
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

«МІКРОПРОЦЕСОРНІ СИСТЕМИ» КОМП'ЮТЕРНИЙ ПРАКТИКУМ

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для студентів,
які навчаються за спеціальністю 171 «Електроніка»,
спеціалізацією «Електронні компоненти і системи»*

Київ
КПІ ім. Ігоря Сікорського
2017

«Мікропроцесорні системи»: Комп'ютерний практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності 171 «Електроніка», спеціалізації «Електронні компоненти і системи» / КПІ ім. Ігоря Сікорського ; уклад.: Т. О. Терещенко, О.В. Хоменко – Електронні текстові дані (1 файл: 153 кбайт). – Київ : КПІ ім. Ігоря Сікорського, 2017. – 24 с.

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № від р.)
за поданням Вченої ради факультету електроніки (протокол № 12/2017 від .12.2017 р.)*

Електронне мережне навчальне видання

«МІКРОПРОЦЕСОРНІ СИСТЕМИ»

КОМП'ЮТЕРНИЙ ПРАКТИКУМ

Укладачі: *Терещенко Тетяна Олександрівна, докт. техн. наук.
Хоменко Олександр Васильович*

Відповідальний редактор *Ямненко Ю. С., завідувач кафедри промислової електроніки, д-р
техн. наук, проф.*

Рецензенти: *Михайлов С.Р., доцент кафедри електронних приладів та
пристроїв, канд. техн. наук, доц.*

Навчальний посібник «Мікропроцесорні системи: комп'ютерний практикум» сприяє набуттю практичних навичок створення програмного забезпечення спеціалізованих та промислових мікропроцесорних систем та методів їх налагодження.

ЗМІСТ

| | |
|-----------------------------|----|
| ВСТУП | 4 |
| ЛАБОРАТОРНА РОБОТА №1 | 5 |
| ЛАБОРАТОРНА РОБОТА №2..... | 7 |
| ЛАБОРАТОРНА РОБОТА №3..... | 10 |
| ЛАБОРАТОРНА РОБОТА №4..... | 14 |
| ЛАБОРАТОРНА РОБОТА №5..... | 18 |
| ЛАБОРАТОРНА РОБОТА №6..... | 21 |
| ЛІТЕРАТУРА | 24 |

ВСТУП

Метою виконання циклу лабораторних робіт є:

- закріплення і експериментальна перевірка теоретичних положень найважливіших розділів і тем навчального матеріалу;
- оволодіння інтегрованими програмними комплексами відпрацювання прикладного програмного забезпечення, та допоміжними програмами, що спрощують цей процес;

Під час виконання робіт студенти відпрацьовують основні етапи створення та налагодження програмного забезпечення мікроконтролерних пристроїв.

ЛАБОРАТОРНА РОБОТА №1

Система переривань МП i8086. Визначення адреси програмної обробки переривань

Мета роботи: Визначити та обґрунтувати адресу переривання, задану викладачем

Порядок роботи:

1. Ознайомитись з теоретичними відомостями по системі переривань процесорів i80x86.
2. В програмному відладчику знайти карту векторів переривань
3. Знайти початкову адресу переривання
4. Визначити значення CS:IP

Зміст звіту

1. Програма з коментарями
2. Висновки

Контрольні питання:

1. Дайте визначення вектора переривань і карти векторів переривань.
2. Які дії виконує МП при переході на підпрограму обробки переривань?
3. Знайдіть початкову адресу переривання. номер якого вкаже викладач.

Теоретичні відомості

Для обробки подій, що відбуваються асинхронно по відношенню до виконання програми, найкраще підходить механізм переривань. Переривання можна розглядати як деякий особлива подія в системі, що вимагає моментальної реакції.

Процесор i8086 може обробляти до 256 типів переривань. Кожному перериванню відповідає свій вектор – подвійне слово, що містить адресу CS: IP підпрограми, що викликається. Під вектори переривань у загальному просторі адрес пам'яті відводиться 1 Кбайт, починаючи з нульової адреси (рис. 1)

При переході на підпрограму обробки переривань *INT n* (*n* – тип переривання) процесор переміщує у стек вміст регістрів *IP*, *CS*, регістр прапорців *F* і скидає прапорець дозволу переривання *IF*; обчислює адресу $4 \times n$ і перше слово за цією адресою переміщує у *IP*, друге – у *CS*. Послідовність цих дій еквівалентна командам:

PUSHF ; Запам'ятовування у стеку прапорців
CALL FAR i_proc_4n ; Далекий виклик підпрограми обробки переривання

Скидання прапорця переривання *IF* не дозволяє перервати виконання підпрограми обробки переривання до її завершення або виконання команди дозволу *STI*. Останньою командою підпрограми обробки переривання є команда ***IRET***. За цією командою процесор вибирає зі стека адресу повернення (адресу команди, наступної за командою *INT*) і вміст регістра прапорців.

