МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ Тернопільський національний технічний університет імені Івана Пулюя

Кафедра автоматизації технологічних процесів та виробництв

Методичні вказівки для виконання лабораторної роботи №1 "Обробка зовнішніх переривань по входу RB0 / INT на програмному симуляторі PIC Simulator IDE" з курсу "Проектування мікропроцесорних систем керування технологічними процесами"

HIAB TANDHO-MIETOANT91H

AITEPATVPA

Тернопіль 2017

Методичні вказівки для виконання лабораторної роботи №1 «Обробка зовнішніх переривань по входу RB0/INT 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 №1

Обробка зовнішніх переривань по входу RB0 / INT 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 I	Rate Tools	Options	Help					
Program Location		C:\Program F	Files\PIC S	imulator IDE	E\demo:	3.hex	2	_
Microcontroller	PIC16F877/	3					-	
Last Instruction			- Next In	struction -				_
GOTO	0x002A			DEC	FSZ Ox	024.F		1
Program Counter and	Working Rej	gister		Instructions	Counter		262	_
PC 002A				lock Cycles	Counte	r 🔽	1536	_
W Register	TOD T			Real Time [Duration	3	84.00	s
Special Function Red	gisters (SFRs)			Genera	l Purpos	e Regist	ers (GF	PR
Address and Name	Hex Value 7	Binary Value 6 5 4 3 2 1	4	Addr.	Hex Value	Addr.	Hex Value	(
001h TMR0	00 1			020h	02	030h	00	2
002h PCL	2A 1			021h	00	031h	00	
003h STATUS	18			022h	OB	032h	00	Г
004h FSR	00			023h	00	033h	00	
005h PORTA	00			024h	OC	034h	00	
006h PORTB	00 [025h	00	035h	00	
007h PORTC	00 1			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	00			02Ch	00	03Ch	00	
OOEh TMR1L	00		0	02Dh	00	03Dh	00	
00Fh TMR1H	00			02Eh	00	03Eh	00	
				and the second s	_		A COLORED	

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 - «З»).

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 assembler 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);

S PIC Simulator ID	E				33	-		S Assen	nbler - l	UNTITLED		. 🗆 🖂
								File Edit	Tools	Options		
Program Location	C:\Program Files\PIC Simulator IDE\rb0int.hex					0001				-		
Microcontroller	PIC16F84	Clock Freque	ency 4.	0 MHz	_							
Last Instruction		Nex	xt Instruction									
Program Counter and	W Register		Instructions	Counter		0	_					
PC 0000			Clock Cycle:	s Countei		0						
W Register			Real Time I	Duration		0.00 µs						
Consid Exection Day	-istan (CEDs)		Course		Denis	(CD						
Special Function Reg	gisters (SFHs)		Genera	I Purpos	e negist	ers (GF	nsj					
Address and Name	Hex Binar Value 7654	3210	Addr.	Hex Value	Addr.	Hex Value						
001h TMR0			00Ch	00	01Ch	00						_
002h PCL			00Dh	00	01Dh	00						•
003h STATUS	18		00Eh	00	01Eh	00		1.1 0.10				
004h FSR			00Fh	00	01Fh	00		Lin 1, Col U			Num	of lines: 1
005h PORTA			010h	00	020h	00						
006h PORTB			011h	00	021h	00		0001				-
008h EEDATA			012h	00	022h	00						
009h EEADR			013h	00	023h	00						
00Ah PCLATH			014h	00	024h	00						_
00Bh INTCON	00		015h	00	025h	00						-
081h OPTION_REG	FF F		016h	00	026h	00		4				+
085h TRISA	1F		017h	00	027h	00		Lin 1, Col 0			Num	of lines: 1
086h TRISB	FF F		018h	00	028h	00						
088h EECON1	00		019h	00	029h	00						
TMR0 Prescaler			01Ah	00	02Ah	00						
			01Bh	00	02Bh	00	•					

