

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

МЕТОДИЧНІ ВКАЗІВКИ
до виконання лабораторних робіт
з навчальної дисципліни
«Програмування мікропроцесорів»

для студентів денної та заочної форм навчання
за спеціальністю «Комп'ютерна інженерія»

Затверджено
Редакційно-видавничою
радою НТУ «ХПІ»,
протокол № 2 від 25.06.2020 р.

Харків
НТУ «ХПІ»
2020

Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни «Програмування мікропроцесорів» для студентів денної та заочної форм навчання за спеціальністю «Комп'ютерна інженерія» / уклад.: Подорожняк А. О., Межерицький С. Г., Гейко Г. В. – Харків: НТУ «ХПІ». – 2020. – 36 с.

Укладачі: А. О. Подорожняк, С. Г. Межерицький, Г. В. Гейко

Рецензент доц. О. Ф. Даниленко

Кафедра обчислювальної техніки та програмування

ВСТУП

Лабораторний практичний цикл «Програмування мікропроцесорів» є актуальним для галузі комп'ютерної інженерії та необхідний для набуття практичних навичок в процесі самостійного вирішення завдань, в яких використовується мікропроцесорна техніка.

Виконання лабораторних робіт передбачає наявність у студентів знань за навчальними дисциплінами: «Основи програмування», «Системне програмування», «Архітектура комп'ютерів», «Комп'ютерна електроніка», «Комп'ютерна схемотехніка». Методичні вказівки містять методику виконання лабораторних робіт, метою яких є отримання студентами спеціальних навичок з основ побудови, програмування та режимів функціонування мікропроцесорів та мікропроцесорних засобів.

Підготовка до лабораторних робіт, в результаті якої проводиться складання попередніх варіантів програмних лістингів, є одним з основних етапів самостійної роботи студентів.

Кожна лабораторна робота містить коротку теоретичну довідку, індивідуальні завдання та приклади програм.

Лабораторні роботи призначені для студентів спеціальності «Комп'ютерна інженерія», але можуть бути корисні для студентів інших спеціальностей та викладачів.

Лабораторна робота 1

ДОСЛІДЖЕННЯ СТРУКТУРИ ТА ФУНКЦІОНУВАННЯ ПРОГРАМОВАНОГО ПАРАЛЕЛЬНОГО ІНТЕРФЕЙСУ МІКРОПРОЦЕСОРНОЇ ЛАБОРАТОРІЇ «МІКРОЛАБ»

Мета роботи: ознайомитись зі структурою, складом та особливістю функціонування мікропроцесорної лабораторії (МПЛ) «Мікролаб» та емулятора її роботи, практично засвоїти режими роботи емулятора. Дослідити методи підключення та організації обміну інформацією з найпростішими пристроями вводу-виводу. Вивчити організацію і функціонування на базі мікропроцесорного комплексу (МПК) K580 організацію апаратних і програмних засобів організації вводу-виводу в мікро-ЕОМ та програмування програмованого паралельного інтерфейсу KP580BB55.

1. Структура та особливості функціонування мікропроцесорної лабораторії «Мікролаб»

На рис. 1.1 показана спрощена схема мікропроцесорної системи (МПС) на базі МПЛ «Мікролаб KP580ИК80». Зовнішній вигляд мікропроцесорної лабораторії «Мікролаб KP580ИК80» показано на рис. 1.2.

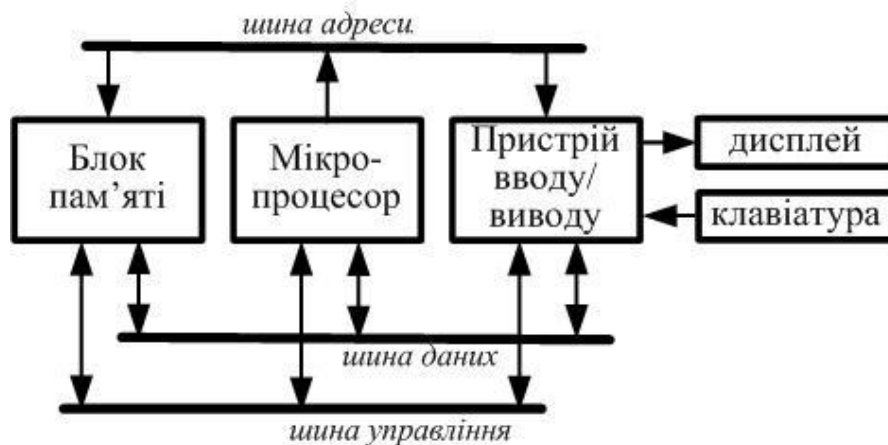


Рисунок 1.1 – Спрощена структурна схема МПС



Рисунок 1.2 – Зовнішній вигляд «Мікролаб KP580ИК80»

МПЛ складається з мікропроцесора (МП) КР580, постійного та оперативного запам'ятовуючого пристроїв (ПЗП та ОЗП) ємністю по 1Кб кожний. У МПЛ є клавіатура, дисплей, три тумблера, які зв'язані з МП по принципу вхід-вихід. МПЛ – це мікро-ЕОМ, яка розроблена спеціально для цілей навчання. Зберігання даних в пам'яті, перевірка змісту пам'яті та використання програм – це основні операції, які виконуються МПЛ. Програми можуть виконуватися безперервно або в покроковому режимі, що дозволяє прослідкувати їх роботу детально.

На рис. 1.3 приведена спрощена структурна схема мікропроцесора КР580ИК80. Арифметико-логічний пристрій (АЛП) виконує всі дії з даними, наприклад, збільшення числа або складання двох чисел. Регістр коду операції, дешифратор коду операції, програмований лічильник та пристрій управління і синхронізації використовуються для вибору команд з пам'яті та їх виконання.

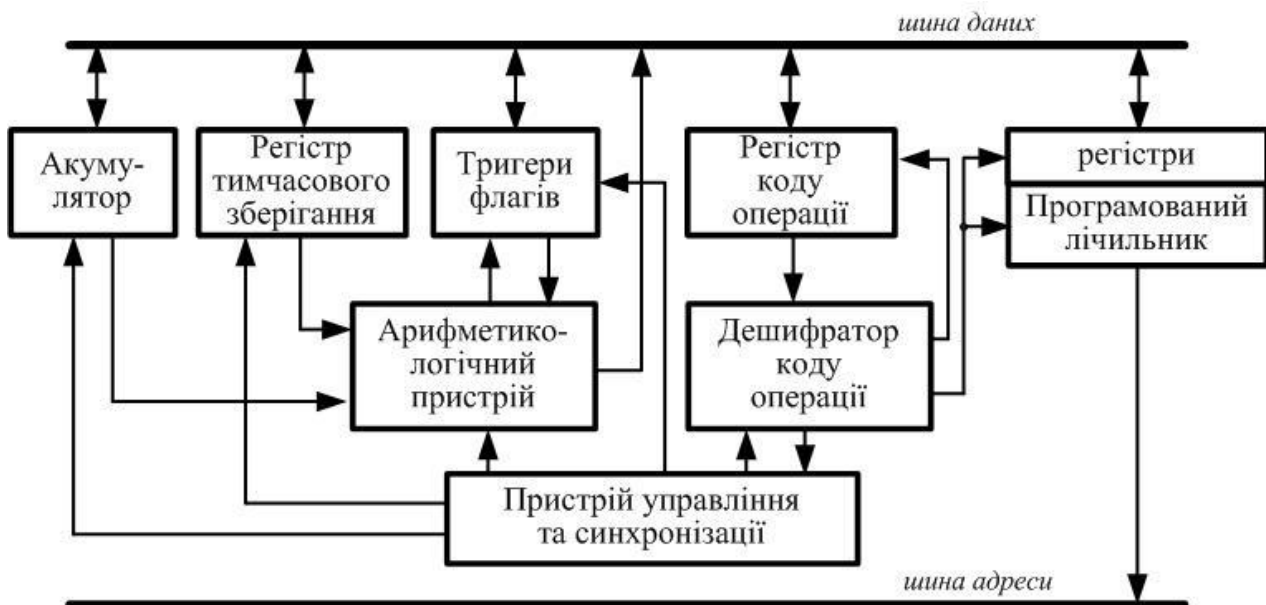


Рисунок 1.3 – Спрощена структурна схема МП КР580ИК80

Робота з емулятором МПЛ

Після завантаження програмного додатку, який являє собою емулятор для роботи з МПЛ, відкривається робоче вікно (рис. 1.4). У лівій частині знаходиться вікно ОЗП емулятора, в правому нижньому куті знаходиться клавіатура для запису команд в адреси комірок пам'яті, VD0...VD7 – світлодіоди, K1...K3 – тумблери, які регулюють роботу світлодіодів.

Для того, щоб записати команду в комірку пам'яті треба:

- 1) натиснути кнопку «Сброс» (ця команда скине програму на початкову адресу 8000);
- 2) натиснути кнопку «Зап» для початку запису;
- 3) на клавіатурі набираємо код, який потрібно записати до комірки пам'яті;
- 4) щоб перейти на наступну адресу треба натиснути кнопку «АД+»;
- 5) коли всі команди будуть записані, натискаємо ще раз кнопку «Сброс»;
- 6) натискаємо кнопку «Пуск».

