**Ministry of Education and Science of Ukraine**

**Lviv Polythehnic National University**

**Institute of computer science and information technology**

**Department of software engineering**



**Report on**

Laboratory work №1

Subject “Microcontrollers Programming”

Theme: “To learn the capabilities of Keil uVision IDE”

**Performed by:**

Student of group SE-41

Dmytro Khizhnyak

**Verified by:**

Associate professor

Tetiana Marusenkova

Lviv – 2018

**Theme:** Study of Keil uVision IDE and libraries CMSIS and SPL by the example of blinking LEDs

**Goal:** To learn the capabilities of Keil uVision IDE

**Task:** The task is to turn on and off the red LED at intervals increasing by 0.5s in each iteration of an endless loop.

**Theoretical background**

1. How can we watch the current value of some variable in Keil?

We need to start debug session and there will be a panel with current variables state, you can make steps with F5 F11 keys. View => Watch Windows = >Watch 1

2. Which tools for tracing does Keil uVision have?

“Step Over”, “Step Out” and “Step”

3. Please describe the typical project structure in Keil uVision and explain your answer.

Typical project contains source code files which are organized in group. Project must have target on which compiled code will be run. There should be a main() function and usually it has an infinite loop so code runs forever till device reset or turn of.

4. Which steps do we need to take before flash programming?

Download needed packages, setup target platform, develop source code and check id device is visible for IDE. Before flashing it’s better to close open files for better performance.

**Source code listings**

#include "stm32f4xx.h"

unsigned short delay\_c = 0;

unsigned short blink\_delay = 500;

void SysTick\_Handler(void){

if(delay\_c > 0)

delay\_c--;

}

void delay\_ms(unsigned short delay\_t){

delay\_c = delay\_t;

while(delay\_c){};

}

void init(void){

SysTick\_Config(SystemCoreClock/1000);

RCC->AHB1ENR |= RCC\_AHB1ENR\_GPIODEN;

GPIOD->MODER |= 0x10000000;

GPIOD->OTYPER = 0;

GPIOD->OSPEEDR = 0;

}

void main\_loop(void) {

GPIOD->ODR = 0x4000;

delay\_ms(blink\_delay);

GPIOD->ODR = 0;

delay\_ms(blink\_delay);

}

int main(void) {

init();

while(1) {

main\_loop();

}

}

**Conclusions**

While performing this work I learned basic concepts of Keil IDE (installing packages, editing files, debugging the code, adjusting configuration). Setup given code sample and compiled it. Got familiar with development and flashing workflow.