**MITTEPATTYPA** HAJBYLA AJBJHO-MUETO AJBY Тернопільський національний технічний університет імені Івана Пулюя

Кафедра автоматизації технологічних процесів і виробництв

Лабораторна робота № 7 з курсу "Мікропроцесорні та програмні засоби автоматизації"

Ввід-вивід даних через порти PIC-контролера

Тернопіль 2018

Методичні вказівки до лабораторної роботи №7 "Ввід-вивід даних через порти РІС-контролера" з курсу "Мікропроцесорні та програмні засоби автоматизації" Медвідь В.Р., Пісьціо В.П., - Тернопіль: ТНТУ, 2018 - 14 с.

Відповідальні за випуск

доцент, к.т.н. Медвідь В.Р., асистент Пісьціо В.П.

Для студентів напряму: 151 "Автоматизація та комп'ютерно-інтегровані технології"

#### Лабораторна робота №7

Вивід/вивід даних через порти PIC-контролера на програмному симуляторі PIC Simulator IDE

# 1. Послідовність роботи з програмним симулятором PIC Simulator IDE

Основне вікно програми PIC Simulator IDE має вигляд, показаний на (рис. 1).

File Simulation R	ate Tools Opti	ons Help	U				
Program Location	C:\Prog	ram Files\PIC	Simulator IDE	\demo3.	hex	2	_
Microcontroller	PIC16F877A	3				$\mathbf{}$	
Last Instruction		Next	Instruction -				
GOTO	0x002A	Г	DECR	SZ 0x0	24.F		1
Program Counter and	Working Register	[	Instructions	Counter		262	
PC 002A			Clock Cycles	Counter	[	1536	
W Register			Real Time D	Juration	3	84.00	15
		······					
Special Function Reg	isters (SFRs)		General	Purpose	Registe	ers (GF	PRs
Address and Name	Hex Binary Value 7654	Value 4	Addr.	Hex Value	Addr.	Hex Value	C
001b TMR0			020b	02	0306	00	
002h PCL			021h	00	031h	00	1.1
003h STATUS	18		022h	OB	032h	00	
004h FSR			023h	00	033h	00	
005h PORTA			024h	OC T	034h	00	
006h PORTB			025h	00	035h	00	
007h PORTC			026h	00	036h	00	
008h PORTD	00 [		027h	00	037h	00	
009h PORTE	00		028h	00	038h	00	
00Ah PCLATH	00		029h	00	039h	00	
00Bh INTCON	00 [[[[[		02Ah	00	03Ah	00	
00Ch PIR1	00 [[[[		028h	00	03Bh	00	
00Dh PIR2			02Ch	00	03Ch	00	
OOFD TMR1			02Dh	00	03Dh	00	
Participant investig			02Eb	00	03Fh	00	
00Rh TMR1H	00 1 1 1 1		0201	00			

Рис. 1. Основне вікно програми PIC Simulator IDE

У верхній частині знаходяться меню, через які можна отримати доступ до основних і додаткових модулів програми (поз. 1)( рис. 1).

В рядку Program Location вказано шлях до обраної програми і її ім'я (поз. 2).

В рядку Microcontrollers, відображається тип обраного мікроконтролера (поз. 3).

У нижній частині вікна є дві панелі (поз.4 і поз.5), в яких відображаються стан програми, вміст регістрів спеціальних функцій (РСФ) і керуючих регістрів обраного МК.

#### Послідовність роботи з програмним симулятором наступний:

• запуск програми PIC Simulator IDE;

• вибір типу мікроконтролера, для якого написана програма;

• вибір частоти кварцового генератора (впливає тільки на відображувані програмою дані про час виконання програми або команди, але не на швидкість роботи програми, що налагоджуються в PIC Simulator IDE);

• завантаження програми у вигляді НЕХ-файлу або запуск вбудованого компілятора мови асемблера і написання в ньому потрібної програми;

- вибір потрібних модулів віртуальних пристроїв;
- вибір швидкості і режиму роботи програми симулятора;
- запуск процесу симуляції роботи програми на обраному МК.

