

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

**ІКНІ**  
**Кафедра ПЗ**

**ЗВІТ**

До лабораторної роботи № 1

**З дисципліни:** *“ Основи програмування вбудованих систем ”*

**На тему:** *“ Дослідження середовища Keil і бібліотек CMSIS і SPL ”*

**Лектор:**

доц. каф. пз

Марусенкова Т.А.

**Виконав:**

ст. гр. ПЗ-32

Бурець В.В.

**Прийняв:**

доц. каф. пз

Крук О.Г.

« \_\_\_\_ » \_\_\_\_\_ 2021 р.

Σ= \_\_\_\_ .

Львів-2021

**Тема роботи:** Дослідження середовища Keil і бібліотек CMSIS і SPL (на прикладі блимання світлодіодами).

**Мета роботи:** Ознайомитися з можливостями середовища Keil uVision.

## ТЕОРЕТИЧНІ ВІДОМОСТІ

### Варіант №1

#### 1. Для яких мікроконтролерів призначена бібліотека CMSIS?

**CMSIS** – бібліотека, стандартна для всіх МК з ядром ARM Cortex. Стандартизується ARM Ltd. Пізні виробники МК з цим ядром доповнюють CMSIS файлами з описом периферійних модулів, специфічних для МК, які вони випускають.

### Завдання

1. Засвічення червоного та синього світлодіодів, після цього – зеленого та жовтого з інтервалом у 1 с, циклічно.

