**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра вычислительной техники**

отчет

**по лабораторной работе №1**

**по дисциплине «Организация ЭВМ и систем»**

Тема: И**сследование внутреннего представления различных форматов данных**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 3353 |  | Шинкарь К. Д. |
| Преподаватель |  | Гречухин М.Н. |

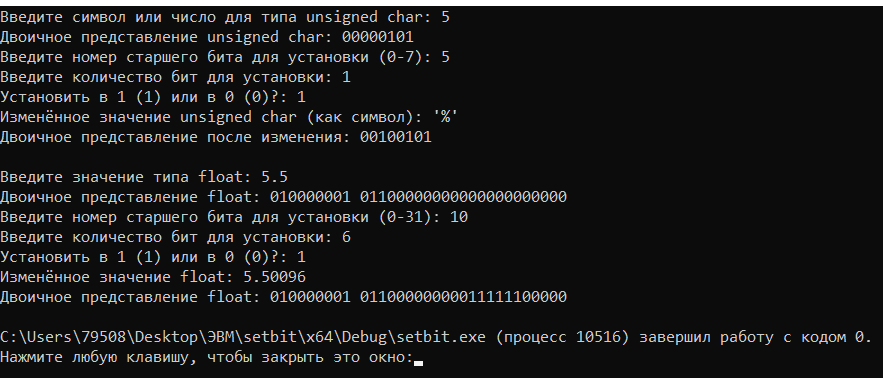
Санкт-Петербург

2024

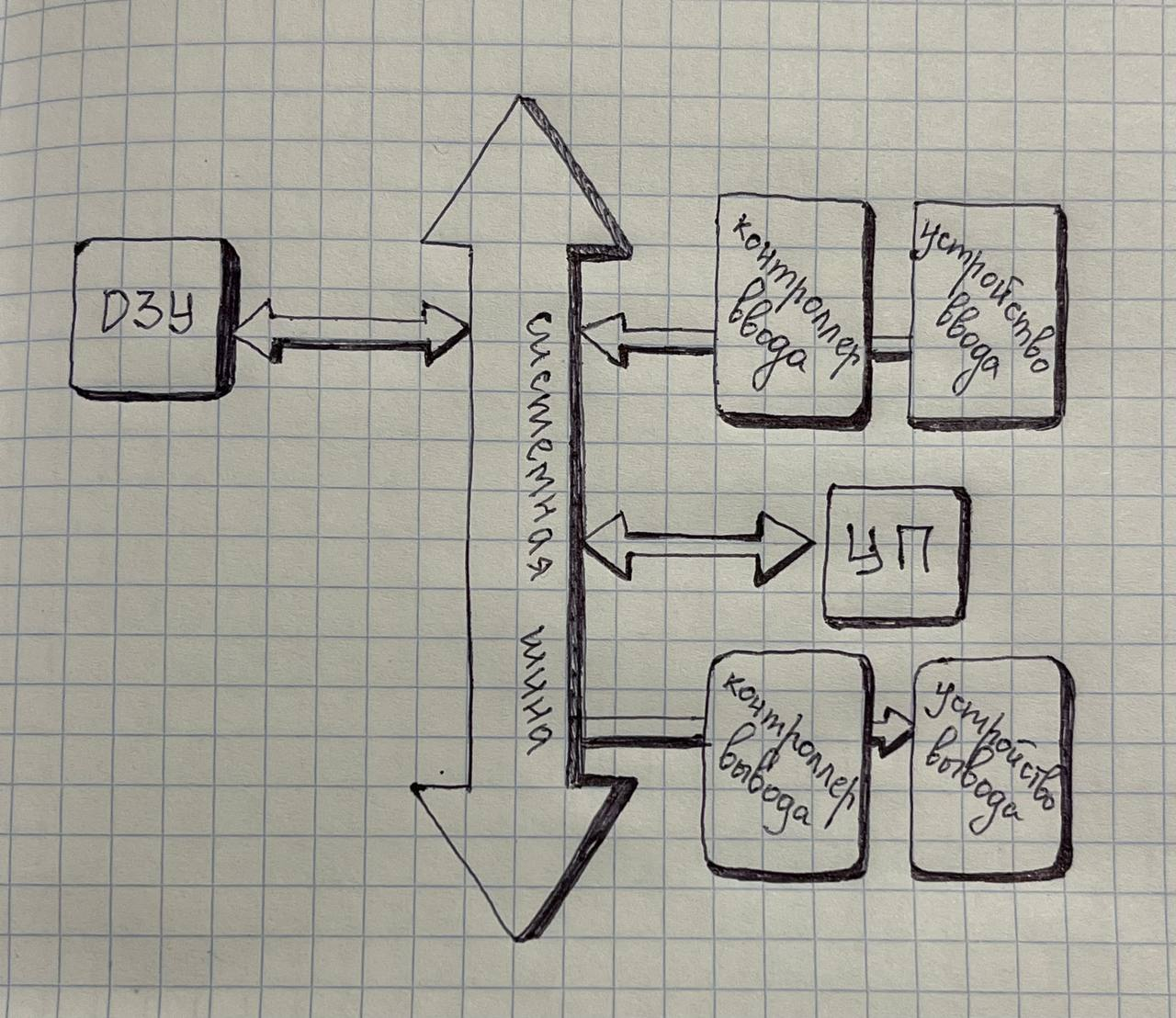
**Цель работы:** ознакомиться с внутренним представлением различных типов данных, используемых компьютером при обработке этих данных.

