

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Институт цифровых технологий, электроники и физики

Кафедра вычислительной техники и электроники (ВТиЭ)

УДК: 004.94

Работа защищена

«___» _____ 2024 г.

Оценка _____

Председатель ГЭК, д.т.н., проф.

_____ С. П. Пронин

Допустить к защите

«___» _____ 2024 г.

Заведующий кафедрой ВТиЭ,

к.ф.-м.н., доцент

_____ В. В. Пашнев

ПРОГРАММНЫЙ ГЕНЕРАТОР СИГНАЛОВ НА ОСНОВЕ
МИКРОКОНТРОЛЛЕРА STM32F103

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К ВЫПУСКНОЙ
КВАЛИФИКАЦИОННОЙ РАБОТЕ

БР 09.03.01.506.294 ПЗ

Студент группы: _____ 506 _____ Д. С. Вебер

Руководитель работы: _____ ст. преп. _____ П. Н. Уланов

Консультанты: _____

Нормоконтролер: _____ к.ф.-м.н., доцент _____ А. В. Калачёв

Барнаул 2024 г.

РЕФЕРАТ

Полный объём работы составляет 15 страниц, включая 7 рисунков и 0 таблиц.

В первой главе были рассмотрены семейства микроконтроллеров, различные среды разработки и изучены методы программной генерации сигналов.

Во второй главе был спроектирован генератор. Проведено моделирование выбранного метода генерации, разработан алгоритм работы и создана схема электрическая принципиальная.

В третьей главе была произведена сборка макета, написана и протестирована результирующая программа.

Ключевые слова: генератор сигналов, микроконтроллер.

Дипломная работа оформлена с помощью системы компьютерной вёрстки \TeX и его расширения \XeTeX из дистрибутива *TeX Live*.

ABSTRACT

The total amount of work is 15 page's, include 7 image's and 0 table's.

In the first chapter, families of microcontrollers, various development environments were considered and methods of software signal generation were studied.

In the second chapter, a generator was designed. The simulation of the selected generation method was carried out, an algorithm of operation was developed and an electrical circuit was created.

In the third chapter, the layout was assembled, the resulting program was written and tested.

Keyword: signal generator, microcontroller.

Thesis is framed using the computer layout system \TeX and its extension \XeTeX from the distribution *TeX Live*.

СОДЕРЖАНИЕ

Введение	4
1 Начало разработки	5
1.1 Обзор микроконтроллеров	5
1.2 Среды разработки для STM32	5
1.3 Методы программной генерации сигнала	5
2 Проектирование	8
2.1 Моделирование прямого цифрового синтеза	8
2.2 Алгоритм работы	10
2.3 Схема генератора	10
3 Реализация и тестирование	12
Заключение	13
Список использованной литературы	14
Приложение 1	15

ВВЕДЕНИЕ

Актуальность

На сегодняшний день разрабатывается достаточно много генераторов сигналов, которые используются для различных исследований или необходимы для настройки каких-либо устройств. Существует два основных вида генераторов сигналов — аналоговый и цифровой.

Ранее практически все лабораторные генераторы были аналоговыми и конструировались на различных схемах. К их достоинствам можно отнести простоту и надёжность, но у них есть существенные недостатки в виде меньшей стабильности и более тщательной настройке. Сейчас практически все генераторы, которые есть на рынке создаются на основе цифровых методов синтеза аналоговых сигналов. Цифровые генераторы легко интегрируются с другими системами и могут управляться через программное обеспечение, что упрощает процесс. Они стабильные и точные, а также могут быть реализованы просто на микроконтроллере. Такого рода генераторы могут найти применение и в промышленности. Возможно являться компонентами в сложных схемах или помогать в настройке и тестировании оборудования. Разработанный в данной работе генератор не претендует на применение в промышленности, но в качестве простого и дешёвого функционального генератора найдёт своё применение.

Применением такого генератора может быть генерация сигналов разных форм, работа с аналоговыми системами для исследования влияния сигналов на них, изучение методов обработки сигнала или основ радиоэлектроники.

Цель выпускной квалификационной работы состоит в создании программы для генерации сигналов на микроконтроллере STM32.

Задачи

1. Рассмотреть семейства микроконтроллеров и осуществить выбор.
2. Выбрать среду разработки.
3. Изучить методы генерации сигналов.
4. Спроектировать генератор.
5. Сконструировать макет.
6. Разработать и протестировать программу.

1. НАЧАЛО РАЗРАБОТКИ

1.1. Обзор микроконтроллеров

1.2. Среда разработки для STM32

1.3. Методы программной генерации сигнала

Основные методы цифровой генерации сигналов — метод аппроксимации и табличный метод.

