

Методичні вказівки для виконання лабораторної роботи №3 «Математичні операції (підпрограма множення) на програмному симуляторі PIC Simulator IDE» з курсу «Проектування мікропроцесорних систем керування технологічними процесами».

Методичні вказівки розглянуті і схвалені кафедрою «Автоматизація технологічних процесів та виробництв», протокол № 4 від 21.11.2016 р.

Відповідальні за випуск

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

Лабораторна робота №3

Математичні операції (підпрограма множення) на програмному симуляторі PIC Simulator IDE

1. Робота з програмним симулятором PIC Simulator IDE

Запустивши на виконання PIC Simulator IDE, побачимо основне вікно цієї програми (рис. 1).

File Simulation F	Rate Tools	Options H	elp	IJ				
Program Location	. (:\Program Files	PIC Sim	ulator IDE	demo	3.hex	2	
Microcontroller	PIC16F877A	3					$\mathbf{}$	
Last Instruction			Next Inst	ruction -				
GOTO	0x002A			DEC	FSZ Ox	024,F		l
Program Counter and	Working Regi	ster	Ins	tructions	Counter		262	
PC 002A			Co	ck Cycles	Counte	er 🗌	1536	-
W Register			R	al Time (Juration		84.00	s
001h TMR0 002h PCL 003h STATUS 004h FSR 005h PORTA 006h PORTB 007h PORTC 008h PORTD 009h PORTE 009h PORTE 004h PCLATH 008h INTCON	00 12A 18 100 00 100 00 100 00 100 00 100 00 100 00 100 00 100 00 100 00 100		•	020h 021h 022h 022h 022h 022h 022h 022h	02 00 00 00 00 00 00 00 00 00 00 00 00 0	030h 031h 032h 033h 034h 035h 036h 036h 038h 038h 039h 039h	88888888888888	121 June
00Ch PIR1 00Dh PIR2 00Eh TMR1L 00Eh TMR1H				02Bh 02Ch 02Dh 02Eh	00 00 00 00 00	03Bh 03Ch 03Dh 03Eh	00 00 00 00 00	

Рис. 1. Основне вікно програми PIC Simulator IDE

У верхній частині знаходяться різні меню, через які можна отримати доступ до різних основних і додаткових модулів програми (на рис. 1 позначено як «1»).

Далі, в рядку Program Location вказано шлях до обраної програми і її ім'я (на рис. 1 - «2»).

Нижче, в рядку Microcontrollers, відображається тип обраного мікроконтролера (на рис. 1 - «3»).

У нижній частині вікна є дві панелі (позначені як «4» і «5»). У них відображаються стан програми, вміст спеціальних 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.

Для цього:

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



Рис. 2 Вікно симулятора з відкритим вікном Assembler

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

S PIC Simulator ID	E				1.66			S Asse	mbler - UNTITLE	D		×
								File Edi	t Tools Options			
Program Location	C:\Program Files\PIC Simulator IDE\rb0int.hex							New	Ctrl+N			•
Microcontroller	PIC16F84	Clock Freque	ncy	4.0 MHz				Open	Ctrl+O			
Last Instruction		Nex	t Instructio	n				Save	As			
								Print 9	Source Ctrl+P			
Program Counter and	IW Register		Instructi	ions Counte	er	0		Print L	listing	1		
PC 0000			Clock Cy	cles Count	er	0						
W Register			Real Ti	me Duratio	n	0.00 p	s					
- Special Function Red	gisters (SFRs)		Ge	neral Purpo	ise Regis	ters (GF	PRsi=					
	Hex Bina	y Value		Hex	-	Hex	ŕ					
Address and Name	Value 7654	43210	Ad	dr. Value	Addr.	Value						
001h TMR0	00		00	Ch 00	01Ch	00	-					•
002h PCL	00		00	Dh 00	01Dh	00			4		+	
003h STATUS			00	En 00	01Eh			Lin 1, Col	0		Num of line:	s: 1
			00		0206	00					 	_
OOSH PORTA			01	1h 00	0201			0001				٠
008h FEDATA			01	2h 00	027h	00						
009h EEADB			01	3h 00	023h	00						
00Ah PCLATH			01	4h 00	024h	00						
00Bh INTCON			01	5h 00	025h	00						•
081h OPTION_REG	FF FF		01	6h 00	026h	00			4) F	
085h TRISA	1F		01	7h 00	027h	00		Lin 1, Col	0		Num of line:	s: 1
086h TRISB	FF		01	8h 00	028h	00						_
088h EECON1	00		01	9h 00	029h	00						
TMR0 Prescaler			01.	Ah 00	02Ah	00						
			01	Bh 00	02Bh	00	-					