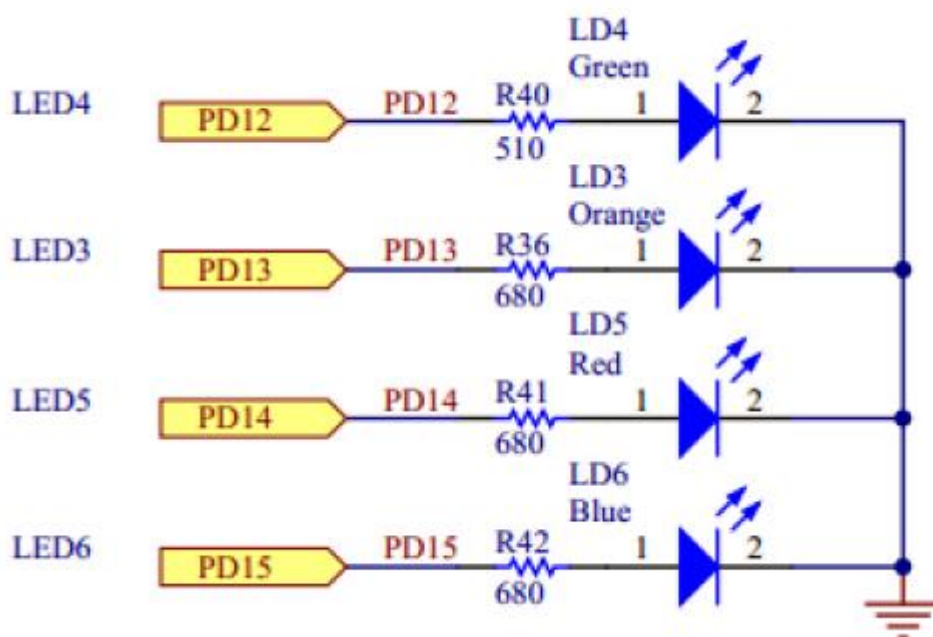


Рис.1 Підключення світлодіодів до STM32F4DISCOVERY

## ХІД РОБОТИ

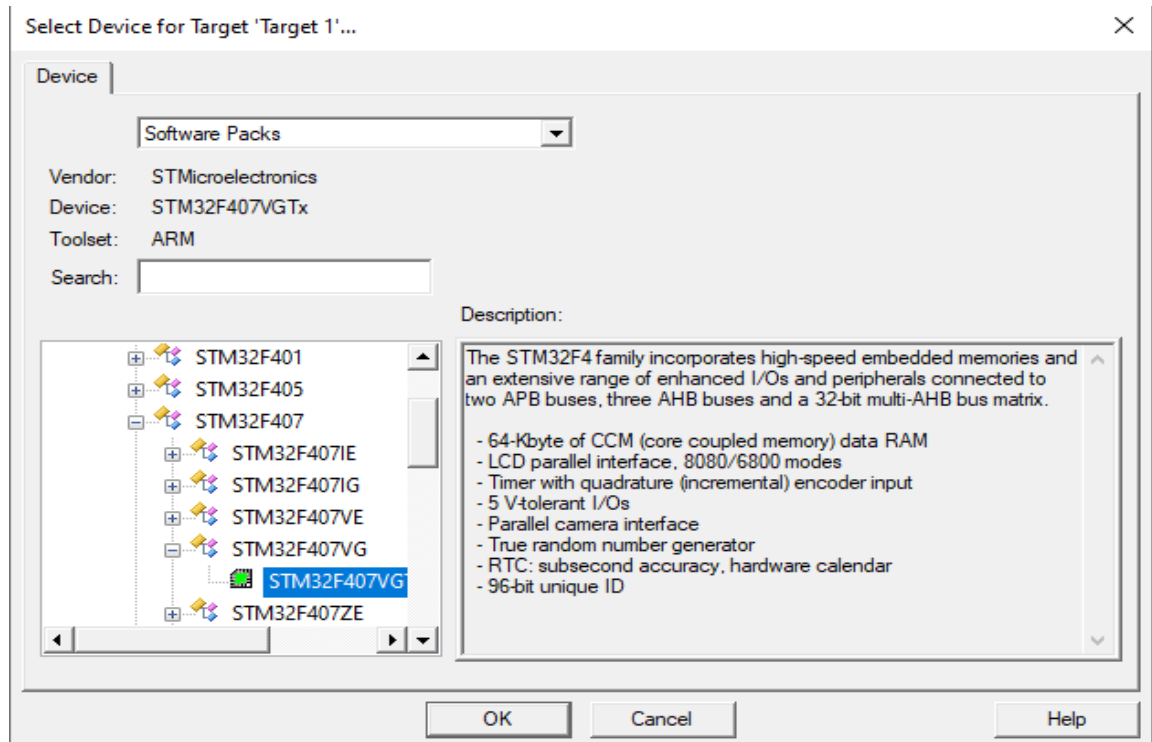


Рис. 2. Вибір мікроконтролера

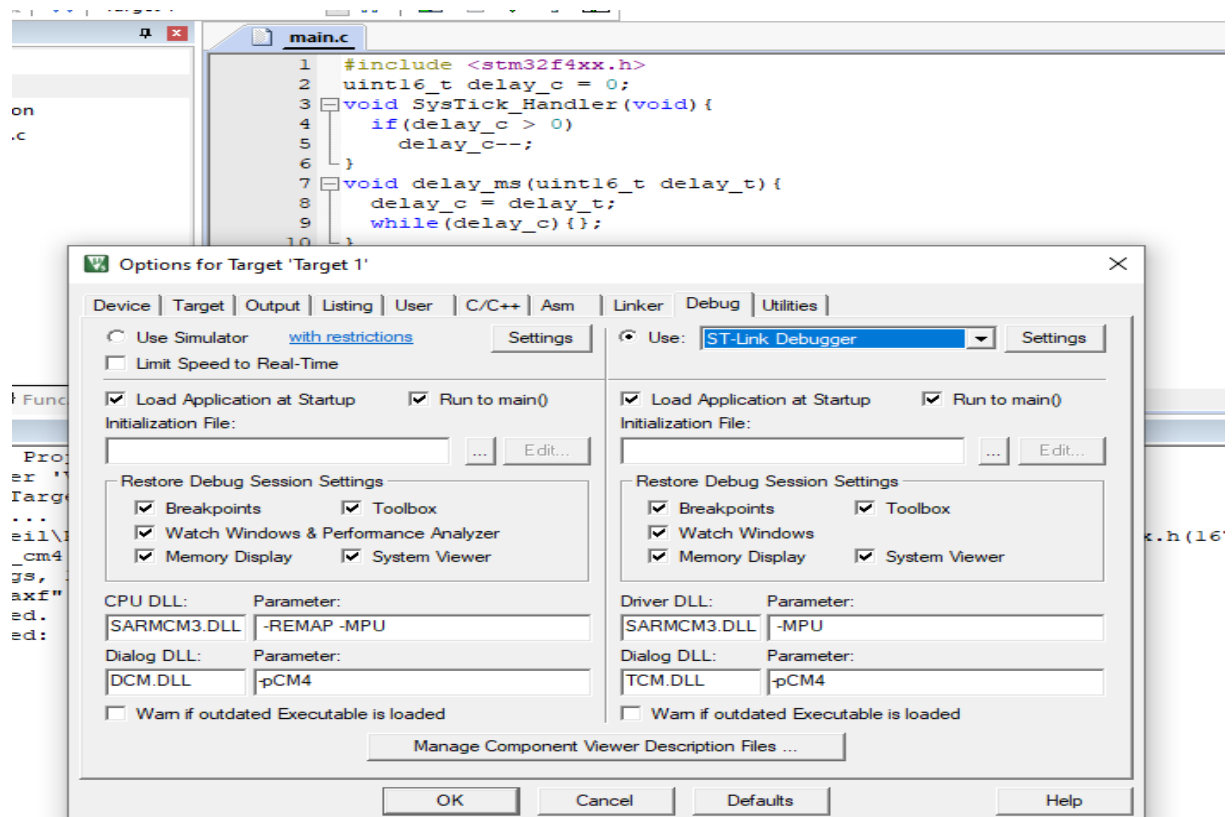


Рис. 3. Вибір debugger`а

The screenshot shows a debugger interface with three main panels: Registers, Disassembly, and GPIOD.

- Registers:** The R15 (PC) register is highlighted with a value of 0x0800031C. Other registers like R0, R1, R2, etc., are also visible.
- Disassembly:** The assembly code for the delay function is shown. The instruction at address 0x0800031C is highlighted: `0x0800031C BF00 NOP`.
- GPIOD:** The GPIOD register values are listed, including MODER, OTYPER, OSPEEDR, PUPDR, IDR, ODR, BSRR, LCKR, AFRL, and AFRH.
- Command Window:** Shows the command `Watch 1` and the variable `delay_c` with a value of `0x03E8` and type `ushort`.

Рис. 4. Значення змінної delay\_c під час відлагодження

The screenshot shows a debugger interface with three main panels: Registers, Disassembly, and GPIOD.

- Registers:** The R15 (PC) register is highlighted with a value of 0x080003A4. Other registers like R0, R1, R2, etc., are also visible.
- Disassembly:** The assembly code for the delay function is shown. The instruction at address 0x080003A4 is highlighted: `0x080003A4 F44F707A MOV r0, #0x3E8`.
