**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №5**

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

Тема: Разработка собственного прерывания

|  |  |  |
| --- | --- | --- |
| Студент гр. 0383 |  | Сабанов П.А. |
| Преподаватель |  | Ефремов М.А. |

Санкт-Петербург

2021

**Цель работы.**

Написать программу, заменяющую обработчик прерывания от системного таймера на обработчик, выводящий звуковой сигнал. По завершении программа должна восстановить старый обработчик прерывания.

Вариант 11.

**Ход работы.**

Была написана функция interfunction, являющаяся обработчиком прерывания. Она выводит звук частоты 100.

Была написана функция disable\_sound, отключающая звук.

В главной функции происходит считывание текущего обработчика прерывания таймера (номер вектора 08h) и установка нового обработчика прерывания (функция interfunction). Затем программа ждёт нажатия пользователем любой кнопки клавиатуры. После этого она возвращает старый обработчик прерывания, вызывает функцию disable\_sound и завершается.

**Выводы.**

Был написан обработчик прерывания interfunction, издающий звук.

Была написана программа, издающая звук при прерывании от системного таймера. Перед завершением программа возвращает старый обработчик прерывания и отключает звук, который мог включить interfunction.

**ПРИЛОЖЕНИЕ А**

**Исходный код программы**

DOSSEG

.MODEL SMALL

.STACK 100h

.DATA

keep\_cs dw 0

keep\_ip dw 0

.CODE

; номер вектора прерывания

vector\_n db 08h

; функция-обработчик прерывания

; void interruption();

interfunction proc far

; сохраняем регистры

push ax

push bx

push cx

push dx

; выводим звук

mov cx, 100 ; частота

in al, 61h ; получаем значение из управляющего регистра порта B PPI (контроллера 8255)

or al, 3 ; устанавливаем биты 0 и 1 (включить спикер и использовать 2-й канал для генерации импульсов спикера)

out 61h, al ; выводим значение в управляющий регистр

mov al, 10110110b ; управляющее слово таймера

out 43h, al ; выводим значение в порт таймера

mov dx, 12h

mov ax, 34ddh ; DX:AX = 1193181 - частота работы таймера

div cx ; значение счётчика таймера AX = DX:AX / CX

out 42h, al ; выводим младший байт счетчика во 2-й канал таймера

mov al, ah

out 42h, al ; выводим старший байт

; разрешение обработки прерываний с более низкими уровнями, чем только что обработанное