Рис. 3

3. Після вибору і завантаження файлу (наприклад. rb0int.asm), його текст з'явиться в верхній половині вікна Assembler (рис. 4).

S PIC Simulator IDE					-		S Ass	embler - rb	0int.as	im.			🛛
							File Ed	dit Tools (Options				
Program Location	C:\Program Files\Pl	IC Simulator ID	E\rb0int.	hex			0001	; Begin					•
Microcontroller	PIC16F84 Clock Freque	ncy 4.	0 MHz				0002		ROL	EQU	OxC		
Last Instruction	Nex	t Instruction					0003		ROH D11	EQU	OXD OXF		
							0005		R1H	EOU	OxF		
							0006		R2L	EQU	0x10		
Program Counter and	W Register	Instructions	Counter		0		0007		R2H	EQU	0x11		
PC 0000		Clock Cycle:	: Counte	r I	0		0008		R3L	EQU	0x12		
W Register		Real Time I	Duration		0.00 µs	:	0009		R3H	EQU	0x13		
0 115 K D	(055.)				(0.5		0010		R4L D4W	EQU	UX14 0v15		
- Special Function Reg	isters (SFHs)	Genera	il Purpos	e Hegisi	ters (GF	'Hsj	0012		R5L	EOII	0x15 0x16		
Address and Name	Hex BinaryValue Value 76543210	Addr	Hex Value	Addr	Hex Value		0013		R5H	EQU	0x17		
				01.01	00		0014		W_TE	MP E	QU Ox4F		
		00Ch	00	01Dh	00	-	0015		STAT	us_т	EMP EQU Ox4	Ε	-
003h STATUS		OOEh	00	01Eh	00			•			•) F
004h FSR		00Fh	00	01Fh	00		Lin 1, Co	10					Num of lines: 60
005h PORTA		010h	00	020h	00								
006h PORTB	00	011h	00	021h	00		0001						-
008h EEDATA		012h	00	022h	00								
009h EEADR		013h	00	023h	00								
OOR6 INTCON		014h	00	024h									-
081h OPTION REG	FF.	016h	00	026h	00			4					
085h TRISA	1F	017h	00	027h	00		Lin 1, Co	10					Num of lines: 1
086h TRISB	FF FF	018h	00	028h	00								
088h EECON1	00	019h	00	029h	00								
TMR0 Prescaler		01Ah	00	02Ah	00								
		018h	00	02Bh	00	-							

Рис. 4 Завантаження файлу rb0int.asm

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

S PIC Simulator IDE				💌 🚫 Ass	emble	er - rb0int.asm	🛛
				File E	dit To	ools Options	
Program Location Microcontroller	C:\Program Files\Pl PIC16F84 Clock Freque	C Simulator IDE \rb0int.l ncy 4.0 MHz	hex	0001	;	Assemble F7 Assemble & Load F8 Assemble With MPASMWIN F9	
Last Instruction		0003		R1L EQU OXE R1H EQU OXF R2L EQU OXF	-		
Program Counter and		Instructions Counter Clock Cycles Counter	0			R2H EQU 0x11 R3L EQU 0x12 R3H EOU 0x13	
Special Function Regi	isters (SFRs)	General Purpose	e Registers (GPR	s) 0010 0011 0012		R4L EQU Ox14 R4H EQU Ox15 R5L EQU Ox16	
Address and Name 001h TMR0 002h PCL	Hex Binary Value Value 7 6 5 4 2 1 0 00 <t< td=""><td>Addr. Value</td><td>Addr. Value</td><td>0013 0014 0015</td><td></td><td>R5H EQU 0x17 W_TEMP EQU 0x4F STATUS_TEMP EQU</td><td>0x4E 💌</td></t<>	Addr. Value	Addr. Value	0013 0014 0015		R5H EQU 0x17 W_TEMP EQU 0x4F STATUS_TEMP EQU	0x4E 💌
003h STATUS 004h FSR 005h PORTA		00Eh 00 00Fh 00 010h 00	01Eh 00 01Fh 00 020h 00	Lin 1, Co	10	Φ	Num of lines: 60
006h PORTB 008h EEDATA 009h EEADR		011h 00 012h 00 013h 00	021h 00 022h 00 023h 00	0001 0002 0003	 PIC Line	ASSEMBLER LISTING Address Opcode	Instruction
00Ah PCLATH 00Bh INTCON 081h OPTION_REG	00 00 FF	014h 00 015h 00 016h 00	024h 00 025h 00 026h 00	0004	0001	L 0000	; Begin 🗸
085h TRISA 086h TRISB 088h EECON1	1F FF 00	017h 00 018h 00 019h 00	027h 00 028h 00 029h 00	Lin 5, Co	010		Num of lines: 66
TMR0 Prescaler		01Ah 00 01Bh 00	02Ah 00 02Bh 00	-			

