2Федеральное государственное автономное образовательное учреждение

высшего образования «Санкт-Петербургский государственный университет аэрокосмического приборостроения» (ГУАП)

ОТЧЕТ ЗАЩИЩЕН

доцент Попов А.А. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Должность Ф.И.О дата, подпись

**Отчет**

**о лабораторной работе №3**

**«Настройка подсистемы тактирования**

**микроконтроллера STM32F303xC»**

по дисциплине «Программирование встроенных приложений»

ОТЧЕТ ВЫПОЛНИЛ:

Студент Д. В. Самарин 4134 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ф.И.О. группа дата, подпись

Санкт-Петербург

2024

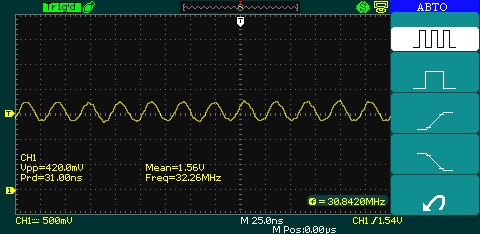
Цель работы: Привитие практических навыков по работе с ИСР Keil, технической документацией, подсистемой тактирования и ПВВ микроконтроллера, библиотекой CMSIS

Задание на лабораторную работу:

1. Номер варианта с заданием. Листинг программы.
2. Осциллограмму вывода MCO (PA8). Подобно рисунку 1.2.
3. Две осциллограммы заданных выводов с указанием типа настройки, подобно рисунку 1.3.
4. Сводную таблицу подобно таблице 1.5.
5. Снимок экрана настройки частоты согласно варианту с помощью инструмента STM32F30x\_Clock\_Configuration\_V1.0.0.xls.
6. Вариант:

|  |  |  |  |
| --- | --- | --- | --- |
| Номер варианта | Частота настройки SYSCLK, МГц | Номер линий | |
| 9 | 64(HSI) | PB12 | PD14 |

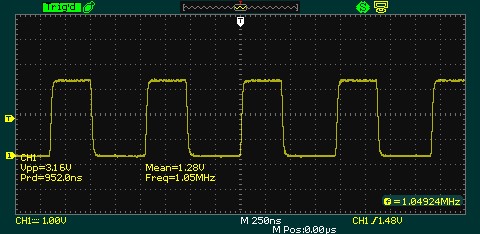
1. Осциллограмма вывода MCO (PA8)

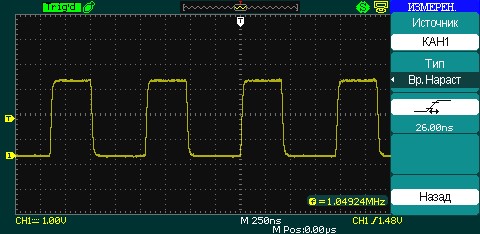


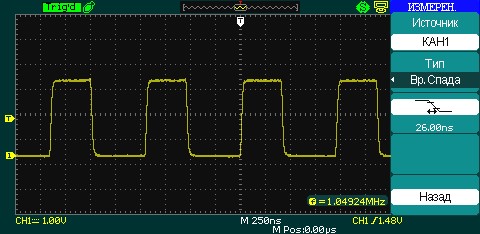
1. Две осциллограммы заданных выводов с указанием типа настройки