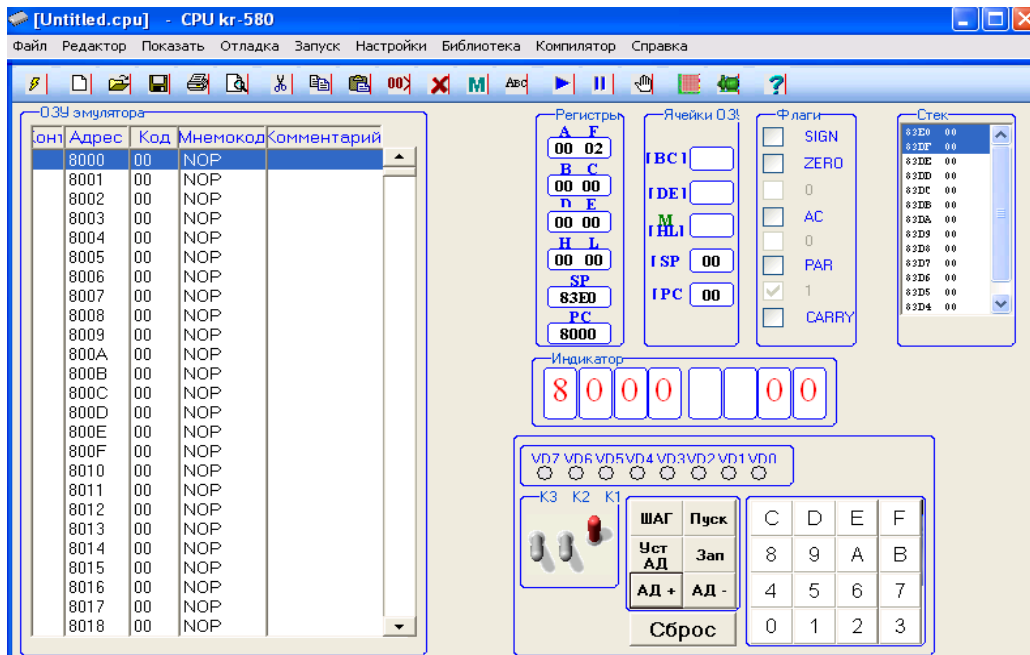


Рисунок 1.4 – Рабочее окно эмулятора

Загальні відомості про організацію вводу-виводу в МПС

При проектуванні МПС найважливіше значення має правильний вибір методів керування вводом-виводом даних. Розрізняють наступні методи організації вводу-виводу:

- програмний обмін даними по командах у режимі чекання готовності пристрою (рис. 1.5);
- обмін даними по командах умовного переходу;
- обмін даними по сигналах переривання роботи МП від зовнішніх пристроїв;
- обмін даними в режимі прямого доступу до пам'яті.



Рисунок 1.5 – Фрагмент схеми алгоритма програмного обміну даними

Ввід та вивід даних може здійснюватися двома способами. При першому – застосовуються спеціальні команди IN (ввід) та OUT (вивід), що забезпечують передачу даних між акумуляторами AL або AX (для K580BM80 – акумулятор A) та адресуючими портами. При виконанні цих команд генерується сигнал M/IO=0, який ідентифікує вибір простору вводу-виводу та у сукупності із сигналами WR і RD дозволяє сформувати системні сигнали IOW та IOR, для керування операціями запису даних у порт та читання даних з нього. Команди IN та OUT можуть використовувати пряму адресацію (BM80 та BM86), коли адреса порта міститься в другому байті команди у виді константи, і непрямую (BM86), коли адреса розташовується в регістрі DX. У першому випадку можна адресувати по 256 портів для вводу і виводу даних. В другому випадку забезпечується адресний простір до 64 восьмибітових, чи адресний простір до 32 шістнадцятибітових портів. При другому способі адреси портів розміщуються в загальному адресному просторі, і звертання до них не відрізняється від звертання до комірок пам'яті. Команди із звертанням до пам'яті мають більший формат і виконуються довше, ніж прості команди IN та OUT. При цьому ускладнюється дешифрування 20-бітової фізичної адреси (BM86) порта і зберігається кількість адрес, що можуть використовуватися для комірок пам'яті. Позитивним моментом є те, що підвищується гнучкість програмування, тому що для вводу-виводу можна використовувати будь-яку команду із звертанням до пам'яті. Як найпростіші пристрої вводу-виводу можуть використовуватися 8-розрядні регістри (наприклад, багаторежимний буферний регістр (ББР) K589IP12). Обмін даними між мікро-ЕОМ і зовнішніми пристроями може викликатися як у визначених місцях у програмі, так і по сигналах переривання. В останньому випадку підпрограма обміну з зовнішнім пристроєм буде викликатися за рахунок переводу мікро-ЕОМ у режим обслуговування переривання. Як пристрої вводу-виводу можуть застосовуватися і більш складні схеми, наприклад програмувальні рівнобіжні інтерфейси.

Програмувальний паралельний інтерфейс KP580BV55

До складу програмувального паралельного інтерфейсу входять:

- двонаправлений 8-розрядний буфер даних (BD), що зв'язує програмувальний паралельний інтерфейс (ППІ) із системною шиною даних;
- блок керування записом/читанням (RWCU), що забезпечує керування зовнішніми і внутрішніми передачами даних, керуючими словами та інформацією про стан ППІ;
- три 8-розрядних порта вводу-виводу (PORT A, B, C) для обміну інформацією із зовнішніми пристроями;
- схема керування каналами A, B, C.

Програмування ППІ

Комбінації режимів роботи ППІ задаються шляхом запису керуючого слова за допомогою команд вводу-виводу за адресою регістра керування. В регістр керування можна тільки записувати керуючі слова, а читання інформації з нього неприпустимо (табл. 1.1).

Таблиця 1.1 – Сигнали керування і функції ППІ

№	Операції	Сигнали керування				
		CS	RD	WR	A1	A0
1	Запис керуючого слова з МП	0	1	0	1	1
2	Запис у канал А	0	1	0	0	0
3	Запис у канал В	0	1	0	0	1
4	Запис у канал С	0	1	0	1	0
5	Читання з каналу А	0	0	1	0	0
6	Читання з каналу В	0	0	1	0	1
7	Читання з каналу С	0	0	1	1	0
8	Відключення ППІ від D (7...0)	1	*	*	*	*

* – довільно

При програмуванні ППІ використовується 8 молодших розрядів адреси; А0, А1 безпосередньо надходять на схему керування ППІ і визначають, це вихідні порти (А1А0=00!01!10) або регістр керування (Ргу) (А0А1=11). Із розрядів А2...А7 у «Мікролаб» апаратним способом формується сигнал CS (вибір кристала). Для кристала (мікросхеми ППІ), код на А2...А7, повинен бути ідентичними для Ргу та портів А, В, С.

Приклад 1. Скласти програму для перезапису даних з порту вводу (С) у порт виводу (В). Вважати, що А7...А2 відповідають комбінації 111110.

Рішення:

Керуюче слово, відповідно до завдання, має наступний вид:

D7									D0
1	0	0	0	0	0	0	0	0	1

Текст програми

```

MVI A, 81      ;запис в акумулятор даних (керуюче слово)
OUT FB        ;програмування інтерфейсу (<А> → Ргу (1111011)
              ;дані з акумулятора пересилаються за адресою FB)
START: IN FA  ;читання даних з порту С 11111010
OUT F9        ;запис даних у порт В 11111001
JMP START     ;цикл
    
```

Режим роботи кожного з каналів ППІ програмується за допомогою керуючого слова. Керуюче слово може задати один із трьох режимів:

- основний режим вводу-виводу (режим 0);
- ввід-вивід, який стробується (режим 1);
- режим двонаправленої передачі інформації.

Одним керуючим словом можна встановити різні режими роботи для кожного з каналів. Розряди D0 – D2 керують групою В (CUB), розряди D3 – D6 керують групою А (CUA). Розряд D7 визначає або установку режимів роботи каналів

(D7 = 1), або роботу ППІ в режимі скидання/установки окремих розрядів каналу С (D7 = 0).

Режим «0» застосовується при синхронному обміні, або при програмній організації асинхронного обміну. У цьому режимі ППІ може розглядатися як пристрій, що складається з чотирьох портів (два 8-розрядних та два 4-розрядних). Вивід інформації з МП здійснюється по команді OUT з фіксацією виведеної інформації в регістрах каналів, а ввід в МП – по команді IN без запам'ятовування інформації. Дані зберігаються в ППІ до виконання МП чергової команди виводу (OUT). Програма для «Мікролаб» приведена в табл. 1.2.

Таблиця 1.2 – Програма перезапису для «Мікролаб»

Адреса	Код	Мітка	Мнемокод
8000	3E		MVI A, 81
8001	81		
8002	D3		OUT FB
8003	FB		
8004	DB	START:	IN FA
8005	FA		
8006	D3		OUT F9
8007	F9		
8008	C3		JMP START
8009	04		
800A	80		

Результати роботи програми в емуляторі наведені на рис. 1.6.