mov al, 20h

out 20h, al

; восстанавливаем регистры

pop dx

pop cx

pop bx

pop ax

iret

interfunction endp

disable\_sound proc near

push ax

in al, 61h

and al, not 3

out 61h, al

pop ax

ret

disable\_sound endp

main proc far

mov ax, @data

mov ds, ax

; сохраняем функцию прерывания

mov ah, 35h ; функция получения вектора

mov al, vector\_n ; номер вектора

int 21h

mov keep\_ip, bx ; запоминание смещения

mov keep\_cs, es ; запоминание вектора прерывания

; устанавливаем нашу функцию прерывания

push ds

mov dx, offset interfunction

mov ax, seg interfunction

mov ds, ax

mov ah, 25h ; функция установки вектора

mov al, vector\_n

int 21h

pop ds

; ждём нажатия клавиши

mov ah, 0

int 16h

; возвращаем сохранённую функцию прерывания

cli

push ds

mov dx, keep\_ip

mov ax, keep\_cs

mov ds, ax

mov ah, 25h ; функция установки вектора

mov al, vector\_n

int 21h

pop ds

sti

; выключаем звук

call disable\_sound

; выход из программы

mov ah, 4ch

xor al, al

int 21h

main endp

end main

**ПРИЛОЖЕНИЕ Б**

**Листинг компиляции программы**

#Microsoft (R) Macro Assembler Version 5.10 12/26/21 12:55:2

Page 1-1

DOSSEG

.MODEL SMALL

.STACK 100h

.DATA

0000 0000 keep\_cs dw 0

0002 0000 keep\_ip dw 0

.CODE

; ÐœÐŸÐŒÐµÑ Ð²ÐµÐºÑÐŸÑÐ° Ð¿ÑÐµÑÑÐ²Ð°ÐœÐžÑ



0000 08 vector\_n db 08h

; ÑÑÐœÐºÑÐžÑ-ÐŸÐ±ÑÐ°Ð±ÐŸÑÑÐžÐº Ð¿ÑÐµÑÑ

Ð²Ð°ÐœÐžÑ

; void interruption();

0001 interfunction proc far

; ÑÐŸÑÑÐ°ÐœÑÐµÐŒ ÑÐµÐ³ÐžÑÑÑÑ

0001 50 push ax

0002 53 push bx

0003 51 push cx

0004 52 push dx

; Ð²ÑÐ²ÐŸÐŽÐžÐŒ Ð·Ð²ÑÐº

0005 B9 0064 mov cx, 100 ; ÑÐ°ÑÑÐŸÑÐ°

0008 E4 61 in al, 61h ; Ð¿ÐŸÐ»ÑÑÐ°ÐµÐŒ Ð·ÐœÐ°ÑÐµÐœÐ

žÐµ ÐžÐ· ÑÐ¿ÑÐ°Ð²Ð»ÑÑÑÐµÐ³ÐŸ ÑÐµÐ³ÐžÑÑÑ

Ð° Ð¿ÐŸÑÑÐ° B PPI (ÐºÐŸÐœÑÑÐŸÐ»Ð»ÐµÑÐ° 82

55)

000A 0C 03 or al, 3 ; ÑÑÑÐ°ÐœÐ°Ð²Ð»ÐžÐ²Ð°ÐµÐŒ Ð±ÐžÑ

Ñ 0 Ðž 1 (Ð²ÐºÐ»ÑÑÐžÑÑ ÑÐ¿ÐžÐºÐµÑ Ðž Ðž

ÑÐ¿ÐŸÐ»ÑÐ·ÐŸÐ²Ð°ÑÑ 2-Ð¹ ÐºÐ°ÐœÐ°Ð» ÐŽÐ»Ñ Ð

³ÐµÐœÐµÑÐ°ÑÐžÐž ÐžÐŒÐ¿ÑÐ»ÑÑÐŸÐ² ÑÐ¿ÐžÐºÐµ

ÑÐ°)

000C E6 61 out 61h, al ; Ð²ÑÐ²ÐŸÐŽÐžÐŒ Ð·ÐœÐ°ÑÐµÐœÐž

Ðµ Ð² ÑÐ¿ÑÐ°Ð²Ð»ÑÑÑÐžÐ¹ ÑÐµÐ³ÐžÑÑÑ

000E B0 B6 mov al, 10110110b ; ÑÐ¿ÑÐ°Ð²Ð»ÑÑÑÐµÐµ

ÑÐ»ÐŸÐ²ÐŸ ÑÐ°Ð¹ÐŒÐµÑÐ°

0010 E6 43 out 43h, al ; Ð²ÑÐ²ÐŸÐŽÐžÐŒ Ð·ÐœÐ°ÑÐµÐœÐž

Ðµ Ð² Ð¿ÐŸÑÑ ÑÐ°Ð¹ÐŒÐµÑÐ°

0012 BA 0012 mov dx, 12h

0015 B8 34DD mov ax, 34ddh ; DX:AX = 1193181 - ÑÐ°ÑÑÐ

ŸÑÐ° ÑÐ°Ð±ÐŸÑÑ ÑÐ°Ð¹ÐŒÐµÑÐ°

0018 F7 F1 div cx ; Ð·ÐœÐ°ÑÐµÐœÐžÐµ ÑÑÑÑÑÐžÐºÐ°

ÑÐ°Ð¹ÐŒÐµÑÐ° AX = DX:AX / CX

001A E6 42 out 42h, al ; Ð²ÑÐ²ÐŸÐŽÐžÐŒ ÐŒÐ»Ð°ÐŽÑÐžÐ¹

Ð±Ð°Ð¹Ñ ÑÑÐµÑÑÐžÐºÐ° Ð²ÐŸ 2-Ð¹ ÐºÐ°ÐœÐ°Ð»