2. Завдання на лабораторну роботу: математичні операції: підпрограма множення..

1. Вивчити програмну модель PIC Simulator IDE.

2. Вивчити команди арифметичних операцій РІС – контролера.

3. Написати і дослідити роботу програми з Прикладу 1 та дослідити вміст регістрів контролера (W, STATUS), які використовуються при виконанні цієї програми.

4. Користуючись вікном "PIC Disassembler", записати перші 7 команд виконуваної програми на Асемблері та в шістнадцяткових кодах.

5. Записати для вибраних команд коментар щодо їх призначення (див. Приклад 2).

Приклад 1.

Демонстрація роботи симулятора при виконанні арифметичних операцій. У цьому прикладі перемножуються два числа: 123 (шістнадцятковий 7В) і 234 (шістнадцятковий ЕА), отримаємо результат 28782 (шістнадцятковий 706Е).

Текст програми з файлу multiply.asm має наступний вигляд:

; Begin

R0L EQU 0xC R0H EQU 0xD R1L EQU 0xE R1H EOU 0xF R2L EQU 0x10 R2H EQU 0x11 R3L EOU 0x12 R3H EQU 0x13 R4L EOU 0x14 R4H EQU 0x15 R5L EQU 0x16 R5H EQU 0x17 ORG 0x0000 BCF PCLATH,3 BCF PCLATH,4 GOTO L0001 ORG 0x0004 RETFIE L0001:

; 1: Dim a As Word 'first number

- ; The address of 'a' is 0x18 a EOU 0x18
- ; 2: Dim b As Word 'second number
- ; The address of 'b' is 0x1A b EQU 0x1A
- ; 3: Dim x As Word 'result
- ; The address of 'x' is 0x1C x EQU 0x1C

; 4:

- ; 5: a = 123 'set first number MOVLW 0x7B MOVWF 0x18 CLRF 0x19
- ; 6: b = 234 'set second number MOVLW 0xEA MOVWF 0x1A

CLRF 0x1B ; 7: x = a * b 'calculate result MOVF 0x18,W MOVWF R3L MOVF 0x19,W MOVWF R3H MOVF 0x1A,W **MOVWF R1L** MOVF 0x1B.W MOVWF R1H CALL M001 MOVWF 0x1C MOVF R2H,W MOVWF 0x1D ; End of program L0002: GOTO L0002 ; Multiplication Routine M001: MOVLW 0x10 MOVWF R4L CLRF R0H CLRF R0L M002: RRF R3H,F RRF R3L,F BTFSS STATUS,C GOTO M003 MOVF R1L,W ADDWF R0L,F MOVF R1H,W BTFSC STATUS,C **INCFSZ R1H,W** ADDWF R0H,F M003: RRF R0H,F RRF R0L,F RRF R2H,F RRF R2L,F DECFSZ R4L,F GOTO M002 MOVF R2L,W RETURN ; End of listing

END

3. Послідовність роботи з симулятором при виконанні програми

Запустимо цю програму в PIC Simulator IDE і виконаємо наступне:

1. Запустити PIC Simulator IDE;

