

АРХІТЕКТУРА КОМП'ЮТЕРІВ 2

ОДНОКРИСТАЛЬНИЙ МІКРОКОНТРОЛЕР КР1816ВЕ48

Лабораторний практикум

Лабораторна робота 1.1

ВИВЧЕННЯ СИСТЕМИ КОМАНД МІКРОКОНТРОЛЕРА МК48

Мета роботи: Вивчення структурної схеми МК48, системи та форматів команд. Отримання навиків складання алгоритмів та розробки програм виконання арифметичних операцій для мікроконтролера МК48.

Теоретичні відомості: 1.2., 1.4

Додаткові теоретичні відомості

Структура програми на мові асемблера МК48

Текст програми на мові асемблера МК48 має певну структуру. Кожна команда (і псевдокоманда) записується в рядок наступної конструкції:

МІТКА: ОПЕРАЦІЯ ОПЕРАНД (И) ; КОМЕНТАР

Поля можуть відділятися один від одного довільним числом пробілів.

Мітка

У полі мітки розміщується символічне ім'я комірки пам'яті, в якій зберігається відмічена команда або операнд. Мітка є буквенно-цифровою комбінацією, що починається з букви. Використовуються тільки букви латинського алфавіту. Асемблер МК48 допускає використання в мітках символу підкреслення (_). Довжина мітки не повинна перевищувати шість символів. Мітка завжди завершується двокрапкою (:).

Операція

В полі операції записується мнемонічне позначення команди МК48 або псевдокоманди асемблера, яке являється скороченням (аббревіатурою) повної англійської назви виконуючої дії.

Наприклад:

MOV	–	<i>Move</i>	–	перемістити;
JMP	–	<i>Jump</i>	–	перейти;
DB	–	<i>Define Byte</i>	–	визначити байт.

Для МК48 використовується строго визначений та обмежений набір мнемонічних кодів. Будь-який інший набір символів, що розміщений в полі операції, сприймається асемблером як помилковий.

Операнди

В цьому полі визначаються операнд (або операнди), які приймають участь в операції. Команди асемблера можуть не мати операндів або мати один чи два операнди. Операнди розділяються комою (,).

Операнд може бути заданий безпосередньо або його адресою (прямою або непрямою).

Використані в якості операндів символічні імена та мітки повинні бути визначені, а числові значення – подані з вказанням на систему числення. Для вказання системи числення використовується суфікс – буква, яка розміщується після числа. Число без суфіксу вважається десятковим. Застосовуються наступні суфікси:

<i>B</i>	–	двійкова система обчислення;
<i>Q</i>	–	вісімкова система обчислення;
<i>D</i>	–	десятькова система обчислення;
<i>H</i>	–	шістнадцятирічна система обчислення.

Обробка виразів в процесі трансляції

Вираз представляє собою сукупність символічних імен та числових значень, що пов'язані операторами асемблера. Оператори асемблера забезпечують виконання в форматі двобайтових слів наступних операцій:

арифметичних операцій:		логічних операцій:	
+	– додавання;	ORL	– АБО;
–	– віднімання;	ANL	– І;
*	– множення;	XRL	– виключне АБО;
/	– ділення;	NOT	– заперечення.
MOD	ділення за модулем;		

Наприклад, запис `ADD A, #((NOT 13)+1)` еквівалентний запису `ADD A, #0F3H` та забезпечує додавання до вмісту акумулятора числа 13, що подане доповнювальним машинним кодом.

Коментар

Поле коментаря застосовується для текстового чи символічного пояснення логічної організації прикладної програми. Поле коментаря ігнорується асемблером, тому тут допускається використання будь-яких символів. За правилами мови асемблера поле коментарію починається після символу крапки з комою (;).

Псевдокоманди асемблера

Спеціальна програма асемблер транслює вихідну програму в об'єктні коди. Асемблер бере на себе багато рутинних задач програміста, таких як присвоєння дійсних адрес, перетворення чисел, присвоєння дійсних значень символічним змінним і таке інше. Програміст вказує асемблеру наступні параметри: початкову адресу прикладної програми, кінець програми асемблера, формати даних. Ця інформація додається у вихідний текст прикладної програми у вигляді псевдокоманд (директив) асемблера, які управляють процесом трансляції і не перетворюються в коди об'єктної програми.

Застосування псевдокоманд:

- псевдокоманда `ORG 10H` задає асемблеру адресу комірки пам'яті `10h`, в якій розміщується наступна команда прикладної програми;
- псевдокоманда `EQU` будь-якому символічному імені, яке використовується в програмі, ставить у відповідність певний операнд; наприклад, запис `PET EQU 13`, означає, що під час асемблювання символічне ім'я `PET`, буде замінено числом 13.
- псевдокоманда `END` дає асемблеру вказівку закінчити трансляцію.

Мікроконтролер оперує командами чотирьох типів, формат яких наведений на рис. 3.33. (див. розділ 3.4).