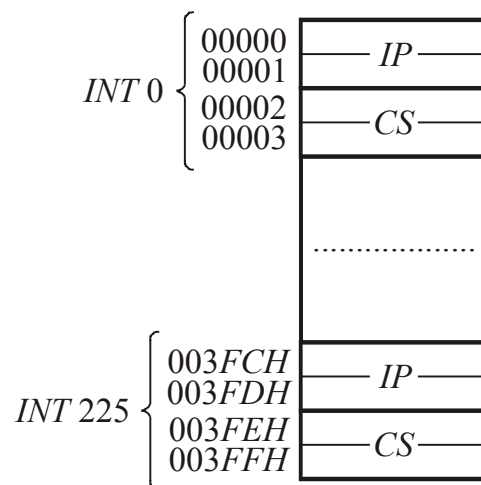


Рисунок 1 – Карта векторів переривань

ЛАБОРАТОРНА РОБОТА №2

Система переривань. Пошук програми обробки переривань INT8 в BIOS

Мета роботи: Визначити програму обробки переривань INT8 в BIOS

Порядок роботи:

1. Ознайомитись з теоретичними відомостями по системі переривань
2. У відладчику визвати програму заданого переривання
3. Прочитати асемблерний код

Зміст звіту

1. Програма з коментарями
2. Висновки

Контрольні питання

1. Які переривання відносяться до зовнішніх?
2. Які внутрішні переривання відносяться до програмних? які до апаратних
3. Поясніть дію переривання *INTO*.
4. В чому особливість переривання INT 3?

Теоретичні відомості

Типи переривань показано на рис. 2. Переривання поділяються на *зовнішні апаратні* і *внутрішні*. Запити *IRQ* і зовнішніх апаратних переривань надходять до системи переривань або на вивід немаскованого переривання *NMI* МП. Система переривання формує сигнал *INTR* маскованого переривання МП. Зазначимо, що масковане переривання відрізняється від немаскованого тим, що перше може бути заборонено програмно – командою скидання прапорця дозволу переривань *IF*. У цьому разі при надходженні запитів переривання вони будуть ігноруватися. Внутрішні переривання процесора поділяють на *програмні* й *апаратні*. Джерелами *внутрішніх програмних* переривань (рис. 2.19) є: помилка ділення (тип 0); покроковий режим (тип 1); команда *INTO* (тип 4).