- GPIOD:** The GPIOD register values are listed, including MODER, OTYPER, OSPEEDR, PUPDR, IDR, ODR, BSRR, LCKR, AFRL, and AFRH. The ODR register is expanded, showing bits 0 to 15, with bit 15 (ODR15) checked.
- Command Window:** Shows the command `Watch 1` and the variable `ODR` with a value of `0x0000C000` and type `ushort`.

Рис. 5. Значення змінної ODR під час відлагодження

The screenshot displays the STM32CubeIDE interface during a debugging session. On the left, the 'Registers' window shows the state of the processor's registers, with the Program Counter (R15) at 0x080003B6. The central 'Disassembly' window shows the assembly code for the delay\_ms function, with the instruction 'MOV r0, #0x3E8' highlighted. The right-hand 'GPIO' window shows the configuration of the GPIO port, with the Output Data Register (ODR) set to 0x00003000. The 'main.c' file is open in the center, showing the C code for the delay\_ms function and the main function.

Рис. 6. Значення змінної ODR під час відлагодження

Код програми:

```
#include <stm32f4xx.h>

uint16_t delay_c = 0;

void SysTick_Handler(void){
    if(delay_c > 0)
        delay_c--;
}

void delay_ms(uint16_t delay_t){
    delay_c = delay_t;
    while(delay_c){};
}

int main (void){
```

```

SysTick_Config(SystemCoreClock/1000);

RCC->AHB1ENR |= RCC_AHB1ENR_GPIODEN; //turn on timing

GPIO->MODER    = 0x55000000;

GPIO->OTYPER = 0;

GPIO->OSPEEDR = 0;

while(1){

    GPIO->ODR = 0xC000;

    delay_ms(1000);

    GPIO->ODR = 0x3000;

    delay_ms(1000);

}

}