В розділі 3.4 наводяться приклади виконання арифметичних та логічних команд, команд роботи з ознаками, команд зсувів та інших команд, які треба використати при розробці програм в даній практичній роботі.

Підготовка до виконання практичної роботи

1. Вивчити архітектуру і систему команд МК48.

2. Записати номер варіанту (номер залікової книжки) у двійковому поданні і виділити шість молодших розрядів $h_6 - h_1$. Значення операндів обрати з табл. 6.1, які подані в шістнадцятиричній системі числення. Операнди ($X1 - X6$) записати в регістри із портів заданих в табл. 6.1.

3. Розробити операційні схеми, цифрові діаграми стану регістрів під час виконання основних операцій для обчислення виразу.

4. Розробити для МК48 функціонально-структурний алгоритм обчислення виразу.

5. Розробити програму обчислення виразу на асемблері МК48. Ознаку переносу, яка виникає під час виконання операцій додавання та зсувів, запам'ятовувати в окремий регістр, для отримання правильного результату, що має розмір два байта.

Таблиця 6.1. Варіанти завдань

h_2	h_1	X1	X2	X3	X4	X5	X6	h_4	h_3	Порт
0	0	33	4A	1C	06	20	05	0	0	P1
0	1	05	2B	32	1A	04	6E	0	1	P2
1	0	45	05	3C	A0	22	12	1	0	BUS
1	1	2C	03	71	CA	11	FF	1	1	P4,P5

Таблиця 6.2. Варіанти завдань

h_5	h_1	Вираз для обчислення
0	0	$F = 4(X1 + X2 - 1) - (X3 + X4) - (X5 \& X6) / 4$
0	1	$F = 4((X1 + X2) \& (X3 - X4 - 1)) / 2 - 8(X5 \vee X6)$
1	0	$F = 4(X1 - X2) + (X3 \& X4) - (X5 - X6 - 1) / 16$
1	1	$F = 4(X1 \& X2) - (X3 - X4 + 1) - (X5 \vee X6 - 1) / 2$

Порядок виконання роботи

1. Використовуючи моделюючий комплекс SCM МК48 (додаток Б) налагодити розроблену програму:

– сформувати початковий текст програми у вікні екранного редактору комплексу SCM МК48, виконати пошук синтаксичних помилок та скопіювати програму;

– налагодити розроблену програму пристрою управління за допомогою програмно-логічної моделі *SCM MK48* з отриманням повної інформації про хід виконання програми.

2. Зробити висновки.

Зміст звіту

Звіт повинен містити короткі теоретичні відомості, операційні схеми, цифрові діаграми стану регістрів, схему алгоритму, налагоджену програму на символічному асемблері, висновки за роботою.

Контрольні питання

1. Які способи адресації використовуються під час звернення до регістрів банків регістрів?
2. Яким чином виконують переключення між банками регістрів?
3. У якій області пам'яті розташовуватися банки регістрів?
4. Які регістри застосовуються як покажчики адреси під час непрямої адресації?
5. Чому регістр *R5* не може бути застосований як покажчик адреси під час непрямої адресації?
6. Зобразити за допомогою мнемонічної схеми формування адреси для ОЗП.
7. У чому полягає основна відмінність ознак, що входять та не входять до складу регістру *PSW*?
8. Як реалізувати додавання двох двобайтових чисел?

Практична робота 2.2

ВИВЧЕННЯ КОМАНД ПЕРЕДАЧІ УПРАВЛІННЯ МК48

Мета роботи: Вивчення системи команд МК48 і отримання навиків розробки програм, що вміщують команди передачі управління.

Теоретичні відомості: 1.2., 1.4

Підготовка до виконання практичної роботи

1. Вивчити архітектуру і систему команд МК48.

2. Записати номер варіанту (номер залікової книжки) у двійковому поданні і виділити шість молодших розрядів $h_6 - h_1$. Значення операндів обрати з табл. 6.3, які подані в шістнадцятиричній системі числення. Операнди ($X1 - X6$) записати в регістри із портів заданих в табл. 6.4. В регістр $R7$ записується кількість ітерацій циклу, яка також задається в таблиці 6.3.

3. Вибрати алгоритм для розробки програми на асемблері за табл. 6.5. Варіанти алгоритмів надані на рис. 6.1.

4. Розробити операційні схеми, цифрові діаграми стану регістрів під час виконання основних операцій для обчислення виразу.

5. Розробити для МК48 функціонально-структурний алгоритм обчислення виразу.

6. Розробити програму обчислення виразу на асемблері МК48. Ознаку переносу, яка виникає під час виконання операцій додавання та зсувів, запам'ятовувати в окремий регістр, для отримання правильного результату, що має розмір два байта.

