**Міністерство Освіти І НАУКИ України**

**Національний університет "Львівська політехніка"**

Інститут **КНІТ**

Кафедра **ПЗ**

### ЗВІТ

До лабораторної роботи № 2

**З дисципліни:** *“Основи інтернету речей”*

**На тему:** *“ Дослідження впливу середовища програмування на розмір та швидкість виконання програмного коду ”*

**Лектор:**

проф. каф. ПЗ

Фечан А. В.

**Виконав:**

ст. гр. ПЗ-42

Бурець В. В.

**Прийняв:**

проф. каф. ПЗ

Фечан А. В.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2021 р.

∑= \_\_\_\_\_ .

Львів – 2021

**Тема роботи:** Ознайомитися з особливостями роботи середовищ програмування Keil uVision та STM32CubeIDE.

**Мета роботи:** Встановити взаємозв’язок між середовищем програмування та розміром і швидкістю виконання програмного коду.

**TЕОРЕТИЧНІ ВІДОМОСТІ**

STM32CubeIDE - це вдосконалена платформа розробки C/C++ з периферійною конфігурацією, генерацією коду, компіляцією коду та функціями налагодження для мікроконтролерів та мікропроцесорів STM32. Він базується на структурі Eclipse та наборі інструментів GCC для розробки та GDB для налагодження. Він дозволяє інтегрувати сотні існуючих плагінів, які доповнюють можливості Eclipse IDE.

Keil uVision – спеціалізоване середовище для програмування мікроконтролерів архітектури ARM, є одним з найпопулярніших.

**ЗАВДАННЯ**

1. Встановити середовища STM32CubeIDE (повинно бути вже встановлено для виконання ЛР№1) та Keil uVision.
2. Трансформувати індивідуальне завдання для забезпечення можливості вимірювання часу виконання коду.
3. Створити проект згідно індивідуального завдання для борду STM32F4 Discovery з застоуванням середовища програмування STM32CubeIDE.
4. Створити проект згідно індивідуального завдання для мікроконтролера STM32F407VG з застоуванням середовища програмування STM32CubeIDE.
5. Створити проект згідно індивідуального завдання для мікроконтролера STM32F407VG з застоуванням середовища програмування Keil uVision.
6. Порівняти розмір отриманого коду та час його виконання у пунктах 3- 5.
7. Оптимізувати розмір коду отриманий в п.3 шляхом вилучення зайвих автоматично згенерованих ділянок .

Варіант №1: Засвічення червоного та синього світлодіодів, після цього – зеленого та жовтого з інтервалом у 1 с, циклічно

**ХІД ВИКОНАННЯ**

Код мовою C, що реалізує індивідуальне завдання згідно варіанту в середовищі STM32CubeIDE:

**int** **main**(**void**)

{

/\* MCU Configuration--------------------------------------------------------\*/

/\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/

HAL\_Init();

/\* Configure the system clock \*/

SystemClock\_Config();

/\* Initialize all configured peripherals \*/

MX\_GPIO\_Init();

/\* USER CODE BEGIN 2 \*/

/\* Infinite loop \*/

/\* USER CODE BEGIN WHILE \*/

**while** (1)

{

/\* USER CODE END WHILE \*/

//turn on red and blue

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_14);

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_15);

//wait 1 second

HAL\_Delay(1000);

//turn off red and blue

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_14);

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_15);

//turn on green and orange

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_12);

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_13);

//wait 1 second

HAL\_Delay(1000);

//turn off green and orange

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_12);

HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_13);

/\* USER CODE BEGIN 3 \*/

}

/\* USER CODE END 3 \*/

}

Код мовою C, що реалізує індивідуальне завдання згідно варіанту в середовищі Keil uVision:

#include "stm32f4xx.h" // Device header

RCC\_ClocksTypeDef RCC\_Clocks;

// For store tick counts in us

static \_\_IO uint32\_t usTicks=0;

// SysTick\_Handler function will be called every 1 us

void SysTick\_Handler()

{

if (usTicks != 0)

{

usTicks--;

}

}

void DelayInit()

{

// Update SystemCoreClock value

// SystemCoreClockUpdate();

// Configure the SysTick timer to overflow every 1 us

// SysTick\_Config(SystemCoreClock / 1000000);

}

void DelayUs(uint32\_t us)

{

// Reload us value

usTicks = us;

// Wait until usTick reach zero

while (usTicks);

}

void DelayMs(uint32\_t ms)

{

// Wait until ms reach zero

while (ms--)

{

// Delay 1ms

DelayUs(100);

}

}

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);

DelayInit();

while(1){

GPIO\_ToggleBits(GPIOD, GPIO\_Pin\_15);

GPIO\_ToggleBits(GPIOD, GPIO\_Pin\_14);

DelayUs(100);

//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);

DelayUs(100);

//Delay(1000);

GPIO\_ToggleBits(GPIOD, GPIO\_Pin\_13);

GPIO\_ToggleBits(GPIOD, GPIO\_Pin\_12);

}

}

**РЕЗУЛЬТАТИ ВИКОНАННЯ**

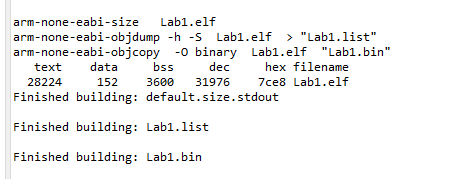


Рис. 1. Результати компіляції проекту створеного у STM32CubeIDE

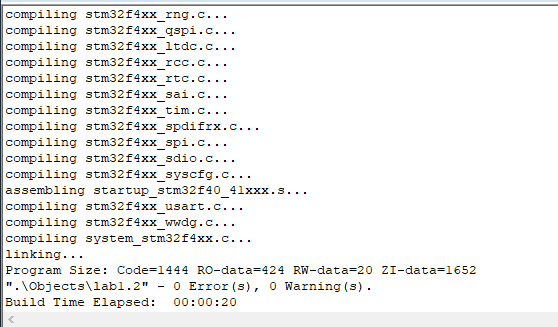


Рис. 2. Результати компіляції проекту створеного у Keil uVision5

**ВИСНОВКИ**

На даній лабораторній роботі я установив середовище розробки STM32CubeIDE та Keil uVision, ознайомився з основними компонентами середовища, створив проект згідно індивідуального варіанту та завантажити його на плату STM32F4 Discovery. Також було створено три аналогічних проекти у середовищі CubeIDE. Було створено проект для мікроконтролера STM32F407VG у середовищі Keil uVision, другий для борди STM32F4 , третій для для борду STM32F4 Discovery у середовищі STM32CubeIDE.

Проект у Keyl uVision був найменший із всіх і був виконаний за найменший час 22.14 секунд.  
 Для мікроконтроллера у STM32CubeIDE час виконання був середнім у порівняння з іншими проектами – 23.61 секунд.

Для борди у STM32CubeIDE довелося із автозгенерованого коду видаляти зайві функції та підключені бібліотеки, за часом виконання цій проект зайняв найбільше часу – 24.98 секунд.