Метод аппроксимации подразумевает собой вычисление отсчётов функции с заданным интервалом. В памяти хранятся только параметры сигнала. Поэтому данный метод позволяет затратить небольшой объём памяти, но его недостаток это затраты на вычисления, что ограничивает максимальную частоту сигнала.

В табличном методе генерации сигналов предполагается, что заранее вычисленные отсчёты хранятся в памяти. То есть никаких вычислений не требуется и генерация сводится к тому, что в порт цифро-аналогового преобразователя нужно вывести ячейку по заданному адресу. Таким образом, время на формирование отсчёта становится меньше и появляется возможность генерировать сигнал с более высокой частотой. Недостатком же являются большие затраты памяти.

Будем рассматривать табличный метод синтеза. Для начала потребуется таблица отсчётов, чтобы её вычислить используем готовый инструмент.

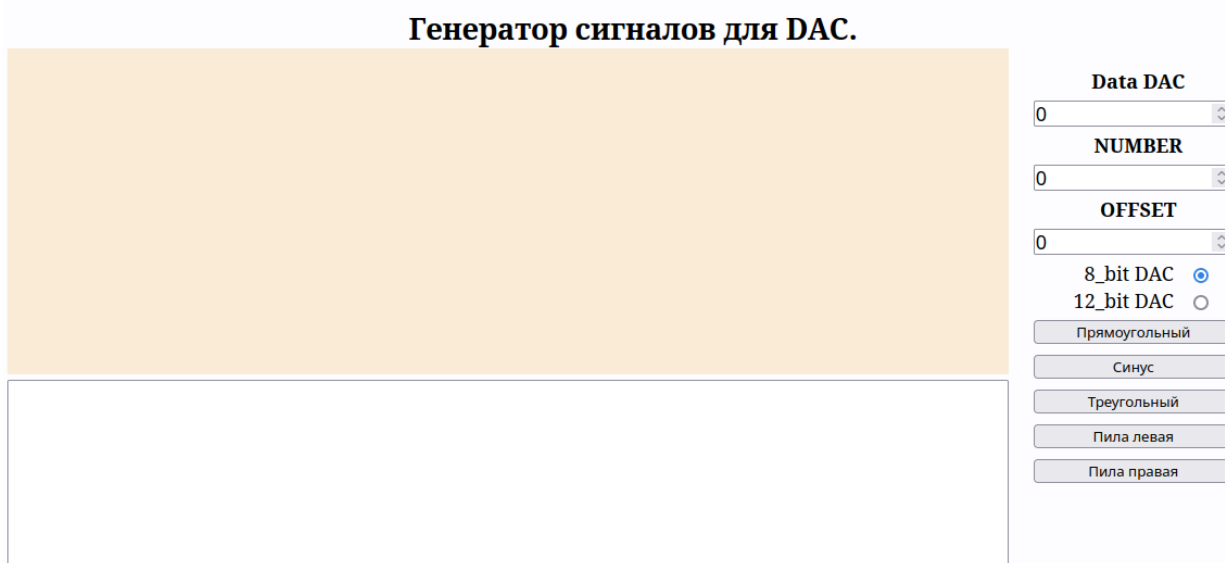


Рис. 1.1 Программа для вычисления значений сигнала.

У таблицы есть 4 параметра:

1. Разрядность ЦАП: 8 или 12 бит.
2. Максимальное значение.
3. Количество значений.
4. Смещение от нуля.

Использовать мы будем 12-битные значения в количестве 256 чисел. Максимальное значение амплитуды сигнала может быть 4095, но так как для улучшения генерации будет задействован встроенный в цифро-аналоговый преобразователь выходной буфер, то он будет срезать сигнал сверху и снизу на 0.2В, поэтому значения тоже следует срезать на эту же величину для корректной генерации.

В документе от ST про работу с цифро-аналоговым преобразователем есть формула для расчета выходного напряжения.

$$DAC_{output} = V_{REF} * \frac{DOR}{DAC_{MaxDigitalValue} + 1},$$
 где DOR — цифровое значение.

Нам нужно найти какое значение соответствует напряжению 0.2В. Выразим DOR и подставим имеющиеся значения.

$$DOR = \frac{3.3}{0.2} * (4095 + 1) = 248$$

Укажем смещение от нуля 248, а максимальное значение 4095 меньше на 248, то есть 3847 и сгенерируем таблицу отсчётов для синусоиды.