2. Натиснути Options | Select Microcontroller;

- 3. Вибрати PIC16F84 і натиснути кнопку Select;
- 4. Натиснути File | Load Program;

5. Вибрати файл multiply.hex і натиснути кнопку Open;

6. Натиснути Tools | Breakpoints Manager (відкриється вікно Breakpoints Manager);

7. Натиснути «Так», щоб використовувати існуючий асемблер, який видав файл;

8. Натиснути рядок, що відповідає адресі 0018, щоб встановити контрольну точку на цій команді;

9. Вибрати Hold PC In Focus option;

10. Вибрати Rate | Extremely Fast simulation rate;

11. Натиснути Simulation | Start (почнеться моделювання).

Коли ця математична підпрограма закінчиться, програма входить в нескінченний цикл за адресою 0018, але через контрольні точки PIC Simulator IDE автоматично переключиться в режим моделювання Step by step.

Роботу PIC Simulator IDE можна зупинити, натиснувши на Simulation | Stop, або продовжити її виконання, очистивши контрольну точку і натиснувши на Rate | Extremely.

Регістри 19 та 18 будуть містити перший параметр: 007В.

Регістри 1ВН і 1АН міститимуть другий параметр: 00ЕА. Результат 706Е знаходиться в регістрах 1DH і 1CH.

Вигляд екрану з виконуваною програмою показаний на рис. 6.

🕒 PIC D	isassembler				DIC Simulator ID	Ē						_	
File Opt	ions				File Simulation Rate	Tools	s Options He	p STEP	Run	n To Next BAS	IC Staten	nent	
				*	Program Location 🖌 1	Step By	y Step Ctrl+F		C Simu	ulator IDE\mu	tiply.hex		
PIC DI	SASSEMBLE	R LIST	ING		Microcontroller 2	Slow	Ctrl+F	gue	ency	4.0 MH	z		
Addres	s Opcode	Instru	uction		- Last Instruction 4	Normai Cash	Ctri+r	-3 Nev	et Instri	uction			
					East misu de dont 4	Fast		-4 146/	- msu	MOVIN	/ 0=10		-
0000	118A		BCF PCLATH, 3		5	Extrem	ely Fast Ctri+r	.5		MUYLY	OXIO		
0001	120A		BCF PCLATH, 4		Program Counter and	W Hec	e Ciri+r Jisici	0	Inst	tructions Cour	nter	18	
0002	2805		NOD						Clas	ak Cueles Cou	ntor	04	_
0003	0000		DETETE							IK Cycles Cou	riter	04	_
0004	307B	1.1.	MOVIN OV7B		W Register	00			He	al Time Durat	ion	21.001	15
0006	0098		MOVWF 0x018		- Special Eurotion Ber	ietore (C	SEBal			General Pur	nose Rea	ietore (G	PBel
0007	0199		CLRF 0x019		Special Function heg	15(615 (5	5rnsj			uenerarru	poserieg	sieis (u	rnsj
0008	30EA		MOVLU OXEA		Address and Name	Value	BinaryVa 2 7 6 5 4 3 3	lue 210		Addr Valu	k Je Addr	Value	
0009	009A		MOVWF Ox01A									00	
000A	019B		CLRF Ox01B			10				000Lh 00		00	-
000B	0818		MOVF 0x018,W		002h PLL	10					0154	00	
000C	0092		MOVWF 0x012			00	FFFFF				0156	00	
0000	0819		MOVF 0x019,W		004h PORTA	00				0106 00	0206	00	
OOOE	0093		MOVWF 0x013		006h PORTR	00				011h 00	0201	00	
OOOF	081A		MOVF OxO1A,W		008h FEDATA	00				012h 7B	022h	00	
0010	008E		MOVWF OxOOE		009h EEADB	00				013h 00	023h	00	
0011	081B		MOVF Ox01B,W		00Ah PCLATH	00				014h 00	024h	00	
0012	008F		MOVWF OxOOF		00Bh INTCON	00				015h 00	025h	00	
0013	2018		CALL L2		081h OPTION REG	FF				016h 00	026h	00	
0014	0090		MOVWF OXOIC		085h TRISA	1F				017h 00	027h	00	
0015	0811		MOVE UXU11,W		086h TRISB	FF				018h 7B	028h	00	
0018	2017	12.	MOVWF UXUID		088h EECON1	00				019h 00	029h	00	
0017	2017	12.	MOVIN OVIO		TMR0 Prescaler	00				01Ah EA	02Ah	00	
0010	0094	L2.	MOVINE OV014						-	01Bh 00	02Bh	00	-
0013	0180		CLRF OXOOD										
001B	018C		CLRF 0x00C		S Breakpoints Man	ager					1.00	-	
001C	0C93	L5:	RRF 0x013.F										
001D	0C92		RRF Ox012,F		Defined A	aares	ss opcode	Instr	uct	lon		_	
001E	1C03		BTFSS STATUS,C		breakpoints			; End	ot	program			
001F	2826		GOTO L4		1 0018 00	217	2817	L0002	:	GOTO LO	0002		
0020	080E		MOVF OxOOE,W		2	M 0	3010	; MUI	тірі	Incation	ROUTH	ne	
0021	078C		ADDWF OxOOC,F		3	<u>л</u> в мо	3010	MUUL:	0.4.7	MOVEW UX	άU		
0022	080F		MOVF OxOOF,W		4	719 719	0094	™ ~		- K4L DAU			
0023	1803		BTFSC STATUS,C		5	ла 11 в	0180		IDE				
0024	OFOF		INCFSZ OxOOF,W		6 01	11C	0_02	M002.		DDE D3H	F		
0025	078D		ADDWF OxOOD,F)1 n	00093	P1002.	RE 6	RKE KJU, RKE KJU,			
0026	OCSD	L4:	RRF OxOOD,F		8 0)1F	1003	B	TESS	S STATUS.	c		
0027	0080		RRF OXOOC,F		10 01)1F	2826	G	ото	M003	-		
0028	0091		RRF UXU11,F		0)20	080E	M	OVE	R1L,W			
0029	0090		REF UXUIU,F		0)21	078C	А	DDWF	F ROL,F			
0028	2810		GOTO LE		Clear All 0)22	080F	M	OVE	R1H,Ŵ			
0028	2010		MOVE DV010 M		Breakpoints)23	1803	В	TESC	STATUS,	C		
0020	0008		RETURN	-	Hold PC)24	0F0F	I	NCFS	SZ R1H,W			
0020	5000		NETONA		In Focus)25	078D	А	.DDWF	F ROH,F			
					0)26	0C8D	M003:		RRF ROH,	F		
					Always 00)27	0C8C	R	RF F	ROL,F			
					Un Top 00)28	0091	R	RF F	R2H,F			