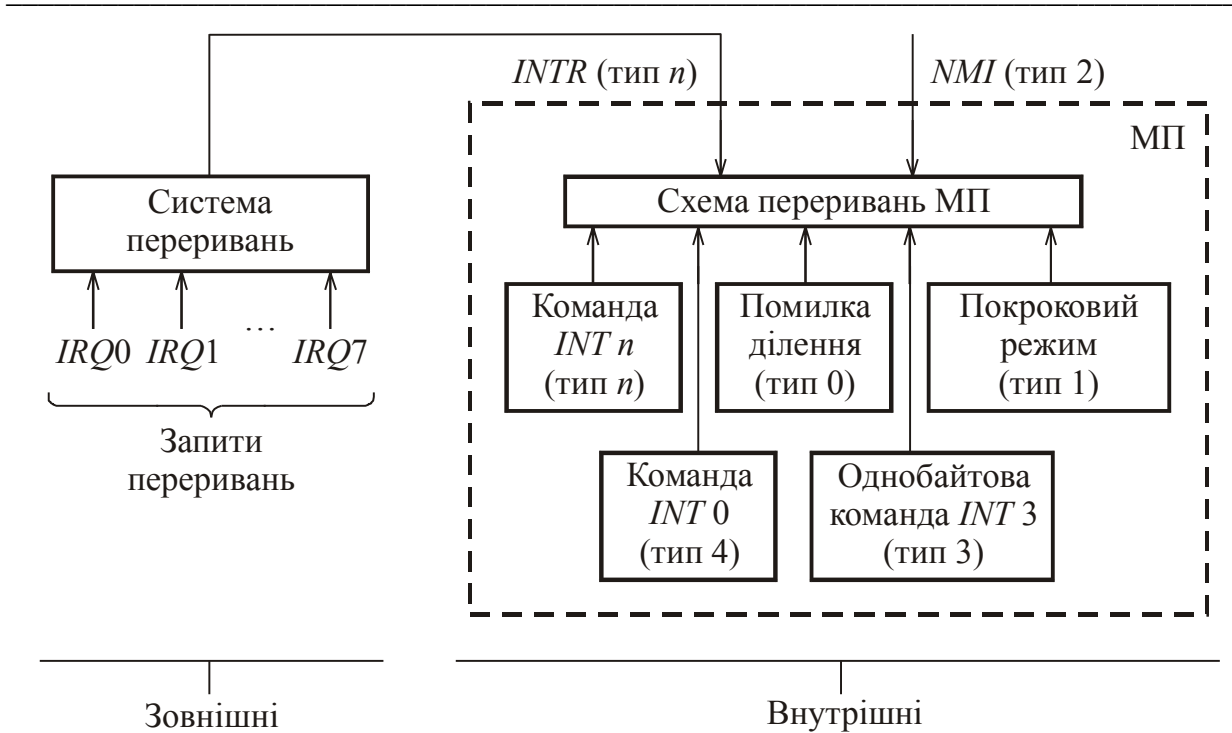


Рисунок 2 – Типи переривань

Внутрішні програмні переривання $INT\ n$ та $INT\ 3$ виконуються за командами переривання і дозволяють викликати підпрограми обробки переривань (наприклад, сервісні підпрограми BIOS і DOS) без застосування дальніх викликів. На відміну від $INT\ n$, переривання $INT\ 3$ є однобайтовою командою і зазвичай використовується для передачі керування підпрограми-налагоджувачу. Виконання програмних переривань не залежить від прапорця дозволу переривань IF .

Внутрішні апаратні переривання процесора виникають у таких особливих випадках:

- при діленні на нуль (тип 0);
- при встановленому прапорці трасування (тип 1). У цьому разі переривання відбувається після виконання кожної команди;
- після команди $INTO$ (тип 4), якщо встановлений прапорець переповнення OF .

Апаратні переривання виникають при активному рівні сигналів на контактах МП- NMI (немасковане переривання – тип 2) і $INTR$ (масковані, типи 5-255).

Масковані переривання виконуються при встановленому прапорці IF . При

переході до підпрограми обробки апаратного переривання процесор формує два цикли підтвердження переривання один за одним, в яких генерується сигнал \overline{INTA} . За другим імпульсом \overline{INTA} контролер переривань передає по шині даних номер вектора переривання n . Далі дії процесора аналогічні виконанню програмного переривання. Обробка поточного переривання може бути перервана немаскованим перериванням або іншим маскованим перериванням вищого пріоритету у тому разі, якщо підпрограма-обробник встановить прапорець дозволу переривання IF. Немасковане переривання виконується незалежно від стану прапорця IF.

ЛАБОРАТОРНА РОБОТА №3

Аналіз програми обробки переривань INT8. Структура системного модуля

Мета роботи: Проаналізувати дії програми INT8

Порядок роботи:

1. Ознайомитись з теоретичними відомостями по системі переривань
2. У відладчику визвати програму заданого переривання
3. Прочитати асемблерний код
4. За допомогою довідкової літератури, визначити, які функції виконує дана програма

Зміст звіту

1. Програма з коментарями
2. Висновки

Контрольні питання

1. Поясніть фрагмент підпрограми обробки переривань INT8 за вказівкою викладача
2. Які функції виконує дана програма?
3. Який пристрій формує запит на переривання INT8?

Теоретичні відомості

Структурна схема системного модуля

Системна плата першої моделі PC містила кілька функціональних вузлів, які завдяки відкритому опису придбали надійний статус недоторканності, гарантований незліченною кількістю програм і програмних продуктів, їх використовують. До таких вузлів відносяться наступні:

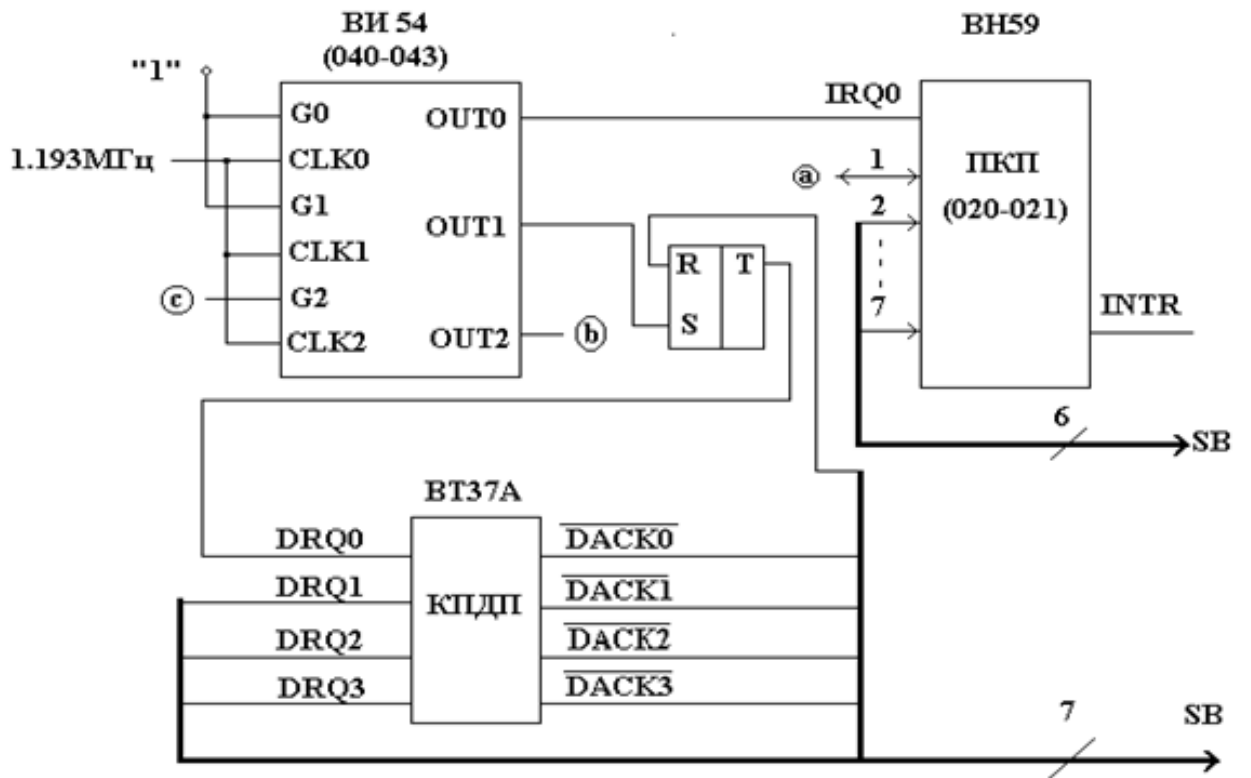
- **Системний таймер**, реалізований на мікросхемі +8253, що використовувався як генератор запитів регенерації пам'яті, інтервальний

