Саратовский Государственный Университет им. Н.Г.Чернышевского

Факультет КНиИТ

Интерфейсы периферийных устройств

Лабораторная работа №1

работу выполнили

студенты 351 группы

Давыдова Анна и Боринос Майя

Саратов**,** 2012

**Assembler**

Блок-схема:

Да

Нет

Печать символа с помощью процедуры Byte\_LPT

Вывод сообщения об ошибке

Ввод символа

Нажата клавиша Esc

Да

Нет

Регистр состояния равен 0d8h

Инициализация LPT порта с помощью процедуры Init\_LPT

Анализ регистра состояния с помощью процедуры Test\_LPT

Сохранение регистров AX и DX в стеке

Загрузка байта регистра управления

Возвращение регистров AX и DX из стека

6 бит = 1

“Printer gotov”

“Printer ne gotov”

Да

Нет

7 бит = 1

“Printer svoboden”

“Printer zanyat”

Сохранение регистра DX, чтение регистра состояния и сохранение его в регистре AX

Анализирование битов регистра состояния

Да

Нет

3 бит = 1

“Net oshibki”

“Oshibka”

Да

Нет

4 бит = 1

“Printer vibran”

“Printer ne vibran”

Да

Нет

5 бит = 1

“Bumagi net”

“Bumaga est”

Да

Нет

2 бит = 1

“Signala podtverjdenia prinyat”

“Signala podtverjdenia ne prinyat”

Да

Нет

Возвращение регистров из стека

Сохранение регистров AX и BX в стек

Загружаем байт из AL в регистр данных

Устанавливает значение стробирующего сигнала в 1

Возвращение регистров AX и BX из стека

Устанавливает значение стробирующего сигнала в 0

Код программы:

.model small

.stack 100h

.data

errormes db 0ah,0dh,'Oshibka!!!',0ah,0dh,'$'

printbusy db 'Printer zanyat',0ah,0dh,'$'

printfree db 'Printer svoboden',0ah,0dh,'$'

printready db 'Printer gotov',0ah,0dh,'$'

printnotready db 'Printer ne gotov',0ah,0dh,'$'

paperyes db 'Bumaga est',0ah,0dh,'$'

paperno db 'Bumagi net',0ah,0dh,'$'

selectno db 'Printer ne vibran',0ah,0dh,'$'

selectyes db 'Printer vibran',0ah,0dh,'$'

erroryes db 'Oshibka',0ah,0dh,'$'

errorno db 'Net oshibki',0ah,0dh,'$'

signalyes db 'Signala podtverjdenia prinyat',0ah,0dh,'$'

signalno db 'Signala podtverjdenia ne prinyat',0ah,0dh,'$'

.code

start:

mov ax,@data

mov ds,ax

call Init\_LPT

lpt1: call Test\_LPT

cmp al,0d8h

je lpt2

lea dx,errormes

mov ah,09h

int 21h

jmp exit

lpt2: mov ah,01h

int 21h

cmp al,27

je exit

call Byte\_LPT

jmp lpt2

exit: mov ah,4ch

mov al,00h

int 21h

Init\_LPT proc

.486

push ax

push dx

mov al,00001100b

mov dx,037Ah

out dx,al

pop dx

pop ax

ret

Init\_LPT endp

Byte\_LPT proc

.486

push ax

push dx

mov dx,0378h

out dx,al

mov al,00001101b

mov dx,037Ah

out dx,al

mov al,00001100b

mov dx,037Ah

out dx,al

pop dx

pop ax

ret

Byte\_LPT endp

Test\_LPT proc

.486

push dx

mov dx,0379h

in al,dx

push ax

test al,10000000b

jz test1

lea dx,printfree

jmp test2

test1: lea dx,printbusy

test2: mov ah,09h

int 21h

pop ax

push ax

test al,01000000b

jz test3

lea dx,printready

jmp test4

test3: lea dx,printnotready

test4: mov ah,09h

int 21h

pop ax

push ax

test al,00100000b

jz test5

lea dx,paperno

jmp test6

test5: lea dx,paperyes

test6: mov ah,09h

int 21h

pop ax

push ax

test al,00010000b

jz test7

lea dx,selectyes

jmp test8

test7: lea dx,selectno

test8: mov ah,09h

int 21h

pop ax

push ax

test al,00001000b

jz test9

lea dx,errorno

jmp test10

test9: lea dx,erroryes

test10: mov ah,09h

int 21h

pop ax

push ax

test al,00000100b

jz test11

lea dx,signalno

jmp test12

test11: lea dx,signalyes

test12: mov ah,09h

int 21h

pop ax

POP dx

ret

Test\_LPT endp

end start

**C++**

Блоксхема:

Создание файла LPT1

Файл корректно создался

Вывод строчки на принтер и закрытие файла

Да

Нет

Код программы:

#include <conio.h>

#include <windows.h>

#include <iostream>

#include <stdio.h>

#include <string>

#include <ctype.h>

using namespace std;

int main ()

{

HANDLE hLPT = CreateFile ( "LPT1" , GENERIC\_WRITE , 0 , NULL , OPEN\_EXISTING , FILE\_ATTRIBUTE\_NORMAL , NULL );

if ( hLPT != INVALID\_HANDLE\_VALUE )

{

cout << "YOUR TEXT\n\n";

DWORD a;

char b;

b = getch();

cout << b;

if ( b == 0x0d ) cout << endl;

WriteFile ( hLPT , &b, 1, &a, NULL );

while ( b != 0x1B )

{

b = getch ();

cout << b;

if ( b == 0x0d ) cout << endl;

WriteFile( hLPT , &b , 1 , &a , NULL );

}

b = 0x0d;

WriteFile ( hLPT , &b , 1 , &a , NULL );

CloseHandle ( hLPT );

}

else cout << "ERROR!!!\n\n";

return 0;

}