```

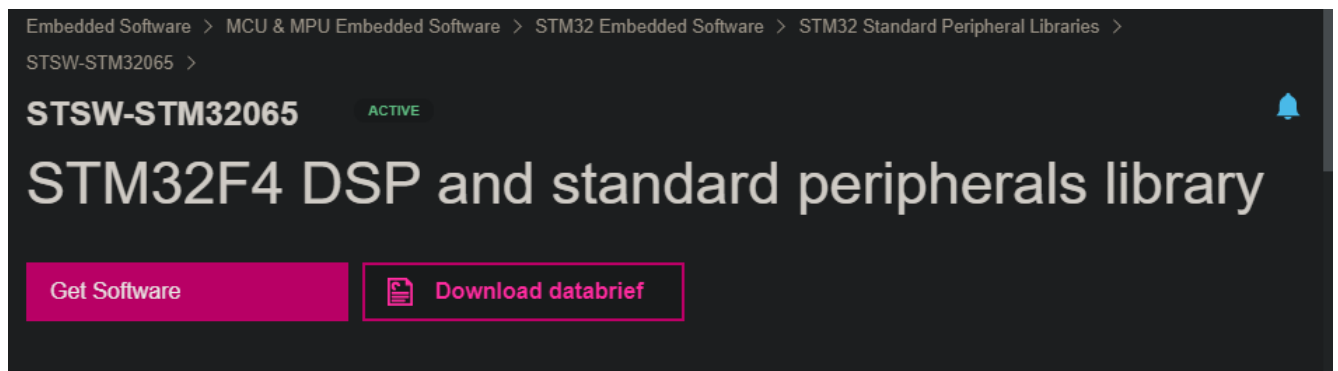


Рис. 7. Завантаження бібліотек для мікроконтроллера

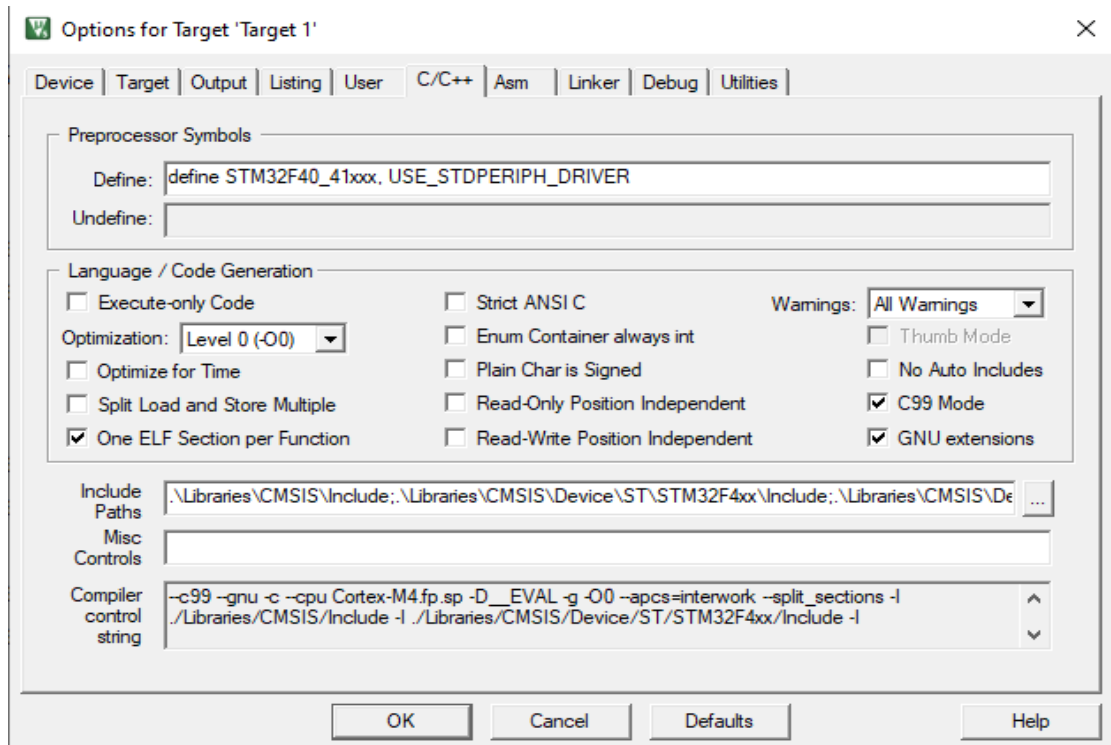


Рис. 7. Задання параметрів роботи бібліотеки "stm32f4xx.h"