Якщо потрібно скористатися для роботи з симулятором власною програмою або внести зміни у вже розроблену, необхідно створити або завантажити для цього файл асемблера, з якого після компіляції буде створений необхідний для роботи з симулятором hex-файл.

S PIC Simulator IDI						12.5			S Asser	nbler - l	JNTITLED	.33	🛛
									File Edit	Tools	Options		
Program Location		C:\Program Files\F	PIC Sin	nulator ID	E\rb0int	hex			0001				•
Microcontroller	PIC16F84	Clock Frequ	iency	4.	0 MHz								
Last Instruction		Ne	ext Inst	ruction									
- Program Counter and	W Register		Ins	tructions	Counter		0	_					
PC 0000			Clo	ck Cycle	s Counte	1	0	_					
W Register			Re	eal Time I	Duration		0.00 µ	s					
Coosial Eurotian Dea	istors (CEDs)			Gener	Durner	o Dogio	hara (Gl						
special Function neg	isteis (orns)			Genera	irupos	e negis	ters (ur	-nsj					
Address and Name	Hex I Value 76	BinaryValue 5543210		Addr.	Hex Value	Addr.	Hex Value						
001h TMR0	00			00Ch	00	01Ch	00	-					
002h PCL	00			00Dh	00	01Dh	00						•
003h STATUS	18			00Eh	00	01Eh	00		1 0.10				) 
004h FSR	00			00Fh	00	01Fh	00		Lin I, LOIL			r	ium or lines:
005h PORTA	00			010h	00	020h	00						
006h PORTB	00			011h	00	021h	00		0001				-
008h EEDATA	00			012h	00	022h	00						
009h EEADR	00			013h	00	023h	00						
00Ah PCLATH	00			014h	00	024h	00						_
00Bh INTCON	00			015h	00	025h	00						•
081h OPTION_REG	FF			016h	00	026h	00		1.1010				· · ·
085h TRISA	1F			017h	00	027h	00		Lin T, Col U			1	ium or lines:
086h TRÍSB	FF			018h	00	028h	00						
088h EECON1	00			019h	00	029h	00						
TMR0 Prescaler	00			01Ah	00	02Ah	00						
1			7	01Bh	00	02Bh	00	-					

Рис. 2 Вікно симулятора з відкритим вікном Assembler

Для цього:

**1.** Натиснути Options | Assembler. Відкриється вікно компілятора Assembler – UNTITLED (рис. 2);

**2.** У вікні Assembler натиснути опцію File. Розкриється закладка (рис. 3), з якої для створення нового файлу потрібно натиснути New, а для завантаження вже створеного – OPEN.

S PIC Simulator IDE							- [		S Assen	nbler -	UNTITLED		🛛
									File Edit	Tools	Options		
Program Location		C:\Program Files\Pl	IC Simula	ator IDE	: Stb0int.l	hex			New	0	Ctrl+N		
Microcontroller	PIC16F84	Clock Freque	ncy	4.0	MHz				Save	(	Ctrl+S		
Last Instruction		Nex	t Instruct	tion					Save A	s	carro		
									Print So	ource (	Ctrl+P		
Program Counter and V	W Register -		Instru	ctions (	Counter	-	0	_	Print Lis	sting			
			Clock	Cucles	Counter		0	_					
W Begister			Beal	Time D	uration		0.00.110	-					
, in riogicion	100		, noa	111100	aradon		0.00 p0						
- Special Function Regi	sters (SFRs)		G	ieneral	Purpose	e Regis	ters (GP	Rs)					
Address and Name	Hex Value 7.6	BinaryValue 6 5 4 3 2 1 0	۵	Addr.	Hex Value	Addr.	Hex Value						
001h TMB0				OOCh	00	01Ch	00	•					
002h PCL				DODh	00	01Dh	00						-
003h STATUS	18		(	00Eh	00	01Eh	00		4				> No
004h FSR	00		(	00Fh	00	01Fh	00		LIN I, LOI U				Num or lines: 1
005h PORTA	00		(	010h	00	020h	00						
006h PORTB	00		(	011h	00	021h	00		0001				-
008h EEDATA	00		(	012h	00	022h	00						
009h EEADR	00		(	013h	00	023h	00						
00Ah PCLATH	00		(	014h	00	024h	00						
00Bh INTCON	00		(	015h	00	025h	00						-
081h OPTION_REG	FF		(	016h	00	026h	00		4				<i>•</i>
085h TRISA	1F			017h	00	027h	00		Lin T, Col U			 	Num of lines: 1
086h TRISB	FF			018h	00	028h	00						
088h EECON1	00			019h	00	029h	00						
TMR0 Prescaler	00			01Ah	00	02Ah	00						
				U1Bh	00	028h	00	-					