ÑÐ°Ð¹ÐŒÐµÑÐ°

001C 8A C4 mov al, ah

#Microsoft (R) Macro Assembler Version 5.10 12/26/21 12:55:2

Page 1-2

001E E6 42 out 42h, al ; Ð²ÑÐ²ÐŸÐŽÐžÐŒ ÑÑÐ°ÑÑÐžÐ¹

Ð±Ð°Ð¹Ñ

; ÑÐ°Ð·ÑÐµÑÐµÐœÐžÐµ ÐŸÐ±ÑÐ°Ð±ÐŸÑÐºÐž Ð

¿ÑÐµÑÑÐ²Ð°ÐœÐžÐ¹ Ñ Ð±ÐŸÐ»ÐµÐµ ÐœÐžÐ·ÐºÐžÐŒÐ

ž ÑÑÐŸÐ²ÐœÑÐŒÐž, ÑÐµÐŒ ÑÐŸÐ»ÑÐºÐŸ ÑÑÐŸ

ÐŸÐ±ÑÐ°Ð±ÐŸÑÐ°ÐœÐœÐŸÐµ

0020 B0 20 mov al, 20h

0022 E6 20 out 20h, al

; Ð²ÐŸÑÑÑÐ°ÐœÐ°Ð²Ð»ÐžÐ²Ð°ÐµÐŒ ÑÐµÐ³ÐžÑ

ÑÑÑ

0024 5A pop dx

0025 59 pop cx

0026 5B pop bx

0027 58 pop ax

0028 CF iret

0029 interfunction endp

0029 disable\_sound proc near

0029 50 push ax

002A E4 61 in al, 61h

002C 24 FC and al, not 3

002E E6 61 out 61h, al

0030 58 pop ax

0031 C3 ret

0032 disable\_sound endp

0032 main proc far

0032 B8 ---- R mov ax, @data

0035 8E D8 mov ds, ax

; ÑÐŸÑÑÐ°ÐœÑÐµÐŒ ÑÑÐœÐºÑÐžÑ Ð¿ÑÐµÑ

ÑÐ²Ð°ÐœÐžÑ

0037 B4 35 mov ah, 35h ; ÑÑÐœÐºÑÐžÑ Ð¿ÐŸÐ»ÑÑÐµÐœ

ÐžÑ Ð²ÐµÐºÑÐŸÑÐ°

0039 2E: A0 0000 R mov al, vector\_n ; ÐœÐŸÐŒÐµÑ Ð²ÐµÐºÑÐŸÑÐ

°

003D CD 21 int 21h

003F 89 1E 0002 R mov keep\_ip, bx ; Ð·Ð°Ð¿ÐŸÐŒÐžÐœÐ°ÐœÐžÐµ Ñ

ÐŒÐµÑÐµÐœÐžÑ

0043 8C 06 0000 R mov keep\_cs, es ; Ð·Ð°Ð¿ÐŸÐŒÐžÐœÐ°ÐœÐžÐµ Ð²

ÐµÐºÑÐŸÑÐ° Ð¿ÑÐµÑÑÐ²Ð°ÐœÐžÑ

; ÑÑÑÐ°ÐœÐ°Ð²Ð»ÐžÐ²Ð°ÐµÐŒ ÐœÐ°ÑÑ ÑÑÐ

œÐºÑÐžÑ Ð¿ÑÐµÑÑÐ²Ð°ÐœÐžÑ

0047 1E push ds

0048 BA 0001 R mov dx, offset interfunction

004B B8 ---- R mov ax, seg interfunction

004E 8E D8 mov ds, ax

0050 B4 25 mov ah, 25h ; ÑÑÐœÐºÑÐžÑ ÑÑÑÐ°ÐœÐŸÐ²

#Microsoft (R) Macro Assembler Version 5.10 12/26/21 12:55:2

Page 1-3

ÐºÐž Ð²ÐµÐºÑÐŸÑÐ°

0052 2E: A0 0000 R mov al, vector\_n

0056 CD 21 int 21h

0058 1F pop ds

; Ð¶ÐŽÑÐŒ ÐœÐ°Ð¶Ð°ÑÐžÑ ÐºÐ»Ð°Ð²ÐžÑÐž

0059 B4 00 mov ah, 0

005B CD 16 int 16h