таймер і тональний генератор для динаміка. У АТ ті ж функції виконувала аналогічна мікросхема 8254.

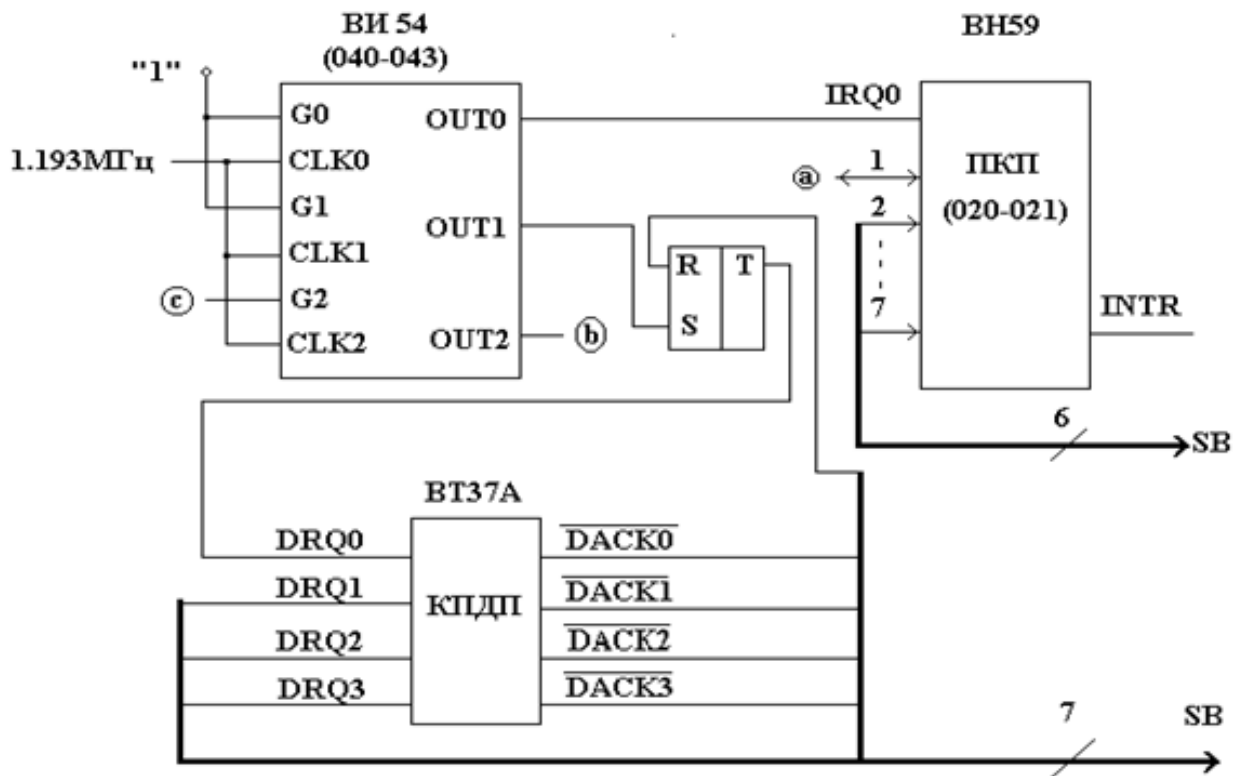
- **Три системних порту** на мікросхемі 8255, використовуваних для інтерфейсу клавіатури, читання перемикачів конфігурації, управління звуком і немаскованим перериванням.
- **Канал управління звуком** - логічна схема, що використовує тональний сигнал таймера і програмно-керовані біти системного порту.
- **Послідовний інтерфейс клавіатури,**
- **Пам'ять конфігурації і годинник-календар - CMOS RTC** - вузол, що з'явився з АТ.