Рис. З Завантаження існуючого або створення нового файлу асемблера

**3.** Після вибору і завантаження файлу (наприклад, rb0int.asm), його текст з'явиться у вікні Assembler (рис. 4).



Рис. 4 Завантаження файлу rb0int.asm

4. Для компіляції створеного або завантаженого і потім зміненого файлу, натисніть Tools і у вікні, що розкриється – Assemble. В нижній половині вікна Assembler з'явиться лістинг відкомпільованого файлу і, одночасно, при відсутності помилок, буде створений одноіменний hex-файл.

S PIC Simulator IDE				100	-		S Ass	eml	bler - rl	b0int.asm			🛛
File Simulation Rate							File E	dit	Tools	Options			
Program Location Microcontroller	C:\Program Files\P PIC16F84 Clock Freque	C Simulator ID	E\rb0int 0 MHz	.hex		_	0001 0002	;	Asse Asse	mble mble & Load	F7 F8		-
Last Instruction	Ne	t Instruction					0003 0004 0005		Asse	R1L EQU R1H EQU	OXE OXF		
Program Counter and		Instructions Clock Cycle:	Counter s Counte	r	0		0006			R2L EQU R2H EQU R3L EQU	0x10 0x11 0x12		
W Register	sters (SFRs)	Real Time I	Duration Il Purpos	e Regisl	0.00 ps ters (GP	'Rs) –	0010			R3H EQU R4L EQU R4H EQU	0x13 0x14 0x15 0x16		
Address and Name 001h TMR0	Hex Binary Value Value 7 6 5 4 3 2 1 0	Addr.	Hex Value	Addr.	Hex Value	-	0012			R5L EQU R5H EQU W_TEMP E	Ox15 Ox17 QU Ox4F	0.47	
002h PCL 003h STATUS 004h FSR		00Dh 00Eh 00Fh	00	01Dh 01Eh 01Fh	00 00 00		Lin 1, Co	∣ I O lo		STATUS_T		UX4E	Num of lines: 60
005h PORTA 006h PORTB 008h EEDATA		010h 011h 012h	00 00 00	020h 021h 022h	00 00 00		0001	 PI(	C ASS	EMBLER LI	STING		
009h EEADR 00Ah PCLATH 00Bh INTCON		013h 014h 015h	00 00 00	023h 024h 025h	00 00 00	Γ	0003 0004 0005	Li1  000	ne  01	Address  0000	Opcode 	Instruction ; Begin	•
081h OPTION_REG 085h TRISA 086h TRISB	FF 1F FF	016h 017h 018h	00 00 00	026h 027h 028h	00 00 00		Lin 5, Co	↓   0 lo		Ð		]	▶ Num of lines: 66
088h EECON1 TMR0 Prescaler		019h 01Ah 01Bh	00	029h 02Ah 02Bh	00 00 00	•							

Рис. 5

2. Завдання на лабораторну роботу: ввід-вивід даних через порти РІС-контролера.

1. Вивчити програмну модель PIC Simulator IDE.

2. Вивчити команди програмування та обміну даними через порти РІС – контролера.

3. Дослідити роботу програми з Прикладу 1 в режимі роботи стимулятора «Normal" та вміст регістрів контролера (W, STATUS...), які використовуються при виконанні цієї програми, в покроковому режимі роботи.

4. Записати для вибраних команд асемблера коментар щодо їх призначення (див. Приклад 1).

### Приклад 1

Програма забезпечує ввід даних з чотирьох молодших розрядів порту В, зсув їх на чотири розряди вліво і вивід через чотири старші розряди цього ж порту.

