**Вектор переривань та процедура обслуговування переривань**

Контрольна робота з дисципліни „Програмне забезпечення автоматизованих систем”

Виконав ст. гр. КНз-21 Босак Андрій Олексійович

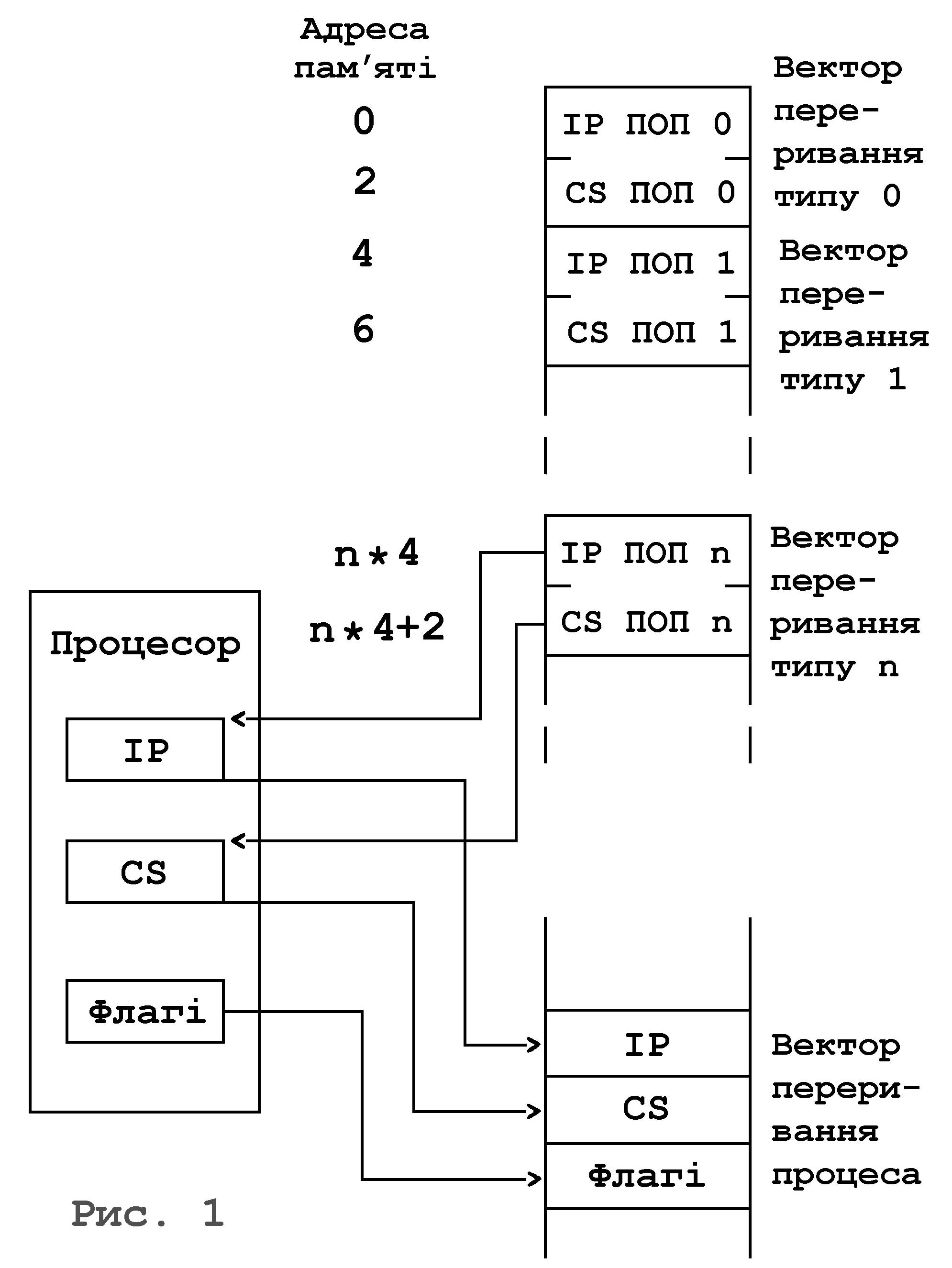
Луцький інститут розвитку людини Відкритого міжнародного університету розвитку людини „Україна”.

Луцьк 2005.

Звернення до функцій DOS і BIOS здійснюється за допомогою програмних переривань (команда ІNT).