З часом елементна база системної плати радикально змінилася, всі функції окремих контролерів взяв на себе чіпсет, але програмна модель цих вузлів збереглася.

Структурна схема системного модуля - блок таймера, програмовного контролера переривань (ПКП) і контролера прямого доступу до пам'яті (КПДП) показана на рис. 3 а ; паралельного порту ВВ55 – на рис.3б



а



б

Рисунок 3 - Структурна схема системного модуля

На рис показано одна ВІС контролера переривання, що було характерно для ХТ. В АТ використовується каскадне включення двох контролерів

ЛАБОРАТОРНА РОБОТА №4**Визначення періоду імпульсної послідовності IRQ0**

Мета роботи: Визначити період слідування імпульсів IRQ0

Порядок роботи:

1. Ознайомитись з теоретичними відомостями по системі переривань
2. У відладчику визвати програму заданого переривання
3. Теоретично обґрунтувати значення періоду імпульсів IRQ0
4. Експериментально визначити період імпульсів і порівняти з теоретичним значення
5. Визначити і обґрунтувати похибку визначення періоду

Зміст звіту

1. Програма з коментарями
2. Висновки

Контрольні запитання

1. Який канал системного таймера генерує запит переривання 0? Яке його призначення?
2. Чому не можна перепрограмувати канал 1 системного таймера?
3. Яким чином задається параметр часової функції таймера

Теоретичні відомості***Системний таймер***

Призначення лічильників-таймерів наступне:

- генерація переривань від системного годинника;
- генерація запитів на регенерацію пам'яті;
- генерація звукових сигналів.

Мікросхема таймера 8254 містить три програмованих 16 розрядних від'ємних лічильників-таймерів CT0-CT2, блок керування читанням/записом RWCU з регістром керуючого слова RCW; тристабільний буфер даних BD. (рис. 4)

На рис. 4 позначено CLK - вхід рахункових імпульсів і GATE - вхід дозволу рахунку і OUT - вихід, що змінює стан після закінчення рахунку

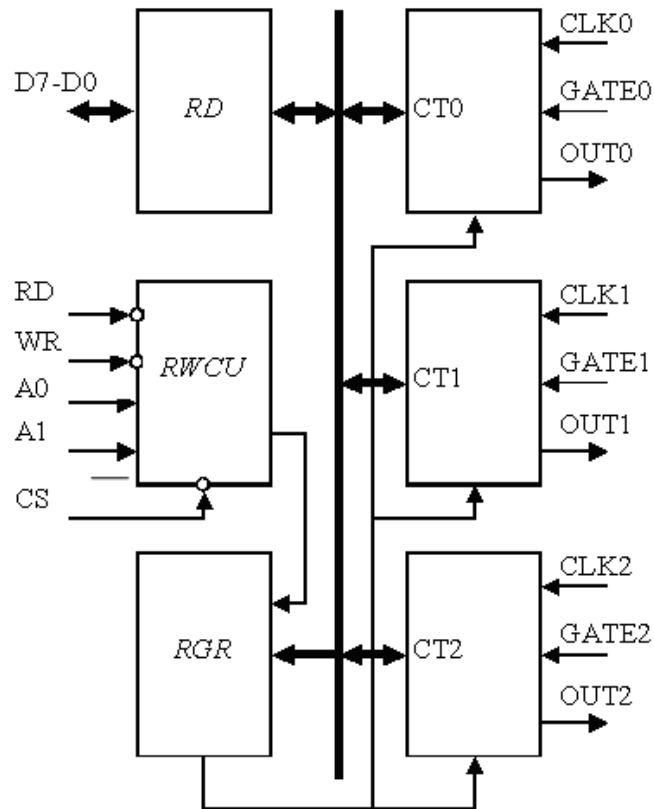


Рисунок 4 - Структурна схема таймера

Канал 0 генерує переривання по входу *IRQ 0* з найвищим пріоритетом. На виході *OUT0* формується стандартна $f = 18,2\text{Гц}$, а підпрограма обробки цього переривання *INT8* забезпечує відлік астрономічного часу. При необхідності створення нових часових залежностей *CT0* можна перепрограмувати, але при цьому функції реального годинника будуть втрачені. Підпрограма обслуговування цього переривання займає певний час. В цьому інтервалі часу не контролюється зовнішнє середовище (тобто не контролюється така тривалість імпульсів).

Якщо це треба зробити, то треба замаскувати переривання системного годинника.

