Белорусский государственный университет информатики и радиоэлектроники

# Кафедра ЭВМ

### Отчет

### по лабораторной работе № 4

### «Случайные методы доступа к моноканалу»

Проверил:

Марцинкевич В.А.

Выполнил:

студент группы 130501

Гнездилов А.М.

Минск 2022

**Цель работы:** написать программу пакетной передачи данных через СОМ-порты упрощенным алгоритмом CSMA/CD в соответствии с требованиями.

**Листинг кода**

**Main.cpp**

#include "USER.h"

string decode(string data)

{

string ret;

string str;

int s = 8;

for (int i = 0; i < data.length() / s; i++)

{

for (int j = i \* s; j < i \* s + s; j++)

{

str += data[j];

}

int st = 0;

int ch = 0;

for (int j = str.length() - 1; j >= 0; j--)

{

if (str[j] == '1') ch += pow(2, st);

st++;

}

char a = ch;

ret += a;

str.clear();

}

return ret;

}

void GetData(HANDLE handle, vector<USER>& USR)

{

DWORD size = 0;

char buff;

string data;

int fl = 0;

do

{

ReadFile(handle, &buff, 1, &size, NULL);

if (buff != '~')

data += buff;

} while (buff != '~');

PurgeComm(handle, PURGE\_RXABORT | PURGE\_TXABORT | PURGE\_RXCLEAR | PURGE\_TXCLEAR);

string str;

string sour;

string pour;

int sum;

for (int i = 0; i < 10; i++)

if (data[i] != '|')

pour += data[i];

for (int i = 10; i < 20; i++)

if (data[i] != '|')

sour += data[i];

for (int i = 20; i < data.length() - 1; i++)

str += data[i];

sum = data[data.length() - 1];

string info = decode(str);

string d;

d += info;

d += " [";

d += str;

d += "]";

for (int i = 0; i < USR.size(); i++)

if (USR[i].GetName() == pour)

{

USR[i].SetDataFrom(d);

USR[i].SetNameFrom(sour);

}

}

void SendData(vector<USER> &USR, HANDLE handle1, HANDLE handle2, int \*coll)

{

string data;

int i, j;

for (int i = 0; i < USR.size(); i++)

{

cout << i + 1 << " - ";

cout << USR[i].GetName() << endl;

}

cout << endl << "Какой пользователь хочет передать данные?: ";

cin >> i;

system("cls");

for (int i = 0; i < USR.size(); i++)

{

cout << i + 1 << " - ";

cout << USR[i].GetName() << endl;

}

cout << endl << "Какому пользователю хочет передать данные?: ";

cin >> j;

system("cls");

cout << "Введите днные для передачи: ";

cin >> data;

USR[i - 1].SetDataTo(data);

if(\*coll == 1)

{

while (\*coll != 0)

{

system("cls");

int r = 1 + rand() % (10);

cout << "Обнаружена коллизия!" << endl

<< "Задержка для пользователя <" << USR[i - 1].GetName() << ">: " << r << endl

<< "Данные для передачи: " << USR[i - 1].GetDataTo() << endl;

\_getch();

if (r > 4) \*coll = 0;

Sleep(r \* 1000);

}

}

USR[i - 1].Write(USR[j - 1].GetName(), handle1, coll, USR[i - 1].GetDataTo());

GetData(handle2, USR);

\_getch();

}

void GetInfo(vector<USER> &USR, int \*coll)

{

for (int i = 0; i < USR.size(); i++)

{

cout << "Пользователь №" << i + 1 << endl

<< "Имя: " << USR[i].GetName() << endl

<< "Последние полученные данные: " << USR[i].GetDataFrom() << endl

<< "Имя отпавителя: " << USR[i].GetNameFrom() << endl

<< "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl << endl;

}

\*coll = 0;

}

void GetList(vector<USER> USR)

{

for (int i = 0; i < USR.size(); i++)

{

cout << "Пользователь №" << i + 1 << " - " << USR[i].GetName() << endl;

}

\_getch();

}

int menu() //меню

{

system("cls");

cout << "1 - Добавить пользователя" << endl

<< "2 - Список пользователей" << endl

<< "3 - Передать данные" << endl

<< "4 - Получить данные" << endl

<< "0 - Выход" << endl;

return \_getch();

}

int main()

{

system("chcp 1251 > null");

HANDLE handle1 = ::CreateFile(L"COM1", GENERIC\_WRITE, 0, 0, OPEN\_EXISTING, FILE\_ATTRIBUTE\_NORMAL, 0);

HANDLE handle2 = ::CreateFile(L"COM2", GENERIC\_READ, 0, 0, OPEN\_EXISTING, FILE\_ATTRIBUTE\_NORMAL, 0);

int i = 0, coll = 0;

vector <USER> USR;

while (1)

{

switch (menu())

{

case '1': { system("cls"); USER user; USR.push\_back(user); i++; break; }

case '2': { system("cls"); GetList(USR); break; }

case '3': { system("cls"); SendData(USR, handle1, handle2, &coll); break; }

case '4': { system("cls"); GetInfo(USR, &coll); \_getch(); break; }

case '0': { system("cls"); cout << "Работа завершена."; return 0; }

}

}

return 0;

}

**USER.h**

#include <windows.h>

#include <iostream>

#include "conio.h"

