

ГУАП
КАФЕДРА № 41

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

доц., к.т.н., доц.

Кононов О. А.

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №10
ВОСПРОИЗВЕДЕНИЕ ЗВУКА С ПОМОЩЬЮ ЦАП

по дисциплине: ОСНОВЫ МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.

4711

Хасанов Б.Р.

Подпись, дата

инициалы, фамилия

Санкт-Петербург 2020

Цель работы

Исследовать возможности использования ЦАП, основные характеристики ЦАП, с помощью микроконтроллера STM32F407VG.

1 Выполнение работы

1.1 Сведения о ЦАП

Цифро-аналоговый преобразователь (ЦАП) — устройство для преобразования цифрового (обычно двоичного) кода в аналоговый сигнал (ток, напряжение или заряд). Цифро-аналоговые преобразователи являются интерфейсом между дискретным цифровым миром и аналоговыми сигналами.

Аналого-цифровой преобразователь (АЦП) производит обратную операцию.

Звуковой ЦАП обычно получает на вход цифровой сигнал в импульсно-кодовой модуляции (англ. PCM, pulse-code modulation). Задача преобразования различных сжатых форматов в PCM выполняется соответствующими кодеками.

ЦАП применяется всегда, когда надо преобразовать сигнал из цифрового представления в аналоговое.

1.2 Создание проекта

Проект был создан с помощью программы STM32CubeMX, где производилась настройка тактирования, необходимых портов ввода/вывода, i2c интерфейс. Пример настройки представлен на рисунках 1 – 4.