Таблиця 6.3. Варіанти завдань

h_2	h_1	$X0$	$X1$	$X2$	$X3$	$X4$	$X5$	$X6$	Кількість ітерацій циклу
		$R0$	$R1$	$R2$	$R3$	$R4$	$R5$	$R6$	
0	0	4A	D3	E5	24	06	20	05	3
0	1	2B	05	27	22	1A	04	6E	4
1	0	55	45	29	20	A0	02	12	5
1	1	03	5C	30	67	0A	11	A0	7

Таблиця 6.4. Варіанти завдань

h_5	h_2	Порти
0	0	$P1$
0	1	$P2$
1	0	BUS
1	1	$P4, P5$

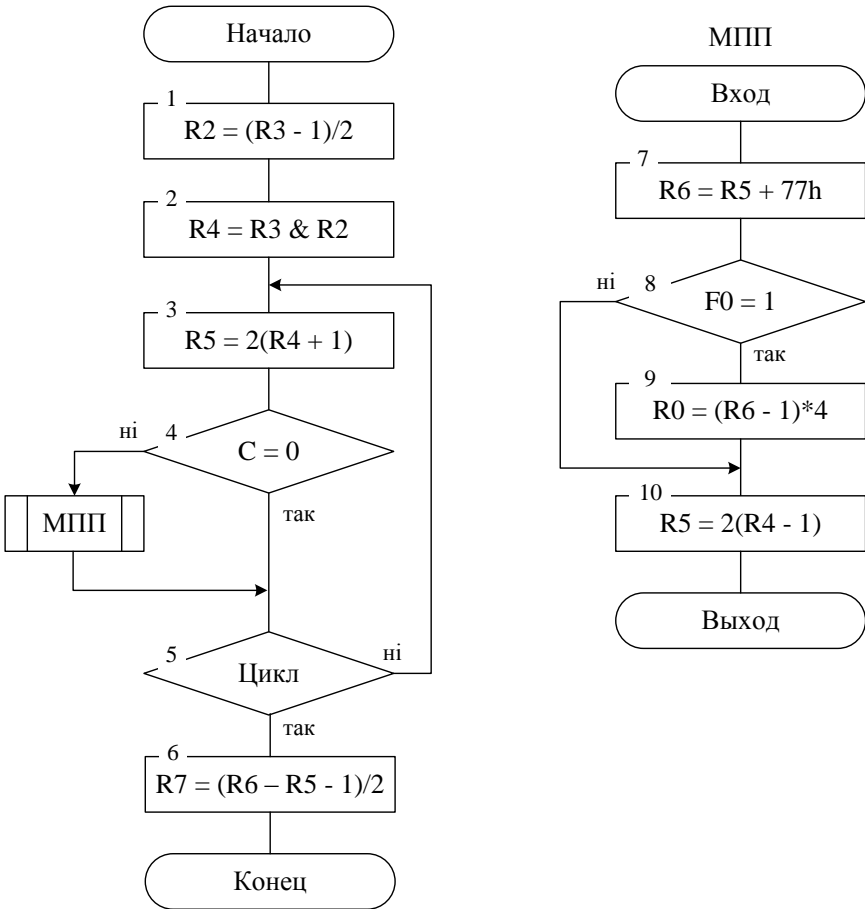
Таблиця 6.5. Варіанти завдань

h_4	h_3	Алгоритм
0	0	рис. 6.1, а
0	1	рис. 6.1, б
1	0	рис. 6.1, в
1	1	рис. 6.1, г

Звіт повинен містити короткі теоретичні відомості, операційні схеми, цифрові діаграми стану регістрів, схему алгоритму, налагоджену програму на символічному асемблері, висновки за роботою.