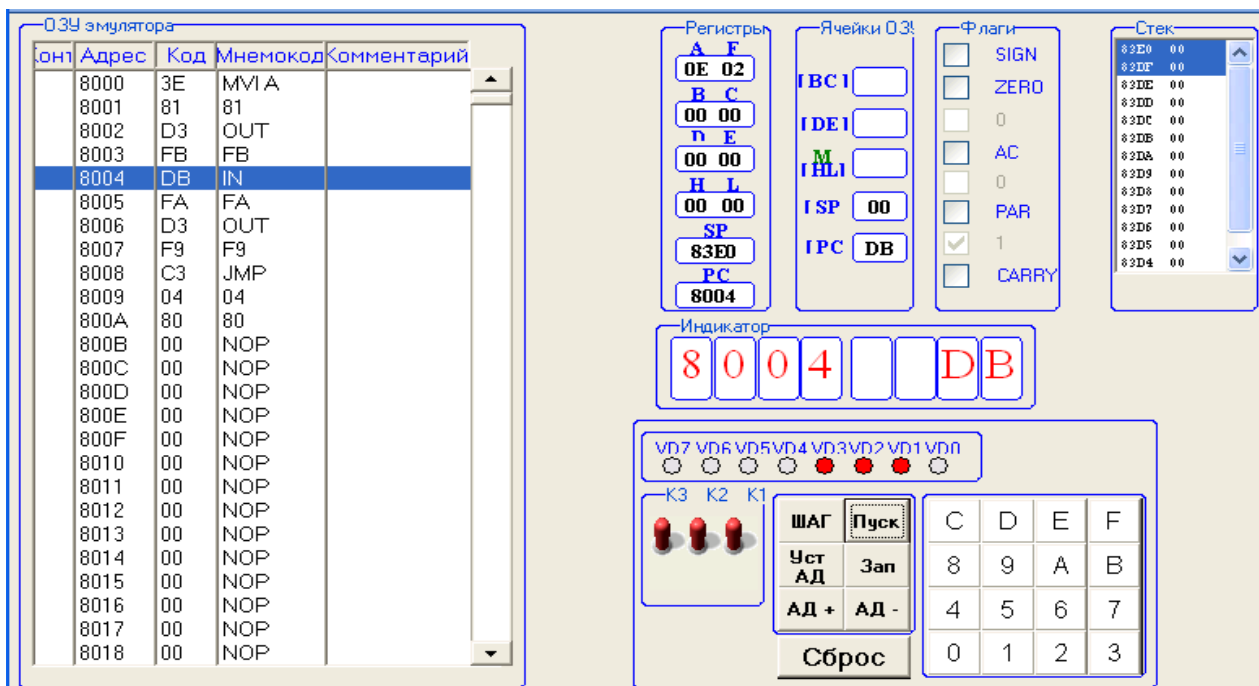


Рисунок 1.6 – Результат роботи програми в емуляторі

В наведеній програмі горінням світлодіодів керують тумблери. Якщо увімкнено усі три тумблери, то будуть горіти усі світлодіоди; якщо буде увімкненим лише перший тумблер – буде горіти світлодіод VD1.

Індивідуальні завдання

1. Вибрати завдання з табл. 1.3 та табл. 1.4 згідно з варіантом.
2. Скласти алгоритм програми.
3. Затвердити алгоритм та програму у викладача.
4. Занести програму в мікро-ЕОМ, перевірити її працездатність в покроковому режимі та отримати результати.
5. Перевірити програму в автоматичному режимі.
6. Оформити звіт про роботу.

Таблиця 1.3 – Варіанти завдань

Номер завдання	Зміст
1	Організувати миготіння світлодіодів.
2	Включити тумблером К світлодіод N.
3	Подати звуковий сигнал заданого тону (f, Гц).

Таблиця 1.4 – Вихідні дані

№ варіанта	№ завдання	№ тумблера	№ світлодіода	Частота звук. сигн.	Примітка
1	1		1, 3, 5		Нумерація світлодіодів і тумблерів згідно принципової схеми «Мікролаб»
2	2	1	4		
3	3			1	
4	1		2, 4, 7		
5	2	2	3		
6	3			0.5	
7	1		1, 6, 8		
2	2	3	5		
9	3			2	

Порядок виконання роботи

1. Ознайомитися з структурою МПЛ «Мікролаб» та її призначенням.
2. Ознайомитися з «SENTINEL KR580Emul» – емулятором роботи МПЛ «Мікролаб КР580ИК80».
3. Ознайомитися з загальними відомостями про організацію вводу-виводу в МПС.
4. Ознайомитися з програмованим паралельним інтерфейсом КР580ВВ55.
5. Ознайомитися з процесом програмування ППІ.
6. Виконати індивідуальне завдання.

Зміст звіту

1. Тема лабораторної роботи.
2. Мета роботи.
3. Опис структурної схеми МПЛ «Мікролаб».
4. Принципові схеми структурних елементів МПЛ «Мікролаб» з поясненнями.
5. Виконання індивідуального завдання.
6. Висновки.

Контрольні питання

1. Поясніть структурну схему МПЛ «Мікролаб КР580ИК80».
2. Поясніть призначення функціональних елементів МПЛ «Мікролаб КР580ИК80».
3. Поясніть, яким чином проводиться програмування МПЛ «Мікролаб КР580ИК80».
4. Поясніть, яким чином проводиться програмування в емуляторі МПЛ «Мікролаб КР580ИК80».
5. Організація вводу-виводу за допомогою БР К580ИР12.
6. Структурна схема ППІ КР580ВВ55.
7. Режими роботи ППІ.
8. Програмування ППІ.
9. Методи організації вводу-виводу в МПЛ.
10. За допомогою яких команд мікро-ЕОМ може здійснювати ввід-вивід інформації?
11. За скільки машинних тактів здійснюється ввід-вивід даних по командах IN та OUT?

Лабораторна робота 2 ДОСЛІДЖЕННЯ СТРУКТУРИ ТА ФУНКЦІОНУВАННЯ КЛАВІАТУРИ У МПЛ «МІКРОЛАБ»

Мета роботи: Вивчення організації і принципів роботи клавіатури та програмно-апаратних методів підключення клавіатури до мікро-ЕОМ.

Теми для попереднього пророблення:

- організація і принцип роботи клавіатури мікро-ЕОМ;
- технічні засоби сполучення клавіатури з мікро-ЕОМ;
- спеціалізований процесор вводу-виводу K580BM89;
- контролер клавіатури/дисплея KP580BV79.

Загальні відомості про принципи організації вводу інформації з клавіатури

Клавіатура є одним із широко розповсюджених пристроїв вводу даних і керуючих впливів у мікро-ЕОМ. В усіх випадках при організації вводу інформації з клавіатури в мікро-ЕОМ перед розроблювачем ставиться ряд задач, з яких до основних можна віднести:

- визначення факту натискання клавіші на клавіатурі;
- визначення номера натиснутої клавіші;
- здійснення передачі керування на відповідну програму.

Найбільш зручно організувати клавіатуру у вигляді матриці розміром $n \times m$, де n та m – відповідно число рядків і стовпців. При такому способі організації до мікро-ЕОМ можна підключити $n \times m$ клавіш. Сполучення клавіатури з мікро-ЕОМ роблять за допомогою пристроїв вводу-виводу даних (інтерфейсів, спеціалізованих процесорів).

При програмному способу дешифровки натиснутої клавіші визначення факту натискання на клавішу може бути здійснене за допомогою такої послідовності операцій:

- 1) записати нулі в розряди вихідного пристрою;
- 2) зчитати вміст розрядів вихідного пристрою;
- 3) повторити знову, якщо в усіх розрядах вхідного пристрою записані одиниці.

Приклад 1. Програма, яка приведена в табл. 2.1, дозволяє визначити факт натискання на одну з клавіш, підключених до трьох молодших розрядів пристрою вводу-виводу, але не вказує номер натиснутої клавіші.

Визначити номер натиснутої клавіші можливо, наприклад, використовуючи алгоритм, що ґрунтується на послідовному записі нулів в кожен з розрядів матриці клавіатури. При наявності нулів у кожному ряду мікро-ЕОМ визначає факт натискання на клавіші, що знаходяться у стовпцях клавіатури, належать аналізованому ряду. Якщо яка-небудь клавіша натиснута, то визначається її номер за номером розряду, у якому записано нуль.

Таблиця 2.1 – Програма, що дозволяє визначити факт натискання на одну з клавіш

Адреса	Код	Мітка	Мнемокод	Коментарі
0800	3E F8		MVI A, 11111000	запис. «0» у три мол. розряди акумулятора
0802	D3 KBDOT		OUT KBDOT	запис. «0» у вих. пристр. (адр. KBDOT)
0804	DB KBDIN		IN KBDIN	одержати число з вх. пристр. (адр. KBDIN)
0806	E6 07	WAITK	ANI 00000111	очистити старші 5 розрядів акумулятора
0808	FE 07		CPI 00000111	чи є в мол. 3-х розрядах акумулятора «0»
080A	CA 0008		JZ WAITK	якщо ні, то перехід на WAITK
080D	C3 0D08		JMP DONE	кінець

Ознайомлення із загальними відомостями про програмну організацію зчитування даних із клавіатури «Мікролаб»

У «Мікролаб» клавіатура з'єднана з системою через ППІ, реалізований на КР580ИК55 та являє собою набір ключів (кнопок) у вигляді матриці 8x3 (окрім кнопки «скидання»).