Рисунок 1 – Схема выводов STM32F407VGT после настройки проекта в STM32CubeMX

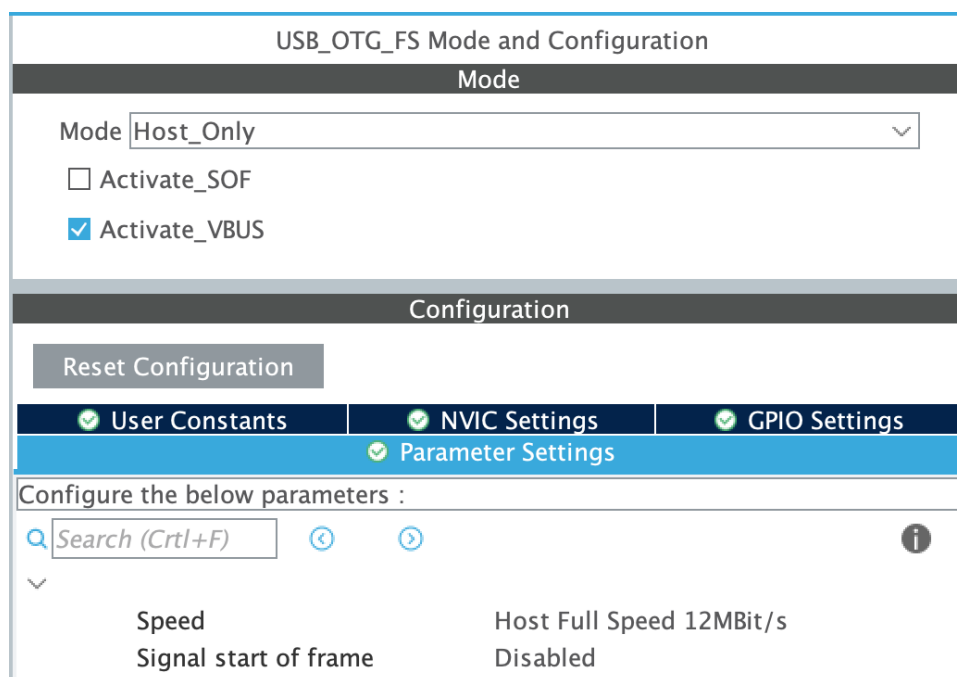


Рисунок 2 – Настройка USB в STM32CubeMX

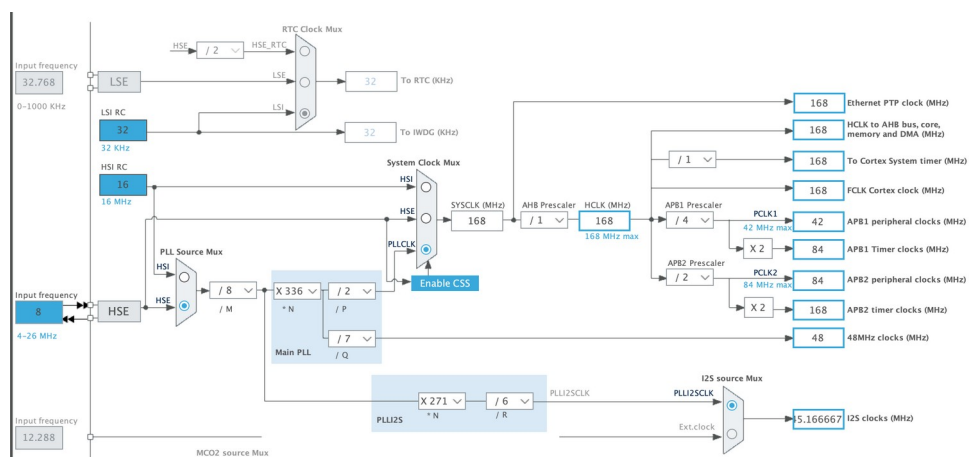


Рисунок 3 – Настройка тактирования в STM32CubeMX

1.3 Код программы на C

Код программы main.c выполнен в среде CubeIDE и представлен ниже. На рисунке 5 представлена структурная схема программы.

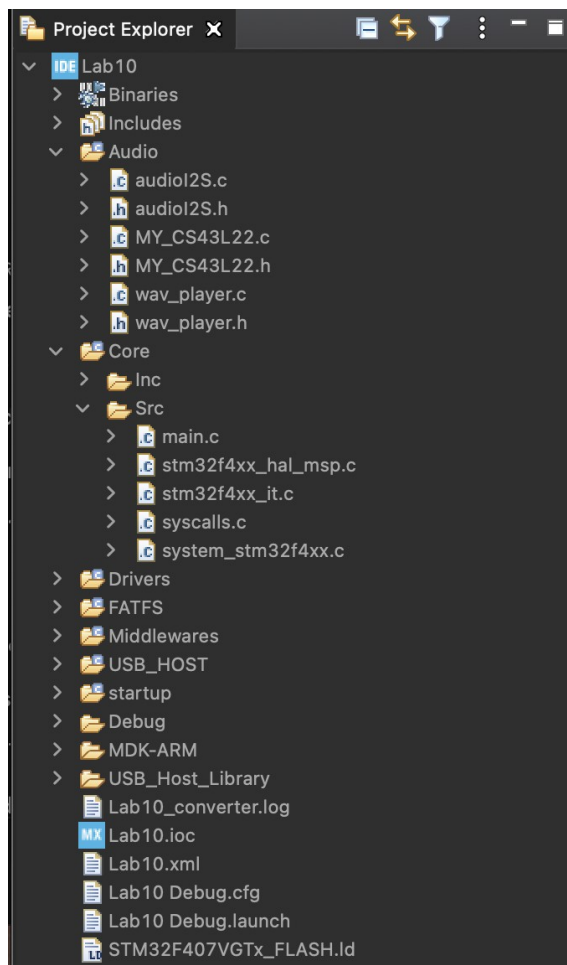


Рисунок 4 – Структурная схема проекта в CubeIDE

audioI2S.h – конфигурация, он начинает DMA передачу и останавливает, плеер отвечает за WAV формат файла чтения с USB-флэшки и делает все необходимые для конфигурации WAV-файл.