#include <vector>

#include <stdlib.h>

#include <stack>

#include <cmath>

using namespace std;

class USER

{

private:

string name;

string nameFrom;

string dataTo;

string dataFrom;

public:

USER();

string GetName();

string GetDataTo();

string GetDataFrom();

string GetNameFrom();

void Write(string name2, HANDLE handle, int\* coll, string data);

void SetDataTo(string data);

void SetDataFrom(string data);

void SetNameFrom(string data);

};

**USER.cpp**

#include "USER.h"

string code(string data)

{

string ret;

for (int i = 0; i < data.length(); i++)

{

stack<int> ST;

int count = 0;

string str;

int a, t, d;

t = 0;

d = 1;

a = data[i];

while (a) { t += (a % 2) \* d; a = a / 2; d = d \* 10; }

do

{

ST.push(t % 10);

t /= 10;

count++;

} while (t);

count = 8 - count;

for (int i = 0; i < count; i++)

str += '0';

while (!ST.empty())

{

t = ST.top();

ST.pop();

char s = { char(t + '0') };

str += s;

}

ret += str;

}

return ret;

}

USER::USER()

{

system("cls"); cout << "Введите имя пользователя: "; cin >> this->name;

}

string USER::GetName() { return name; }

string USER::GetDataTo() { return dataTo; }

string USER::GetDataFrom() { return dataFrom; }

string USER::GetNameFrom() { return nameFrom; }

void USER::Write(string name2, HANDLE handle, int \*coll, string data)

{

char data1[256] = "";

char data2[256] = "";

string NameTo, NameFrom;

string str;

string b;

for (int i = 0; i < data.length(); i++) data1[i] = data[i];

system("cls");

char s = strlen(data1);

NameTo = name2;

if (NameTo.length() < 10)

for (int i = NameTo.length(); i < 10; i++)

NameTo += '|';

NameFrom = name;

if (NameFrom.length() < 10)

for (int i = NameFrom.length(); i < 10; i++)

NameFrom += '|';

str += NameTo;

str += NameFrom;

b = code(data1);

str += b;

str += s;

str += '~';

this->SetDataTo(str);

for (int i = 0; i < str.length(); i++) data2[i] = str[i];

DWORD size = strlen(data2);

DWORD bytes;

BOOL Ret = WriteFile(handle, &data2, size, &bytes, NULL);

system("cls");

cout << "Байт записано в COM\_1: " << strlen(data1) << endl << "Данные: " << data1 << " [" << b << "]" << endl;

\*coll = 1;

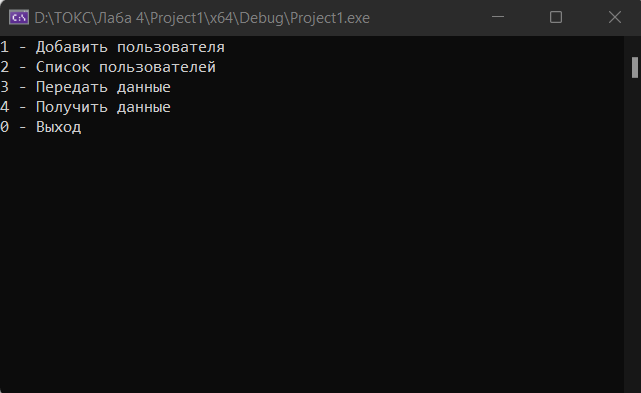
}

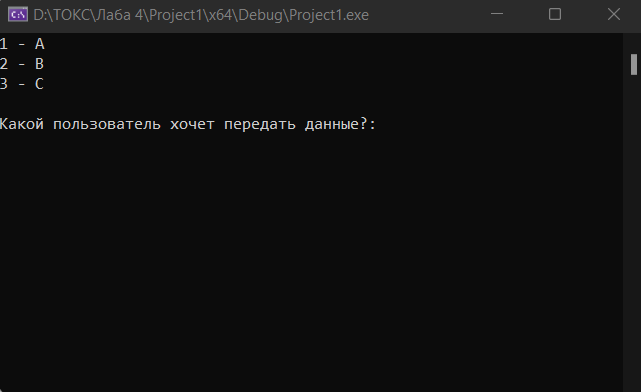
void USER::SetDataTo(string data) { this->dataTo = data; }

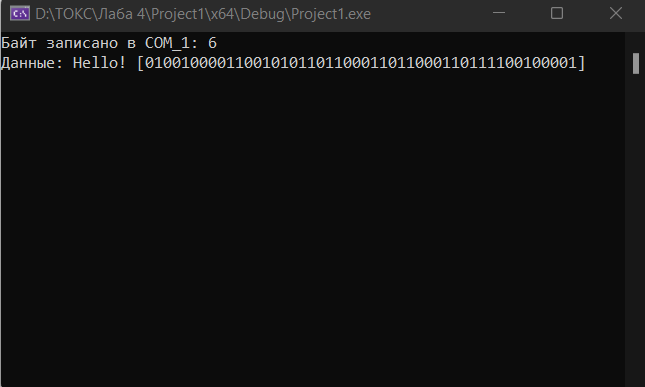
void USER::SetDataFrom(string data) { this->dataFrom = data; }

void USER::SetNameFrom(string data) { this->nameFrom = data; }

**Скриншоты работы программы**





****