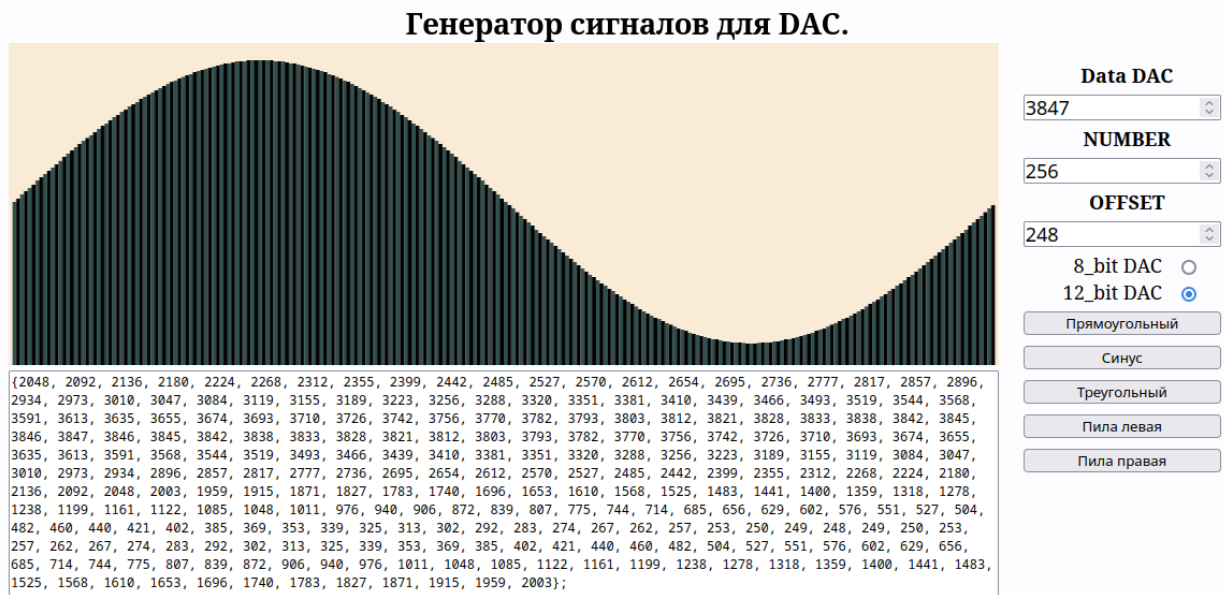


Рис. 1.2 Вычисление таблицы сигнала.

Теперь у нас есть данные для генерации сигнала, но теперь нужно продумать как передавать их в цап и как вообще работать с цапом.

2. ПРОЕКТИРОВАНИЕ

2.1. Моделирование прямого цифрового синтеза

Смоделируем алгоритм метода прямого цифрового синтеза на языке Си для дальнейшей реализации на микроконтроллере.

Листинг 2.1 Метод DDS.

```

1  int main() {
2      uint16_t p_acc, p_step;
3      uint8_t addr = 0; // адрес ячейки
4
5      p_acc = 0;        // аккумулятор фазы
6      p_step = 128;     // код частоты
7
8      while(1)
9      {
10         addr = p_acc >> 8; // выделение старшей части
           ↪ аккумулятора фазы
11         p_acc += p_step;   // шаг
12         printf("%d 0x%X\n", addr, sinus[addr]); // вывод
           ↪ отсчёта
13     }
14
15     return 0;
16 }
```

Алгоритм программы представлен следующей блок-схемой.



Рис. 2.1 Алгоритм метода DDS.

Код частоты задаёт выходную частоту генератора. При значении 256 вывод будет следующий:

```

kenny@desktop:~/workspace/vkr/dds
> gcc dds.c -o dds && ./dds
0 0x800
1 0x82C
2 0x858
3 0x884
4 0x8B0
  
```

Рис. 2.2 Формирование отсчётов при коде частоты 256.

Увеличим код частоты в два раза и получим следующее:

```

kenny@desktop:~/workspace/vkr/dds
> gcc dds.c -o dds && ./dds
0 0x800
2 0x858
4 0x8B0
6 0x908
8 0x95F
  
```

Рис. 2.3 Формирование отсчётов при коде частоты 512.

Как можно заметить отсчёты стали формироваться через один, соответственно частота вырастит в два раза. Теперь уменьшим частоту в два раза выставив код частоты 128.



```
kenny@desktop:~/workspace/vkr/dds
> gcc dds.c -o dds && ./dds
0 0x800
0 0x800
1 0x82C
1 0x82C
2 0x858
2 0x858
```

Рис. 2.4 Формирование отсчётов при коде частоты 128.

Программа стала выводить каждый отсчёт по два раза тем самым, понизив частоту.

В данном виде модуляции код частоты просто абстрактное число, которое добавляется к аккумулятору фазы и узнать реальную частоту проблематично. Результат синтеза будет проверен опытным путём на микроконтроллере.