Код программы

main.c – код основной программы

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

```
#include "fatfs.h"
```

```
#include "usb_host.h" // библиотека для работы с USB
```

```
#include "audioI2S.h"
```

```
#include "MY_CS43L22.h" // библиотека для настройки громкости
```

```
#include "wav_player.h"
```

```
extern ApplicationTypeDef Appli_state;
```

```
I2C_HandleTypeDef hi2c1;
```

```
I2S_HandleTypeDef hi2s3;
```

```
DMA_HandleTypeDef hdma_spi3_tx;
```

```
#define WAV_FILE1 "audio/94150.wav" // путь где лежит файл со звуком и его название
```

```
void SystemClock_Config(void);
```

```
static void MX_GPIO_Init(void);
```

```
static void MX_DMA_Init(void);
```

```
static void MX_I2C1_Init(void);
```

```
static void MX_I2S3_Init(void);
```

```
void MX_USB_HOST_Process(void);
```

```
int main(void) {
```

```
    HAL_Init();
```

```
    SystemClock_Config();
```

```
    MX_GPIO_Init();
```

```
    MX_DMA_Init();
```

```
    MX_I2C1_Init();
```

```
    MX_I2S3_Init();
```

```
    MX_FATFS_Init();
```

```
    MX_USB_HOST_Init();
```

```
    CS43_Init(hi2c1, MODE_I2S);
```

```
    CS43_SetVolume(180); //0-255 – настройка громкости
```

```
    CS43_Enable_RightLeft(CS43_RIGHT_LEFT);
```

```
audioI2S_setHandle(&hi2s3);
```

```
bool isSdCardMounted=0;
```

```
bool pauseResumeToggle=0;
```

```
while (1) {
```

```
    MX_USB_HOST_Process();
```

```
    if(Appli_state == APPLICATION_START) {// проверка на подключения USB носителя
```

```
        HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, GPIO_PIN_SET); // загорается зелёный светодиод
```

```
    }
```

```
    else if(Appli_state == APPLICATION_DISCONNECT) {// если USB не задействован, светодиод гаснет/не горит
```

```
        HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, GPIO_PIN_RESET);
```

```
        f_mount(NULL, (TCHAR const*)"", 0);
```

```
        isSdCardMounted = 0;
```

```
    }
```

```
    if(Appli_state == APPLICATION_READY) {
```

```
        if(!isSdCardMounted) {
```

```
            f_mount(&USBHFS, (const TCHAR*)USBHPath, 0);
```

```
            isSdCardMounted = 1;
```

```
        }
```

```
        if(HAL_GPIO_ReadPin(GPIOA, GPIO_PIN_0)) {
```

```
            // нажатие на кнопку запускает музыкальный файл,
```

```
            //загорается оранжевый светодиод
```

```
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_13, GPIO_PIN_SET);
```

```
            HAL_Delay(500);
```

```
            wavPlayer_fileSelect(WAV_FILE1);
```

```
            wavPlayer_play();
```

```
            while(!wavPlayer_isFinished()) {
```

```
                wavPlayer_process();
```



```

if(HAL_GPIO_ReadPin(GPIOA, GPIO_PIN_0)) {

    // еще одно нажатие на кнопку, ставит композицию на паузу

    //загорается красный светодиод

    pauseResumeToggle^=1;

    if(pauseResumeToggle) {

        HAL_GPIO_WritePin(GPIOD, GPIO_PIN_14, GPIO_PIN_SET);

        wavPlayer_pause();

        HAL_Delay(200);

    } else {

        HAL_GPIO_WritePin(GPIOD, GPIO_PIN_14, GPIO_PIN_RESET);

        HAL_Delay(1000);

        if(HAL_GPIO_ReadPin(GPIOA, GPIO_PIN_0)) {

            wavPlayer_stop();

        }

        {

            wavPlayer_resume();

        }

    }

}

HAL_GPIO_WritePin(GPIOD, GPIO_PIN_13, GPIO_PIN_RESET);

HAL_Delay(1000);

}

}

}

}

