Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Архітектура комп'ютерів-2. Процесори

Домашня

Модульна контрольна робота

Виконала

студентка групи ІВ-71

Молчанова В.С.

Залікова книжка №7110

Перевірила

доцент

Ткаченко В.В.

Київ – 2019

Зміст

[Завдання 1 3](#_Toc27529617)

[1. Опис мікропроцесорної системи 3](#_Toc27529618)

[2. Карта розподілу пам’яті даних 5](#_Toc27529619)

[Завдання 2 6](#_Toc27529620)

[1. Задача №69 6](#_Toc27529621)

[2. Задача №228 8](#_Toc27529622)

[Висновки 8](#_Toc27529623)

# Завдання 1

## 1. Опис мікропроцесорної системи

На схемі *ІАЛЦ 462600 01.Е1* зображена мікропроцесорна система на основі мікроконтролера МК51, який складається з:

1. Резидентної пам’яті програм (РПП) об’ємом *4Кб*. З адреси 0000h розпочинає виконуватися програма під час системного скидання. Інші призначені для зберігання початкових адрес підпрограм обслуговування переривань від зовнішніх сигналів, таймерів-лічильників або послідовного інтерфейсу.
2. Резидентної пам’яті даних (РПД) об’ємом *128б*, яка поділяється на
   1. Оперативний запам’ятовуючий пристрій (ОЗП), що містить чотири банки регістрів загального призначення, до регістрів яких команди можуть звертатися за їх іменами *R7* – *R0* та шістнадцять байт (*20h* – *2Fh*), що утворюють область комірок, до яких можливе застосування прямої побітової адресації.
   2. Регістри спеціальних функцій *SFR*, до складу якого входять
      1. Восьмирозрядний акумулятор *ACC*, який використовується як допоміжний регістр під час виконання операцій
      2. Регістр-розширювач аккумулятора *B*
      3. Регістр слова стану програми *PSW*
      4. Покажчик стека *SP*
      5. Регістр-покажчик даних DPTR
      6. Регістри таймерів-лічильників
      7. Буфер прийомопередавача
      8. Регістри управління
3. Арифметико-логічного пристрою (АЛП), завдяки якому МК виконує арифметичні та логічні дії
4. Блока таймерів-лічильників
5. Пристрою управління та синхронізації
6. Блока переривань і послідовного порту
7. Чотирьох портів вводу-виводу (*P0, P1, P2, P3*).

До МК підключено:

1. *16* сторінок **зовнішньої пам’яті даних** (ЗПД) об’ємом *64Кб* . Кожна сторінка пам’яті даних обирається через порт *P1* безпосередньо через вихід порту (або через дешифратор). Першасторінка використовується для зберігання адрес *56* зовнішніх пристроїв. Зовнішні пристрої належать загальному адресному простору пам’яті даних.
2. *6* сторінок **зовнішньої пам’яті програм** (ЗПП) об’ємом *16Кб*.
3. **Контролер пріоритетних переривань** (КПП) – реалізує зовнішні векторні переривання. Коли зовнішній пристрій став готовим до взаємодії, він надсилає на контролер запит на переривання *IRQ*. По цих запитах КПП видає на процесор сигнал вимоги переривання *INT*. Після його отримання процесор перериває виконання поточної програми наступним чином:
   1. Завершує виконання поточної команди.
   2. Зберігає у стеку стан програми, що переривається, та адресу повернення до неї.
   3. Видає на КПП сигнал підтвердження переривання *IACK*
   4. Зчитує вектор, що виставляє на шину даних КПП, по ньому обчислює адресу переходу до підпрограми обробки переривання та переходить на її першу команду.
   5. Виконує підпрограму, остання команда якої відновлює стан перерваної основної програми за інформацією, збереженою в стеці.
4. **Контролер прямого доступу до пам’яті даних** (КПДП) – реалізує режим прямого доступу до пам’яті, який допомагає прискорити обмін даними між елементами системи, адже під час роботи КПДП може паралельно виконувати завдання, що не вимагають доступу до пам’яті. Відбувається це наступним чином:
   1. КПДП надсилає сигнал вимоги захоплення шини (*HRQ -* Hold Request)
   2. Процесор отримує цей сигнал, виконує попередню ініціалізацію контролера для чого пересилає по шині даних в контролер прямого доступу інформацію, необхідну для управління обміном (адресу комірки пам’яті, в якій розміщується перший байт даних, що записуються або зчитуються, адресу порту, загальну кількість даних, що передаються, напрям передачі та інше) та відключається від шини даних і шини адреси, надаючи їх контролеру для організації обміну за допомогою сигналу підтвердження захоплення шини (*HLDA* – Hold Acknowledge).
   3. КПДП виконує обмін даними між ЗП та СП, який координується за допомогою сигналів вимоги ПДП (*DRQ* – DMA Request) та підтвердження ПДП (*DACK* – DMA Acknowledge).
   4. Процесор отримує сигнал від контролера і переходить до виконання основної програми.
5. **Програмований периферійний адаптер** (ППА) - забезпечує ввід/вивід за трьома додатковими восьмирозрядними портами РА, РВ, РС. по шині даних відбувається не тільки обмін даними, але і пересилання з МК в ППА управляючих слів, генерованих програмним забезпеченням процесора, а також передача в МК інформації про стан периферійного обладнання. Налаштування внутрішньої організації ППА – тобто підключення його до конкретного обладнання виконується за допомогою управляючого слова режиму роботи (УСРР), що міститься у регістрі управляючого слова (РУС)
6. **Селектори адрес** (СА), що за адресою, виставленою на шині визначають, до якого з елементів системи треба підключитись.