СТ1 призначений для регенерації динамічної пам'яті ПК. Після скидання нульовий канал ПДП встановлюється на виконання одиночних циклів ПДП з автоінкрементацією адреси, тобто виконує регенерацію динамічної пам'яті. Період імпульсів СТ1 визначає швидкість автоінкрементації адреси динамічного ОЗП. Перепрограмувати СТ1 не має сенсу, а випадкове звертання до нього призводить до втрати пам'яті.

СТ2 – призначений для генерації звукових сигналів і за винятком піка при запуску операційної системи доступний для користувача. Керується як програмною зміною коефіцієнта ділення, так і по входу G2 з боку системного порту (розряд RB0).

Призначення регістрів наведено в табл. 1

Таблиця 1

| Порт, R/W | Призначення | Параметри |
|--------------|---|---|
| 040 RW | Лічильник 0 — системний годинник | Генерація IRQO кожні 54,936 мс — частота 18,206 Гц. Режим 011, LSB/MSB, Binary, константа лічильника =0 (відповідає коефіцієнту ділення 65536) |
| 041 RW | Лічильник 1 — регенерація пам'яті | DRQO для XT, логіка регенерації — для AT. Режим 010, LSB, Binary, константа лічильника -12h (18) |
| 042 RW | Лічильник 2 — генератор звука. | Вхід GATE від біта 0 порту B 8255 (061). Режим 011, LSB/MSB, Binary, значення лічильника визначає висоту тона |
| 043 RW | Керуючий регістр | Біти 7, 6 — вибір лічильника 0, 1, 2. Біти 5, 4 — режим звернення 00 — фіксація поточного значення, |

| | | |
|--|--|--|
| | | <p>01 — LSB — тільки молодший байт,</p> <p>10 — MSB — тільки старший байт,</p> <p>11 — LSB/MSB — спочатку молодший, потім старший Біти 3-1 — режим лічильника:</p> <p>000 — переривання по лічильнику,</p> <p>001 — одновібратор,</p> <p>x10 - генератор коротких імпульсів заданої частоти,</p> <p>x11 - генератор меандру,</p> <p>100 - Лічильник подій з дозволом,</p> <p>101 - Лічильник подій з перезапуском.</p> <p>Біт 0 - режим рахунк 0 = Binary, 1 = BCD</p> |
|--|--|--|

ЛАБОРАТОРНА РОБОТА №5

Маскування і демаскування переривання IRQ0

Мета роботи: Написати програму для маскування і демаскування переривання IRQ0

Порядок роботи:

1. Ознайомитись з теоретичними відомостями по системі переривань
2. У відладчику визвати програму заданого переривання
3. Написати програму для маскування переривання IRQ0
4. Впевнитися, що комп'ютер не реагує на переривання
5. Написати програму для демаскування переривання IRQ0
6. Впевнитися, що комп'ютер реагує на переривання

Зміст звіту

1. Програма з коментарями
2. Висновки

Контрольні питання

1. Які функції виконує контролер переривань?
2. Напишіть підпрограму маскування/демаскування переривання, номер якого задає викладач
3. Поясніть, чому переривання IRQ0 викликає підпрограму INT 8

Теоретичні відомості

Система переривань IBM Структурна схема системи переривань ПК являє собою каскадне включення двох контролерів переривань ПКП1 та ПКП2 при чому ПКП1 являється ведучим, а ПКП2 веденим. Режим ведучого/веденого задається апаратно шляхом подачі логічної одиниці на вхід SP (0 для веденого). Ведучий контролер формує сигнал переривання INT на мікропроцесор (рис. 5)