Засобом керування всіма операціями системи є програма монітору. Усі функції цієї програми задаються за допомогою простих операцій клавіатури. Дані, що зчитуються, перетворюються у код, що відповідає натиснутій кнопці. Це перетворення виконує частина програми «монітор» – підпрограма «KEYIN» (адреса 0216). Після повернення з підпрограми «KEYIN» в акумуляторі знаходиться код натиснутої кнопки. У табл. 2.2 наведені коди кнопок «Мікролаб».

Таблиця 2.2 – Коди кнопок мікро-ЕОМ «Мікролаб»

Кнопка	Код	Кнопка	Код	Кнопка	Код	Кнопка	Код
0	00	6	06	С	0С	Вст. AD	12
1	01	7	07	D	0D	AD+	13
2	02	8	08	E	0E	AD–	14
3	03	9	09	F	0F	ЗП	15
4	04	A	0A	Пуск	10	Вивід	16
5	05	B	0B	Повернення	11	Ввід	17

Принцип роботи клавіатури мікро-ЕОМ «Мікролаб»

До восьми розрядів порту В підключені світлодіодні індикатори. Цей порт програмується як порт виводу. До трьох розрядів порту С (які не використовуються при скануванні клавіатури) приєднані тумблери. Тобто, запрограмувавши ППІ певним чином, можна зчитувати дані з перемикачів і

виводити дані на світлодіодні індикатори. До нульового розряду порту *B* окрім світлодіода підключений динамік, тобто, організувавши певним чином програму, можна «змусити» динамік звучати.

Програма монітору видає по черзі на три виходи мікросхеми ППІ негативні імпульси, зчитуючи після кожного вводу інформацію з порту *A*. Кожен розряд порту *C* з трьох виходів приходять через буферний каскад на 8 кнопок (ряд). У випадку, якщо жодна з кнопок ряду, на який подано низький рівень, не натиснута, у порт *A* надійде код FF. Якщо натиснута хоча б одна кнопка, то у відповідному розряді порту *A*, в якому при введенні був логічний «0», програма монітору формує код натиснутої кнопки та звертається до підпрограми, яка виконує відповідну цій кнопці операцію.

Читання ряду кнопок є двокрокова операція: запис даних у скануючий порт (FA) та читання байта даних портом, який зчитує (F8). За один цикл зчитуються дані одного ряду з восьми кнопок. Щоб вибрати визначений ряд кнопок, відповідний біт скануючого порту (PC4, PC5 чи PC6) встановлюється в логічну «1», інші байти залишаються рівними логічному «0».

Наприклад, щоб вибрати ряд кнопок 8...F, потрібно відправити в порт FA код 10101111 (AF), а щоб вибрати кнопки ПУСК...ВВІД – код 11001111 (CF). Отже, щоб зчитати дані з ряду 0...7 у порт FA вводиться байт 9F, потім виконується читання вхідним портом інформації про стан стовпчиків. При натисканні кнопки відповідний біт порту, який зчитує, буде встановлюватися в «0» (табл. 2.3).

Таблиця 2.3 – Коди кнопок, що зчитуються з клавіатури мікро-ЕОМ «Мікролаб»

Кнопка	Код, що читається	Код, що читається, (h)	Кнопка	Код, що читається	Код, що читається, (h)
не натиснута	11111111	FF	4	11101111	EF
0	11111110	FE	5	11011111	DF
1	11111101	FD	6	10111111	BF
2	11111011	FB	7	01111111	7F
3	11110111	F7			

Програма KEYIN не тільки сканує клавіатуру, але й перетворює коди, що зчитуються зі стовпчиків (у залежності від обраного ряду), у коди, які наведені в табл. 2.2.

Приклад 2. Програма для перевірки стану кнопки «2» (використовується програма KEYIN) наведена у табл. 2.4. Якщо кнопка «2» натиснута, генерується звуковий сигнал. Але звуковий сигнал не залежить від натискання кнопки у симуляторі. Звук все одно буде генеруватися (такою є особливість роботи в стимуляторі).

Таблиця 2.4 – Програма для перевірки стану кнопки «2»

Адреса	Код	Мітка	Мнемокод	Коментарі
8000	3E		MVI A, 9F	вибір ряду кнопок 0...7
8001	9F			
8002	D3		OUT FA	
8003	FA			
8004	BB	READ	IN F8	читання стану колонок порівняння акумулятора з кодом (кнопка «2» – код FB)
8005	F8		CPI FB	
8006	FE			
8007	FB			
8008	C2		JNZ READ	читати знову, якщо кнопка «2» не натиснута
8009	04			
800A	80			
800B	CD		CALL BEEP	якщо натиснута кнопка «2», з'явиться звуковий сигнал
800C	50			
800D	03			
800E	C3		JMP READ	
800F	04		CALL READ	
8010	80			повторення програми

Приклад виконання лабораторної роботи

Розробимо програму, в якій по натисненню клавіші «5» повинен спрацювати звуковий сигнал. Лістинг програми з використанням підпрограми KEYIN наведений у табл. 2.5.

Таблиця 2.5 – Програма для перевірки стану кнопки «5» (використовується програма KEYIN)

Адреса	Код	Мнемокод	Коментарі
8000	CD	READ: CALL KEYIN	виклик підпрограми, читання з клавіатури
8001	16		
8002	02		
8003	FE	CPI 05	порівняння KEYIN з 05
8004	05		
8005	C2	JNZ READ	повернення, якщо кнопка «5» не натиснута
8006	00		
8007	80		
8008	CD	CALL BEEP	звуковий сигнал, якщо кнопка «5» натиснута
8009	50		
800A	03		
800B	C3	JMP READ	повторення програми
800C	00		
800D	80		

Лістинг програми, яка реалізує поставлене завдання без використання підпрограми KEYIN наведений у табл. 2.6.

Таблиця 2.6 – Програма для перевірки стану кнопки «5» (програма KEYIN не використовується)

Адреса	Код	Мнемокод	Коментарі
8000	3E	MVI A, 8F	вибір ряду кнопок 8F
8001	8F		
8002	D3	OUT FA	вивід даних у порт FA
8003	FA		
8004	DB	READ: IN F8	читання стану стовпців
8005	F8		
8006	FE	CPI 05	порівняння акумулятора з кодом клавіші 05
8007	05		
8008	C2	JNZ READ	читання знову, якщо кнопка не натиснута
8009	04		
800A	80		
800B	CD	CALL BEEP	виклик звукового сигналу
800C	50		
800D	03		
800E	C3	JMP READ	повторення програми
800F	04		
8010	80		

Індивідуальне завдання

Розробити програму спрацьовування звукового сигналу при натисканні клавіші згідно варіанта та перевірити її роботу. Використати два способи – з використанням програми KEYIN та без її використання.

Таблиця 2.7 – Варіанти завдань

№ варіанту	Найменування кнопки
1	A
2	B
3	C
4	D
5	E
6	F
7	9
8	8
9	7

Порядок виконання роботи

1. Ознайомитися з основними принципами організації вводу інформації з клавіатури.

2. Ознайомитися з загальними відомостями про програмну організацію зчитування даних з клавіатури «Мікролаб».
3. Розробити програму спрацьовування звукового сигналу при натисканні кнопки згідно завдання (табл. 2.7) із використанням програми KEYIN.
4. Ввести програму в «Мікролаб» та перевірити її роботу.
5. Розробити програму спрацьовування звукового сигналу при натисканні клавіші згідно завдання (табл. 2.7) без використання програми KEYIN.
6. Ввести програму в «Мікролаб» та перевірити її роботу.
7. Перевірити правильність функціонування програм.
8. Оформити звіт про роботу.

Зміст звіту

1. Тема лабораторної роботи.
2. Мета роботи.
3. Індивідуальне завдання.
4. Перелік команд вводу-виводу для МП VM80.
5. Алгоритми і тексти програм.
6. Структурна схема підключення клавіатури до мікро-ЕОМ.
7. Результати роботи програм.
8. Висновки.

Контрольні питання

1. Пристрої вводу-виводу інформації.
2. Основні принципи організації вводу інформації з клавіатури.
3. Перелічіть режими роботи програмувального пристрою вводу/виводу інформації.
4. Способи організації опитування клавіатури (сканування клавіатури, організація сканування).
5. Як здійснюється зчитування інформації з клавіатури в «Мікролаб»?

Лабораторна робота 3

ДОСЛІДЖЕННЯ ОРГАНІЗАЦІЇ ПРЯМОГО ДОСТУПУ ДО ПАМ'ЯТІ У МПЛ «МІКРОЛАБ»

Мета роботи: Вивчення принципів організації прямого доступу до пам'яті (ПДП), засобів прямого доступу та схемотехнічну реалізацію ПДП у мікро-ЕОМ. Придбання практичних навичок програмування контролера мікро-ЕОМ.

Теми для попереднього пророблення:

- загальні принципи організації ПДП;
- програмування контролера ПДП (КПДП), режими роботи КПДП;
- організація ПДП у «Мікролаб».

Загальні положення

Важливою властивістю, що поліпшує продуктивність процесора, є здатність виконувати пересилання з прямим звертанням до пам'яті. У звичайних операціях вводу та виводу, процесор сам керує передачею даних. Інформація, що повинна бути розміщена в пам'яті, пересилається з пристрою вводу в процесор, а потім із процесора в зазначену комірку пам'яті.

