



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное  
образовательное учреждение высшего образования  
“Национальный исследовательский университет ИТМО”

**ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ  
И КОМПЬЮТЕРНОЙ ТЕХНИКИ**



I/O.

0000—0?

## ЛАБОРАТОРНАЯ РАБОТА №1

“Принципы организации ввода/вывода без операционной системы”

по дисциплине  
“Системы ввода-вывода”

Вариант: 1.

Работу выполнил:

Студент потока 1.1, группы Р3311

— Болорболд Аригуун

Лектор:

Быковский Сергей Вячеславович

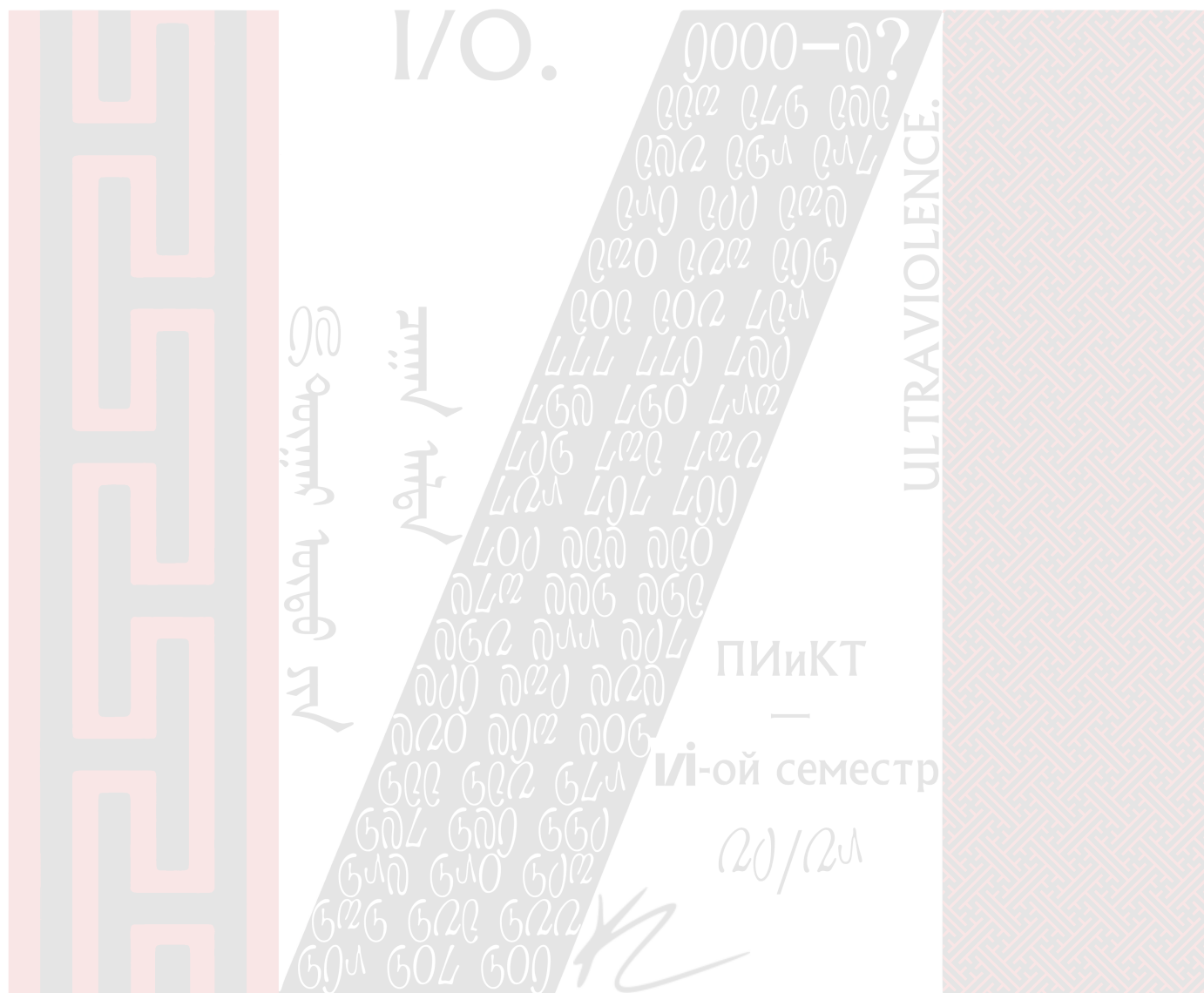
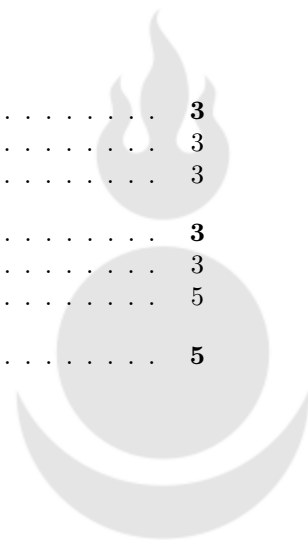
Практик:

