Министерство науки и высшего образования РФ

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №8

по дисциплине «ЭВМ и переферийные устройства»

на тему «Таймер и звук»

Выполнили: студенты группы 22ВВП1

Беляев Д. И.

Демин М. С.

Сергунов М. Р.

Приняли:

Никишин К. И.

Патунин Д. В.

Пенза 2025

**Цель работы**

Изучить принцип управления системным таймером и способы генерации звука.

**Задание**

Разработать и отладить программу генерации звукового сигнала. Генерация должна прекращаться с нажатием любой клавиши.  
\* Модернизировать программу, обеспечивая изменение тона сигнала при каждом нажатии клавиши. Определить с помощью модернизированной программы значения коэффициента пересчета таймера, при которых обеспечивается высший и низший тон услышанного звукового сигнала.

**Ход работы**

Листинг:

stseg segment para stack

dw 100h dup(?)

stseg ends

dseg segment para

; коэффициент текущего делителя PIT

curCount dw 0

; таблица делителей для цифр '0'..'9'

toneTable dw 5000, 4500, 4000, 3600, 3200, 2800, 2400, 2000, 1600, 1200

; длительность звука (итераций задержки)

durCount dw 0ffffh

dseg ends

cseg segment para public 'code'

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

; main – основная программа

main proc far

push ds

xor ax, ax

push ax

mov ax, dseg

mov ds, ax

key\_loop:

call waitKey ; ждём клавишу, ASCII в AL

cmp al, 27 ; ESC?

je exit\_prog

cmp al, '0'

jb key\_loop ; не цифра

cmp al, '9'

ja key\_loop ; не цифра

; индекс цифры = AL - '0'

mov bl, al

sub bl, '0'

xor bh, bh ; BX = индекс

; получаем делитель из таблицы toneTable[BX]

lea si, toneTable

mov ax, bx

shl ax, 1 ; байтовый шаг

add si, ax

mov ax, [si]

mov [curCount], ax

mov cx, [durCount] ; установить длительность

call playToneDuration

jmp key\_loop

exit\_prog:

pop ds

mov ah, 4ch

int 21h

main endp

; waitKey – ожидает нажатия клавиши

; выход:

; AL = ASCII-код нажатой клавиши

waitKey proc near

mov ah, 0

int 16h

ret

waitKey endp

; playTone – включает спикер с текущим делителем

playTone proc near

mov ax, [curCount]

mov dx, ax

mov al, 0b6h

out 43h, al

mov al, dl

out 42h, al

mov al, dh

out 42h, al

in al, 61h

or al, 03h

out 61h, al

ret

playTone endp

; stopTone – выключает спикер

stopTone proc near

in al, 61h

and al, 0fch

out 61h, al

ret

stopTone endp

; delayStep – простая задержка одного шага

delayStep proc near

nop

ret

delayStep endp

; playToneDuration – воспроизводит тон в течение cx шагов задержки

; вход:

; cx = количество итераций delayStep

playToneDuration proc near

call playTone

play\_loop:

call delayStep

loop play\_loop

call stopTone

ret

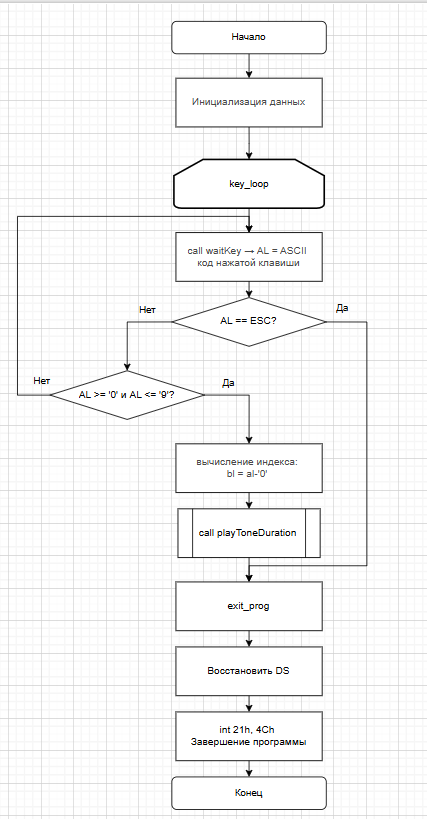
playToneDuration endp

cseg ends

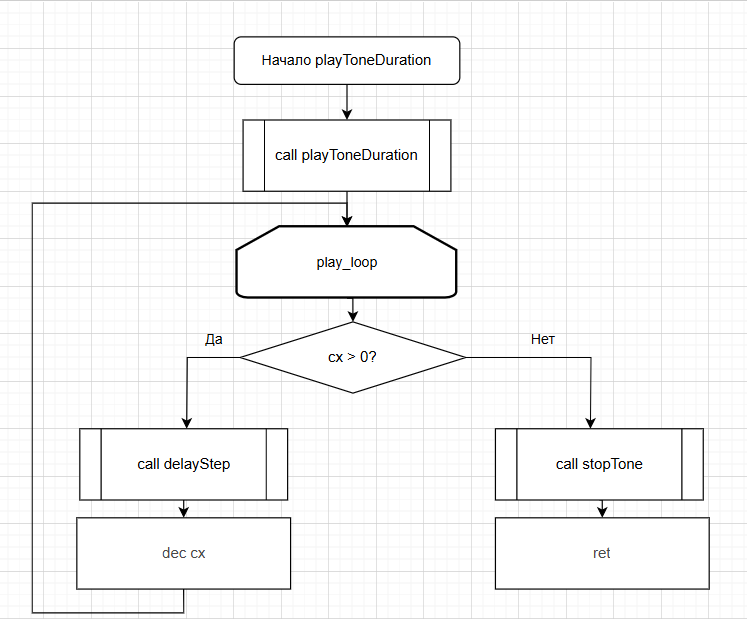
end main

Блок-схема программы:

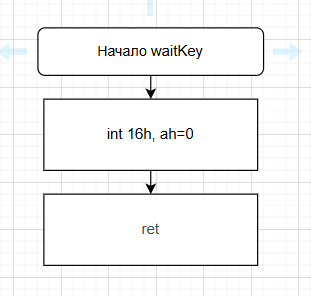
Main



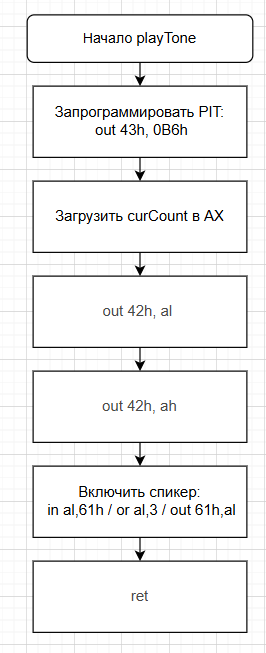
playToneDuration



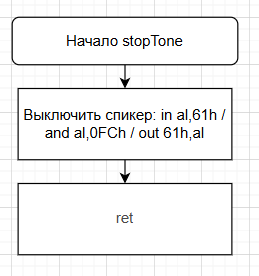
waitKey



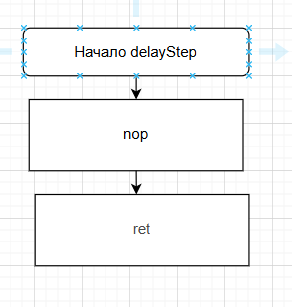
playTone



stopTone



deleyStep



Результат работы программы:

****

**Вывод**

Изучили принцип управления системным таймером и способы генерации звука.