**Оптимизация компилятора отключена ‘Level 0(-O0)’**

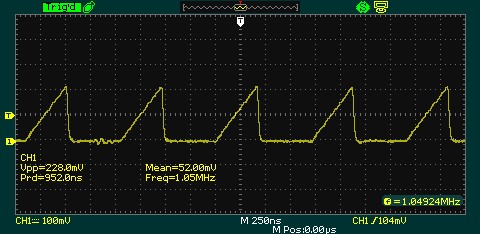
Линия PB12

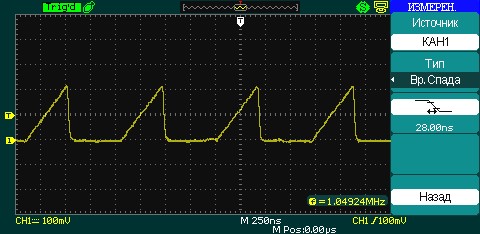


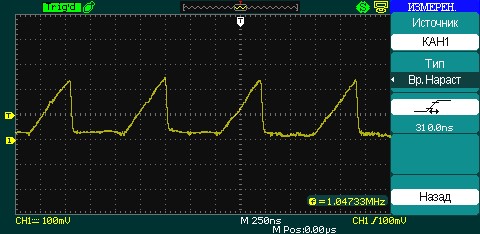




Линия PD14

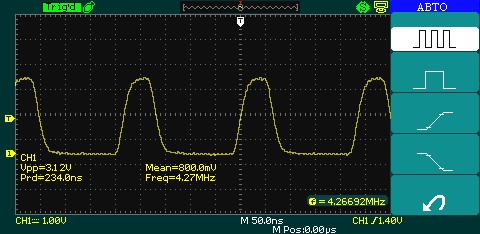


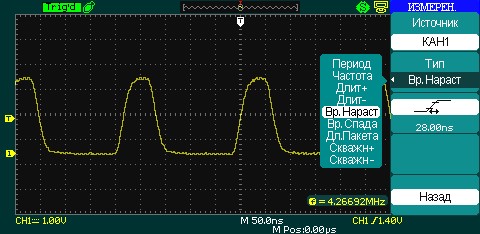


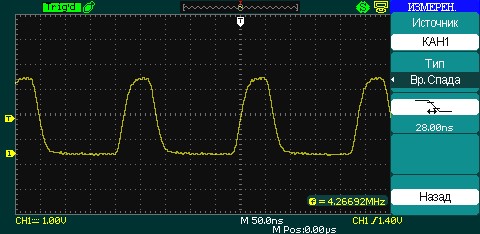


**Оптимизация компилятора включена ‘Level 3(-O3)’**

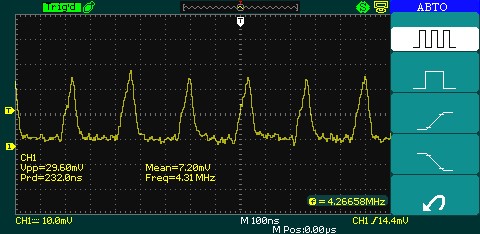
Линия PB12

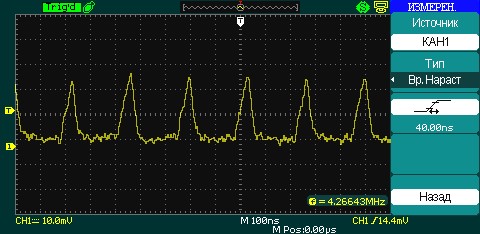


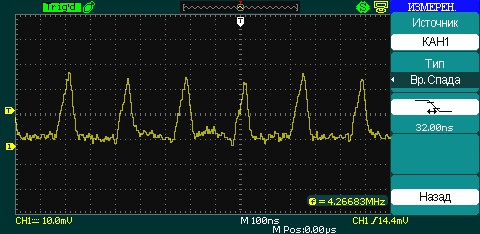




Линия PD14

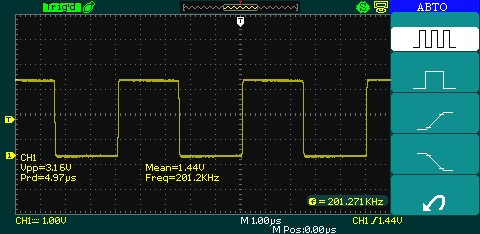


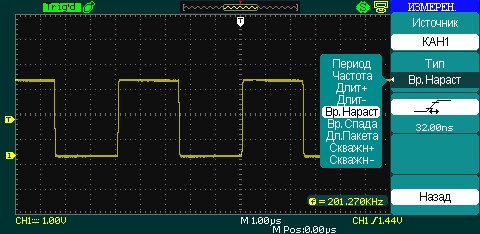


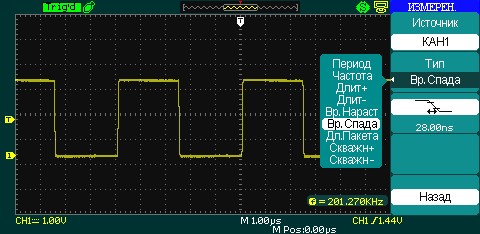


**Добавлена задержка**

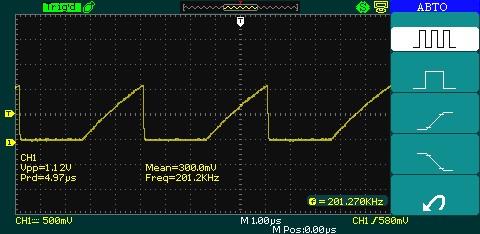
Линия PB12

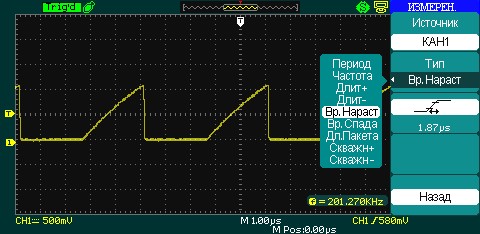


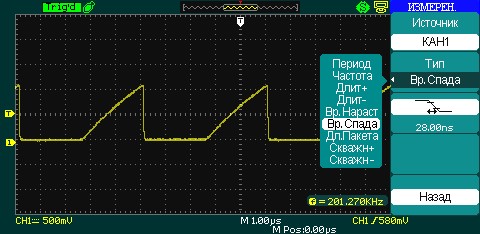




Линия PD14







1. Сводная таблица характеристик сигналов на заданных выходах

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Линия | Измеряемый параметр | | | | |
| Частота следования импульсов, КГц | Амплитуда, В | Длительность импульса, нс | Время нарастания фронта, нс | Время спада среза, нс |
| Оптимизация компилятора отключена ‘*Level 0(-O0)*’ | | | | | |
| PB12 | 1050 | 3.16 | **950** | 26 | 28 |
| PD14 | 1050 | 0.228 | **950** | 310 | 28 |
| Оптимизация компилятора включена ‘*Level 3(-O3)*’ | | | | | |
| PB12 | 4267 | 3.12 | **240** | 28 | 28 |
| PD14 | 4267 | 0.029 | **240** | 40 | 32 |
| Добавлена задержка | | | | | |
| PB12 | 201 | 3.16 | **5000** | 32 | 28 |
| PD14 | 201 | 1.12 | **5000** | 1870 | 28 |

