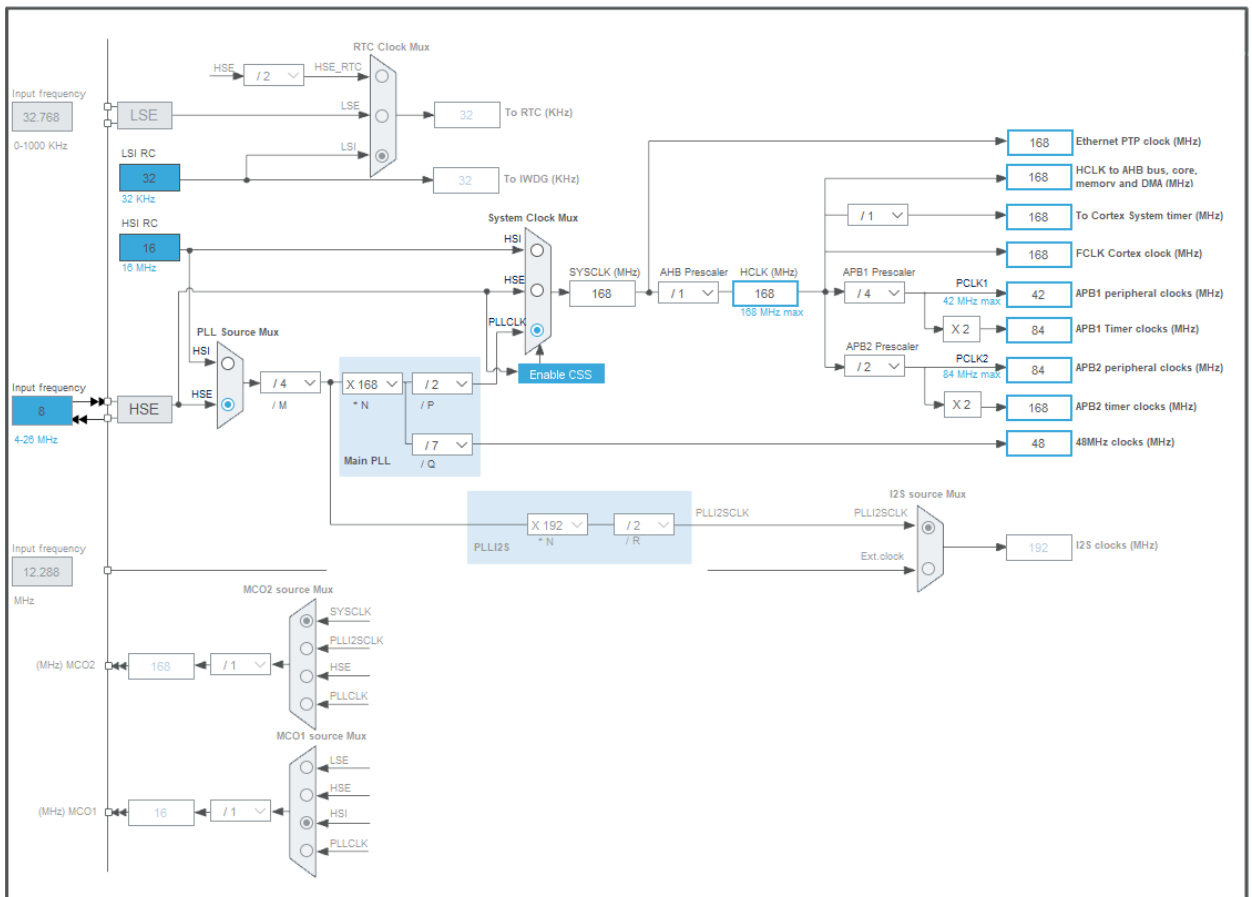


Рисунок 4 - Clock Configuration

Напишем же теперь программу:

```
#include "main.h"
```

```
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_USART2_UART_Init(void);
static void MX_RNG_Init(void);
```

```
__STATIC_INLINE void DelayMicro(__IO uint32_t micros)
{
    micros *=(SystemCoreClock / 1000000) / 5;
    while (micros--);
}
int main(void)
{
    uint8_t str[8];
    uint32_t rng_number = 0;
```

```

HAL_Init();

MX_GPIO_Init();
MX_USART2_UART_Init();
MX_RNG_Init();

while (1)
{
    HAL_RNG_GenerateRandomNumber(&rng, &rng_number);
    //Генерируем случайное число
    sprintf(str, "%x", rng_number); //Преобразуем случайное число в строку
    // Передаём случайное число по uart
    while( HAL_UART_Transmit_IT(&huart2, "Our rng number = ", 17) ==