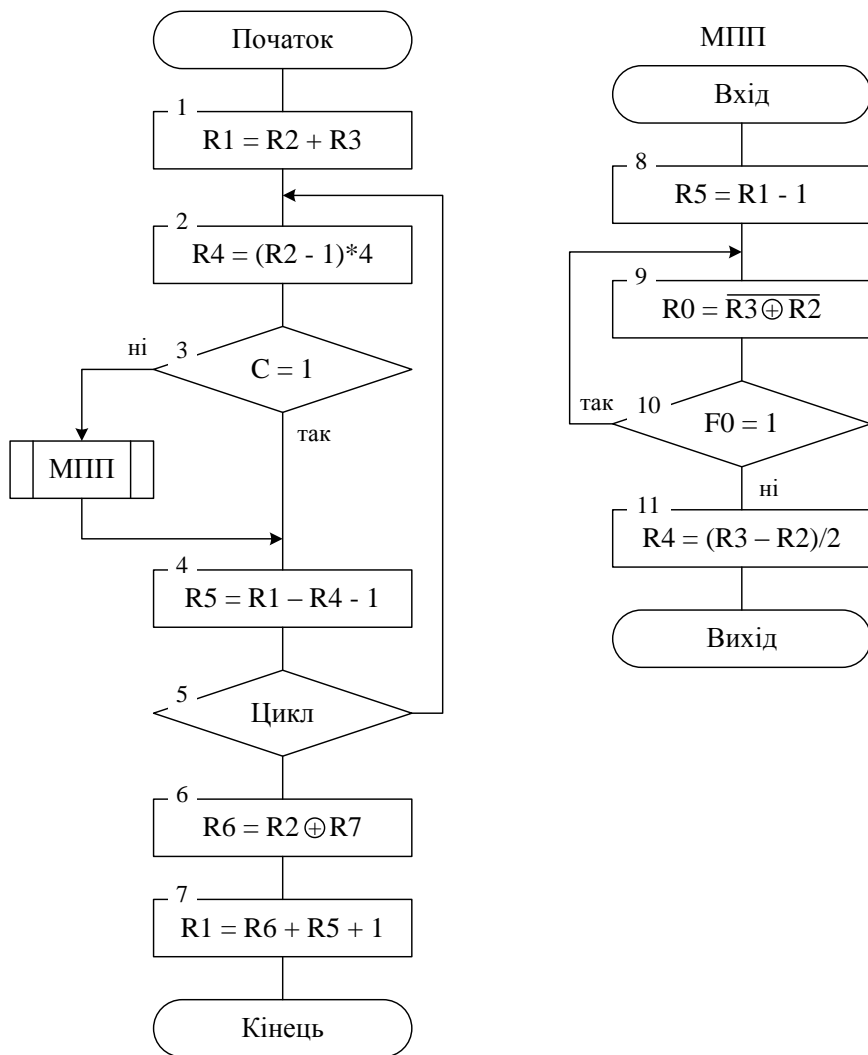
Контрольні питання

1. Які способи адресації використовуються під час звернення до ОЗП?
2. Які регістри являються покажчиками адреси при непрямій адресації під час звернення до резидентної та зовнішньої пам'яті даних?
3. Привести приклад зчитування констант з пам'яті програм. Яким чином можна переключатися між банками пам'яті програм?
4. Охарактеризувати команди зчитування з пам'яті програм. Яким чином виконуються умовні та безумовні переходи на підпрограми? Який регістр використовується як покажчик стеку?
5. Яким чином виконується запис інформації в стек? Яка глибина стеку?
6. Яким чином підключаються сторінки зовнішньої пам'яті даних?
7. Привести приклад запису інформації в зовнішню пам'ять даних. Який спосіб адресації при цьому використовується? Які регістри є покажчиками адреси?
8. Привести приклад підключення до МК48 десяти сторінок ПД та вибору п'ятої сторінки.