Текст програми з файлу «Input\output.asm» має наступний вигляд:

; Begin

ORG 0x0000 BCF PCLATH,3 BCF PCLATH,4 BSF STATUS,RP0 MOVLW 0x80 MOVWF OPTION\_REG MOVLW 0x0F MOVWF TRISB BCF STATUS,RP0

SHIFTING:

SWAPF PORTB,W MOVWF PORTB GOTO SHIFTING END

#### 3. Послідовність роботи з симулятором при виконанні програми

Виконаємо цю програму в PIC Simulator ID, для чого необхідно:

1. Запустити PIC Simulator IDE;

2. Натиснути Options | Select Microcontroller;

3. Вибрати PIC16F84 і натиснути кнопку Select;

4. Натиснути Tools і у вікні, що розкриється, вибрати «Assembler». Відкриється вікно компілятора «Assembler – UNTITLED» (рис. 2);

5. Набрати текст програми Прикладу 1 у вікні «Assembler»;

6. Натиснути Tools і у вікні, що розкриється – Assemble. В нижній половині вікна Assembler з'явиться лістинг відкомпільованого файлу (рис. 6);

7. Одночасно, при відсутності помилок, буде створений файл «Input\output.hex», для якого можна вибрати ім'я та шлях для запису. Записати його на «Робочий стіл» комп'ютера;

8. Вибрати File | Load Program і завантажити створений файл «Input/output.hex»;

9. Натиснути Tools | 8 х LED Board. Відкриється вікно з панеллю, що містить вісім світлодіодів (рис. 7);

S PIC Simulator ID				< S A5	sembler - 115.asm	×
File Simulation Rate				File E		
Program Location Microcontroller	C:\Documents and Settin PIC16F84A Clock Freque	gs\home\Рабочий стол ency 4.0 MHz	\\115.hex	0001	ORG 0x0000 BCF PCLATH,3	<u>•</u>
Last Instruction	Ne	xt Instruction	_	0003	BCF FCLAIR,4 BSF STATUS,RPO MOVLW 0x80	
Program Counter and PC 0000 W Register	W Register	Instructions Counter Clock Cycles Counter Real Time Duration	0 0 0.00 µs	0008	NOVUF OFIION_REG NOVUW OXOF NOVWF TRISB BCF STATUS,RPO SHIFTING:	
Special Function Reg Address and Name 001h TMR0 002h PCL	isters (SFRs) Hex Binary Value Value 7 6 5 4 3 2 1 0 00	General Purpose Hex Addr. Value 00Ch 00 00Dh 00	Registers (GPRs) Hex Addr. Value 01Ch 00 ^ 01Dh 00	0011 0012 0013 0014	SWAPF PORTB, W MOVWF PORTB GOTO SHIFTING END	
003h STATUS 004h FSR 005h PORTA 006h PORTB 008h EEDATA	18           00           00           00           00           00	00Eh 00 00Fh 00 010h 00 011h 00 012h 00	01Eh 00 01Fh 00 020h 00 021h 00 022h 00	Lin 4, C	د ol 1	Num of lines: 14
009h EEADR 00Ah PCLATH 00Bh INTCON 081h OPTION_REG 085h TRISA 086h TRISB 088h EECON1 TMB0 Pressaler	00 00 00 FF FF FF FF FF 00 00	013h 00 014h 00 015h 00 016h 00 017h 00 018h 00 019h 00 019h 00	023h 00 024h 00 025h 00 026h 00 027h 00 028h 00 029h 00 029h 00	0015 0016 0017 0018 0019 0020 0021	0011 0008 0E06 0012 0009 0086 0013 000A 2808 0014 000B Number of errors = 0	SWAPF PORTB MOVWF PORTB GOTO SHIFTI END
		01Bh 00	02Bh 00 •	Lin 21, 0	< Col 0	Num of lines: 21

Рис. 6 Вигляд інтерфейсу симулятора з програмою «Input\output»