Прямий доступ до пам'яті – це метод перенесення даних безпосередньо між пам'яттю та периферійним пристроєм. Дані через МП не проходять. Це забезпечується керуючим входом «HOLD» (захоплення шин) МП.

Коли ця лінія знаходиться в стані логічної «1», МП закінчує машинний цикл, потім зупиняється, щоб перевести лінію «HLDA» (підтвердження захоплення шини) у стан логічної «1». Усі виходи шин МП (адресної, даних і керування) переводяться у високоімпендансний стан (z-стан). Тепер периферійні пристрої можуть керувати цими шинами і виконувати всі необхідні перенесення даних. Коли лінія «HOLD» встановлюється периферійним пристроєм у стан логічного «0», МП продовжує роботу з тієї точки, де він зупинився. Для видачі адресних і керуючих сигналів периферійний пристрій повинен мати керуючий пристрій для прямого доступу до пам'яті. Якщо МПС має ПДП, то всі буфери шин повинні мати три стани. Лінія «HLDA» забороняє виходи буферів і дозволяє контролеру ПДП використовувати ці шини.

Загальні принципи організації ПДП

Режим ПДП є самим швидкісним методом обміну, що реалізується за допомогою спеціальних апаратних засобів контролерів ПДП без використання програмного забезпечення.

Для забезпечення режиму ПДП контролер повинен виконувати ряд послідовних операцій:

- 1) прийняти запит DREQ на ПДП від зовнішнього пристрою;
- 2) сформував запит MRQ на захоплення шин для центрального процесора (ЦП);
- 3) прийняти сигнал HLDA, що підтверджує цей факт після того, як ЦП ввійде в стан захоплення шин адреси (ША), даних (ШД), управління (ШУ) у z-стан);

- 4) сформувати сигнал DACK, що повідомляє зовнішньому пристрою про початок виконання циклів ПДП;
- 5) сформувати на ША адресу комірки пам'яті призначеної для обміну;
- 6) виробити сигнали MR, IOW, MW, IOR, які забезпечують керування обміном;
- 7) по закінченні ПДП або повторити цикл ПДП, змінивши адресу, або припинити ПДП, знявши запити на ПДП.

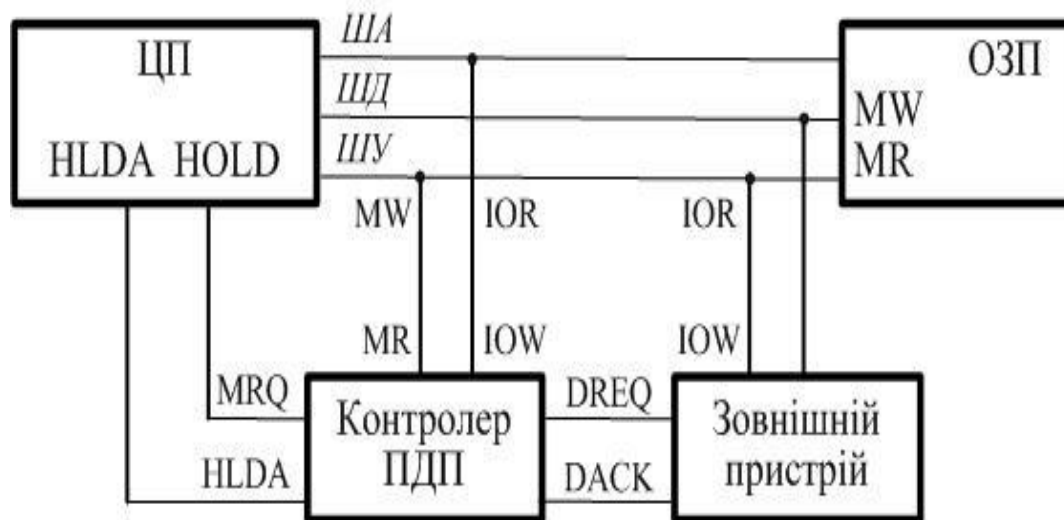


Рисунок 3.1 – Структурна схема МПС з контролером ПДП

Цикли ПДП виконуються з послідовно розташованими комірками пам'яті, тому КПДП повинен мати лічильник адреси ОЗП. Число циклів ПДП визначається спеціальним лічильником. Керування обміном здійснюється спеціальною логічною схемою, що формує в залежності від типу обміну, пари керуючих сигналів: MR, IOR (цикли читання), MW, IOW (цикли запису). Контролер ПДП по запиту бере на себе керування системними шинами і виконує сполучені цикли читання-виводу або запису/вводу доти, поки вміст лічильника циклів ПДП не буде дорівнювати нулю.

Прямий доступ до пам'яті в «Мікролаб» використовується для організації відображення внутрішнього стану системи за допомогою пристрою індикації. Пристроєм індикації, що служить для візуального спостереження в мікро-ЕОМ є 8 восьмисегментних індикаторів VD1-VD8. Інформація відображається на індикаторах у 16-річному коді. Індикація інформації динамічна. Дані на індикатори передаються з восьми старших комірок ОЗП з адресами 83F8-83FF за допомогою ПДП без участі ЦП. У кожній з цих комірок ОЗП знаходиться вісьмисегментний код, що відповідає своєму індикатору.

Приклади використання режиму ПДП при організації індикації

Приклад 1. Розробити програму висвітлювання символів 1, 2, 3, 4, А, С, 3, Е, що використовує підпрограму SEGCG (початкова адреса 01C0). Початкові дані для підпрограми SEGCG повинні знаходитися в комірках пам'яті 83F4-83F7.

Програма приведена в табл. 3.1, а результат її роботи – на рис. 3.2.

Таблиця 3.1 – Програма висвітлювання символів

Адреса	Код	Мнемокод	Коментарі
8000	3E	MVI A, 12	12 -> A
8001	12		
8002	32	STA 83F4	A -> (83F4)
8003	F4		
8004	83		
8005	3E	MVI A, 34	34 -> A
8006	34		
8007	32	STA 83F5	A -> (83F5)
8008	F5		
8009	83		
800A	3E	MVI A, AC	AC -> A
800B	AC		
800C	32	STA 83F6	A -> (83F6)
800D	F6		
800E	83		
800F	3E	MVI A, 3E	3E -> A
8010	3E		
8011	32	STA 83F7	A -> (83F7)
8012	F7		
8013	83		
8014	CD	CALL SEGCG	виклик SEGCG
8015	C0		
8016	01		
8017	76	HLT	зупинка

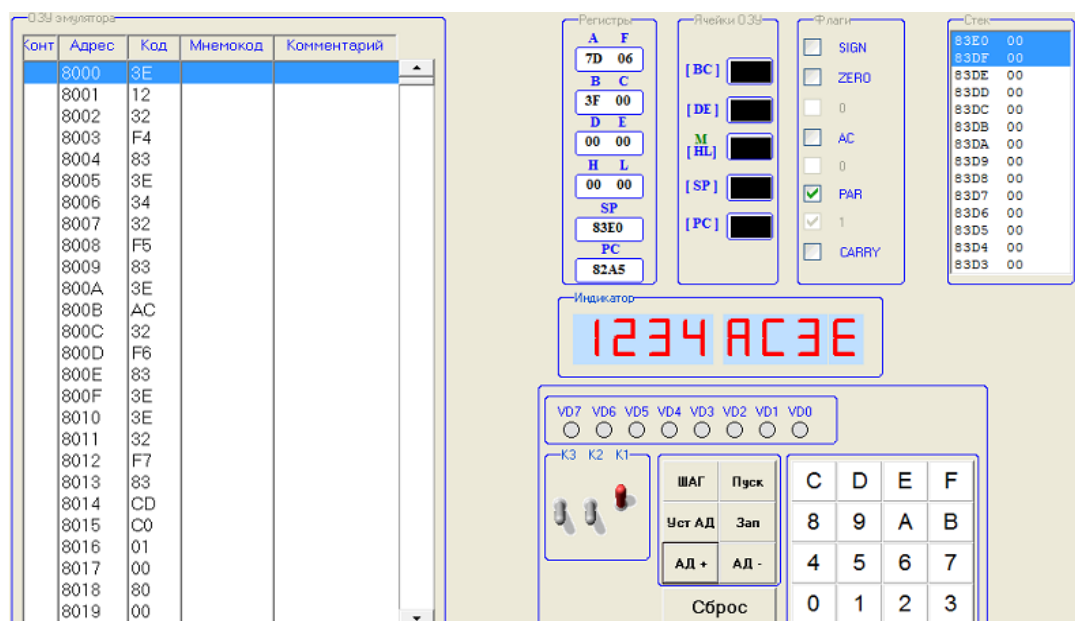


Рисунок 3.2 – Результат роботи програми

Приклад 2. Розробити програму, що виводить на індикатори напис «СС60» без використання підпрограми SEGCG. Семисегментні коди знаходяться в комірках пам'яті 83F8-83FF. Відповідність розрядів і сегментів приведена на рис. 3.3. Програма виводу символів наведена в табл. 3.2.