Рис. 6 Вигляд екрану з програмою «Математичні операції: підпрограма множення »

З вікна "PIC Disassembler" вибираємо сім перших команд з їх шістнадцятковими кодами і знаходимо з таблиці кодів асемблера PIC-контролера коментар щодо призначення цих команд (див. Приклад 2, де наведено такий запис для однієї команди).

Приклад 2		
Код команди	Команда	Виконувана операція (коментар)
118A	BCF PCLATH, 3	; скинути в "0" 3-ій біт регістра РСLАТН

і т.д.

Вміст регістрів контролера, які використовуються при виконанні програми, знаходимо з області pericтрів Adress and Name, яка розташована в лівій нижній частині основного вікна симулятора (виділені рожевим кольором). Всі регістри возьмирозрядні.

В процесі виконання програми по зміні кольору комірок видно, вміст яких регістрів змінюється. Забарвлення комірки відповідного розряду регістру помаранчовим кольором означає наявність "1", білим - "0". Вміст регістрів записуємо в шістнадцятковому коді за Прикладом 3.

Приклад 3	
Регістр	Вміст регістра
PORTA	1F

і т. д.

4. Контрольні запитання

1. Призначення регістра W мікроконтролера.

2. Арифметичні команди мікроконтролера.

3. Формат регістра STATUS.

4. Призначення та позначення основних елементів програмної моделі мікроконтролера

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 РІС- контроллере./ Майкл Предко; Пер. с английского Земского Ю.В. – М.: ДМК Пресс, 2006. – 408 с.: ил. – (В помощь радиолюбителю).

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