S 8 x LED Board - X S PIC Simulator IDE - I X Assembler - 115.asm	X
File Simulation Rate Tools Options Help File Edit Tools Options	
PORTB, 0 Program Location C:\Documents and Settings\home\Pa6owuki cron\115.hex 0001 ORG 0x0000	<u> </u>
PIBTE 1 Microcontroller PIC16F84A Clock Frequency 4.0 MHz 0002 BCF PCLATH, 3	
Last Instruction Next Instruction 0004 BSF STATUS, PP0	
PORTB.2 0005 MOVLW 0x80	
0006 NOVWF OPTION REG	
POBTB.3 Program Counter and W Hegister Instructions Counter 0 0007 MOVLW 0x0F	
PC 0000 Clock Cycles Counter 0 00008 MOVUF TRISB	
PORTB, 4 W Register 00 Real Time Duration 0.00 µs	
COLLER (C. D. C. L. C. L	
PORTB, 5 Special Function Registers (SFRs) General Purpose Registers (GPRs) 0012 NOVUE PORTB	
Hex Binary Value 7 A 3 2 1 0 Hex Hex 0013 GOTO SHIFTING	
PORTB, 6 00 0014 END	
PORTE, 7 B 003h STATUS 18 000H 00 01Eh 00	
004h FSB 00 00Fh 00 01Fh 00	
Always On Top Close 005h PORTA 00 010h 00 020h 00	•
006h PORTB 00 011h 00 021h 00 Uin 4 Col 1	Num of lines: 14
008h EEDATA 00 012h 00 022h 00	Humor mos. 14
009h EEADR 00 011 002h 00 0015 0011 0008 0E06	SWAPF PORTB
0048 PLATH 00 0149 00 025 00 0016 0012 0009 0086	MOVWF PORTB
001 001 00 000 000 000 000 000 0017 0013 000A 2808	GOTO SHIFTI
005h TRISA 1F 017h 00 027h 00 0018 0014 000B	END
086h TRISB FF 018h 00 028h 00 0019	
088h EECON1 00 019h 00 029h 00 0020 Number of errors = 0	_
TMR0 Prescaler 00 01Ah 00 02Ah 00 0021	· ·
	Num of lines: 21

Рис. 7 Вигляд інтерфейсу симулятора з програмою та панеллю «8 x LED Board»

10. Натиснути Tools | Microcontroller Viev PIC16F84. Відкриється вікно з виводами мікроконтролера (рис. 8);

11. У вікні «Select Pin» почергово натиснути поле «PORTB» і далі «0», після чого натиснути на поле «Select», яке розташоване внизу вікна. Таким чином, вибрано порт В та його вивід 0. Це повторити для всіх ліній вибраного порту;

12. Вибрати Rate | Normal;



Рис. 8 Вигляд інтерфейсу симулятора з програмою та панеллю «Microcontroller Viev PIC16F84»



Рис. 9 Налаштування виводів порту В

13. Натиснути Simulation | Start (почнеться виконання програми). Якщо при цьому курсором клацнути на одному з виводів мікросхеми (панель «Microcontroller Viev PIC16F84») за номером n (це відповідає появі на цьому виводі логічної «1» - світлодіод світиться), то засвітиться світлодіод на виводі мікросхеми з номером n+4 (рис. 10);

14. Щоб зупинити виконання програми, потрібно натиснути Simulation | Stop.