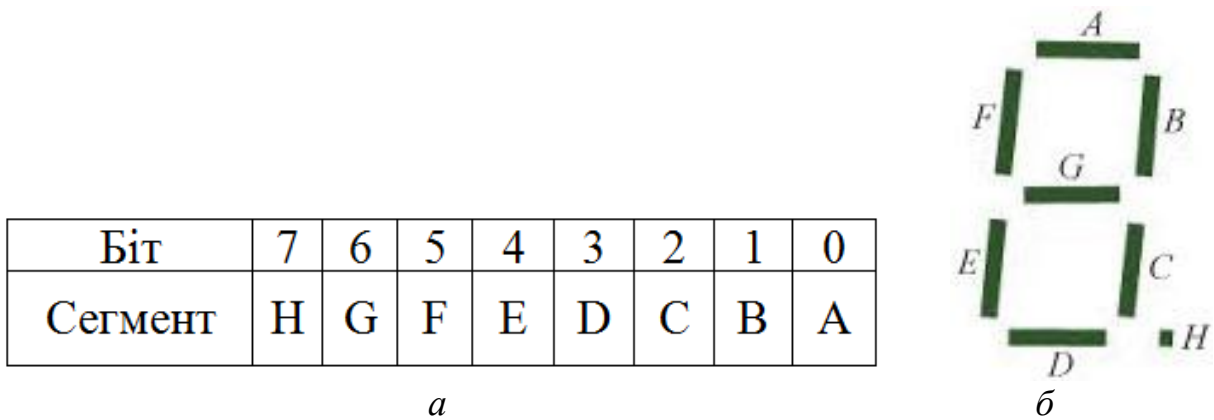


Рисунок 3.3 – Відповідність розрядів (а) та сегментів (б)

Таблиця 3.2 – Програма виводу символів

Мнемокод	Коментарі
MVI A,39	семисегментний код C → A
STA 83F8	A → (83F8)
STA 83F9	A → (83F9)
MVI A,7D	семисегментний код б → A
STA 83FA	A →(83FA)
MVIA,3F	семисегментний код 0 → A
STA 83FB	A →(83FB)
XRAA	0→ A
STA 83FC	A →(83FC) гасіння індикаторів
STA 83FD	A →(83FD)
STA 83FE	A →(83FE)
STA 83FF	A →(83FF)
HLT	зупинка

Приклад виконання завдання

Лістинг програми, яка виконує вивід на екран мікро-ЕОМ «10b ГА» по натисненню клавіші «3» приведений в табл. 3.3. Результат виконання програми наведений на рис. 3.4.

Таблиця 3.3 – Програма виводу на екран символів

Адреса	Код	Мнемокод	Коментарі
1	2	3	4
8000	CD	READ: CALL KEYIN	Виклик процедури KEYIN
8001	16		
8002	02		
8003	FE	CPI 03	Порівняння ключа з 03
8004	03		
8005	C2	JNZ READ	Повернення, якщо не 03
8006	00		
8007	80		
8008	3E	MVI A,06	Семисегментний код «1»
8009	06		
800A	32	STA 83F8	A -> [83F8]
800B	F8		
800C	83		
800D	3E	MVI A,3F	Семисегментний код «0»
800E	3F		
800F	32	STA 83F9	A -> [83F9]
8010	F9		
8011	83		
8012	3E	MVI A,7C	Семисегментний код «b»
8013	7C		
8014	32	STA 83FA	A -> [83FA]
8015	FA		
8016	83		
8017	3E	MVI A,0	Гасіння
8018	00		
8019	32	STA 83FB	
801A	FB		
801B	83		
801C	32	STA 83FC	A -> [83FC]
801D	FC		
801E	83		
801F	32	STA 83FD	A -> [83FD]
8020	FD		
8021	83		
8022	3E	MVI A,31	Семисегментний код «Г»
8023	31		
8024	32	STA 83FE	A -> [83FE]
8025	FE		
8026	83		
8027	3E	MVI A,77	Семисегментний код «А»

Продовження табл. 3.3

1	2	3	4
8028	77		
8029	32	STA 83FF	A -> [83FF]
802A	FF		
802B	83		
802C	C3	JMP READ	Перехід на мітку READ
802D	00		
802E	80		

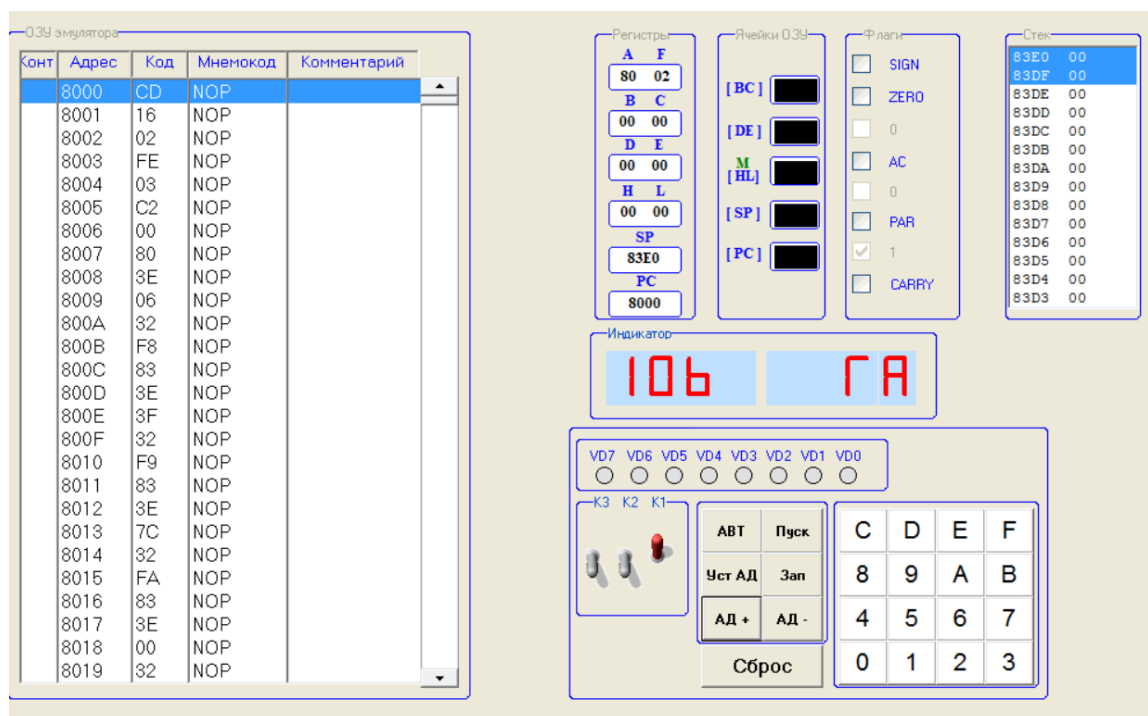


Рисунок 3.4 – Результат виконання програми

Порядок виконання роботи

1. Ознайомитися з загальними відомостями про принципи організації ПДП.
2. Використовуючи вихідні дані, наведені в табл. 3.4, розробити програму виводу інформації на восьмисегментний індикатор з використанням підпрограми SEGCG.
3. Ввести програму в «Мікролаб» та перевірити її роботу.
4. Використовуючи вихідні дані, наведені в табл. 3.4, розробити програму виводу інформації на восьмисегментний індикатор без використання підпрограми SEGCG.
5. Ввести програму в «Мікролаб», запустити та перевірити її роботу.
6. Розробити програму, що виводить на два правих індикатори значення кожної кнопки «X» (відповідно до табл. 3.4), а на інші індикатори – нулі. При цьому використовувати підпрограму KEYIN (поч. адреса 0216) та SEGCG.

Таблиця 3.4 – Варіанти завдань

№ вар.	Інформація до п.3	Інформація до п.1	Кнопка «Х» до п. 5	№ індикатору до п. 5
1	00124В00	1, 5, В, D, F	1	1, 2
2	НГ190000	2, 4, С, .F, А	2	3, 4
3	ЛАБ10000	3, 6, F, А, В	3	5, 6
4	00ГРС012	3, 4, 5, 6, 3	4	7, 8
5	НАПВГПО1	В, С, D, 3, 8	5	1, 2
6	ВГПНАП81	D, F, F, 1, 2	6	3, 4
7	ОННП1110	С, С, F, 8, 9	7	5, 6
8	ГРУПАААА1	А, А, В, F, 6	8	7, 8
9	ПОГОКОО6	9, 8, 7, В, А	9	1, 2

7. Ввести програму в «Мікролаб» та перевірити її роботу.
8. Оформити звіт про роботу.

Зміст звіту

1. Тема лабораторної роботи.
2. Мета роботи.
3. Індивідуальне завдання.
4. Алгоритми і тексти програм.
5. Результати роботи програми.
6. Висновки.

Контрольні питання

1. Принципи організації ПДП у МПС типу VM80.
2. Як утворюється семисегментний код букв, цифр?
3. Схемотехнічна реалізація ПДП. Приклади.
4. Призначення підпрограми «SEGCG».
5. Використання КПДП для організації передачі даних.
6. Структурна схема організації ПДП у «Мікролаб».
7. Схема підключення індикаторів у «Мікролаб».
8. Формування сигналу HOLD (захоплення шини) у «Мікролаб».