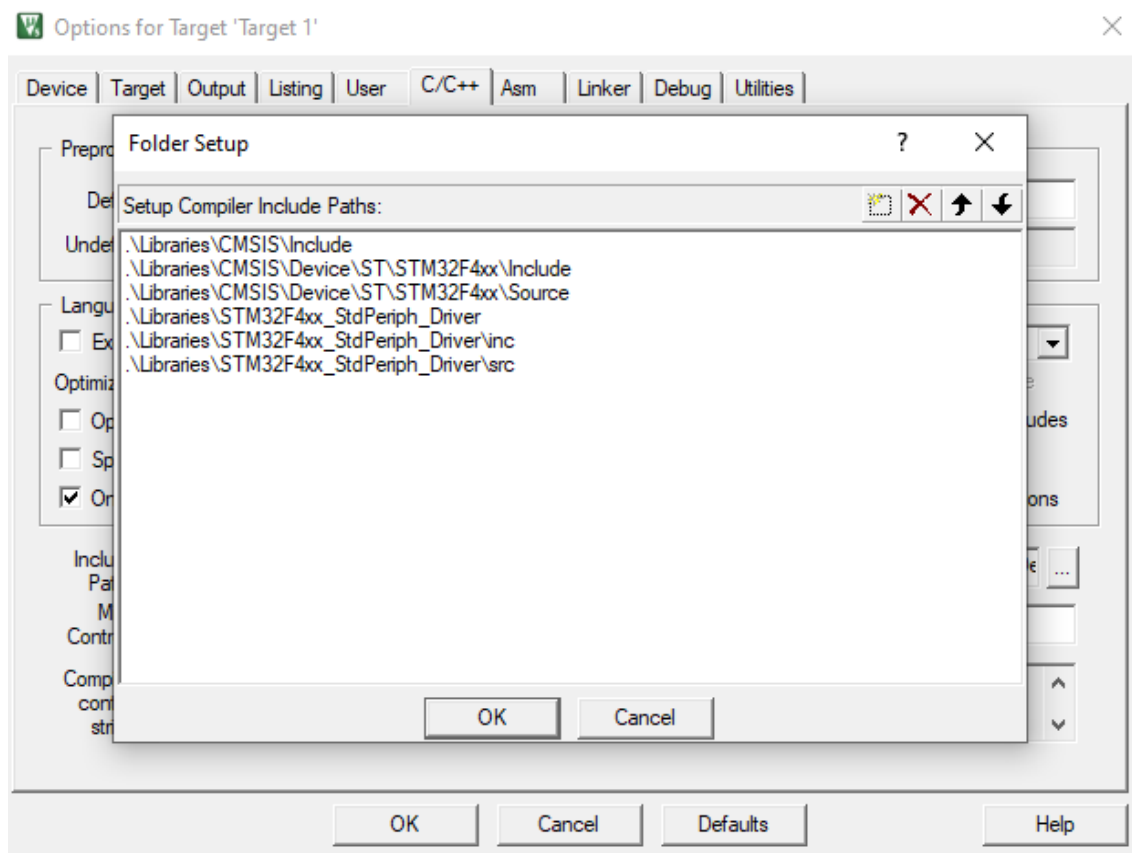


Рис. 8. Встановлення шляхів до файлів

Код програми з використанням SPL бібліотек

```
#include "stm32f4xx.h"          // Device header

RCC_ClocksTypeDef RCC_Clocks;
static __IO uint32_t TimingDelay;

void Delay(__IO uint32_t nTime)
{
    TimingDelay = nTime;

    while(TimingDelay != 0);
}

void TimingDelay_Decrement(void){
    if (TimingDelay != 0x00){
        TimingDelay--;
    }
}

void SysTick_Handler(void)
{
    TimingDelay_Decrement();
}

int main(void)
{
    GPIO_InitTypeDef GPIO_InitStructure;
    GPIO_StructInit(&GPIO_InitStructure);
    GPIO_InitStructure.GPIO_Pin = GPIO_Pin_15 | GPIO_Pin_14;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT;
    GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
    GPIO_InitStructure.GPIO_Speed = GPIO_Speed_2MHz;
```



```

GPIO_Init( GPIOC, &GPIO_InitStructure);

RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOD, ENABLE);

RCC_GetClocksFreq(&RCC_Clocks);

SysTick_Config(RCC_Clocks.HCLK_Frequency / 1000);

while(1){

    GPIO_ToggleBits(GPIOD, GPIO_Pin_15);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_14);
    Delay(1000);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_15);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_14);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_13);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_12);
    Delay(1000);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_13);
    GPIO_ToggleBits(GPIOD, GPIO_Pin_12);

}
}

```

## ВИСНОВКИ

На даній лабораторній роботі я встановив середовище Keil, розробив програму для блимання червоного і синього світлодіодів кожних 1 с., а потім зеленого і жовтого теж з інтервалом 1 с. Під час відлагодження програми було досліджено, що я правильно встановив значення регістрів для виконання заданого завдання. А також було створенно два проекти для виконання встановленої задачі. Перший проект з використанням бібліотеки SPL, а другий з використанням CMSIS