

1. Цель работы

Целью работы является изучение поразрядных операций типа НЕ, И, ИЛИ, исключающее ИЛИ и операций сдвига.

2. Задание

Согласно варианту №3 необходимо:

- а) ввод десятичных чисел и формирование из них заданной упакованной структуры как 16-ричного числа

№ разряда	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Значение	N	N	N	N	E	C	C	C	C	C	C	C	C	C	0	P

- в) из заданной упакованной структуры, представленной как 16-ричное число выделить значения отдельных её составных частей (в виде десятичных чисел).

3. Описание созданных функций

3.1. ПРОЕКТ I

Имя: request_data

Назначение: Запрашивает данные(N, E, C, P) для получения 16-го числа

Входные данные:

- -

Выходные данные:

- struct code – Структура из 4 unsigned short int

Побочный эффект: Отсутствует

Тестовые данные:

	x
12	Составляющая P неверна!
1 189 0 12	cafl

Прототип: CODE request_data()

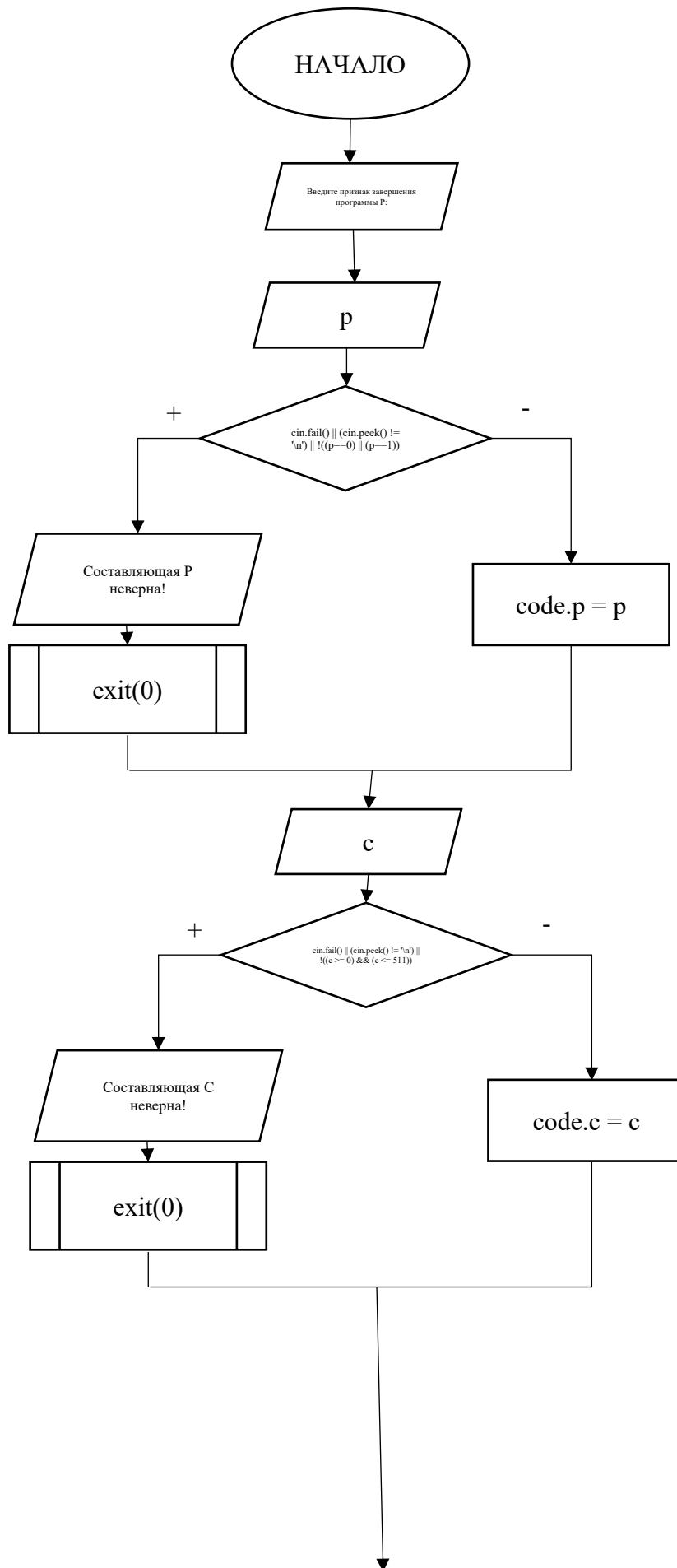
Псевдокод

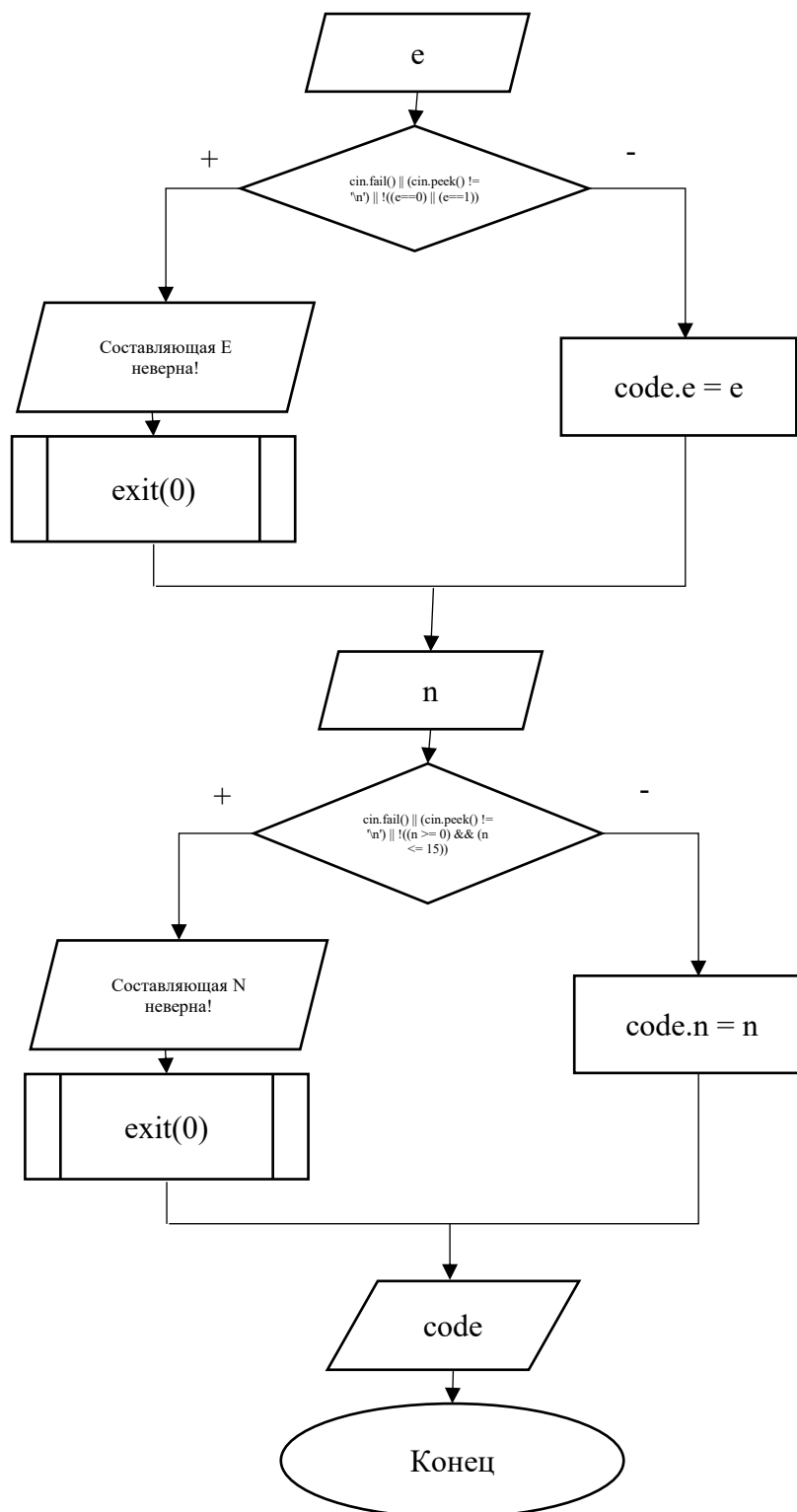
Получить составляющие

Проверить их на корректность

Вернуть составляющие

Блок-схема





3.2. ПРОЕКТ 2

Имя: request_data

Назначение: Запрашивает данные как 16-го число для расшифровки