Табунщик Сергей Михайлович

г. Санкт-Петербург  
2025 г.

## Содержимое

<b>1</b>	<b>Текст задания</b>	<b>3</b>
1.1	Цель задания	3
1.2	Задачи	3
<b>2</b>	<b>Выполнение</b>	<b>3</b>
2.1	Описания функции	3
2.2	Демонстрация	5
<b>3</b>	<b>Вывод</b>	<b>5</b>



# 1 Текст задания

## 1.1 Цель задания

Познакомиться с принципами организации ввода/вывода без операционной системы на примере компьютерной системы на базе процессора с архитектурой RISC-V и интерфейсом OpenSBI с использованием эмулятора QEMU.

## 1.2 Задачи

1. Реализовать функцию putchar вывода данных в консоль;
2. Реализовать функцию getchar для получения данных из консоли;
3. На базе реализованных функций putchar и getchar написать программу, позволяющую вызывать определенные варианты функции OpenSBI посредством взаимодействия пользователя через меню;
4. Запустить программу и выполнить вызов пунктов меню, получив результаты их работы;
5. Оформить отчет по работе в электронном формате.

№ варианта	Пункты меню
1	<ol style="list-style-type: none"><li>1. Get SBI specification version</li><li>2. Get number of counters</li><li>3. Get details of a counter(должно быть возможно задавать номер счетчика)</li><li>4. System Shutdown</li></ol>

## 2 Выполнение

Ссылка на репозиторию: [GitHub](#)

### 2.1 Описания функции

#### 1. putchar():

```
#define SBI_ECALL_0_1_PUTCHAR 0x01

void
putchar(char ch)
{
    sbi_call(ch, 0, 0, 0, 0, 0, 0, SBI_ECALL_0_1_PUTCHAR);
}
```

Легаси расширение, которое выводит символ, принятый из аргумента. Является блокирующей функцией, т.е. до вывода текущего символа не будет выведен следующий.

#### 2. getchar():

```
#define SBI_ECALL_0_1_GETCHAR 0x02

int
getchar(void)
{
```

```

    struct sbiret ret;
    do {
        ret = sbi_call(0, 0, 0, 0, 0, 0, 0, 0, SBI_ECALL_0_1_GETCHAR);
    } while (ret.error == -1);
    char ch = (char) ret.error;
    return (int)ch;
}

```

Легаси расширение, который принимает символ, введённый пользователем. В отличие от предыдущей, эта функция является неблокирующей, т.е. она будет ждать до тех пор (путём возврата -1), пока пользователь не вводит символ через клавиатуру.

### 3. Интерфейс вызова функций OpenSBI:

```

struct sbiret
sbi_call(long arg0, long arg1, long arg2, long arg3, long arg4, long arg5, long fid, long eid)
{
    register long a0 __asm__("a0") = arg0;
    register long a1 __asm__("a1") = arg1;
    register long a2 __asm__("a2") = arg2;
    register long a3 __asm__("a3") = arg3;
    register long a4 __asm__("a4") = arg4;
    register long a5 __asm__("a5") = arg5;
    register long a6 __asm__("a6") = fid;
    register long a7 __asm__("a7") = eid;

    __asm__ __volatile__("ecall"
                        : "=r"(a0), "=r"(a1)
                        : "r"(a0), "r"(a1), "r"(a2), "r"(a3), "r"(a4), "r"(a5), "r"(a6), "r"(a7)
                        : "memory");
    return (struct sbiret){.error = a0, .value = a1};
}

