

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное автономное образовательное учреждение  
высшего образования «Санкт-Петербургский политехнический университет  
Петра Великого»

Институт компьютерных наук и технологий

Высшая школа технологий искусственного интеллекта

Направление: 02.03.01 Математика и компьютерные науки

Отчет о выполнении лабораторной работы №1  
«Создание нового проекта в Keil Vision5»  
Курс «Программирование микроконтроллеров»

Выполнил студент группы

№5130201/30001 \_\_\_\_\_ Мелещенко С. И.

Преподаватель \_\_\_\_\_ Вербова Н. М.

«\_\_\_\_\_» \_\_\_\_\_ 2025г.

Санкт-Петербург, 2025

# Введение

Тема: Создание нового проекта в Keil Vision5.

Цель: Ознакомится с основными приемами работы с документацией при составлении программ для микроконтроллеров.

Постановка задачи: создать новый проект в Keil Vision5 и разработать программу для микроконтроллера (МК) STM32F200, которая включает и выключает светодиод.

Задание на лабораторную работу.

1. Установить и настроить интегрированную среду разработки Keil µVision5.
2. Создать новый проект для микроконтроллера с ядром ARM Cortex-M3.
3. Разработать простую программу на языке С, выполняющую арифметические и логические операции.
4. Скомпилировать программу и провести отладку с использованием встроенного симулятора.
5. Проанализировать результаты выполнения программы и сделать соответствующие выводы.

## 1 Основная часть

Был создан проект и сконфигурированы параметры инструментария. Созданы файлы источников на С++. С помощью менеджера проектов построено приложение.

Сначала работоспособность программы была проверена на приведенном коде.

### Листинг 1: Код

```
/*
 * Name:      Blinky.c
 * Purpose:  LED PG7 Flasher for MCBSTM32F200
 */
/*
 Main function
 */
int main ()
{
//—— Declaration of type of variables ———

int i; //counter for get ready delay
unsigned long int j; //counter for blinky delay

//—— Initialization of variables ———

i=0;
j=0;

//—— Main cycle of algorithm ———
```

```

*(unsigned long*)(0x40023830) |= 0x40; //Enable port G clocking

for( i=0; i<4; i++){} //small delay for GPIOG get ready

*(unsigned long*)(0x40021800) = (*(unsigned long*)(0x40021800) & (~0x00008000))
| (0x00004000); //Set PG7 as General purpose output

while(1)
{
    *(unsigned long*)(0x40021814) |= 0x80; //Turn LED ON!

    for( j=0; j<2000000 ;j++ ){} //Delay

    *(unsigned long*)(0x40021814) &= ~0x80; //Turn LED OFF

    for( j=0; j<2000000 ; j++){} //Delay
}

```

Алгоритм программы.

1. Инициализация переменных.
2. Выполнение арифметических операций: сложение, вычитание, умножение и деление.
3. Выполнение логических операций: AND, OR, XOR.
4. Сохранение результатов операций в соответствующие переменные.
5. Вывод результатов на консоль или сохранение в памяти для последующего анализа.

Далее я модифицировала текст программы BlinkyLed.c добавив в нее определения адресов и данных.

Для последовательного и непрерывного зажигания и гашения диодов внутри цикла while (который выполняется бесконечно) выполняются следующие шаги: сначала в бит регистра, отвечающий за левый диод на микроконтроллере, записывается «1», затем выполняется задержка в виде цикла на один миллион итераций, чтобы диод оставался включенным некоторое время. После этого в тот же бит записывается «0», а в бит регистра, соответствующий следующему диоду, устанавливается «1». Этот процесс повторяется последовательно для всех диодов. После зажигания последнего диода выполняется дополнительная пауза с помощью цикла на два миллиона итераций, после чего весь цикл начинается заново.

## Листинг 2: Код

```

int main(){
    int i;
    unsigned long int j;

    i = 0;
    j = 0;

    *(unsigned long*)(0x40023830) |= 0x40;
    *(unsigned long*)(0x40023830) |= 0x80;

```

```

*(unsigned long*)(0x40023830) |= 0x80;

for ( i = 0; i < 4; i++) {}

*(unsigned long*)(0x40021800)=(*(unsigned long*)
(0x40021800)&(~0x00020000)) | (0x00010000);
*(unsigned long*)(0x40021800)=(*(unsigned long*)
(0x40021800)&(~0x00008000)) | (0x00004000);
*(unsigned long*)(0x40021800)=(*(unsigned long*)
(0x40021800)&(~0x00000200)) | (0x00001000);

*(unsigned long*)(0x40021C00) = (* (unsigned long*)(0x40021C00)
& (~0x00000020)) | (0x00000010);
*(unsigned long*)(0x40021C00) = (* (unsigned long*)(0x40021C00)
& (~0x00000080)) | (0x00000040);
*(unsigned long*)(0x40021C00) = (* (unsigned long*)(0x40021C00)
& (~0x00002000)) | (0x00001000);
*(unsigned long*)(0x40021C00) = (* (unsigned long*)(0x40021C00)
& (~0x00008000)) | (0x00004000);

*(unsigned long*)(0x40022000) = (* (unsigned long*)(0x40022000)
& (~0x00200000)) | (0x00100000);

while(1){
    *(unsigned long*)(0x40021C14) |= 0x04;
    for ( j = 0; j < 1000000; j++) {}
    *(unsigned long*)(0x40021C14) &= ~0x04;

    *(unsigned long*)(0x40021814) |= 0x100;
    for (j=0;j<1000000;j++){}
    *(unsigned long*)(0x40021814) &= ~0x100;

    *(unsigned long*)(0x40021814) |= 0x80;
    for (j=0;j<1000000;j++){}
    *(unsigned long*)(0x40021814) &= ~0x80;

    *(unsigned long*)(0x40021814) |= 0x40;
    for (j=0;j<1000000;j++){}
    *(unsigned long*)(0x40021814) &= ~0x40;

    *(unsigned long*)(0x40022014) |= 0x400;
    for (j=0;j<1000000;j++){} //
    *(unsigned long*)(0x40022014) &= ~0x400;

    *(unsigned long*)(0x40021C14) |= 0x80;
    for (j=0;j<1000000;j++){}
    *(unsigned long*)(0x40021C14) &= ~0x80;

    *(unsigned long*)(0x40021C14) |= 0x40;
    for (j=0;j<1000000;j++){}
    *(unsigned long*)(0x40021C14) &= ~0x40;

    *(unsigned long*)(0x40021C14) |= 0x08;
    for ( j = 0; j < 1000000; j++) {}
    *(unsigned long*)(0x40021C14) &= ~0x08;

    for ( j = 0; j < 2000000; j++) {}
}
}

```

## **2 Вывод**

В ходе выполнения лабораторной работы были достигнуты следующие результаты:

1. Успешно установлена и настроена среда разработки Keil µVision5.
2. Создан и настроен новый проект для микроконтроллера с ядром ARM Cortex-M3.
3. Разработана и скомпилирована программа на языке C, выполняющая заданные арифметические и логические операции.
4. Проведена отладка программы с использованием встроенного симулятора, подтверждена корректность работы программы.

В результате выполнения лабораторной работы были приобретены практические навыки работы с интегрированной средой разработки Keil µVision5, включая создание проектов, написание и отладку программ для микроконтроллеров с ядром ARM Cortex-M3. Разработанная программа успешно выполняет заданные арифметические и логические операции, что подтверждает правильность выбранного алгоритма и корректность реализации. Полученные знания и навыки могут быть применены для дальнейшей разработки и отладки программного обеспечения для встраиваемых систем на базе микроконтроллеров.