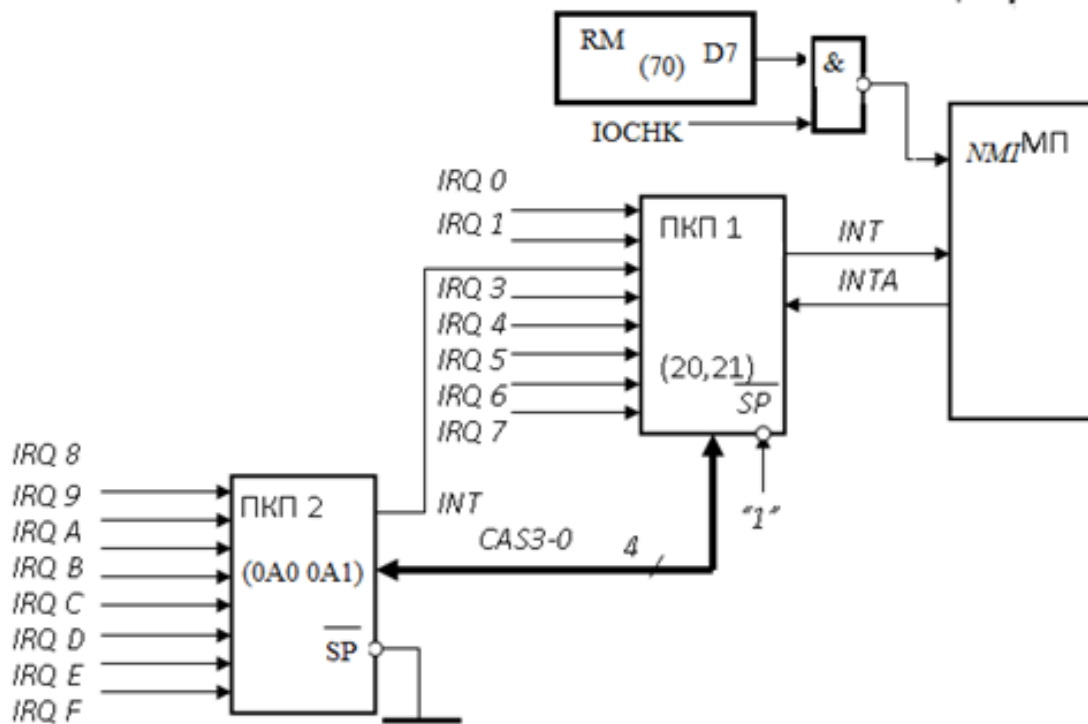


Рисунок 5 - Структурна схема системи переривань ПК

Немасковане переривання NMI обслуговує в IBM PC сигнал помилки каналу введення/виведення ІОСНК

Дозвіл сигналу реагування на сигнал помилки NMI полягає в записі «1» у розряд D7 порту RM з адресою 70H.

MOU AL, 80H ; записати «1» (D7)

OUT [70H], AL ; у регістр маски NMI.

Всі інші зовнішні переривання маскуються через ПКП

Апаратні переривань, розташованих в порядку убунання пріоритету наведено в табл. 2

Таблиця 2

| Номер | Описание |
|-------|--|
| 8 | IRQ0 - переривання інтервального таймера, виникає 18,2 рази на секунду |
| 9 | IRQ1 - переривання від клавіатури |
| A | IRQ2 - використовується для каскадування апаратних перериваь |
| 70 | IRQ8 - переривання від годинника реального часу |
| 71 | IRQ9 - переривання від контролера EGA |
| 72 | IRQ10 - зарезервовано |
| 73 | IRQ11 - зарезервовано |
| 74 | IRQ12 - зарезервовано |
| 75 | IRQ13 - переривання від арифметичного сопроцесора |
| 76 | IRQ14 - переривання від контролера жорсткого диска |
| 77 | IRQ15 - зарезервовано |
| B | IRQ3 - переривання асинхронного порту COM2 |
| C | IRQ4 - переривання асинхронного порту COM1 |
| D | IRQ5 - переривання від контролера жорсткого диска (тільки в комп'ютерах IBM PC / XT) |
| E | IRQ6 - переривання від контролера НГМД |
| F | IRQ7 - переривання принтера |

ЛАБОРАТОРНА РОБОТА №6

Дослідження системи переривань та генерації звуку за допомогою таймера

Мета роботи: Написати програму, яка генерує звук на динаміці. Загальна тривалість звучання – t_1 с. Тривалість тиші – t_2 с. Частота звуку - f Гц.

Порядок роботи:

Для виконання цього завдання необхідно написати програму і довантажити її у вільну область пам'яті. Після чого замінити адресу переривання int 1с на адресу, в який була завантажена програма. Зміна адреси повинно відбуватися в момент, коли переривання замасковані, тому в іншому випадку система не дозволить цього зробити

Контрольні запитання

1. Як заборонити надходження імпульсів на динамік за допомогою системного порту?
2. Як заборонити надходження імпульсів на динамік за допомогою системного таймера?

Системний порт

ВІС КР580ВВ55 призначена для організації введення/виведення паралельної інформації у 8-байтовому форматі і дозволяє реалізувати більшість відомих протоколів обміну по паралельних каналах.

До складу ВІС входять: двонапрямлений 8-розрядний буфер даних Bufer of Data (BD), що пов'язує лінії даних ВІС із системною шиною даних; блок керування читанням/записом Read/Write Control Unit (RWCU), що забезпечує керування зовнішнім і внутрішнім передаванням даних та керуючих слів; три 8-розрядних порти введення/виведення (Port A, Port B, Port C) для обміну інформацією, рис. 6.