; Ð²ÐŸÐ·Ð²ÑÐ°ÑÐ°ÐµÐŒ ÑÐŸÑÑÐ°ÐœÑÐœÐœÑ

Ñ ÑÑÐœÐºÑÐžÑ Ð¿ÑÐµÑÑÐ²Ð°ÐœÐžÑ

005D FA cli

005E 1E push ds

005F 8B 16 0002 R mov dx, keep\_ip

0063 A1 0000 R mov ax, keep\_cs

0066 8E D8 mov ds, ax

0068 B4 25 mov ah, 25h ; ÑÑÐœÐºÑÐžÑ ÑÑÑÐ°ÐœÐŸÐ²

ÐºÐž Ð²ÐµÐºÑÐŸÑÐ°

006A 2E: A0 0000 R mov al, vector\_n

006E CD 21 int 21h

0070 1F pop ds

0071 FB sti

; Ð²ÑÐºÐ»ÑÑÐ°ÐµÐŒ Ð·Ð²ÑÐº

0072 E8 0029 R call disable\_sound

; Ð²ÑÑÐŸÐŽ ÐžÐ· Ð¿ÑÐŸÐ³ÑÐ°ÐŒÐŒÑ

0075 B4 4C mov ah, 4ch

0077 32 C0 xor al, al

0079 CD 21 int 21h

007B main endp

end main

#Microsoft (R) Macro Assembler Version 5.10 12/26/21 12:55:2

Symbols-1

Segments and Groups:

N a m e Length Align Combine Class

DGROUP . . . . . . . . . . . . . GROUP

\_DATA . . . . . . . . . . . . 0004 WORD PUBLIC 'DATA'

STACK . . . . . . . . . . . . 0100 PARA STACK 'STACK'

\_TEXT . . . . . . . . . . . . . 007B WORD PUBLIC 'CODE'

Symbols:

N a m e Type Value Attr

DISABLE\_SOUND . . . . . . . . . N PROC 0029 \_TEXT Length = 0009

INTERFUNCTION . . . . . . . . . F PROC 0001 \_TEXT Length = 0028

KEEP\_CS . . . . . . . . . . . . L WORD 0000 \_DATA

KEEP\_IP . . . . . . . . . . . . L WORD 0002 \_DATA

MAIN . . . . . . . . . . . . . . F PROC 0032 \_TEXT Length = 0049

VECTOR\_N . . . . . . . . . . . . L BYTE 0000 \_TEXT

@CODE . . . . . . . . . . . . . TEXT \_TEXT

@CODESIZE . . . . . . . . . . . TEXT 0

@CPU . . . . . . . . . . . . . . TEXT 0101h

@DATASIZE . . . . . . . . . . . TEXT 0

@FILENAME . . . . . . . . . . . TEXT lab5

@VERSION . . . . . . . . . . . . TEXT 510

112 Source Lines

112 Total Lines

23 Symbols

47952 + 457258 Bytes symbol space free

0 Warning Errors

0 Severe Errors