## 2. Карта розподілу пам’яті даних

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | | | | ЗПД0 | |
| FFFFh | 1111 | 1111 | 1111 | 1111 |  | |
| -------- | ------ | ------ | ------ | ------ |
| F7E1h | 1111 | 0111 | 1110 | 0001 | КПДП | |
| F7E0h | 1111 | 0111 | 1110 | 0000 |
| F7DFh | 1111 | 0111 | 1101 | 1111 | КПП | |
| F7DEh | 1111 | 0111 | 1101 | 1110 |
| F7DDh | 1111 | 0111 | 1101 | 110**1** | ЗП55 (РД) | Зовнішні пристрої |
| F7DCh | 1111 | 0111 | 1101 | 110**0** | ЗП55 (РС) |
| F7DBh | 1111 | 0111 | 1101 | 101**1** | ЗП54 (РД) |
| F7DAh | 1111 | 0111 | 1101 | 101**0** | ЗП54 (РС) |
| -------- | ------ | ------ | ------ | ------ |  |
| F775h | 1111 | 0111 | 0111 | 010**1** | ЗП3 (РД) |
| F774h | 1111 | 0111 | 0111 | 010**0** | ЗП3 (РС) |
| F773h | 1111 | 0111 | 0111 | 001**1** | ЗП2 (РД) |
| F772h | 1111 | 0111 | 0111 | 001**0** | ЗП2 (РС) |
| F771h | 1111 | 0111 | 0111 | 000**1** | ЗП1 (РД) |
| F770h | 1111 | 0111 | 0111 | 000**0** | ЗП1 (РС) |
| F76Fh | 1111 | 0111 | 0110 | 111**1** | ЗП0 (РД) |
| F76Eh | 1111 | 0111 | 0110 | 111**0** | ЗП0 (РС) |
| -------- | ------ | ------ | ------ | ------ |  | |
| 0023h | 0000 | 0000 | 0010 | 00**11** | РУС | ППА |
| 0022h | 0000 | 0000 | 0010 | 00**10** | РС |
| 0021h | 0000 | 0000 | 0010 | 00**01** | РВ |
| 0020h | 0000 | 0000 | 0010 | 00**00** | РА |
| -------- | ------ | ------ | ------ | ------ |  | |
| 0000h | 0000 | 0000 | 0000 | 000 |  | |

# Завдання 2

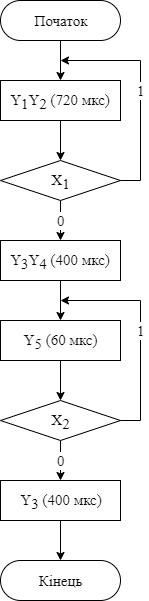
## 1. Задача №69

Умова: Для МК 1816ВЕ51 розробити програму реалізації алгоритму керування Н ↓1 (У1 У2 ) Х1↑1 ( У3 У4)↓2 У5 Х2 ↑2 У3 ДО.