```

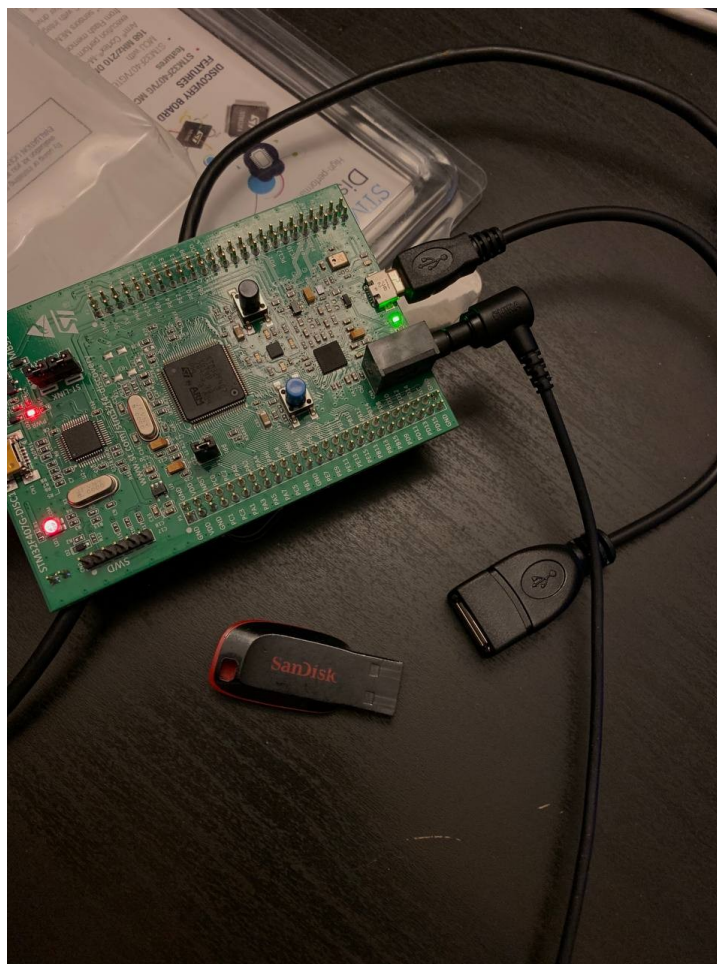


Рисунок 5 – Не подключена USB-накопитель, плеер ждем подключения, светодиоды не горят

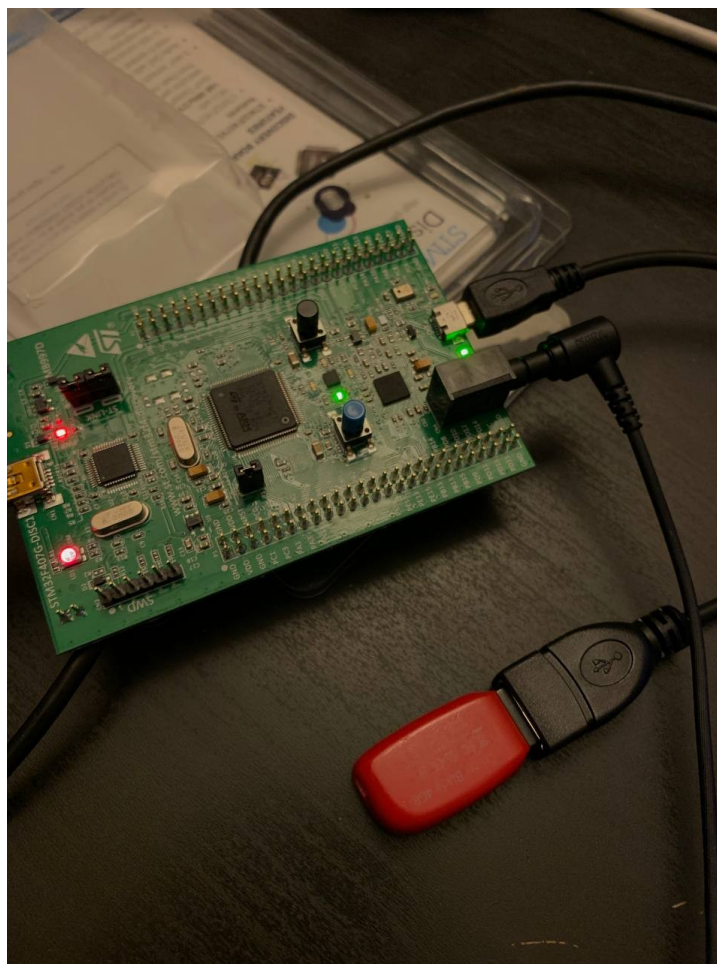


Рисунок 6 – USB-накопитель подключен, загорелся зелёный светодиод, плеер готов воспроизводить музыку по нажатию на кнопку

