

Санкт-Петербургский политехнический университет Петра
Великого

Институт компьютерных наук и кибербезопасности
Высшая школа технологий искусственного интеллекта
Программирование микроконтроллеров

Отчет по лабораторной работе №1

Создание нового проекта в Keil uVision 5

Работу выполнила:
Михайлова Алёна
Студент группы 5130201/30002

Проверила:
Вербова Н.М.

Санкт-Петербург – 2025г.

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

Постановка задачи: создать новый проект в Keil uVision5 и разработать программу для микроконтроллера (МК) STM32F200, которая включает и выключает светодиод. Разработать программу, которая последовательно включает и выключает каждый из 8 светодиодов.

Код программы:

```
int main ()
{
    int i; //counter for get ready delay
    unsigned long int j; //counter for blinky delay

    i=0;
    j=0;

    *(unsigned long*)(0x40023830) |= 0x40; // GPIOG
    *(unsigned long*)(0x40023830) |= 0x80; // GPIOH
    *(unsigned long*)(0x40023830) |= 0x100; // GPIOI

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

        *(unsigned long*)(0x40021800) = (*(unsigned long*)(0x40021800)
        & (~0x00002000)) | (0x00001000); // PG6
        *(unsigned long*)(0x40021800) = (*(unsigned long*)(0x40021800)
        & (~0x00008000)) | (0x00004000); // PG7
        *(unsigned long*)(0x40021800) = (*(unsigned long*)(0x40021800)
        & (~0x00020000)) | (0x00010000); // PG8

        *(unsigned long*)(0x40021C00) = (*(unsigned long*)(0x40021C00)
        & (~0x00000020)) | (0x00000010); // PH2
        *(unsigned long*)(0x40021C00) = (*(unsigned long*)(0x40021C00)
        & (~0x00000080)) | (0x00000040); // PH3
        *(unsigned long*)(0x40021C00) = (*(unsigned long*)(0x40021C00)
        & (~0x00002000)) | (0x00001000); // PH6
```

```
*(unsigned long*)(0x40021C00) = (*(unsigned long*)(0x40021C00)
& (~0x00008000)) | (0x00004000); // PH7
```

```
*(unsigned long*)(0x40022000) = (*(unsigned long*)(0x40022000)
& (~0x00200000)) | (0x00100000); // PI10
```

```
while(1)
```

```
{
```

```
    *(unsigned long*)(0x40021814) |= 0x40; //ON PG6
```

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

```
    *(unsigned long*)(0x40021814) &= ~0x40; //OFF PG6
```

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

```
    *(unsigned long*)(0x40021814) |= 0x80; //ON PG7
```

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

```
    *(unsigned long*)(0x40021814) &= ~0x80; //OFF PG7
```

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

```
    *(unsigned long*)(0x40021814) |= 0x100; //ON PG8
```

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

```
    *(unsigned long*)(0x40021814) &= ~0x100; //OFF PG8
```

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

```
    *(unsigned long*)(0x40021C14) |= 0x4; //ON PH2
```

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

```
    *(unsigned long*)(0x40021C14) &= ~0x4; //OFF PH2
```

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

```

*(unsigned long*)(0x40021C14) |= 0x8; //ON PH3

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

*(unsigned long*)(0x40021C14) &= ~0x8; //OFF PH3

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

*(unsigned long*)(0x40021C14) |= 0x40; //ON PH6

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

*(unsigned long*)(0x40021C14) &= ~0x40; //OFF PH6

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

*(unsigned long*)(0x40021C14) |= 0x80; //ON PH7

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

*(unsigned long*)(0x40021C14) &= ~0x80; //OFF PH7

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

*(unsigned long*)(0x40022014) |= 0x400; //ON PI10

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

*(unsigned long*)(0x40022014) &= ~0x400; //OFF PI10

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

```

Полученные результаты: изначально получилось сделать так, чтобы мигал светодиод PG7, далее получилось включить все светодиоды по очереди.

Анализ результатов: чтобы зажечь светодиод PG7 необходимо перевести седьмой разряд порта GPIOG в состояние «1», а для того чтобы «потушить» светодиод необходимо седьмой разряд порта GPIOG перевести в состояние «0».

За включение тактирования периферийных блоков отвечают регистры

включения тактирования периферии RCC XXX. В нашем случае это RCC AHB1. Тактирование периферийного блока GPIOG включается установкой шестого бита, GPIOH -седьмого, а GPIOI - восьмого (по таблице 3а).

Далее определяем адреса GPIOx по таблице:

0x4002 2000 - 0x4002 23FF	GPIOI
0x4002 1C00 - 0x4002 1FFF	GPIOH
0x4002 1800 - 0x4002 1BFF	GPIOG

И по указанным адресам переводим соответствующие биты каждой лампочки в положение 01

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Далее прописываем в цикле последовательное включение/выключение каждого индикатора путем смены бита.

Выводы: в ходе лабораторной работы у меня получилось ознакомиться с основными приемами работы с документацией при составлении программ для микроконтроллеров. Я научилась создавать новый проект в Keil uVision5 и разработала программу для микроконтроллера (МК) STM32F200, которая включает и выключает светодиод. Также была выполнена модификация задания: программа, которая последовательно включает и выключает каждый из 8 светодиодов.