Прийняти: TУ1 = TУ2 > 720 мкс; TУ5 > 60 мкс; TУ3 = TУ4 > 400 мкс.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Розряд порту | *P1*[7] | *P1*[6] | *P1*[5] | *P1*[4] | *P1*[3] | *P1*[2] | *P1*[1] | *P1*[0] |
| Сигнал | Х1 | Х2 | 0 | Y5 | Y4 | Y3 | Y2 | Y1 |

*Табл.1 Відповідність виходів порту та сигналів*



*Рис. 1 Графічне зображення алгоритму*

Код програми:

;Налаштування порту P2 (х1, х2)

    ANL P2 #C0h

    ORL P2, #07h

; M1,M0 := 00 , C/T := 0, GATE := 0 – режим 0

    ANL 89h, #0000 1111h

    MOV TMOD, #00h  ; Встановлення режиму 0 для таймера

BLOCK1:

; Формування затримки 720 мкс

    ANL 8Dh, #00h   ; Обнуління молодших розрядів таймера TL1

    MOV 8Bh, #EAh   ; 720/32=22.5~22, (-22)ДК= 11101010b = EAh

    ANL P2, #C3h    ; установка Y1, Y2 на порт Р2

    SETB TR1        ; Запуск таймера

    CYCLE\_720:      ; Цикл відліку затримки 720 мкс

        JNB TF1, CYCLE\_62   ; Перехід після переповнення таймера

; Перевірка умови Х1

checkX1:

    JB P2.6, BLOCK1 ; Перевірка шостого біта порту, що відповідає за умову Х1

BLOCK2:

; Формування затримки 400 мкс

    ANL 8Dh, #00h   ; Обнуління молодших розрядів таймера TL1

    MOV 8Bh, #F4h   ; 400/32=12.5~12, (-12)ДК= 11110100b = F4h

    ANL P2, #CCh    ; установка Y1, Y2 на порт Р2

    SETB TR1        ; Запуск таймера

    CYCLE\_400:      ; Цикл відліку затримки 720 мкс

        JNB TF1, CYCLE\_400  ; Перехід після переповнення таймера

BLOCK3:

; Формування затримки 60 мкс

    ANL 8Dh, #00h   ; Обнуління молодших розрядів таймера TL1

    MOV 8Bh, #FEh   ; 60/32=1.875~2, (-2)ДК= 11111110b = FEh

    ANL P2, #D0h    ; установка Y5 на порт Р2

    SETB TR1        ; Запуск таймера

    CYCLE\_60:       ; Цикл відліку затримки 720 мкс

        JNB TF1, CYCLE\_60   ; Перехід після переповнення таймера

; Перевірка умови Х2

checkX2:

    JB P2.7, BLOCK3 ; Перевірка сьомого біта порту, що відповідає за умову Х2

BLOCK4:

; Формування затримки 400 мкс

    ANL 8Dh, #00h   ; Обнуління молодших розрядів таймера TL1

    MOV 8Bh, #F4h   ; 400/32=12.5~12, (-12)ДК= 11110100b = F4h

    ANL P2, #C4h    ; установка Y3

    SETB TR1        ; Запуск таймера

    CYCLE\_400\_2:        ; Цикл відліку затримки 720 мкс

        JNB TF1, CYCLE\_400\_2    ; Перехід після переповнення таймера

END

## 2. Задача №228

Умова: Розробити для МК51 операційну схему й програму виконання зсувів 64-розрядних слів на шість розрядів вправо (слова розміщені в РПД, починаючи з комірки *4Fh*, зсуви робити в банку регістрів 0).



*Рис. 2*

;Вибір БР0

ANL PSW #11100111b

SET B PSW.3

MOV R7, #6  ; Внесення кількості зсувів

MOV R6, #8  ; Внесення кількості байт (64/8=8)

LABL1:

    MOV R0, #4Fh    ; Внесення адреси початку слова

    CLR C           ; Очищення ознаки переносу

    LABL2:

        MOV A, @R0

        RRC A

        MOV @R0, A

        INC R0

        DJNZ R6 LABL2

    DJNZ R7 LABL1

END

# Висновки

У даній роботі було побудовано мікропроцесорну систему на основі мікроконтролера МК51, зображену на схемі *ІАЛЦ 462600 01.Е1* та було розв’язано 2 задачі, пов’язані з використанням моделі програміста та певних елементів МПС.