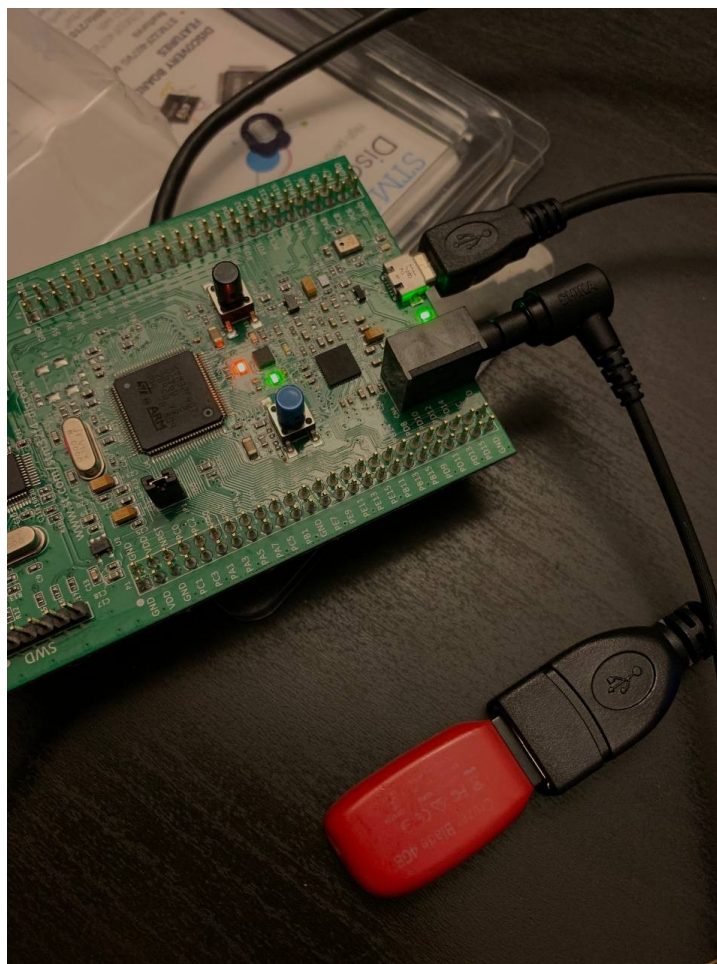


Рисунок 7 – По нажатию на кнопку загорается оранжевый светодиод, плеер воспроизводит музыку

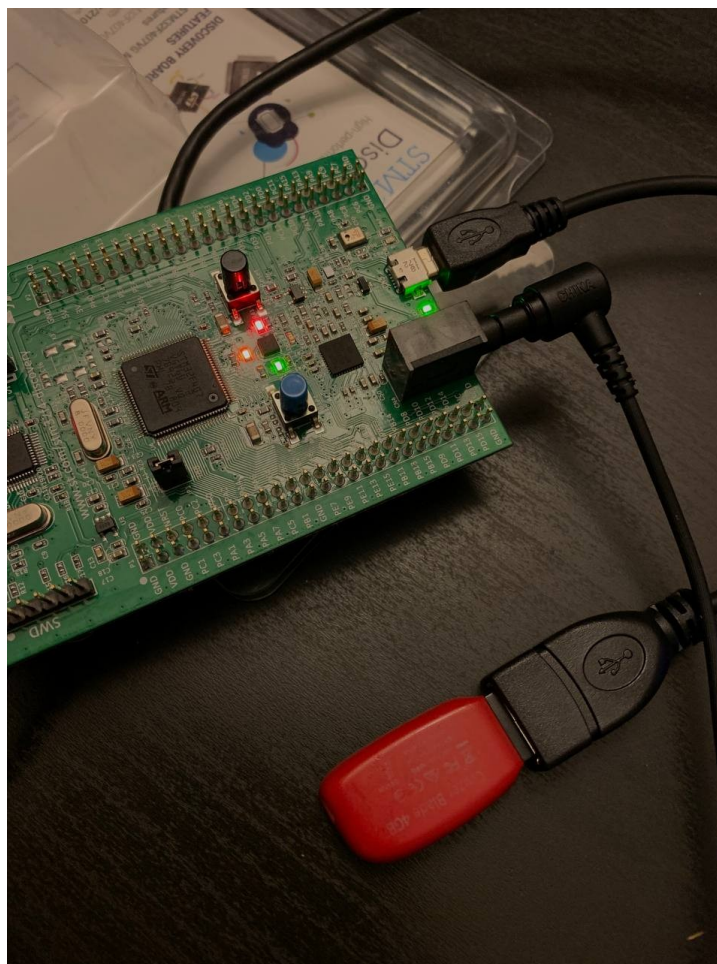


Рисунок 8 – По нажатию на кнопку еще раз, загорается красный светодиод, музыка ставится на паузу

Выводы

Был реализован простейший аудио плеер, на основе внешнего ЦАП расположенного на плате с использованием i2s интерфейса.

В результате выполнения лабораторной работы были получены навыки:

- работы в программной среде CubeIDE, CubeMX;
- программирования на языке C.

Была произведена проверка программы на плате STM32F4 Discovery. Исходя из результатов работы платы, можно сделать вывод, что программа работает верно.