#include "RTE\_Components.h"

#include CMSIS\_device\_header

// PB12 PD14 64 (HSI)

//PE11 PE13 shablon

void delay(volatile uint32\_t count){//1 002 560 = 1sec

while(count--)

\_\_NOP();

}

int main(void)

{

// Вспомогательные переменные

volatile uint32\_t StartUpCounter = 0, HSEStatus = 0;

SET\_BIT(RCC -> CR,RCC\_CR\_HSEON);// включаем HSE

do {// ждем вхождения в работу HSE

HSEStatus = RCC->CR & RCC\_CR\_HSERDY;

StartUpCounter++;

} while((HSEStatus == 0) && (StartUpCounter != 0x5000));

//если за 0x5000 итераций, HSE не запустился, то проблемы в аппаратуре

if ((RCC->CR & RCC\_CR\_HSERDY) != RESET) {

// настраиваем буфер FLASH предварительной выборки команд

FLASH->ACR = FLASH\_ACR\_LATENCY\_1;

// настройка PLL на 4 МГц = 8 МГц(HSE)/4\*2

// сначала выключаем чтобы изменить биты PLL, после настройки включим

CLEAR\_BIT(RCC -> CR,RCC\_CR\_PLLON);

// PLLSRC=HSE/4

// PLLCLK=PLLSRC \* 2

RCC->CFGR2 |= (uint32\_t)RCC\_CFGR2\_PREDIV\_DIV2;

RCC->CFGR |= (uint32\_t)(RCC\_CFGR\_PLLSRC\_HSE\_PREDIV | RCC\_CFGR\_PLLMUL16);

SET\_BIT(RCC -> CR,RCC\_CR\_PLLON); // включаем PLL

// ждём запуск и стабилизацию PLL

while((RCC->CR & RCC\_CR\_PLLRDY) == 0){}

// выбираем выход PLL источником тактирования МК

RCC->CFGR |= (uint32\_t)RCC\_CFGR\_SW\_PLL;

//Ожидаем установки PLL источником тактирования МК

while ((RCC->CFGR & (uint32\_t)RCC\_CFGR\_SWS) != (uint32\_t)RCC\_CFGR\_SWS\_PLL){}

}

else

{ while(1){} } // HSE не запустился

//контролируем частоту тактирования 4/1=4 МГц

SystemCoreClockUpdate();//устанавливается в глобальной переменной SystemCoreClock

// Настройка MCO на PLLCLK/2

SET\_BIT(RCC -> CFGR, RCC\_CFGR\_MCO\_PLL);

SET\_BIT(RCC -> AHBENR,RCC\_AHBENR\_GPIOAEN); //разрешаем тактирование GPIOA

//для PA8 устанавливаем Alternate function mode

SET\_BIT(GPIOA -> MODER,GPIO\_MODER\_MODER8\_1);

//для восьмой линии выбираем AF0 режим работы MCO

CLEAR\_BIT(GPIOA -> AFR[1], GPIO\_AFRH\_AFRH0\_Msk);

//устанавливаем наши пины

//разрешаем тактирование GPIOD

SET\_BIT(RCC->AHBENR,RCC\_AHBENR\_GPIODEN);

//устанавливаем работу линий PE11,PE13 на вывод

SET\_BIT(GPIOD->MODER,GPIO\_MODER\_MODER14\_0);

//линию PE13 в режим вывода с открытым стоком

SET\_BIT(GPIOD->OTYPER,GPIO\_OTYPER\_OT\_14);

//подтягиваем PE13 к питанию Pull-up

SET\_BIT(GPIOD->PUPDR, GPIO\_PUPDR\_PUPDR14\_0);

//разрешаем тактирование GPIOE

SET\_BIT(RCC->AHBENR,RCC\_AHBENR\_GPIOBEN);

//устанавливаем работу линий PE11,PE13 на вывод

SET\_BIT(GPIOB->MODER,GPIO\_MODER\_MODER12\_0);

while(1){

//устанавливаем (1) на выходе линий PE11,PE13

GPIOB->BSRR= GPIO\_ODR\_12;

GPIOD->BSRR= GPIO\_ODR\_14;

//delay(10); // задержка

//сбрасываем в (0) выходы линий PE11,PE13

GPIOB->BRR= GPIO\_ODR\_12;

GPIOD->BRR= GPIO\_ODR\_14;

//delay(10);

}

4.Снимок экрана настройки частоты согласно варианту с помощью инструмента STM32F30x\_Clock\_Configuration\_V1.0.0.xls