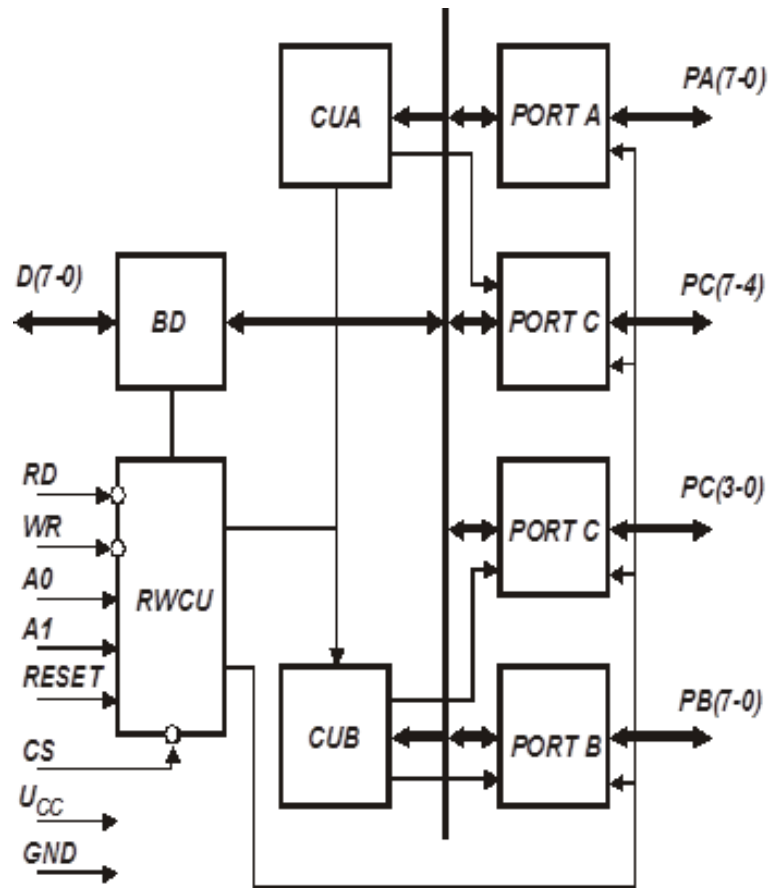


Рисунок 6 – Структурная схема VIC

Порт С поділений на два 4-розрядних: С' (PC7-PC4) і С''(PC3-PC0). Порти А і С' об'єднані у групу В, порти В і С'' - у групу В. Блоки керування групою А Control Unit A (CUA) та групою В (CUB) виробляють сигнали керування для відповідних груп. Блок RWCU (Register of Control Word Unit) містить реєстр керуючого слова, який зберігає керуючі слова, що надходять від МПІ. Призначення програмованого інтерфейсу введення / виводу і8255 в персональному комп'ютері наступне:

- читання скан-кодів клавіатури (через порт А надходять дані; порт PB6 – дозвіл синхронізації; PB7 – сигнал сканування / скидання клавіатури);
- управління звуком - лінії PB0, PB1; контроль виходу СТ2 таймера - PC5;
- дозвіл та ідентифікація джерел NMI;

-
- читання байта конфігурації.
 - Дозвіл контролю помилки каналу введення-виведення PB5 (EN IO СНК) і власне, її контроль PC6 (IO ERROR).

Користувач може використовувати три лінії системного порту PB0, PB1, PC5 (лічильник 2 таймера (PC5), і генерацію звуку PB0, PB1).

ЛІТЕРАТУРА

1. Дистанційний курс Спеціалізовані та промислові мікропроцесорні системи; Сертифікат УЦДО від 15.05.2012; № НМП №2536 - режим доступу до ресурсу:
<http://moodle.udec.ntu-kpi.kiev.ua/moodle/course/view.php?id=309>
2. Жуйков В.Я., Терещенко Т.О., Петергеря Ю.С. Електронний підручник «Мікропроцесори і мікроконтролери» - 2009 Гриф надано Міністерством освіти і науки України (лист № 1.4_18-Г-114 від 10.01.2009 р. - режим доступу до ресурсу: <http://www.kaf-pe.ntu-kpi.kiev.ua>
3. Мікропроцесорна техніка. Друге видання. Доповнене./ Ю.І. Якименко, Т.О. Терещенко, Є.І. Сокол, В.Я. Жуйков, Ю.С. Петергеря. За ред. Т.О. Терещенко. – Київ, 2004. – 440 с
4. Жуйков В.Я., Терещенко Т.О., Ямненко Ю.С. Заграничний А.В. Електронний підручник "Мікропроцесорна техніка". - Рекомендовано до друку Вченою Радою НТУУ «КПІ» протокол №6 від 16.05.2016 р. режим доступу до ресурсу: http://kaf-pe.kpi.ua/?page_id=675,
<http://ela.kpi.ua/handle/123456789/18969>
5. Мікропроцесорна техніка : підручник / В. Я. Жуйков, Т. О. Терещенко, Ю. С. Ямненко – 3-тє вид., перероб. і допов. – Київ: НТУУ «КПІ» Вид-во «Політехніка», 2015. – 440
6. Мікропроцесорна техніка: Навчальний посібник / В.Я. Жуйков, О.І. Захожай, Ю.Е. Паеранд, Т. О. Терещенко Алчевськ: ДонДГУ, 2013 – 497 с.
7. Абель П. Язык Ассемблера для IBM PC и программирование / Пер. с англ. Ю.В. Сальникова. – М.: Высш.шк., 1992. – 447 с.