Лабораторна робота 4

ДОСЛІДЖЕННЯ ОРГАНІЗАЦІЇ ЧАСОВИХ ФУНКЦІЙ КЕРУВАННЯ У МПЛ «МІКРОЛАБ»

Мета роботи: Дослідження методу реалізації типової функції керування сигналом будь-якої природи – формування часових інтервалів різної тривалості й особливості їхнього програмування на мікро-ЕОМ. Одержання практичних навичок генерації програмно-керованого вихідного сигналу об'єкта керування мікро-ЕОМ.

Теми для попереднього пророблення

- способи реалізації часових функцій керування;
- програмований таймер КР580ВИ53 (призначення, функціонування);
- програмування таймера КР580ВИ53.

Способи реалізації часових функцій керування

Управління периферійними пристроями часто вимагає від мікропроцесорної системи точного завдання часових інтервалів між керуючими сигналами. Зробити це програмно часто неможливо, до того ж вставка тактів очікування в програму гальмує роботу процесора. Такі завдання доручають програмованим таймерам (ПТ). За допомогою ПТ процесор може формувати часові інтервали довільної тривалості, робити синхронізацію зовнішніх пристроїв, організовувати лічильники подій, вести рахунок поточного часу та ін.

Таймер КР580ВИ53 містить три незалежних 16-бітових лічильника з максимальною частотою рахунку 2 МГц. Кожен з трьох лічильників може бути запрограмований на різні режими роботи і рахунки (двійковий або двійково-десятковий).

Таймер КР580ВИ53 може працювати як програмований тактовий генератор; лічильник подій; бінарний перемножувач; цифровий одновібратор; годинник реального часу.

У сучасних ЕОМ таймер може виконувати досить різноманітні функції:

1) відраховувати інтервали регенерації динамічної пам'яті. У цьому випадку вибір оптимального періоду дозволяє підвищити продуктивність обчислювальної системи за рахунок зменшення простоїв процесора при регенерації пам'яті;

2) у мультизадачних системах таймер дозволяє відслідковувати час виконання того чи іншого завдання.

3) можливе використання таймера в системах реального часу, щоб встановити тривалість виконання технологічних операцій на виробництві;

4) лічильники таймера можуть бути використані для генерації псевдовипадкових чисел;

5) іноді таймери застосовуються при створенні систем з закритими алгоритмами обробки інформації для захисту від розкриття алгоритму трасуванням;

6) таймер може бути використаний в якості сторожа при роботі процесора в захищеному режимі для виходу з тупикових ситуацій;

- 7) досить часто таймер застосовують при генерації різних звукових ефектів в ігрових системах;
- 8) широке застосування таймер має як джерело опорної частоти при роботі інших пристроїв комп'ютера.

Ознайомлення із загальними відомостями про програмований таймер КР580ВИ53

ПТ може використовуватися в наступних стандартних режимах роботи: лічильник зовнішніх подій, програмувальний мултивібратор що чекає, дільник частоти, генератор меандру тощо. Велика інтегральна схема (ВІС) містить три незалежних канали. У кожному каналі є регістр керуючого слова, 16-розрядний програмувальний лічильник, що працює на вирахування в двійковому чи двійково-десятковому коді, двобайтний буферний регістр, у який по спеціальній команді (CLC) зчитується поточний код лічильника. Програмування кожного каналу таймера зводиться до наступних операцій:

- 1) запис у регістр керуючого слова керування каналом;
- 2) запис у 16-розрядний програмувальний лічильник (регістр констант) необхідного коду перерахування.

Після виконання цих операцій канал готовий до роботи під керуванням зовнішніх сигналів.

Особливості програмування ПТ:

- 1) запис індивідуального слова керування каналу відбувається по єдиній для всіх каналів адресі (A0A1=11). Указівка конкретного каналу, до якого відноситься керуюче слова, міститься в самому керуючому слові;
- 2) необхідно обов'язково завершити цикл звертання до лічильника цілком, тобто якщо запрограмовано два звертання, то два і потрібно виконати;
- 3) звертання до адрес лічильників каналів може відбуватися у будь-якій послідовності;
- 4) зчитування регістра керуючого слова заборонено.

Формат слів та команд таймера наведено на рис. 4.1 та рис. 4.2 відповідно.

D7				D0			
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

Рисунок 4.1 – Формат слова ПТ: SC1, SC0 – адреса регістра каналу; RW1, RW0 – розмірність і порядок завантаження констант; M2, M1, M0 – режим роботи (0-5); BCD – режим рахунку (0 – двійковий, 1 – двійково-десятковий).

SC1	SC0	0	0	X	X	X	X
-----	-----	---	---	---	---	---	---

Рисунок 4.2 – Формат команди CLC: SC1, SC0 – адреса каналу; 0,0 – код команди CLC; X – не використовується.

Приклад 1. Програма початкової установки таймера:

- канал 0, режим 5, у лічильник завантажити 15H;
- канал 1, режим 1, у лічильник завантажити 06H;
- канал 2, режим 5, у лічильник завантажити 0BH,
має вигляд:

TIME: MVI A, 3AH	52H = 01010010B – режим двійкового
OUT PORT 53+3	рахунку, режим 1, канал 1,
MVI A, 52H	звертання до молодшого байту
OUT PORT 53+3	
MVI A, 9AH	9AH = 10011010B – режим двійкового
OUT PORT 53+3	рахунку, режим 5, канал 2,
MVI A, 13H	звертання до молодшого байту
OUT PORT 53	
MVI A, 15H	PORT 53 + i – адреса порту
OUT PORT 53	
MVI A, 06H	
OUT PORT 53+1	
MVI A, 0BH	
OUT PORT 53+2	
RET	

Зчитування МП числа, що міститься в лічильнику, може бути здійснено командою читання («читання лічильника») чи командою CLC («читання на льоту»). При першому способі на час виконання звичайної операції читання зовнішньою логікою повинна бути заборонена подача синхронізуючих імпульсів на даний канал таймера. При другому (командою CLC) зчитування вмісту лічильника каналу не порушує виконання рахунку у каналі.

Приклад 2. Підпрограма читання вмісту лічильника каналу 0 без порушення рахунку у каналі:

```
TIME: MVI A, 0000XXXXB
      OUT PORT 53 + 3
      IN PORT 53
      MOV L, A
      IN PORT 53
      MOV H, A
      RET
```

Ознайомлення з способами реалізації часових функцій керування програмованим таймером КР580ВИ53

Одним з основних видів керування сигналом будь-якої природи є керування за часом чи формування часових затримок сигналу. Реалізація такого роду керування може бути здійснена як з використанням апаратних засобів (програмувального таймера), так і програмним шляхом.

Використовуючи один із шести режимів роботи ПТ, можна реалізувати практично будь-яку часозадаючу функцію, зв'язану як з керуванням МПС, так і специфічні функції керування.

При використанні програмного шляху реалізації часозадаючих функцій, варто продумати спосіб його програмної реалізації.

Програмна реалізація часових затримок використовує метод програмних циклів, при якому в деякий робочий регістр МП завантажується число, що потім у кожному проході циклу зменшується на одиницю. Час затримки визначається числом, завантаженим у регістр-лічильник, і часом виконання команд, що утворюють цикл.

На рис. 4.3 приведена схема алгоритма генерації часової затримки (найпростіший спосіб).



Рисунок 4.3 – Схема алгоритма генерації часової затримки (найпростіший спосіб)

В табл. 4.1 приведена програма, що реалізує алгоритм генерації часової затримки (найпростіший спосіб). Щоб використовувати приведену програму, в акумуляторі встановлюється потрібне значення затримки і викликається програма затримки. Вміст акумулятора зменшується доти, поки не досягає нуля, після чого керування повертається на програму виклику. Час затримки, отриманий при виконанні цієї програми, можна визначити виходячи з числа періодів синхроімпульсів (тактів), необхідних для виконання кожної команди, частоти синхроімпульсів і числа циклів.

Таблиця 4.1 – Програма для реалізації алгоритму генерації часової затримки

Мітки	Мнемокод	Коментарі	Час виконання команди(тактів)
DEL1:	DCR A	Зменшення (A) на одиницю	5
	JNZ DEL1 RET	Продовж. рахунку, якщо (A) != 0	7/10
		Повернення	10
	CALL DEL1	Виклик підпрограми DEL1 з основної програми	17

Для приведеної програми виконання одного циклу складає 15 тактів. Для багаторазового прогону необхідно $N*5$ тактів, (N – первинний вміст акумулятора). Час виконання приведенного фрагмента програми з урахуванням часу звертання до нього:

$$T = (17+(N*15)+10)*0.5 \text{ мкс} = 1926 \text{ мкс (при } N=255).$$

Максимальний час затримки при використанні підпрограми (табл. 4.1) складає 1926 мкс. Для одержання більш тривалої затримки доцільно застосувати команди, що впливають на пари регістрів. Програма, що використовує коди регістрів, приведена в табл. 4.2.

Таблиця 4.2 – Програма для реалізації алгоритму генерації часової затримки (для довших затримок)

Мітки	Мнемокод	Коментарі	Час виконання команди(тактів)
DEL2: M1	LX1 B, NH	Завантаження числа N у пари	10
		регістрів B/C	5
	DCX B	Зменшення вмісту регістрової пари на 1	7
	MOV A, B	Перевірка регістрової пари B/C на 0	4
	ORA C		10
	JNZ M1	Якщо вміст не 0, повторення рахунка	10
RET	Повернення		

Використання команди DCX B, що зменшує вміст пари регістрів B/C дозволяє збільшити рахунок затримки до значення $256*256 = 65536$. Внаслідок того, що команда DCX B не впливає на нульовий прапор МП, для цього використовується команда ORA C.