Система переривань машин типу IBM PC у принципі не відрізняється від будь-якої іншої системи векторизованих переривань. Самий початок оперативної пам'яті від адреси 0000h до 03FFh відводиться під вектори переривань - чотирьохбайтові області, в яких зберігаються адреси програм обробки переривань (ПОП). У два старші байти кожного вектора записується сегментна адреса ПОП, в два молодші - відносна адреса точки входу в ПОП в сегменті. Вектори, як і відповідні їм переривання, мають номери, які називаються типами, причому вектор з номером 0 (вектор типу 0) розташовується починаючи з адреси 0, вектор типу 1−з адреси 4, вектор типу 2− з адреси 8 і т.д. Вектор з номером N займає, таким чином, байти пам'яті від N\*4 до N\*4+2. Всього у виділеній під вектори області пам'яті поміщається 256 векторів.

Одержавши сигнал на виконання процедури переривання з певним номером, процесор зберігає в стеку виконуваної програми слово прапорів, а також сегментну і відносну адресу програмного сегменту (вміст CS і IP) і завантажує CS і IP з відповідного вектора переривань, здійснюючи тим самим перехід на ПОП (рис 1).



Програма обробки переривання зазвичай закінчується командою повернення з переривання IRET, що виконує зворотні дії - завантаження IP, CS і регістра прапорів із стека, що приводить до повернення в основну програму в точку, де вона була перервана.

Запити на виконання процедури переривань можуть мати різну природу. Перш за все розрізняють апаратні переривання від периферійних пристроїв або інших компонентів системи і програмні переривання, що викликаються командою INT, яка використовується, зокрема, для програмного звернення до функцій DOS і BIOS. Сигнали, що збуджують апаратні переривання, можуть ініціюватися ланцюгами самого процесора, наприклад, при спробі виконання операції ділення на нуль (такі переривання називаються внутрішніми, або відмовами), а можуть приходити з периферійного устаткування (зовнішні переривання). Незалежно від джерела, процедура переривання, описана вище, завжди виконується однаково, як для апаратних, так і для програмних переривань.

Велика частина векторів переривань зарезервована для виконання визначених дій; частина з них автоматично заповнюється адресами системних програм при завантаженні системи. Вектори з наступними номерами можуть представляти особливий інтерес для користувача:

00h - ділення на 0;

0lh - покрокове виконання (при TF=1);

02h - немасковане переривання (виведення NMI процесора)

03h - команда INT без числового параметра;

04h - INTO - переривання по переповненню (ініціюється апаратно, але лише за наявності в програмі команди INTO);

08h - переривання від таймера (апаратне);

09h - переривання від клавіатури (апаратне);

0Ah - зарезервовано для підключення нестандартного пристрою (апаратне);

0Bh - другий послідовний порт COM2 (апаратне);

0Сh - перший послідовний порт СОМ1 (апаратне);

0Dh - жорсткий диск (апаратне); 0Eh - гнучкий диск (апаратне);

0Fh - паралельний порт (принтер LPT1) (апаратне);

10h - відеодрайвер BIOS;

13h - драйвер BIOS диска;

16h - драйвер BIOS клавіатури;

17h - драйвер BIOS принтера;

1Bh - обробник переривань по <Ctrl>/<Break>;

1Ch - вхід в програму BIOS обробки переривань від системного таймера (18,2 переривань в секунду);

1Dh - адреса таблиці відеопараметрів, BIOS;

lEh - адреса таблиці параметрів дискети, BIOS;

1Fh - адреса другої половини таблиці шрифтів графічних режимів 4...6, BIOS;

21n - диспетчер функцій DOS;

22h - обробник завершення процесу, DOS;

23h - обробник переривань по <Ctrl>/C;

24h - обробник переривань по критичній помилці;

25h - абсолютне читання диска;

26h - абсолютний запис на диск;

2Fh - мультиплексне переривання DOS;

43h - адреса таблиці шрифтів графічних режимів, BIOS;

60h...66h - переривання користувача;

67h - драйвер додаткової пам'яті LIM EMS;

68h...6Fh - вільні вектори;

70h...7Fh - апаратні переривання ведомого контроллера машин типу AT;

78h...7Fh - вільні вектори;

Flh...FFh - не використовуються.

Як видно з таблиці, вектори переривань можна умовно розбити на наступні групи:

вектори апаратних переривань (08h...0Fh і 70h...77h);

драйвери BIOS (10h, 13h, 16h і т.д.);

програми DOS (21h, 22h, 23h і т.д.);

адреси системних таблиць DOS або BIOS (1Dh, lEh, 43h і т.д.).

Системні програми, адреси яких зберігаються у векторах переривань, в більшості своїй є всього лише диспетчерами, що відкривають доступ до великих груп програм, що реалізують системні функції. Так, відеодрайвер BIOS (векторl0h) включає програми зміни відеорежиму, управління курсором, задання колірної палітри, завантаження шрифтів і багато інших. Особливо характерний в цьому відношенні вектор 21h, через який здійснюється виклик практично всіх функцій DOS: введення з клавіатури і виведення на екран, обслуговування файлів, каталогів і дисків, управління пам'яттю і процесами, служби часу і т.д.

