НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ І ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

**Лабораторна робота №1.2**

*з дисципліни* ***«****Архітектура комп’ютерів 3****»***

Виконав:

студент 3 курсу

групи ІВ-81

Бойківський А.О.

Перевірив:

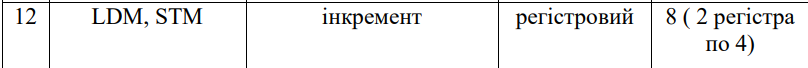
Нікольский С.С.

Київ 2021 р.

**Тема:** Завантажувач основної програми. Обробка виключень. Вивід даних на відлагуваджувальний порт або консоль.

**Мета:** Навчитися працювати з оперативною пам’яттю, використовувати інструкції спеціального призначення, використовувати виключення процесора Cortex-M4. Створення мінімального завантажувача системи. Навчитися користуватися виводом даних через відлагоджувальний порт (або консоль).

Варіант : 8108%16 = 12



**kernel.S**

.syntax unified

.cpu cortex-m4

.thumb

#define A #25

#define B #3

#define C #4

.global vtable\_kernel

.global \_\_kernel\_reset\_\_

.type vtable\_kernel, %object

.type \_\_kernel\_reset\_\_, %function

.section .interrupt\_vector

vtable\_kernel:

.word \_\_stack\_start

.word \_\_kernel\_reset\_\_+1

.size vtable\_kernel, .-vtable\_kernel

.section .rodata

data: .asciz "kernel started!\n"

final: .asciz "Value in register #3: "

.section .text

\_\_kernel\_reset\_\_:

ldr r0, =data

bl dbgput\_line

mov r0, A

mov r1, B

mov r2, C

cmp r0, r1

ITE GE

addGE r3, r0, r1

subLT r3, r0, r1

sdiv r3, r2

ldr r0, =final

bl dbgput\_line

mov r0, r3

bl dbgput\_num

end:

b end

**bootloader.S**

.syntax unified

.cpu cortex-m4

//.fpu softvfp

.thumb

.global bootload

.section .rodata

image: .incbin "kernel.bin"

end\_of\_image:

str\_boot\_start: .asciz "bootloader started"

str\_boot\_end: .asciz "bootloader end"

str\_boot\_indicate: .asciz "#"

.section .text

bootload:

ldr r0, =str\_boot\_start

bl dbgput\_line

ldr r0, =end\_of\_image

ldr r1, =image

ldr r2, =\_ram\_start

loop:

ldm r1!, {r3-r5}

stm r2!, {r3-r5}

cmp r0, r1

bhi loop

bl newline

ldr r0, =str\_boot\_end

bl dbgput\_line

ldr lr, =bootload\_end

add lr, #1

ldr r2, =\_ram\_start

add r2, #4

ldr r0, [r2]

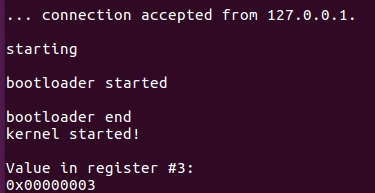
bx r0

bootload\_end:

b bootload\_end

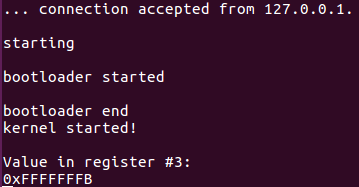
**Результати виконання:**

a = 16, b = 5, c = 7



Отриманий результат: r3 = 3, збігається з очікуваним

a = 3, b = 32, c = 5



Отриманий результат: r3 = 0xfffffffb = -5, збігається з очікуваним

**Висновок:**

В ході виконання лабораторної роботи було вивчено асемблерні інструкції ядра Согіех-МА, що використовуються для доступу до пам'яті та створено проект на асемблері, що виконує завантаження програми, визначеної в файлі bootloader.S в оперативну пам'ять для її виконання (розрахунку за заданою формулою). Виконання проекту було перевірено за допомогою відлагоджувача gdb. Також було створено Макебіе для автоматизації створення прошивки.