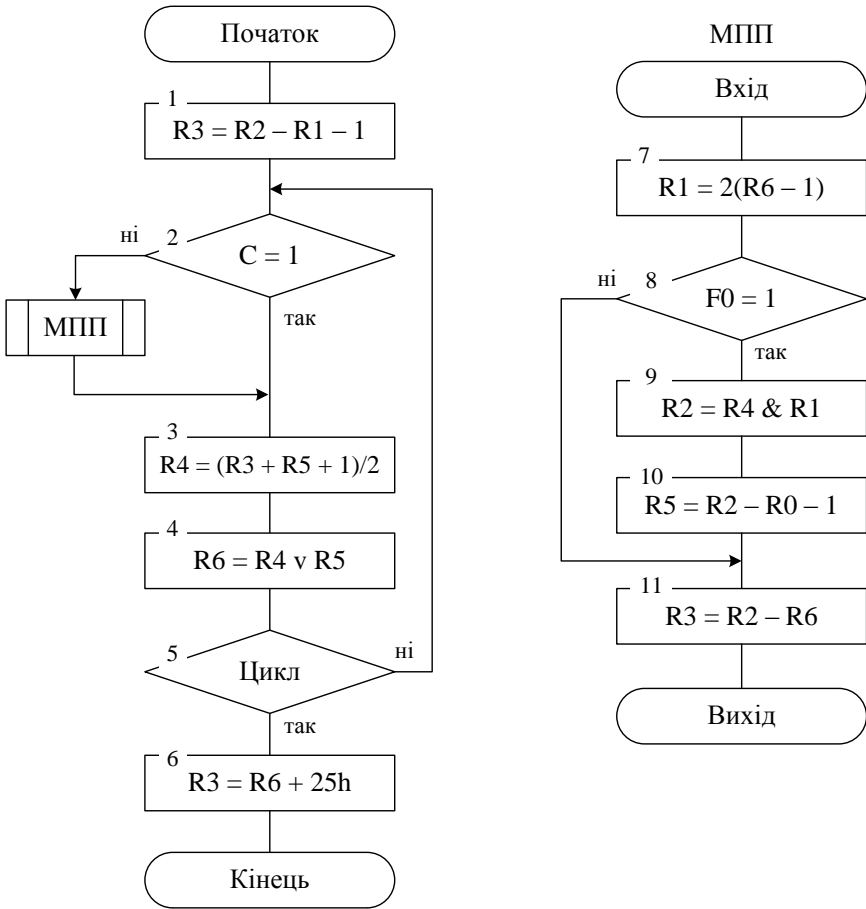
a

Рис. 6.1. Вихідний алгоритм



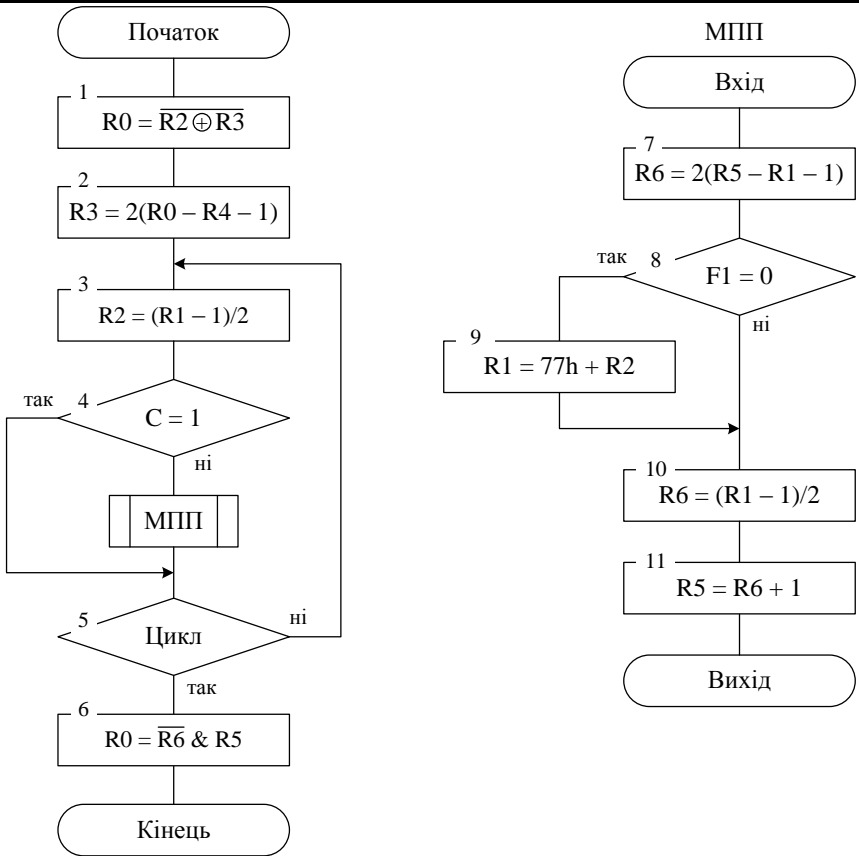
б

Продовження рис. 6.1



6

Продовження рис. 6.1



2

Продовження рис. 6.1

Лабораторна робота 1.3

ПРОГРАМНЕ ФОРМУВАННЯ ЧАСОВОЇ ЗАТРИМКИ В МК48

Мета роботи: Вивчення структури, режимів роботи, системи команд і отримання навиків розробки програм, що управляють, для мікроконтроллера КМ1816ВЕ48.

Теоретичні відомості: 1.2., 1.4

Додаткові теоретичні відомості

Формування часової затримки малої тривалості

Процедура реалізації часової затримки основана на методі програмних циклів. При цьому в довільний робочий регістр завантажуються число, яке в кожній ітерації циклу зменшується на 1. Так продовжується до тих пір, поки вміст робочого регістра не стане рівним нулю, що інтерпретується програмою як вихід з циклу. Час затримки визначається величиною, завантаженою в робочий регістр, і часом виконання команд, що утворюють програмний цикл.

Для отримання заданої часової затримки необхідно визначити значення X , яке завантажуються в робочий регістр. Обчислення значення X базується на часі виконання команд, що утворюють дану підпрограму. При цьому необхідно враховувати, що команди MOV і RET виконуються одноразово, а число повторень команди DJNZ дорівнює числу X . Крім того, звернення до підпрограми часової затримки здійснюється за командою CALL, час виконання якої також необхідно враховувати під час обчислення часової затримки. При тактовій частоті 6 МГц кожен машинний цикл виконується за 2,5 мкс. Під час завантаження в регістр R2 числа 17 реалізується часова затримка 100 мкс. В підпрограму можуть бути включені додаткові команди NOP, час виконання кожною з яких рівне 2,5 мкс.

Формування часової затримки великої тривалості

Недоліком програмного способу реалізації часової затримки є нерациональне використання ресурсів МК48. Під час формування затримки МК48 практично простоює, оскільки не може вирішувати ніяких задач управління об'єктом. В той же час апаратурні засоби МК48 дозволяють реалізувати часові затримки на фоні основної програми роботи.