**Задание:** необходимо разработать алгоритм ввода с клавиатуры требуемых типов данных (в данном случае unsigned char и float) и показать на экране их внутреннее представление в двоичной системе счисления. Написать и отладить программу на языке С++, реализующую разработанный алгоритм. В соответствии с заданием дополнить разработанный ранее алгоритм блоками для выполнения преобразования двоичного полученного кода исходного типа данных и последующего вывода преобразованного кода в двоичной системе счисления и в формате исходного данного.

**Пример работы программы**

****

**Структурная схема аппаратных средств, используемых при выполнении программы с необходимой степенью детализации содержимого блоков.**



**Системная шина**:

Системная шина необходима для обмена данными между процессором, памятью и устройствами ввода/вывода. Она объединяет остальные задействованные элементы компьютера для передачи данных и команд.

**Оперативная память (ОЗУ)**:

Оперативная необходима для временного хранения данных и инструкций, которые использует процессор во время выполнения программы. Процессор обладает возможностью извлекать инструкции и данные из памяти для их обработки.

**Центральный процессор (ЦП)**:

Центральный процессор выполняет инструкции программы. Он способен обмениваться данными с оперативной памятью, управляющими устройствами и устройствами ввода/вывода.

**Контроллеры ввода/вывода**:

**Контроллеры ввода/вывода** позволяют обмениваться данными между процессором и периферийными устройствами. Они подключены к системной шине и обеспечивают связь с внешними устройствами.

**Устройства ввода**:

К устройствам ввода можно отнести клавиатуру, мышь, сканеры и другие устройства, с помощью которых возможен ввод данных в систему. Данные с устройств ввода поступают в процессор через контроллеры ввода/вывода.

**Устройства вывода**:

К устройствам ввода можно отнести мониторы, принтеры и колонки. Они получают данные от процессора через контроллер вывода и служат для вывода обработанной информации пользователю.

**Текст программы**

#include <iostream>

#include <cmath> // Для floor()

using namespace std;

// вывод двоичного представления unsigned char

void printBinary(unsigned char value) {

for (int i = sizeof(unsigned char) \* 8 - 1; i >= 0; --i) {

cout << ((value >> i) & 1); //сдвиг вправо (от старшего к младшему)

}

cout << endl;

}

// вывод двоичного представления целого числа

void printBinary(int value) {

for (int i = sizeof(int) \* 8 - 1; i >= 0; --i) {

cout << ((value >> i) & 1);

}

cout << endl;

}

// для вывода двоичного представления float с пробелом между частями

void printBinaryFloat(float value) {

union {

float input;

uint32\_t bits; // для доступа к float как к целому числу

} data;

data.input = value;

cout << "Двоичное представление float: ";

for (int i = 31; i >= 0; --i) {

cout << ((data.bits >> i) & 1);

if (i == 31 - 8) {

cout << ' '; // пробел перед экспонентой

}

if (i == 31 - 8 - 23) {

cout << ' '; // пробел перед мантиссой

}

}

cout << endl;

}

// установка битов в unsigned char

unsigned char setBits(unsigned char value, int startBit, int numBits, bool bitValue) {

for (int i = startBit; i > startBit - numBits; --i) {

if (bitValue) {

value |= (1 << i); // устанавливаем бит в 1

}

else {

value &= ~(1 << i); // устанавливаем бит в 0

}

}

return value;

}

// установка битов в float

float setBits(float value, int startBit, int numBits, bool bitValue) {

union {

float input;

uint32\_t bits;

} data;

data.input = value;

for (int i = startBit; i > startBit - numBits; --i) {

if (bitValue) {

data.bits |= (1 << i); // устанавливаем бит в 1

}

else {

data.bits &= ~(1 << i); // устанавливаем бит в 0

}

}

return data.input;

}

int main() {

setlocale(LC\_ALL, "ru");

// unsigned char

unsigned char ucharValue;

cout << "Введите символ или число для типа unsigned char: ";

char tempChar;

cin >> tempChar;

if (isdigit(tempChar)) {

ucharValue = static\_cast<unsigned char>(tempChar - '0'); // Если введена цифра

}

else {

ucharValue = static\_cast<unsigned char>(tempChar); // Если введён символ

}

cout << "Двоичное представление unsigned char: ";

printBinary(ucharValue);

int startBit, numBits;

bool bitValue;

cout << "Введите номер старшего бита для установки (0-7): ";

cin >> startBit;

cout << "Введите количество бит для установки: ";

cin >> numBits;

cout << "Установить в 1 (1) или в 0 (0)?: ";

cin >> bitValue;

ucharValue = setBits(ucharValue, startBit, numBits, bitValue);

cout << "Изменённое значение unsigned char (как символ): '" << ucharValue << "'" << endl;

cout << "Двоичное представление после изменения: ";

printBinary(ucharValue);

// float

float floatValue;

cout << "\nВведите значение типа float: ";

cin >> floatValue;

printBinaryFloat(floatValue);

cout << "Введите номер старшего бита для установки (0-31): ";

cin >> startBit;

cout << "Введите количество бит для установки: ";

cin >> numBits;

cout << "Установить в 1 (1) или в 0 (0)?: ";

cin >> bitValue;

floatValue = setBits(floatValue, startBit, numBits, bitValue);

cout << "Изменённое значение float: " << floatValue << endl;

printBinaryFloat(floatValue);

return 0;

}