Національний Технічний Університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет Інформатики та Обчислювальної Техніки

Кафедра обчислювальної техніки

**Лабораторна робота №3**

з курсу «Архітектура комп’ютерів»

Виконав:

студент 3 курсу

групи ІВ-81

Буцев Б. А.

Перевірив:

Нікольський С. С.

Київ 2021

**Тема:** Завантажувач основної програми. Обробка виключень. Вивід даних на

відлагоджувальний 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: 8107%16 = 11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 11 | LDRH, STRH | декремент\* | регістровий | 2 |

**Код програми:**

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:

ldrh r3, [r1], #2

strh r3, [r2], #2

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

kernel.S

.syntax unified

.cpu cortex-m4

.thumb

#define a #4

#define b #6

#define c #2

.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

// calculate

mov r0, a

and r0, b

lsr r0, r0, #1

mov r1, #1

mov r2, c

.fact:

mul r1, r2

subs r2, #1

bne .fact

add r0, r1

mov r1, #0

ldr r0, =final

bl dbgput

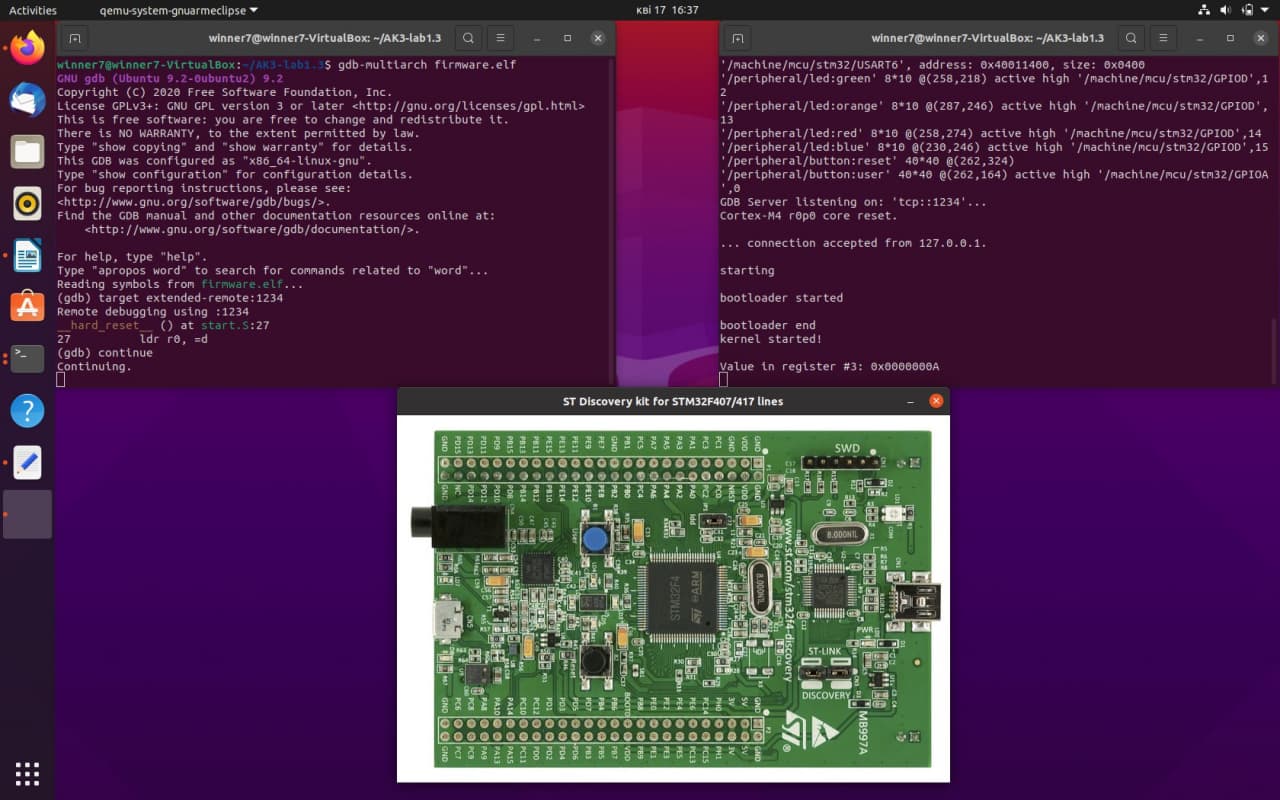
mov r0, r3

bl dbgput\_num

end:

b end

**Приклади роботи програми:**



**Висновки:** У ході виконання лабораторної роботи було створено простий завантажувач за варіантом, який завантажує програму обчислення виразу в оперативну пам’ять. Роботу програми було протестовано для всіх можливих випадків. Програма працює коректно.