Під час використання таймера можна отримати часові затримки тривалістю від 80 мкс до 20 мс. Наприклад, для реалізації часової затримки 240 мкс необхідно виконати наступні дії:

	MOV	A, #FDh	; завантаження в А (– 3)дк
	MOV	T, A	; завантаження таймера
	ORL	BUS, ...	; установка управляючого сигналу
	STRT	T	; запуск таймера
LL1:	JTF	LL2	; перевірка ознаки переповнення від T
	JMP	LL1;	; знаття управляючого сигналу
LL2:	ANL	BUS, ...	

Поява сигналу переривання від таймера відповідає закінченню часового інтервалу 240 мкс. Погрішність може складати 4-5 мкс.

Підготовка до виконання практичної роботи

1. Вивчити архітектуру і систему команд МК48.
2. Записати номер варіанту (номер залікової книжки) у двійковому поданні і виділити шість молодших розрядів $h_6 - h_1$. За отриманими значеннями двійкових розрядів вибрати варіант алгоритму управління з табл. 6.6. Вихідні алгоритми відповідно до варіантів надані на рис. 6.2. В операторні вершини заданого алгоритму управління записати управляючі сигнали, надані в табл. 6.7.

Таблиця 6.6. Варіанти завдань

h_2	h_1	Вихідний алгоритм
0	0	рис. 6.2, а
0	1	рис. 6.2, б
1	0	рис. 6.2, в
1	1	рис. 6.2, г

Таблиця 6.7. Варіанти завдань

h_2	h_3	Управляючі сигнали			
		Номер операторної вершини			
		1	2	3	4
0	0	$y_1 y_2$	$y_1 y_3$	$y_4 y_5$	y_3
0	1	y_5	$y_1 y_3 y_4$	y_1	$y_3 y_4$
1	0	$y_1 y_2 y_3$	y_2	$y_4 y_1$	y_4
1	1	$y_1 y_5$	y_2	$y_2 y_3 y_4$	$y_1 y_2 y_5$

3. Розробити програму управління для МК48, що реалізує заданий алгоритм. Для вводу/виводу даних використати порти, задані в табл. 6.8. Часові параметри управляючих сигналів задані в табл. 6.9.

4. Побудувати часові діаграми управляючих сигналів для всіх комбінацій значень логічних умов.

Таблиця 6.8. Варіанти завдань

h_2	h_4	Порт
0	0	<i>BUS</i>
0	1	<i>P1</i>
1	0	<i>P2</i>
1	1	<i>P4, P5</i>

Таблиця 6.9. Варіанти завдань

h_4	h_5	h_1	Часові затримки				
			Управляючі сигнали				
			y_1	y_2	y_3	y_4	y_5
0	0	0	720	25	400	10	800
0	0	1	560	12	700	330	280
0	1	0	100	725	15	22	45
0	1	1	40	240	100	50	150
1	0	0	18	280	80	720	60
1	0	1	400	35	800	180	12
1	1	0	560	45	20	150	22
1	1	1	222	15	720	500	560

Порядок виконання роботи

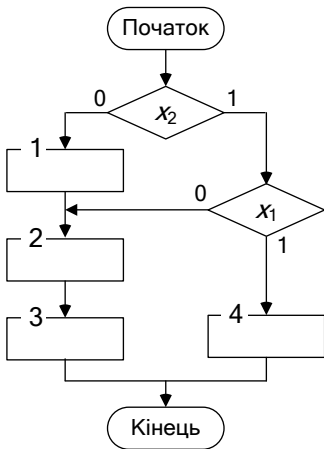
1. Використовуючи моделюючий комплекс *SCM MK48* налагодити розроблену програму:

– сформував початковий текст програми у вікні екранного редактору комплексу *SCM MK48*, виконати пошук синтаксичних помилок та скомпілювати програму;

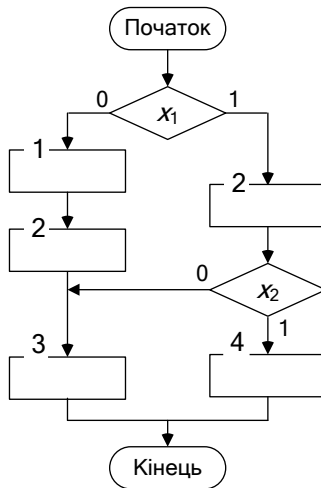
– налагодити розроблену програму пристрою управління за допомогою програмно-логічної моделі *SCM MK48* з отриманням повної інформації про хід виконання програми.

2. Отримати часові діаграми управляючих сигналів на виході заданого порту.