Після цього може бути використана команда JNZ. Час затримки, отриманий при виконанні приведеної програми обчислюється по виразу:

$$T = ((N-1)*24+10)*0.5 = 0.786 \text{ сек.}$$

Для отримання більш тривалої затримки організовується лічильник. Наприклад, в регістрі *D*, що визначає скільки разів виконується затримка DEL2 (0.786 сек.). Максимальна затримка, отримана таким чином, дорівнює $T = 256*0.786 = 201$ (с.).

Ознайомлення із способами управління керуючими сигналами заданих світлодіодів на мікро-ЕОМ «Мікролаб»

При розробці програми необхідно використовувати алгоритм реалізації заданої послідовності горіння світлодіодів, приведений на рис. 4.4, принцип кодування світлодіодів (табл. 4.3) та основну програму почергового горіння різних груп світлодіодів (табл. 4.4).



Рисунок 4.4 – Алгоритм програми реалізації заданої послідовності горіння світлодіодів

Таблиця 4.3 – Принцип кодування світлодіодів

VD1	VD2	VD3	VD4	VD5	VD6	VD7	VD8	код
1	0	0	1	0	0	1	0	92
0	1	0	0	1	0	0	1	49
0	0	1	0	0	1	1	0	26

Таблиця 4.4 – Основна програму почергового горіння різних груп світлодіодів

Мітки	Мнемокод	Коментарі
SEQ	MOV A, 92	установка горіння світлодіодів VD (1,4,7)
	MVI D, I	час горіння VD (1,4,7)
	CALL DELL	
	MOV A, 49	установка горіння світлодіодів VD (2,5,8)
	MVI D,I	час горіння світлодіодів VD (2,5,8)
	CALL DELL	

Приклад виконання завдання

Розробити алгоритм та програму керування світлодіодами. Перша послідовність світлодіодів – 1, 5, 8 із затримкою 1.1 с. Друга послідовність – 4, 6, 7 із затримкою 4.0 с.

Текст програми, набраний у емуляторі роботи мікро-ЕОМ «Мікролаб» наведено на рис. 4.5. Результати роботи програми наведено на рис. 4.6 та рис. 4.7.

Адрес	Код	Мнемокод			
8000	3E	MVI A			
8001	81	81			
8002	D3	OUT			
8003	FB	FB			
8004	3E	MVI A			
8005	89	89			
8006	D3	OUT	801F	C2	JNZ
8007	F9	F9	8020	1C	1C
8008	16	MVI D	8021	80	80
8009	03	03	8022	15	DCR D
800A	CD	CALL	8023	C2	JNZ
800B	19	19	8024	19	19
800C	80	80	8025	80	80
800D	3E	MVI A	8026	C9	RET
800E	16	16	8011	16	MVI D
800F	D3	OUT	8012	08	08
8010	F9	F9	8013	CD	CALL
801A	25	25	8014	19	19
801B	00	00	8015	80	80
801C	0B	DCX BC	8016	C3	JMP
801D	78	MOV A,B	8017	04	04
801E	B1	ORA C	8018	80	80
			8019	01	LXI BC

Рисунок 4.5 – Текст програми



Рисунок 4.6 – Перша послідовність світлодіодів



Рисунок 4.7 – Друга послідовність світлодіодів

Порядок виконання роботи

1. Ознайомитися з загальними відомостями про формування часових інтервалів за допомогою програмованого таймера.
2. Ознайомитися з способами реалізації часових функцій керування програмованим таймером КР580ВИ53.
3. Ознайомитися з способами управління керуючими сигналами вибраних світлодіодів на мікро-ЕОМ «Мікролаб».
4. Розробити програму горіння заданих груп світлодіодів на визначений час згідно свого варіанту (табл. 4.5).
5. Ввести програму в «Мікролаб» та перевірити її роботу.
6. Оформити звіт про роботу.

Таблиця 4.5 – Варіанти завдань

№ вар.	Час затримки T1 (с)	Час затримки T2 (с)	Перша послідовність світлодіодів	Друга послідовність світлодіодів
1	0.5	1.0	1, 3, 5	2, 4, 6
2	0.6	1.5	2, 4, 6	1, 3, 5
3	0.7	2.0	3, 5, 7	8, 6, 4
4	0.8	2_5	1, 2, 3	6, 7, 8
5	0.9	3.0	3, 4, 5	1, 2, 6
6	1.0	3.5	3, 4, 6	1, 5, 8
7	1.1	4.0	1, 5, 8	4, 6, 7
8	1.2	4.5	2, 5, 7	3, 6, 8
9	1.3	5.0	6, 7, 8	3, 4, 5

Зміст звіту

1. Тема лабораторної роботи.
2. Мета роботи.
3. Індивідуальне завдання.
4. Алгоритми і текст програми.
5. Результати роботи програми.
6. Висновки.

Контрольні питання

1. Призначення програмованого таймеру КР580ВИ53.
2. Режими роботи програмованого таймеру КР580ВИ53.
3. Функції, що може виконувати програмований таймер КР580ВИ53.
4. Методи реалізації типових часових функцій управління.
5. Методи реалізації часових затримок управляючих сигналів.
6. Програмні методи реалізації коротких часових затримок.
7. Програмні методи реалізації довгих часових затримок.

Список літератури

1. Скороделов В. В. Цифрові пристрої та мікропроцесори. Архітектура та програмування мікроконтролерів: навчальний посібник / В. В. Скороделов, О. М. Рисований, О. Ф. Даниленко, М. В. Ліпчанський. – Харків : ХВУ, 2004. – 318 с.
2. Козимов В. В. Железо 2011. Путеводитель по компьютерным устройствам и комплектующим / В. В. Козимов, И. В. Коттер, Р. Г. Прогди и др. – СПб. : Наука и техника, 2011. – 400 с.
3. Зиков І. С. Програмування мікропроцесорів у захищеному режимі: Навчально-методичний посібник / І. С. Зиков, С. Г. Межерицький, А. О. Подорожняк, І. П. Хавіна. – Харків : ТОВ ДІСА ПЛЮС. – 2018. – 264 с.
4. Поворознюк А. І. Архітектура комп'ютерів: навчальний посібник / А. І. Поворознюк. – Харків : Современная печать, – 2009. – 356 с.
5. Рисований О. М. Системне програмування: підручник для студентів напряму «Комп'ютерна інженерія» вищих навчальних закладів / О. М. Рисований. – Харків : НТУ «ХПІ», 2010. – 912 с.
6. Палагута К. А. Микропроцессоры INTEL 8080, 8085 и их программирование / К. А. Палагута. – Москва : МГИУ, 2007. – 104 с.
7. Кравец В. А. Микропроцессоры и микропроцессорные системы. Книга 1. Архитектура и функционирование. Учебное пособие / В. А. Кравец, А. Н. Рисованный, Ф. А. Домнин и др. – Харьков : ХВУ, 2000. – 350 с.
8. Кравец В. А. Микропроцессоры и микропроцессорные системы. Книга 2. Программирование, разработка устройств и систем. Учебное пособие / В. А. Кравец, А. Н. Рисованный, Ф. А. Домнин и др. – Харьков : ХВУ, 2000. – 350 с.
9. Харрис Д. М. Цифровая схемотехника и архитектура компьютера / Д. М. Харрис, С. Л. Харрис. – М. : Morgan Kaufman, 2015. – 1676 с.

ЗМІСТ

Вступ	3
Лабораторна робота 1. Дослідження структури та функціонування програмованого паралельного інтерфейсу мікропроцесорної лабораторії «Мікролаб».....	4
Лабораторна робота 2. Дослідження структури та функціонування клавіатури у МПЛ «Мікролаб».....	12
Лабораторна робота 3. Дослідження організації прямого доступу до пам'яті у МПЛ «Мікролаб»	18
Лабораторна робота 4. Дослідження організації часових функцій керування у МПЛ «Мікролаб».....	25
Список літератури	34

Навчальне видання

Методичні вказівки до виконання лабораторних робіт
з навчальної дисципліни «Програмування мікропроцесорів»
для студентів денної та заочної форм навчання
за спеціальністю «Комп'ютерна інженерія»

Укладачі:

ПОДОРОЖНЯК Андрій Олексійович,
МЕЖЕРИЦЬКИЙ Сергій Геннадійович,
ГЕЙКО Геннадій Вікторович

Відповідальний за випуск проф. Семенов С. Г.
Роботу до видання рекомендував проф. Заполовський М. Й.

В авторській редакції

План 2020 р., поз. 260
Підп. до друку 04.09.2020 р. Формат 60x84 1/16.
Папір офсет. Друк ризографічний. Ум. друк. арк. 2,1.
Наклад 50 прим. Замовлення № 904-20

Видавець:
Видавничий центр НТУ «ХПІ»,
вул. Кирпичова, 2, м. Харків, 61002
Свідоцтво суб'єкта видавничої справи ДК № 5478 від 21.08.2017 р.