2.2. Алгоритм работы

2.3. Схема генератора

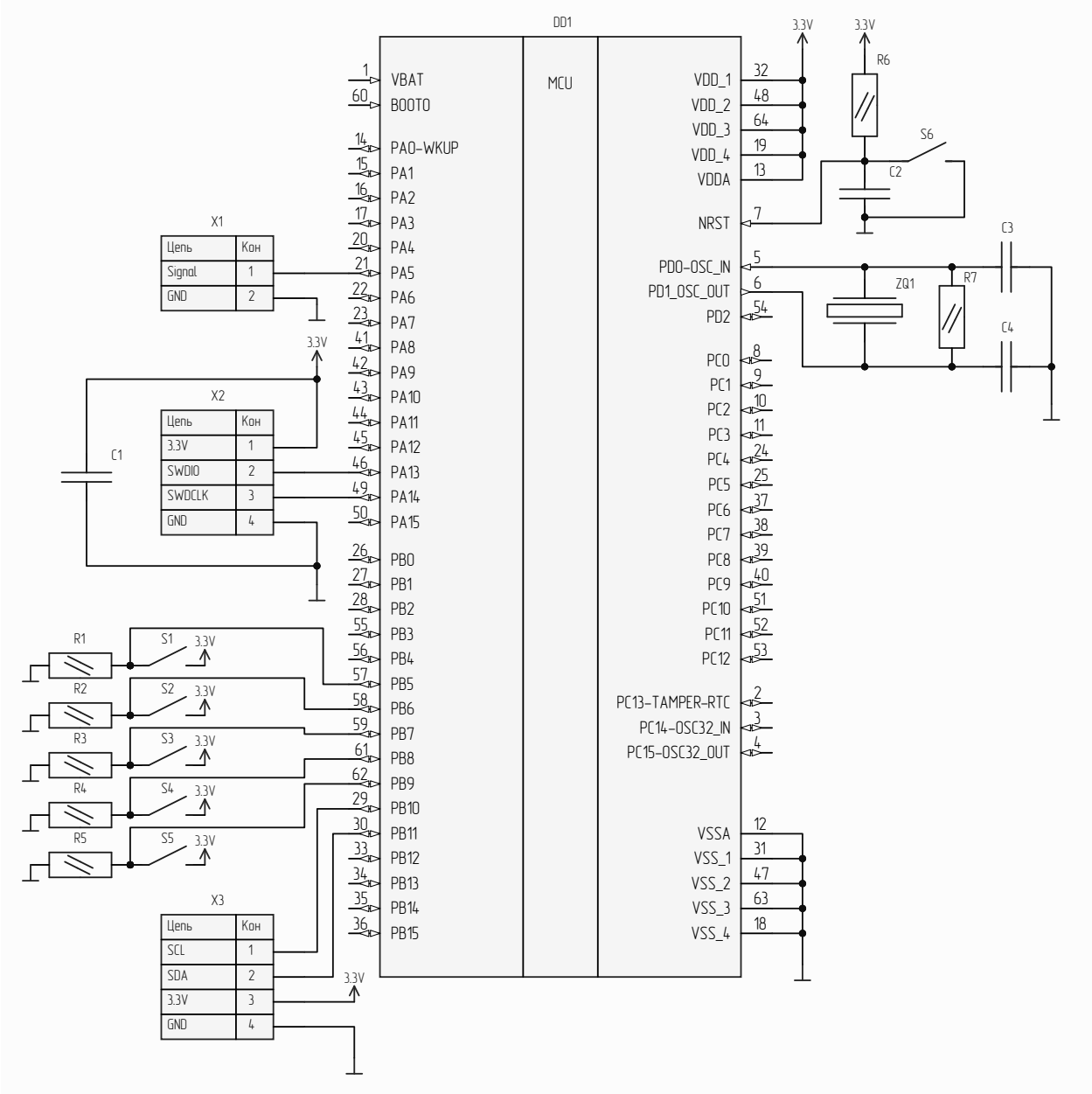


Рис. 2.5 Схема электрическая принципиальная.

3. РЕАЛИЗАЦИЯ И ТЕСТИРОВАНИЕ

ЗАКЛЮЧЕНИЕ

1. Пример ссылки на литературу [1].
 2. Пример ссылки на литературу [2].
 3. Пример ссылки на литературу [3].
- Всего рисунков в документе 7.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. [Электронный ресурс] Bitbucket — Википедия. — URL: <https://ru.wikipedia.org/wiki/Bitbucket> (дата обр. 28.03.2020).
2. [Электронный ресурс] Id Software — Википедия. — URL: https://ru.wikipedia.org/wiki/Id_Software (дата обр. 31.03.2020).
3. [Электронный ресурс] GitHub — Википедия. — URL: <https://ru.wikipedia.org/wiki/GitHub> (дата обр. 28.03.2020).

ПОСЛЕДНИЙ ЛИСТ ВКР

Выпускная квалификационная работа выполнена мной совершенно самостоятельно. Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

«___» _____ 2024 г.

_____ Д. С. Вебер