МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

кафедра САПР



**ЗВІТ**

до лабораторної роботи №4

**ПРОГРАМУВАННЯ ПАРАЛЕЛЬНОГО ІНТЕРФЕЙСУ**

Виконав:

студент групи КНз-21

Чалий Михайло

Перевірив

Мазур В.В.

Львів-2014

**Мета роботи**

Навчитися використовувати паралельний порт за допомогою функцій BIOS

**Короткі теоретичні відомості**  
  
Порт паралельного інтерфейсу був введений в РС для підключення принтера — звідси і пішла його назва LPT-порт (Line Printer — відрядковий принтер). Традиційний, він же стандартний, LPT-порт (так званий Spp-порт) орієнтований на виведення даних, хоча з деякими обмеженнями дозволяє і вводити дані. Існують різні модифікації LPT-порту — двонаправлений, ЕРР, ЕСР та інші, що розширюють його функціональні можливості, що підвищують продуктивність і що знижують навантаження на процесор. Спочатку вони були фірмовими продуктами окремих виробників, пізніше був прийнятий стандарт IEEE 1284.  
  
Із зовнішнього боку порт має 8-бітову шину даних, 5-бітову шину сигналів стану і 4-бітову шину сигналів, що управляють, виведені на роз'єм-розетку Db-25s. У LPT-порту використовуються логічні рівні ТТЛ, що обмежує допустиму довжину кабелю із-за невисокої перешкодозахищеної ТТЛ-інтерфейсу. Гальванічна розв'язка відсутня — схемна земля пристрою, що підключається, з'єднується з схемною землею комп'ютера. Через це порт є уразливим місцем комп'ютера, страждаючим при порушенні правил підключення і заземлення пристроїв. Оскільки порт зазвичай розташовується на системній платі, в разі його «випалювання» частенько виходить з ладу і його найближче оточення, аж до вигорання всієї системної плати.  
  
З програмного боку LPT-порт є набір регістрів, розташованих в просторі вводу-виводу. Порт може використовувати лінію запиту апаратного переривання. У розширених режимах може використовуватися і канал DMA. Порт має підтримку на рівні BIOS — пошук встановлених портів під час тесту POST і сервіси друку забезпечують виведення символу (по опиту готовності, не використовуючи апаратних переривань), ініціалізацію інтерфейсу і принтера, а також опит стану принтера.  
  
Практично всі сучасні системні плати (ще починаючи з pci-плат для процесорів 486) мають вбудований адаптер LPT-порту. Існують карти ISA з LPT-портом, де він найчастіше є сусідами з парою СОМ-портів, а також з контролерами дискових інтерфейсів (Fdc+IDE). LPT-порт зазвичай присутній і на платі дисплейного адаптера MDA (монохромний текстовий) і HGC (монохромний графічний «Геркулес»). Є і карти PCI з додатковими LPT-портами.  
  
До LPT-портів підключають принтери, плоттери, сканери, комунікаційні пристрої і пристрої зберігання даних, а також електронні ключі, програматори і інші пристрої. Інколи паралельний інтерфейс використовують для зв'язку між двома комп'ютерами — виходить мережа, «зроблена на коліні» (Laplink).

**Програма для роботи з портом LPT1**

title lpt

.286

sseg segment stack

db 256 dup(?)

sseg ends

dseg segment stack

dseg ends

cseg segment

assume ss:sseg,cs:cseg,ds:dseg

start:

jmp main

main:

push ds

xor ax,ax

push ax

mov ax,dseg

mov ds,ax

mov si,0

; Get port address LPT1 from BIOS 0040:0008

mov ax,40h

mov es,ax

mov dx,es:[8] ; 378h

mov al,0

out dx,al

inc dx

in al,dx

call print\_port

mov ah,8 ; Wait for input

int 21h

dec dx

mov al,11111111b

out dx,al

inc dx

in al,dx

call print\_port

mov ah,8

int 21h

inc dx

mov al,0

out dx,al

dec dx

in al,dx

call print\_port

mov ah,8

int 21h

inc dx

mov al,1

out dx,al

dec dx

in al,dx

call print\_port

mov ah,8

int 21h

exit:

mov ax,4c00h

int 21h

print\_port proc near

push dx

mov dh,10000000b

ci:

mov dl,'0'

push ax

test al,dh

jz m0

mov dl,'1'

m0:

mov ah,2

int 21h

pop ax

shr dh,1

jnz ci

pop dx

ret

print\_port endp

cseg ends

end start

**Лістінг програми для роботи з портом LPT1**

JWasm v2.11, Oct 20 2013

.\L4LPT.ASM

title lpt

.286

00000000 sseg segment stack

00000000 000000000000000000 db 256 dup(?)