Звернення з прикладної програми до системних функцій здійснюється одноманітно. У регістр АН засилається номер функції (не слід плутати з типом переривання!), в інші регістри − вихідні дані, необхідні для виконання конкретної системної програми. Після цього виконується команда INT з числовим аргументом, що вказуює тип (номер) переривання, наприклад, INT 21h.

Більшість функцій DOS і багато функцій BIOS повертають в флазі перенесення CF код завершення. Якщо функція виконалась успішно CF=0, у разі ж будь-якої помилки CF=1. У останньому випадку в одному з регістрів (найчастіше в АХ) повертається ще і код помилки. Таким чином, типова процедура звернення до системних засобів виглядає таким чином:

mov АН,fun ; fun - номер функції

;Заповнення тих чи інших регістрів (AL, BX, ES,

ВР і ін.)

;параметрами, необхідними для виконання

даної функції

...

int 21h ; Перехід у MS-DOS

jc error

;Продовження програми

...

error:cmp AX,1 ; Аналіз коду завершення

je err1

cmp AX2

je err

...

Аналогічно викликаються і функції BIOS.

**Задача на звуковий сигнал**

У найпростішому випадку в IBM PC для генерації звуку використовується мікросхема інтегрального таймера 8253 або 8254. Ця мікросхема має три незалежні канали, кожний з яких може програмуватися для роботи в режимі поділювача частоти або генератора одиночних імпульсів. Кожен канал містить 16-розрядний лічильник, в який записується значення дільника частоти або коефіцієнта перерахунку (залежно від режиму роботи). Кожен канал має вхід частоти (clk) і вхід дозволу (gate). На вхід частоти всіх каналів подається імпульсний сигнал частотою 1,19 Мгц. Канал 0 мікросхеми таймера використовується для вироблення сигналу переривання по таймеру (частотою 18,2 Гц). Канал 1 працює в режиму генерації одиночних імпульсів через кожні 15 мкс. Цей сигнал використовується для регенерації динамічної пам'яті ЕОМ.

Канал 2 мікросхеми початково програмується для роботи в режимі дільника частоти. Вихід каналу використовується для генерації звуку через вбудований динамік. Для управління звуком використовуються біти 0 і 1 системного порту В (мікросхема 8255). Біт 0 використовується для дозволу проходження сигналу на вихід каналу 2 таймера. Сигнал з виходу каналу 2 подається на схему "І", на другий вхід який подається сигнал біта 1 системного порту В. Цей сигнал може вирішувати або забороняти проходження сигналу з виходу каналу 2 таймера, а при закритому каналі 2 (бітом 0 порту В) сигнал біта 1 порту В може використовуватися для безпосередньої генерації звуку в динаміку.

Адреса системного порту В - 61h, адреси каналів таймера - 40h, 41h, 42h, 43h - для каналів 0, 1, 2 і регістра, що управляє, відповідно. Нижче розглянуті приклади генерації звуку за допомогою сигналу біта 1 системного порту В, а також за допомогою таймера. Розглянутий випадок генерації звуку з використанням переривання.

**Приклади генерації звуку**

Завдання 1. Написати програму, що видає різні звуки при натисненні на клавіші '1' і '2'. Для генерації звуку слід використовувати сигнал біта 1 системного порту В. Виход з програми повинен здійснюватися по натисненню клавіші 'q'.

Assume CS: Code, DS: Code

Code SEGMENT

org 100h

frequency1 equ 300 ; Затримка переключення 1

frequency2 equ 500 ; Затримка переключення 2

number\_cycles1 equ 1000 ; Кількість циклів (трив.)

number\_cycles2 equ 600 ; Кількість циклів (трив.)

port\_b equ 61h ; Адреса системного порта В

.286

Start proc near

mov ax,cs

mov ds,ax ; DS = CS

beg1: call kbin ; Опитування клавіатури

cmp al,'1' ; = '1' ?

jnz beg2 ; Ні

call ton1 ; Звук висоти 1

jmp beg1 ; Перехід на початок циклу

beg2: cmp al,'2' ; = '2' ?

jnz beg3 ; Ні

call ton2 ; Звук висоти 2

jmp beg1 ; Перехід на початок циклу

beg3: cmp al,'q' ; = 'q' ?