Входные данные:

- -

Выходные данные:

- unsigned short int – 16е число

Побочный эффект: Отсутствует

Тестовые данные:

Вход	Выход
caf4	caf4
5h	Неверное 16е число

Прототип: unsigned short int request_data()

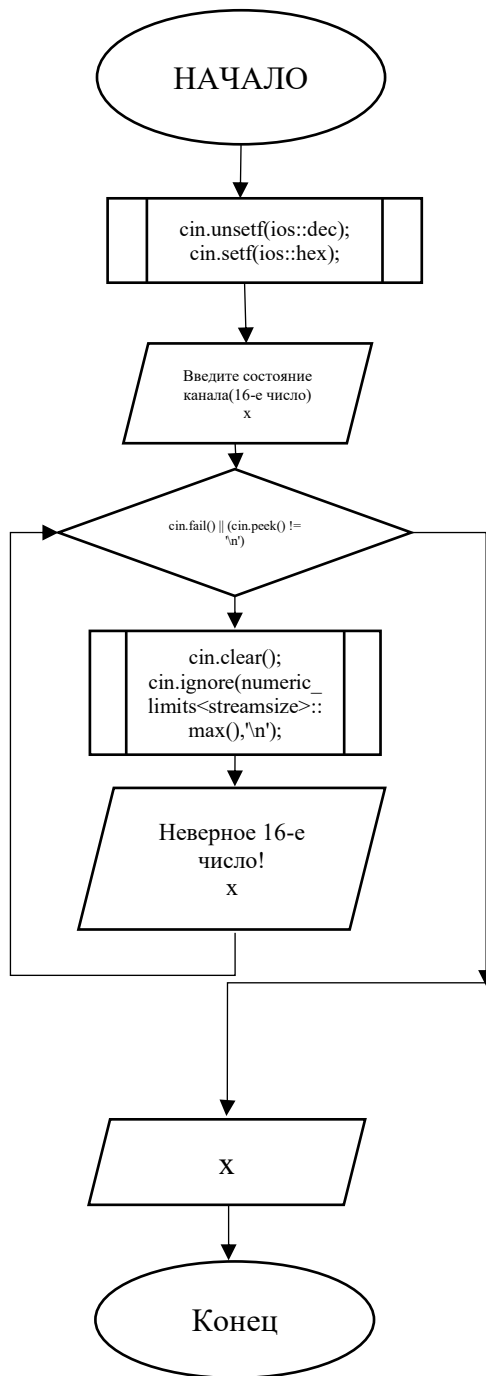
Псевдокод

Получить 16е число

 Проверить его на корректность

Вернуть 16е число

Блок-схема



4. Текст программ

4.1 ПРОЕКТ 1

```

#include <iostream>

using namespace std;

struct CODE // Структура из 4 переменных
{
    unsigned short int n;
    unsigned short int e;
    unsigned short int c;
    unsigned short int p;
};
  