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 IDI					1.5			S Asse	mbler -	UNTITLED		-	
File Simulation Rate								File Edi	t Tools	Options			
Program Location	C:*	Program Files\PI	C Simulator II	Simulator IDE\rb0int.hex				New	(Etrl+N			-
Microcontroller	PIC16F84	Clock Frequer	ncy 4	0 MHz				Save	(
- Last Instruction		Next	Instruction -					Save	As				
								Print S	Source (Etrl+P			
Program Counter and	W/ Desister		1	. .	_		_	Print L	isting				
Flogram Counter and	wheyister		Instruction	s Lounte		U							
PC 0000			Clock Cycle	es Counte	er	0							
W Register	00		Real Time	Duration		0.00 ps	8						
- Special Function Beg	isters (SEBs)		Gener	al Purno	se Benis	ters (GE	Bsla						
opooldin anotorining	Lou Din	anu) (alua	Gono	Lou	oonogio	Lou	,						
Address and Name	Value 765	43210	Addr.	Value	Addr.	Value							
001h TMR0			00Ch	00	01Ch	00							_
002h PCL			00Dh	00	01Dh	00							•
003h STATUS	18		00Eh	00	01Eh	00		Lin 1. Call	•			Maria	• • • • • • • •
004h FSR	00		00Fh	00	01Fh	00		LIN I, COI	U			Num	or lines: I
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				+
085h TRISA	1F		017h	00	027h	00		Lin 1, Col	0			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							
			01Bh	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			185	- • ×	S Asser	mbler - rb0int.a	sm	.33	🗙
File Simulation Rate					File Edit	Tools Options	;		
Program Location	C:\Program Files\P	IC Simulator IDE\rb0int.	C Simulator IDE\rb0int.hex						•
Microcontroller	PIC16F84 Clock Freque	ncy 4.0 MHz			0002	ROL	EQU OxC		
Last Instruction	Nex	t Instruction			0003	ROH	EQU OXD		
				_	0004	R1H	EQU OXE FOU OXE		
					0006	R2L	EQU 0x10		
Program Counter and	W Register	Instructions Counter		0	0007	R2H	EQU Ox11		
PC 0000		Clock Cycles Counte	r	0	0008	R3L	EQU Ox12		
W Register		Real Time Duration	0.1	00 µs	0009	R3H	EQU Ox13		
Linesson					0010	R4L	EQU Ox14		
 Special Function Regi 	isters (SFRs)	General Purpos	e Register	s (GPRs)	0011	R4H	EQU Ox15		
	Hex Binary Value	Hex		łex	0012	RSL	EQU UX16		
Address and Name	Value 76543210	Addr. Value	Addr. V	alue	0013	коп	MD FOU OVAR		
001h TMR0		00Ch 00	01Ch	00 🔺	0015	STAT	TUS TEMP EOU O	X4E	
002h PCL	00	00Dh 00	01Dh	00					
003h STATUS	18	00Eh 00	01Eh	00	Lin 1. Col 0			N	um of lines: 60
004h FSR		00Fh 00	U1Fh	00					
UUSh PURTA		010h 00	020h	00	0001				•
		012h 00	0216	00					
009h EEADB		013h 00	022h						
00Ah PCLATH		014h 00	024h	00					
00Bh INTCON		015h 00	025h	00					-
081h OPTION_REG	FF FF	016h 00	026h	00					•
085h TRISA	1F	017h 00	027h	00	Lin 1, Col 0)		١	lum of lines: 1
086h TRISB	FF	018h 00	028h	00					
088h EECON1	00	019h 00	029h	00					
TMR0 Prescaler		01Ah 00	02Ah	00					
		01Bh 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				🗵 🚫 Assembler - rb0int.asm	🛛
				File Edit Tools Options	
Program Location Microcontroller Last Instruction	C:\Program Files\F PIC16F84 Clock Freque Ne	1C Simulator IDE \rb0int.1 ency 4.0 MHz xt Instruction	hex	0001 Assemble F7 0002 Assemble & Load F8 0003 Assemble With MPASMWIN F9 0004 B1L F0L 0XE	-
Program Counter and	W Register	Instructions Counter	0	0005 R1H EQU 0XF 0006 R2L EQU 0X10 0007 R2H EQU 0X11	_
PC 0000 W Register		Clock Cycles Counter Real Time Duration	0 0.00 µs	0008 R3L EQU 0x12 0009 R3H EQU 0x13 0010 R4L EQU 0x14 0011 R4L EQU 0x14	
Address and Name	isters (SFRs) Hex Binany Value Value 76543210	General Purpose Hex Addr. Value	e Registers (GPRs) – Hex Addr. Value	0011 R4H EQU 0x15 0012 R5L EQU 0x16 0013 R5H EQU 0x17 0014 W TEMP FOU 0x4F	
001h TMR0 002h PCL 003h STATUS		00Ch 00 00Dh 00 00Eh 00	01Ch 00 01Dh 00 01Eh 00 01Fl 00	0015 STATUS_TEMP EQU 0x4E	Num of lines: 60
004h FSH 005h PORTA 006h PORTB		00Fh 00 010h 00 011h 00	020h 00 021h 00 022h 00	0001 0002 PIC ASSEMBLER LISTING	·
009h EEADR 009h EEADR 00Ah PCLATH		012h 00 013h 00 014h 00 015h 00	022h 00 023h 00 024h 00 025h 00	- 0003 Line Address Opcode Instruction 0004 0005 0001 0000 ; Begin	
081h OPTION_REG 085h TRISA 086h TRISB	FF 1F FF	016h 00 017h 00 018h 00	026h 00 027h 00 028h 00	4 Lin 5, Col 0	Num of lines: 66
088h EECON1 TMR0 Prescaler		019h 00 01Ah 00 01Bh 00	029h 00 02Ah 00 02Bh 00	2	

Рис. 5

2. Завдання на лабораторну роботу: обробка зовнішніх переривань мікроконтролера по входу RB0 / INT.

1. Вивчити програмну модель PIC Simulator IDE.

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

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

4. Користуючись вікном "Program Memory Editor" 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

Ця програма змінює значення порту PORTA при зміні стану входу RB0 / INT по фронту імпульсу.