jnz beg1 ; Ні

int 20h ; Вихід з програми

start endp

ton2 proc near ; Процедура генерації звука 2

mov dx,number\_cycles2 ; Тривалість 2

mov di,frequency2 ; Затримка 2

jmp ton0 ; Перехід на універсальну процедуру генерації звука

ton1 proc near ; Процедура генерації звука 1

mov dx,number\_cycles1 ; Тривалість 1

mov di,frequency1 ; Затримка 1

; Універсальна процедура генерації звука

; DX - кількість циклів, DI - затримка

ton0 proc near

cli ; Заборона переривань

in al,port\_b ; Читанння стану системн. порту В

and al,11111110b ; Відкл. динаміка від таймера

ton01: or al,00000010b ; Включення динаміка

out port\_b,al ; Запис в системний порт В

mov cx,di ; Лічильник циклу затримки

loop $ ; Затримка

; Вимкнення звука

and al,11111101b ; Вимкнення динаміка

out port\_b,al ; Запис в системний порт В

mov cx,di ; Лічильник циклу затримки

loop $ ; Затримка

dec dx ; Декремент лічильника кількість циклів

jnz ton01 ; Перехід на початок нового періода

sti ; Дозвіл переривань

ret ; Вихід з процедури

ton0 endp ; Кінець універсальної процедури

ton1 endp ; Кінець процедури генерації звука 1

ton2 endp ; Кінець процедури генерації звука 2

kbin proc near ; Ввод з клавіатури з очікуванням

mov ah,0 ; Функція 0

int 16h ; клавіатурного переривання

ret ; Вихід з процедури

kbin endp ; Кінець процедури вводу з клавіатури

code ends ; Кінець сегмента (кодового)

END Start ; Вказівка на точку входа

Завдання 2. Написати програму, що видає різні звуки при натисненні на клавіші '1' і '2'. Для генерації звуку слід використовувати вихід каналу 2 таймера. Вихід з програми повинен здійснюватися по натисненню клавіші 'q'.

Assume CS: Code, DS: Code

Code SEGMENT

org 100h

frequency1 equ 1000 ; Коефіц. ділення 1

frequency2 equ 3000 ; Коефіц. ділення 2

duration equ 50000 ; Тривалість

port\_b equ 61h ; Адреса системного порта В

.286

Start proc near ; Основна процедура

mov ax,cs

mov ds,ax ; DS = CS

beg1: call kbin ; Опитування клавіатури

cmp al,'1' ; = '1' ?

jnz beg2 ; Ні

call ton1 ; Звук висоти 1

jmp beg1 ; Перехід на початок циклу

beg2: cmp al,'2' ; = '2' ?

jnz beg3 ; Ні

call ton2 ; Звук висоти 2

jmp beg1 ; Перехід на початок циклу

beg3: cmp al,'q' ; = 'q' ?

jnz beg1 ; Ні

int 20h ; Вихід з програми

start endp ; Кінець основної процедури

ton2 proc near ; Процедура генерації звука 2

mov dx,duration ; Тривалість

mov di,frequency2 ; Коефіцієнт ділення 2

jmp ton0 ; Перехід на універсальну процедуру

ton1 proc near ; Процедура генерації звука 1

mov dx,duration ; Тривалість

mov di,frequency1 ; Коефіцієнт ділення 2

; Універсальна процедура генерації звука

; DX - тривалість, DI - коефіц. ділення

ton0 proc near

cli ; Заборона переривань

; Включення динаміка і таймера

in al,61h ; Читання стану системного порта В

or al,3 ; Дозвіл звучання (біти 0 и 1)

out 61h,al ; Запис в системний порт В

; Програмування поділювача частоти 2 канала

mov ax,di ; Поділювач частоти

out 42h,al ; Мол.байт частоти  канал 2 таймера

xchg al,ah ; AH  AL

out 42h,al ; Ст.байт частоти  канал 2 таймера

; Формування затримки

mov cx,dx ; Лічильник циклу затримки

ton01: push cx ; Команди, що використовуються тільки для

pop cx ; збільшення трив. цикла затримки

loop ton01 ; Затримка

; Виключення звука

in al,61h ; Читання стану системного порта В

and al,0fch ; Заборона звучання (биты 0 и 1)

out 61h,al ; Запис в системний порт В

sti ; Дозвіл переривань

ret ; Вихід з піроцедури

ton0 endp ; Кінець універсальної процедури

ton1 endp ; Кінець процедури генерації звука 1

ton2 endp ; Кінець процедури генерації звука 2

kbin proc near ; Ввод з клавіатури і перевірка на вибір гри

kbin endp

code ends ; Кінець сегмента (кодового)

END Start ; Вказівка на точку входу

**Список литературы**

Финогенов К. Г. «Самоучитель по системным функциям MS-DOS». М.: 1993.

Пирогов В. Ю. «Ассемблер. Учебный курс».

В контрольній роботі також використовувались матеріали сайту http://www.BankReferatov.ru ,перекладені на українську мову.