```

- a0-a5 (arg0-arg5) — аргументы, передаваемые вызову.
- a6 (fid) — ID функции.
- a7 (eid) — ID расширения.
- `get_sbi_version()` получает версию спецификации. Minor-версия закодирована в последних 24 битах численного значения, возвращаемого вызовом, а Major — в предстоящих 7 битах. Для этого в функции есть дополнительные битовые операции.
- `get_num_counters()` получает число счётчиков в ОС. Они доступны в read-only режиме.
- `get_counter_details()` получает специфичные детали одного счётчика. Пользователь вводит номер счётчика, который передаётся в вызов интерфейса. В итоге выводится такая информация, как подлежащее число CSR (Control and Status Register), ширина счётчика и тип (аппаратура или прошивка).
- `system_shutdown()` завершает работу системы.

## 2.2 Демонстрация

```
platinum@XVII-STAR-PLATINUM:~/io/lab1$ ./run.sh
+ QEMU=qemu-system-riscv32
+ CC=clang
+ CFLAGS=-std=c11 -O2 -g3 -Wall -Wextra --target=riscv32 -ffreestanding -nostdlib
+ clang -std=c11 -O2 -g3 -Wall -Wextra --target=riscv32 -ffreestanding -nostdlib -Wl,-Tkernel.ld -Wl,-Map=kernel.map -o kernel.elf kernel.c
+ qemu-system-riscv32 -machine virt -bios default -nographic -serial mon:stdio --no-reboot -kernel kernel.elf

OpenSBI v1.2

Platform Name      : riscv-virtio,qemu
Platform Features  : medeleg
Platform HART Count : 1
Platform IPI Device : aclint-mswi
Platform Timer Device : aclint-mtimer @ 100000000Hz
Platform Console Device : semihosting
Platform HSH Device : ---
Platform PMU Device : ---
Platform Reboot Device : sifive_test
Platform Shutdown Device : sifive_test
Firmware Base      : 0x00000000
Firmware Size      : 208 KB
Runtime SBI Version : 1.0

Domain0 Name       : root
Domain0 Boot HART  : 0
Domain0 HARTs      : 0*
Domain0 Region00    : 0x02000000-0x0200ffff (I)
Domain0 Region01    : 0x00000000-0x0003ffff (C)
Domain0 Region02    : 0x00000000-0xffffffff (R,W,X)
Domain0 Next Address : 0x02000000
Domain0 Next Arg1    : 0x07000000
Domain0 Next Mode    : S-mode
Domain0 SysReset     : yes

Boot HART ID       : 0
Boot HART Domain   : root
Boot HART Priv Version : v1.10
Boot HART Base ISA  : rv32imafdc
Boot HART ISA Extensions : time
Boot HART PMP Count : 16
Boot HART PMP Granularity : 4
Boot HART PMP Address Bits: 32
Boot HART MHPM Count : 0
Boot HART MIDELEG   : 0x00000222
Boot HART MEDELEG   : 0x0000b109

Меню OpenSBI:
1. Get SBI specification version
2. Get number of counters
3. Get details of a counter
4. System shutdown
Введите опцию:
1

Версия SBI: 1.000

Меню OpenSBI:
1. Get SBI specification version
2. Get number of counters
3. Get details of a counter
4. System shutdown
Введите опцию:
2

Число счётчиков: 19

Меню OpenSBI:
1. Get SBI specification version
2. Get number of counters
3. Get details of a counter
4. System shutdown
Введите опцию:
3

Введите число счётчика: 5

Счётчик: 5
Детали:
Тип: Прошивка
CSR и ширина не применимы для прошивки.

Меню OpenSBI:
1. Get SBI specification version
2. Get number of counters
3. Get details of a counter
4. System shutdown
Введите опцию:
2

Введите число счётчика: 2

Счётчик: 2
Детали:
Тип: Аппаратура
CSR: 3074
Ширина: 64 биты

Меню OpenSBI:
1. Get SBI specification version
2. Get number of counters
3. Get details of a counter
4. System shutdown
Введите опцию:
4

Завершение работы системы...
platinum@XVII-STAR-PLATINUM:~/io/lab1$ |
```

## 3 Вывод

В рамках этой лабораторной работы я ознакомился с процессом создания драйверов устройств без использования ОС.