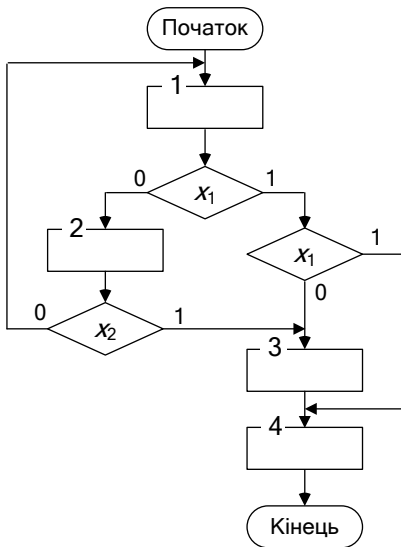
3. Зробити висновки.



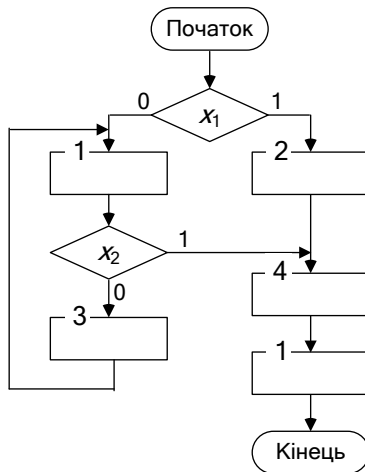
a



б



в



г

Рис. 6.2. Вихідні мікроалгоритми

Зміст звіту

Звіт повинен містити короткі теоретичні відомості, схему алгоритму, налагоджену програму на символічному асемблері, часові діаграми управляючих сигналів, висновки за роботою.

Контрольні питання

1. Охарактеризуйте основні групи системи команд МК48.
2. Наведіть формат команди МК48.
3. Які способи адресації даних застосовуються в МК48?
4. Наведіть модель програміста МК48.
5. Як здійснюється обмін даними з зовнішніми пристроями?
6. Як налаштувати виходи портів мікроконтролера на введення даних?
7. Розробіть програму формування управляючого сигналу заданої тривалості.
8. Як здійснити запуск таймера/лічильника в різних режимах?
9. Охарактеризуйте можливості додаткових портів МК48.
10. Приведіть приклади використання в програмі ознак користувача $F0$ і $F1$.
11. Накресліть структурну схему підключення до МК48 зовнішньої пам'яті програм і даних.
12. У чому полягає основна відмінність ознак, що входять до складу регістру PSW і ознак, що не входять до складу цього регістру?
13. У якій області пам'яті розміщуються підпрограми обслуговування переривань?
14. Вкажіть призначення кожного виходу мікросхеми мікроконтролера МК48.

Лабораторна робота 1.4

ВИКОНАННЯ АРИФМЕТИЧНИХ ОПЕРАЦІЙ В МІКРОКОНТРОЛЕРІ МК48

Мета роботи: Вивчення структури, режимів роботи, системи команд і отримання навиків розробки програм виконання операції множення в мікроконтролері МК48.

Теоретичні відомості: 1.2., 1.4

Підготовка до виконання практичної роботи

1. Записати номер варіанту (номер залікової книжки) у двійковому поданні і виділити шість молодших розрядів $h_6 - h_1$.

2. За отриманими значеннями двійкових розрядів вибрати спосіб множення та розрядність операндів з табл. 6.10. Машинні коди, в яких повинні бути подані операнди X та Y у вихідному стані, та результат Z наприкінці виконання обчислень, визначені в табл. 6.11, де ПК – прямий код, ДК – доповнювальний код.

3. Розробити операційну схему, цифрову діаграму стану регістрів, схему алгоритму виконання операції множення двійкових чисел $Z = X \times Y$ (де $X < 1, Y < 1$) у заданих машинних кодах. Значення операндів вибрати самостійно.

Таблиця 6.10. Варіанти завдання

h_6	h_5	h_4	Спосіб множення	Розрядність операндів
0	0	0	1	16
0	0	1	2	24
0	1	0	3	16
0	1	1	4	8
1	0	0	1	24
1	0	1	2	16
1	1	0	3	24
1	1	1	4	16

Таблиця 6.11. Варіанти завдання

h_5	h_4	h_3	X	Y	Z
0	0	0	ДК	ДК	ПК
0	0	1	ПК	ПК	ДК
0	1	0	ПК	ДК	ПК
0	1	1	ДК	ПК	ПК
1	0	0	ПК	ПК	ДК
1	0	1	ДК	ДК	ПК
1	1	0	ДК	ПК	ДК
1	1	1	ПК	ДК	ДК

4. Розробити програму реалізації операції множення на асемблері МК48. Порти для вводу/виводу даних, надані в табл. 6.12.

5. Визначити час виконання операції при опорній частоті $F=6\text{МГц}$.

Таблиця 6.12. Варіанти завдань

h_2	h_4	Порт
0	0	<i>P4, P5</i>
0	1	<i>BUS</i>
1	0	<i>P1</i>
1	1	<i>P2</i>

Порядок виконання роботи

1. Використовуючи моделюючий комплекс *SCM МК48* налагодити розроблену програму:

- сформувати початковий текст програми у вікні екранного редактору комплексу *SCM МК48*, виконати пошук синтаксичних помилок та скомпілювати програму;
- налагодити розроблену програму множення за допомогою програмно-логічної моделі *SCM МК48* з отриманням повної інформації про хід виконання програми.