🚫 8 x LED Board 🖃 🗖 🗙	S PIC Simulator ID	E			-		S Ass	embler - :	115.asm		
	File Simulation Rate						File Ed				
PORTB, 0	Program Location	C:\Documents and	Settings\h	ome\Paбoчий сто	v/\115.hex		0001		ORG 01	x0000	•
	Microcontroller	PIC16F84A Clock	Frequency	4.0 MHz			0002		BCF PC	CLATH, 3	
	Last Instruction		Next Ins	struction			0003		BSF ST	TATUS.RPO	
PORTB, 2	MOVW	F PORTB		GOTO 0x	008		0005		MOVLW	0x80	
	Program Counter and	W Register			11	-	0006		MOVWF	OPTION_REG	
(C)) PORTB, 3				structions Counter		0	0007		MOVLW	TRICR	
ă	PC J UUUA			ock Cycles Counte	r 604	4	0008		BCE ST	TATUS PPO	
(( )) PORTB, 4	W Register		F	Real Time Duration	151.0	0 µs	0010	SHIFTI	VG:	TATOS, NO	
ž	Consist Eurotion Day	ristors (CEDs)		General Purpos	o Rogistoro (l	CPRA	0011		SWAPF	PORTB, W	
(C) PORTB, 5	special Function Reg	yisters (orns)		General Pulpos	e negisters (i	urnsj	0012		MOVUF	PORTB	
	Address and Name	Value 7 6 5 4 3 2 1	0	Addr Value	Addr Valu	x	0013		GOTO S	SHIFTING	
( PORTB, 6 📃							0014		END		
	UUTh TMRU	49		00Lh 00	01Dh 00						
PORTB, 7		19		00000 000	0156 00	-					
-	004h ESB		÷	00Eh 00	01Eh 00	-					_
Always On Ton Close	005h POBTA			010h 00	020h 00	-					•
	006h PORTB			011h 00	021h 00	-		4			· · · ·
	008h EEDATA			012h 00	022h 00	-	Lin 4, Co	11			Num of lines: 14
	009h EEADR			013h 00	023h 00		DOM 5	0044	0000	0807	-
	00Ah PCLATH			014h 00	024h 00		0015	0011	0008	0086	SWAPF PORTS
	00Bh INTCON			015h 00	025h 00		0017	0012	0003	2808	GOTO SHIFTI
	081h OPTION_REG	80		016h 00	026h 00		0017	0013	OOOR	2000	FND
	085h TRISA	1F		017h 00	027h 00		0019				
	086h TRISB	OF	_	018h 00	028h 00		0020	Number	of erro	ors = 0	
	088h EECON1	00	-	019h 00	029h 00	_	0021				-
	TMRU Prescaler	91		01Ah 00	02Ah 00			4		Ũ	,
				101Bh 100	028h   00	-	Lin 21, C	ol O			Num of lines: 21
		Sam DICICEDIA	_		3						
	S Microcontroller v	New - PILIbro4A									
	T OFF T OFF T OFF	RA2 1 18 RA1 RA3 2 17 RA0 TOCKI/RA4 3 16 0SC \MCLR 4 15 0SC Vss 5 14 Vdd	1/CLKIN 2/CLKOUT	OFF T OFF T							
	T ON T OFF T OFF T OFF	INT/RB0 6 13 RB7 RB1 7 12 RB6 RB2 8 11 RB5 RB3 9 10 RB4		OFF OFF OFF ON							
	Always On Top			Close							

Рис. 10 Зображення інтерфейсу стимулятора з виконуваною програмою

Для того, щоб мати змогу контролювати вміст регістрів після виконання стимулятором кожної команди, перейти на виконання програми в кроковому режимі роботи.

Для цього:

1. В основному вікні симулятора натиснути Rate | Step By Step, а далі вибрати опцію Simulation і натиснути Start. Симулятор готовий до виконання програми в кроковому режимі;

2. Для виконання наступної команди програми потрібно натиснути на закладку STEP, яка з'явиться справа від закладки HELP вгорі основного вікна симулятора після вибору крокового режиму його роботи.

Вміст регістрів контролера, які використовуються при виконанні команд програми, знайти в області perictpiв Adress and Name, яка розташована в лівій нижній частині основного вікна симулятора (виділені рожевим кольором). Всі регістри восьмирозрядні.

В процесі виконання програми по зміні кольору комірок видно, вміст яких регістрів змінюється. Забарвлення комірки відповідного розряду регістру помаранчевим кольором означає наявність "1", білим - "0".

# 4. Завдання для виконання лабораторної роботи Завдання 1

1. Виконати програму (Приклад 1) в режимі «Normal». В процесі виконання програми клацнути курсором на одному з виводів чотирьох молодших розрядів порту В мікросхеми (панель «Microcontroller Viev PIC16F84») за номером n=0...3 (це відповідає появі на цьому виводі логічної «1» - світлодіод світиться). При цьому засвітиться світлодіод на виводі мікросхеми з номером n+4 (рис. 10);

2. Виконати програму в кроковому режимі виконання програми. Вміст тих регістрів, значення яких змінюється в процесі виконання команд програми, записати в шістнадцятковому коді в табл.1.