```

```

CODE request_data() // Функция запроса данных
{
    unsigned short int n, e, c, p;

    CODE code;

    cout << "Введите признак завершения программы P: ";
    cin >> p;
    if (cin.fail() || (cin.peek() != '\n') || !((p==0) || (p==1))) // Проверка на
корректность и ограничения
    {
        cout << "Составляющая P неверна!";
        exit(0); // Прекращение программы
    }
    code.p = p; // Добавление p в структуру

    cout << "Введите код причины прерывания(C): ";
    cin >> c;
    if (cin.fail() || (cin.peek() != '\n') || !((c >= 0) && (c <= 511))) //
Проверка на корректность и ограничения
    {
        cout << "Составляющая C неверна!";
        exit(0); // Прекращение программы
    }
    code.c = c; // Добавление c в структуру

    cout << "Введите признак ошибки(E): ";
    cin >> e;
    if (cin.fail() || (cin.peek() != '\n') || !((e == 0) || (e == 1))) //
Проверка на корректность и ограничения
    {
        cout << "Составляющая E неверна!";
        exit(0); // Прекращение программы
    }
    code.e = e; // Добавление e в структуру

    cout << "Введите номер канал(N): ";
    cin >> n;
    if (cin.fail() || (cin.peek() != '\n') || !((n >= 0) && (n <= 15))) //
Проверка на корректность и ограничения
    {
        cout << "Составляющая N неверна!";
        exit(0); // Прекращение программы
    }
    code.n = n; // Добавление n в структуру

    return code;
}

int main()
{
    setlocale(LC_ALL, "rus");

    unsigned short int x = 0;

    CODE code = request_data(); // Получение стукрутры

    x = code.p; // Добавление составляющей p
    x = 0 << 1 | x; // Добавление 0 в структуру
    x = code.c << 2 | x; // Добавление составляющей c
    x = code.e << 11 | x; // Добавление составляющей e
    x = code.n << 12 | x; // Добавление составляющей n

    cout << "Состояние канала: " << hex << x << endl;
}

```

4.2 ПРОЕКТ 2

```
#include <iostream>

using namespace std;

unsigned short int request_data() // Запрос 16го числа
{
    unsigned short int x;

    cin.unsetf(ios::dec); // Настройка ввода на 16е числа
    cin.setf(ios::hex);

    cout << "Введите состояние канала(16-е число)" << endl;

    cin >> x;

    while (cin.fail() || (cin.peek() != '\n') ) // Проверка корректности
    {
        cin.clear();
        cin.ignore(numeric_limits<streamsize>::max(), '\n');
        cout << "Неверное 16-е число!" << endl;
        cin >> x;
    }

    return x;
}

int main()
{
    setlocale(LC_ALL, "rus");

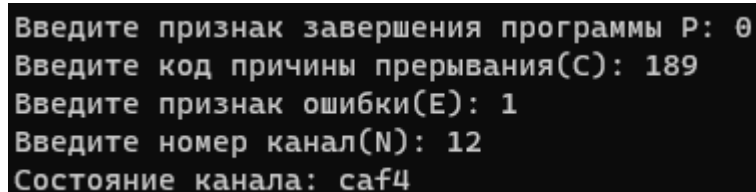
    unsigned short int x;
    unsigned short int n, e, c, p;

    x = request_data(); // Запрос 16го числа

    n = x >> 12; // Составляющая n
    e = x >> 11 & 1; // Составляющая e
    c = x >> 2 & 0x1ff; // Составляющая c
    p = x & 1; // Составляющая p

    cout << "Составная N: " << dec << n << endl; // Вывод каждой составляющей
    cout << "Составная E: " << dec << e << endl;
    cout << "Составная C: " << dec << c << endl;
    cout << "Составная P: " << dec << p << endl;
}
```

5. Пример работы программы



```
Введите признак завершения программы Р: 0
Введите код причины прерывания(С): 189
Введите признак ошибки(Е): 1
Введите номер канал(N): 12
Состояние канала: saf4
```

Рисунок 1 – Пример работы программы

```
Введите состояние канала(16-е число)
saf4
Составная N: 12
Составная E: 1
Составная C: 189
Составная P: 0
```

Рисунок 2 – Пример работы программы

Полученные данные совпадают с действительными

6. Анализ результатов и выводы

В процессе лабораторной работы были изучены поразрядные операции типа НЕ, И, ИЛИ, исключающее ИЛИ и операций сдвига.

Достоинства программы:

- Проверка данных на корректность
- Возможность использования подпрограмм в других разработках

Недостатки:

- Не было обнаружено