00000100 sseg ends

00000000 dseg segment stack

00000000 dseg ends

00000000 cseg segment

assume ss:sseg,cs:cseg,ds:dseg

00000000 start:

00000000 EB00 jmp main

00000002 main:

00000002 1E push ds

00000003 33C0 xor ax,ax

00000005 50 push ax

00000006 B80000 mov ax,dseg

00000009 8ED8 mov ds,ax

0000000B BE0000 mov si,0

; Get port address LPT1 from BIOS 0040:0008

0000000E B84000 mov ax,40h

00000011 8EC0 mov es,ax

00000013 268B160800 mov dx,es:[8] ; 378h

00000018 B000 mov al,0

0000001A EE out dx,al

0000001B 42 inc dx

0000001C EC in al,dx

0000001D E83000 call print\_port

00000020 B408 mov ah,8 ; Wait for input

00000022 CD21 int 21h

00000024 4A dec dx

00000025 B0FF mov al,11111111b

00000027 EE out dx,al

00000028 42 inc dx

00000029 EC in al,dx

0000002A E82300 call print\_port

0000002D B408 mov ah,8

0000002F CD21 int 21h

00000031 42 inc dx

00000032 B000 mov al,0

00000034 EE out dx,al

00000035 4A dec dx

00000036 EC in al,dx

00000037 E81600 call print\_port

0000003A B408 mov ah,8

0000003C CD21 int 21h

0000003E 42 inc dx

0000003F B001 mov al,1

00000041 EE out dx,al

00000042 4A dec dx

00000043 EC in al,dx

00000044 E80900 call print\_port

00000047 B408 mov ah,8

00000049 CD21 int 21h

0000004B exit:

0000004B B8004C mov ax,4c00h

0000004E CD21 int 21h

00000050 print\_port proc near

00000050 52 push dx

00000051 B680 mov dh,10000000b

00000053 ci:

00000053 B230 mov dl,'0'

00000055 50 push ax

00000056 84C6 test al,dh

00000058 7402 jz m0

0000005A B231 mov dl,'1'

0000005C m0:

0000005C B402 mov ah,2

0000005E CD21 int 21h

00000060 58 pop ax

00000061 D0EE shr dh,1

00000063 75EE jnz ci

00000065 5A pop dx

00000066 ret

00000067 print\_port endp

00000067 cseg ends

end start

Binary Map:

Segment Pos(file) RVA Size(fil) Size(mem)

---------------------------------------------------------------

<header> 0 0 30 0

sseg 30 0 100 100

dseg 130 100 0 0

cseg 130 100 67 67

---------------------------------------------------------------

197 167

Macros:

N a m e Type

@CatStr . . . . . . . . . . . . Func

@Environ . . . . . . . . . . . . Func

@InStr . . . . . . . . . . . . . Func

@SizeStr . . . . . . . . . . . . Func

@SubStr . . . . . . . . . . . . Func

Segments and Groups:

N a m e Size Length Align Combine Class

cseg . . . . . . . . . . . . . . 16 Bit 0067 Para Private ''

dseg . . . . . . . . . . . . . . 16 Bit 0000 Para Stack ''

sseg . . . . . . . . . . . . . . 16 Bit 0100 Para Stack ''

Procedures, parameters and locals:

N a m e Type Value Segment Length

print\_port . . . . . . . . . . . P Near 0050 cseg 0017 Public

ci . . . . . . . . . . . . . . L Near 0053 cseg

m0 . . . . . . . . . . . . . . L Near 005C cseg

Symbols:

N a m e Type Value Attr

exit . . . . . . . . . . . . . . L Near 4Bh cseg

main . . . . . . . . . . . . . . L Near 2h cseg

start . . . . . . . . . . . . . L Near 0h cseg Public

.\L4LPT.ASM: 84 lines, 2 passes, 0 ms, 0 warnings, 0 errors

# Висновки

Виконуючи роботу я навчитися використовувати паралельний порт за допомогою функцій BIOS