Таблиця 1

Регістр	PC	W	TMR0	STATUS	PCL	TRISA	TRISB	PCLATH	EEADR	EEDATA	FSR
Команда 1											
Команда 2											
Команда п											

3. З програми Прикладу 1 вибрати десять команд і за таблицею команд асемблера для РІС-контролера (табл. 1) записати коментар щодо призначення цих команд (див. Приклад 2, де наведено такий запис для однієї команди).

# Приклад 2КомандаВиконувана операція (коментар)Код командиКомандаВиконувана операція (коментар)118АВСF РСLАТН, 3; скинути в "0" 3-ій біт регістра РСLАТН

і т.д.

# Завдання 2

1. Скласти програму, яка забезпечує ввід даних з **чотирьох старших розрядів порту В**, зсув їх на чотири розряди вправо і вивід через **чотири молодші розряди цього ж порту**.

2. Виконати програму в режимі «Normal».

3. В процесі виконання програми клацнути курсором на одному з виводів чотирьох старших розрядів порту В мікросхеми (панель «Microcontroller Viev PIC16F84») за номером n=4...7. При цьому повинен засвітитися світлодіод на виводі мікросхеми з номером n-4 (рис. 10);

4. Виконати програму в кроковому режимі виконання програми.

5. Вміст тих регістрів, значення яких змінюється в процесі виконання команд програми, записати в шістнадцятковому коді в табл. 2.

6. З виконуваної програми вибрати десять команд і за таблицею команд асемблера для РІС-контролера (таблиця 3) записати коментар щодо призначення цих команд (див. Приклад 2, де наведено такий запис для однієї команди).

# Завдання 3

1. Скласти програму, яка забезпечує ввід даних з чотирьох молодших розрядів порту А, та вивід через чотири молодші розряди порту В.

2. Виконати програму в режимі «Normal».

3. В процесі виконання програми клацнути курсором на одному з виводів чотирьох молодших розрядів порту A мікросхеми (панель «Microcontroller Viev PIC16F84») за номером n=0...3. При цьому повинен засвітитися світлодіод на виводі мікросхеми з номером n порту В (рис. 10);

4. Виконати програму в кроковому режимі виконання програми.

5. Вміст тих регістрів, значення яких змінюється в процесі виконання команд програми, записати в шістнадцятковому коді в табл. 2.

6. З виконуваної програми вибрати десять команд і за таблицею команд асемблера для РІС-контролера (таблиця 3) записати коментар щодо призначення цих команд (див. Приклад 2, де наведено такий запис для однієї команди).

# Завдання 4

Програма забезпечує ввід даних з чотирьох молодших розрядів порту А і вивід через чотири старші розряди порту В.

Текст програми має наступний вигляд:

; Begin

ORG 0x0000 BCF PCLATH,3 BCF PCLATH,4 BSF STATUS,RP0 MOVLW 0x80 MOVWF OPTION\_REG MOVLW 0x0F MOVWF TRISA MOVLW 0xF0 MOVWF TRISB BCF STATUS,RP0 SHIFTING: MOVF PORTA,W MOVWF PORTB GOTO SHIFTING

END

#### 5. Контрольні запитання

1. Будова РІС-контролерів серії РІС16Х8Х.

2. Призначення регістрів спеціальних функцій мікроконтролера.

3. Як запрограмувати порти мікроконтролера на ввід або вивід?

4. Формат регістра STATUS. Призначення бітів в процесі програмування портів мікроконтролера.

5. Які файли створюються в результаті компілювання програми на асемблері?

### 5. Література

1. Данилин А. Программа-симулятор PIC Simulator IDE / Данилин А. // Современная электроника. 2006.- №4. -С. 68-76.

2. Тавернье К. РІС-микроконтроллеры. Практика применения. М.: ДМК, 2002.

3. Предко М. Создайте робота своими руками на PIC- контроллере./ Майкл Предко; Пер. с английского Земского Ю.В. – М.: ДМК Пресс, 2006. – 408 с.: ил.

4. Кениг А. и М. Полное руководство по РІС-микроконтроллерам.: Пер. с нем.-К.: МК- Пресс", 2007.-256 с., ил.