HAL_BUSY );
    while( HAL_UART_Transmit_IT(&huart2, str, 8) == HAL_BUSY );
        while( HAL_UART_Transmit_IT(&huart2, "\n", 1) ==
HAL_BUSY );
        HAL_Delay(100);
    }
}

```

3 Результаты работы программы

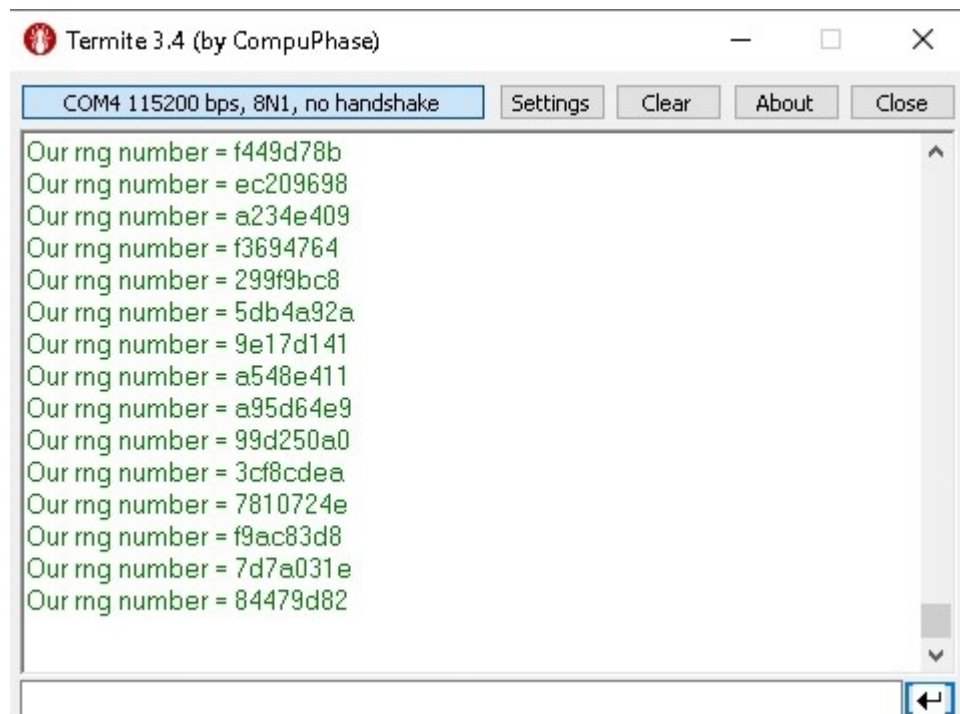


Рисунок 5 – Сгенерированные и переданные по uart случайные значения

Вывод

В результате выполнения лабораторной работы был изучен генератор случайных чисел на плате STM32F407VG. Были получены случайные числа - из рисунка 5 видно, что числа не повторяются, каждое новое значение не совпадает с предыдущим.

Список источников

1 Техническая документация по STM32F405xx STM32F407xx/
STMicroelectronics STM32: 2020 – 203 с.

2 Reference manual/ STMicroelectronics STM32: 2019 – 1749 с.