НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ І ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

**Лабораторна робота №3**

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

Виконав:

студент 3 курсу

групи ІВ-83

Денисюк Т. Ю.

Перевірив:

Нікольський С. С.

Київ 2021 р.

Варіант 8308 % 16 = 8

**Лістинг програми:**

**print.S**

|  |
| --- |
| .thumb |
|  | .syntax unified |
|  | .cpu cortex-m4 |
|  | #define SEMIHOSTING\_SYS\_WRITE0 #0x04 |
|  | #define SEMIHOSTING #0xAB |
|  |  |
|  | .section .data |
|  | str\_hex: .asciz "0xXXXXXXXX\n" |
|  |  |
|  | .text |
|  | .global dbgput\_line |
|  | .global dbgput |
|  | .global newline |
|  | .global dbgput\_num |
|  | // param: @str |
|  | dbgput: |
|  | push {lr} |
|  | // move str to r1 |
|  | mov r1, r0 |
|  | mov r0, SEMIHOSTING\_SYS\_WRITE0 |
|  | bkpt SEMIHOSTING |
|  | pop {pc} |
|  |  |
|  | \_newline\_sym: .asciz "\n\r" |
|  | .align 4 |
|  | dbgput\_line: |
|  | push {lr} |
|  | // move str to r1 |
|  | mov r1, r0 |
|  | mov r0, SEMIHOSTING\_SYS\_WRITE0 |
|  | bkpt SEMIHOSTING |
|  | ldr r1,=\_newline\_sym |
|  | mov r0, SEMIHOSTING\_SYS\_WRITE0 |
|  | bkpt SEMIHOSTING |
|  | pop {pc} |
|  | newline: |
|  | push {lr} |
|  | ldr r1,=\_newline\_sym |
|  | mov r0, SEMIHOSTING\_SYS\_WRITE0 |
|  | bkpt SEMIHOSTING |
|  | pop {pc} |
|  | dbgput\_num: |
|  | push {lr} |
|  | mov r2, #9 |
|  | mov r3, #0x0000000F |
|  | ldr r1, =str\_hex |
|  | next: |
|  | push {r0} |
|  | and r0, r3 |
|  | add r0, #48 |
|  | cmp r0, #58 |
|  | blo store |
|  | add r0, #7 |
|  | store: |
|  | strb r0, [r1, r2] |
|  | pop {r0} |
|  | lsr r0, r0, #4 |
|  | sub r2, #1 |
|  | cmp r2, #2 |
|  | bge next |
|  |  |
|  | ldr r1, =str\_hex |
|  | mov r0, SEMIHOSTING\_SYS\_WRITE0 |
|  | bkpt SEMIHOSTING |
|  | pop {pc} |

**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 // go to \_\_reset\_kernel\_\_ |
|  | ldr r0, [r2] |
|  | bx r0 |
|  |  |
|  | bootload\_end: |
|  | b bootload\_end |

**kernel.S**

|  |
| --- |
| .syntax unified |
|  | .cpu cortex-m4 |
|  |  |
|  | //.fpu softvfp |
|  | .thumb |
|  |  |
|  | .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 |
|  | start: .asciz "kernel start\n" |
|  | finish: .asciz "result in reg #3: " |
|  |  |
|  | .section .text |
|  | \_\_kernel\_reset\_\_: |
|  | ldr r0, =start |
|  | bl dbgput\_line |
|  |  |
|  | // calc part |
|  |  |
|  | mov r0, #2 |
|  | mov r1, #2 |
|  | mov r2, #1 |
|  | cmp r0,r1 |
|  | ITE GE |
|  | addGE r3,r0,r1 |
|  | SUBLT r3,r0,r1 |
|  | sdiv r3,r2 |
|  |  |
|  | ldr r0, =finish |
|  | bl dbgput |
|  |  |
|  | mov r0, r3 // Move result to r0 |
|  | bl dbgput\_num |
|  |  |
|  | end: |
|  | b end |

**Скріншоти**