Позначення	Функція	Цикли	Код команли	Біти стану	Прим.
ADDLW	Додавання константи і W	1	11111x	C, DC, Z	
ADDWF	Додавання W с f	1	00 0111 dfff ffff	C, DC, Z	1, 2
ANDLW	Логічне I константи і W	1	11 1001 kkkk kkkk	Ζ	
ANDWF	Логічне I W і f	1	00 0101 dfff ffff	Ζ	1, 2
BCF	Скидання біту в регістрі f	1	01 00bb bfff ffff		1, 2
BSF	Встановлення біту в регістрі f	1	01 01bb bfff ffff		1, 2
BTFSC	Пропустити команду, якщо біт у f дорівнює нулю	1 (2)	01 10bb bfff ffff		3
BTFSS	Пропустити команду, якщо біт у f дорівнює одиниці	1 (2)	01 11bb bfff ffff		3
CALL	Виклик підпрограми	2	10 0kkk kkkk kkkk		
CLRF	Скидання регістру f	1	00 0001 1 fff ffff	Ζ	2
CLRW	Скидання регістра W	1	00 0001 0xxx xxxx	Z	
CLRWDT	Скидання сторожового таймера WDT	1	00 0000 0110 0100		
COMF	Інверсія регістру f	1	00 1001 dfff ffff	Z	1, 2
DECF	Декремент регістру f	1	00 0011 dfff ffff	Z	1, 2
DECFSZ	Декремент f, пропустити команду, якщо 0	1 (2)	00 1011 dfff ffff		1, 2, 3
GOTO	Перехід за адресою	2	10 1kkk kkkk kkkk		
INCF	Інкремент регістру f	1	00 1010 dfff ffff	Z	1, 2
INCFSZ	Інкремент f, пропустити команду, якщо 0	1 (2)	00 1111 dfff ffff		1, 2, 3
IORLW	Логічне АБО константи і W	1	11 1000 kkkk kkkk	Z	
IORWF	Логічне АБО W і f	1	00 0100 dfff ffff	Z	1, 2
MOVF	Пересилання регістру f	1	00 1000 dfff ffff	Z	1, 2
MOVLW	Пересилання константи в W	1	11 00xx kkkk kkkk		
MOVWF	Пересилання W у f	1	00 0000 1 fff ffff		
NOP	Пуста команда	1	00 0000		

OPTION	Завантаження регістру ОРТІОN	1	00 0000		
RETFIE	Повернення з переривання	2	00 0000 0000 1001		
RETLW	Повернення з підпрограми з заванта- женням константи в W	2	11 01xx kkkk kkkk		
RETURN	Повернення з підпрограми	2	00 0000 0000 1000		
RLF	Зсув f вліво через перенесення	1	00 1101 dfff ffff	С	1, 2
RRF	Зсув f вправо через перенесення	1	00 1100 dfff ffff	С	1, 2
SLEEP	Перехід у режим SLEEP	1	00 0000 0110 0011		
SUBLW	Вирахування W з константи	1	11 110x kkkk kkkk	C, DC, Z	
SUBWF	Вирахування W з f	1	00 0010 dfff ffff	C, DC, Z	1, 2
SWAPF	Обмін місцями тетрад в f	1	00 1110 dfff ffff		1, 2
TRIS	Завантаження регістру TRIS	1	00 0000 0110 0fff		
XORLW	Виключаюче АБО константи і W	1	11 1010 kkkk kkkk	Z	
XORWF	Виключаюче АБО W i f	1	00 0110 dfff ffff		1, 2

\*Якщо в результаті виконання команди змінюється лічильник команд, або виконується перехід по перевірці умови, то команда виконується за два цикли. Другий цикл виконується як NOP.

# Додаток 1

# Завдання 3

Програма забезпечує ввід даних з чотирьох молодших розрядів порту А і вивід через чотири старші розряди порту В.

Текст програми має наступний вигляд:

# ; Begin

ORG 0x0000 BCF PCLATH,3 BCF PCLATH,4 BSF STATUS,RP0 MOVLW 0x80 MOVWF OPTION\_REG MOVLW 0x0F MOVWF TRISA MOVLW 0xF0 MOVWF TRISB BCF STATUS,RP0 SHIFTING:

MOVF PORTA,W MOVWF PORTB GOTO SHIFTING END