2. 3. Зробити висновки.

Зміст звіту

Звіт повинен містити короткі теоретичні відомості, операційну схему, цифрову діаграму стану регістрів, схему алгоритму, налагоджену програму на символічному асемблері *МК48*, висновки за роботою.

Контрольні питання

1. Охарактеризуйте основні способи множення чисел.
2. Поясніть в яких випадках і як виконується округлення результату під час виконання арифметичних операцій?
3. Розробіть алгоритм та підпрограму виконання наступних арифметичних операцій у *МК48*:
 - додавання,
 - віднімання,
 - зсув шістнадцятирозрядних слів,
 - якщо операнди знаходяться у внутрішній пам'яті даних,

- у зовнішній пам'яті даних.
- 4. Поясніть як здійснити перевірку значення довільного розряду регістра
 - у внутрішній пам'яті даних,
 - у зовнішній пам'яті даних.

Лабораторна робота 1.5

ОБЧИСЛЕННЯ АРИФМЕТИЧНОЇ ФУНКЦІЇ В МІКРОКОНТРОЛЕРІ КМ1816BE48

Мета роботи: Вивчення структури, режимів роботи, системи команд і отримання навиків розробки програм виконання широкого набору арифметичних операцій в мікроконтролері КМ1816BE48.

Теоретичні відомості:

Підготовка до виконання практичної роботи

1. Записати номер варіанту (номер залікової книжки) у двійковому поданні і виділити шість молодших розрядів $h_6 - h_1$.
2. За отриманими значеннями двійкових розрядів вибрати спосіб ділення з табл. 6.13 та задану арифметичну функцію з табл. 6.14. Розрядність операндів прийняти рівній шістнадцяти. Машинні коди, в яких повинні бути подані операнди X та Y у вихідному стані, та результат Z наприкінці виконання обчислень, визначені в табл. 6.15, де ПК – прямий код, ДК – доповнювальний код.
3. Розробити операційну схему, цифрові діаграми стану регістрів, схему алгоритму обчислення заданої функції $F = (X1, X2)$ (де $X1 < 1, X2 < 1, X2 < X1, X1 \neq 0$). Значення операндів вибрати самостійно.

Таблиця 6.13. Варіанти завдання

h_1	Спосіб ділення
0	Ділення правильних дробів з відновленням залишку
1	Ділення правильних дробів без відновлення залишку

4. Розробити програму обчислення заданої функції на асемблері МК48. Порти для вводу/виводу даних, надані в табл. 6.16.

5. Визначити час виконання операції при опорній частоті $F=6\text{МГц}$.

Таблиця 6.14. Варіанти завдання

h_2	h_1	Арифметична функція
0	0	$F = X_2 / X_1 + 2(X_2 + X_1)$
0	1	$F = 4(X_2 + X_1) - X_2 / X_1$
1	0	$F = X_2 / X_1 + (X_2 + X_1) / 2$
1	1	$F = (X_2 - X_1) / 4 + X_2 / X_1$

Таблиця 6.15. Варіанти завдання

h_1	h_4	h_5	X_1	X_2	F
0	0	0	ДК	ДК	ПК
0	0	1	ПК	ПК	ДК
0	1	0	ПК	ДК	ПК
0	1	1	ДК	ПК	ПК
1	0	0	ПК	ПК	ДК
1	0	1	ДК	ДК	ПК
1	1	0	ДК	ПК	ДК
1	1	1	ПК	ДК	ДК

Таблиця 6.16. Варіанти завдань

h_2	h_4	Порт
0	0	P_2
0	1	P_4, P_5
1	0	BUS
1	1	P_1

Порядок виконання роботи

1. Використовуючи моделюючий комплекс *SCM МК48* налагодити розроблену програму:

– сформувати початковий текст програми у вікні екранного редактору комплексу *SCM МК48*, виконати пошук синтаксичних помилок та скомпілювати програму;

– налагодити розроблену програму пристрою управління за допомогою програмно-логічної моделі *SCM* МК48 з отриманням повної інформації про хід виконання програми.

2. Зробити висновки.

Контрольні питання

1. Охарактеризуйте основні методи ділення чисел.
2. Поясніть в яких випадках і як виконується округлення результату під час виконання операції ділення?
3. Поясніть які обмеження накладаються на операнди під час ділення чисел з фіксованою комою?
4. Поясніть, як організувати цикл
 - з виходом із циклу за умовою,
 - з виходом із циклу за заданій кількості проходів.
5. Яким чином здійснити запис результату ділення в задану сторінку зовнішньої пам'яті даних.
6. Намалювати структурну схему підключення до МК48 п'яти сторінок ЗПД.