Текст програми з файлу rb0int.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 W TEMP EQU 0x4F STATUS TEMP EQU 0x4E ORG 0x0000 BCF PCLATH,3 BCF PCLATH,4 GOTO L0001 ORG 0x0004 MOVWF W TEMP SWAPF STATUS,W **CLRF STATUS** MOVWF STATUS_TEMP CALL L0002 SWAPF STATUS TEMP,W MOVWF STATUS SWAPF W TEMP,F SWAPF W TEMP,W RETFIE L0001 ; 1: TRISA = 0x00 'set all PORTA pins as outputs BSF STATUS, RP0 CLRF 0x05 BCF STATUS, RP0 ; 2: PORTA = 0xff 'make all PORTA pins high MOVLW 0xFF

MOVWF 0x05

- ; 3: INTCON.INTE = 1 'enable RB0/INT interrupts
- BSF 0x0B,4
- ; 4: INTCON.GIE = 1 'enable all un-masked interrupts PGE = 0.007
- BSF 0x0B,7
- ; 5: End
- L0003: GOTO L0003
- ; 6:
- ; 7: On Interrupt 'interrupt routine
- L0002:
- ; 8: PORTA = PORTA 1 'decrement the value on PORTA
- MOVLW 0x01
- SUBWF 0x05,W
- MOVWF 0x05
- ; 9: INTCON.INTF = 0 'enable new RB0/INT interrupts
- BCF 0x0B,1
- ; 10: Resume
- RETURN
- ; End of program
- L0004: GOTO L0004
- ; End of listing
- END

3. Послідовність роботи з симулятором при виконанні програми

Запустімо цю програму в PIC imulator IDE в такій послідовності:

- 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 rb0int.hex і натиснути кнопку Ореп (програмне забезпечення завантажено);

6. Натиснути Tools | Microcontroller View (відкриється вікно Microcontroller View із схематичним зображенням мікроконтролера та його виводів, а також станом виводів ON/OFF);

7. Вибрати Rate | Step By Step (покроковий режим виконання програми);

8. Натиснути Tools | Program Memory Editor (відкриється вікно з виконуваною програмою на асемблері та в кодах - бінарному та шістнадцятковому);

8. Натиснути Simulation | Start (почнеться виконання завантаженої програми). Причому, програма буде виконуватися по одній команді. Для виконання кожної команди необхідно натиснути на закладку STEP, яка з'явиться вгорі основного вікна після натискання "Start";

9. Для виконання програми в автоматичному режимі потрібно вибрати Rate | Extremely Fast simulation rate.

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

Вигляд екрану з виконуваною програмою показано на рис. 2 (натискання кнопки «Т», пов'язаної з ніжкою RB0 / INT, переключить логічний стан цього виводу).

Вигляд екрану з виконуваною програмою «Обробка зовнішніх переривань на прикладі RB0/INT» показано на рис. 6.

S PIC Simulator IDE			🛛 🛛	S Microc	ontroller Vie	w - PIC16F84	- - X	
File Simulation Rate	Tools Options Help			ON		BA2 1 18 BA1	ON	
Program Location	C:\Program Files\	PIC Simulator IDE\rb0int.h	nex		TO	RA3 2 17 RA0 CKI/RA4 3 16 OSC1/		
Microcontroller	PIC16F84 Clock Frequ	ency 4.0 MHz				MCLR 4 15 OSC2/	CLKOUT	
Last Instruction	Ne	ext Instruction		I OFF		NT/RB0 6 13 RB7	OFF I	
GOTO	0x015	GOTO 0x0	15	T OFF		RB2 8 11 RB5 RB3 9 10 RB4	OFF T OFF T	
- Program Counter and	W Register	Instructions Counter	9951				Claus	
PC 0015		Clock Cycles Counter	79576	Always	Uniop		CIOSE	
W Register	FF	Real Time Duration	19894.00 µs	S Progra	m Memory B	ditor		×
- Special Eurotion Regi	otoro (SEBo)	General Purpose	Bagistars (GPRs)	Address	Hex Value	Binary Value	Instruction	
Address and Name	Hex BinaryValue Value 76543210	Hex Addr. Value A	Hex Addr. Value	0000h 0001h	118Ah 120Ah	01000110001010 01001000001010	BCF PCLATH, 3 BCF PCLATH, 4	•
001h TMR0 002h PCL 003h STATUS 004h FSR 005h PORTA 006h PORTB 008h EEDATA 009h EEDATA 009h EEADR 004h PCLATH 008h INTCON 081h 0PTION_REG 085h TRISA 096h TRISA	00 15 1C 00 1F 00 00 00 00 90 FF 00 00 00 00 00 00 00 00 00 00 00 00	000h 00 00Dh 00 00Eh 00 00Fh 00 010h 00 011h 00 012h 00 013h 00 013h 00 014h 00 015h 00 015h 00 016h 00 017h 00	01Ch 00 01Dh 00 01Eh 00 02Dh 00 02Dh 00 02Dh 00 02Dh 00 022h 00 022h 00 023h 00 023h 00 023h 00 023h 00 026h 00 026h 00 026h 00 027h 00 029b 00	0002h 0003h 0004h 0005h 0007h 0008h 0008h 0008h 0008h 0008h 0008h 0008h 0008h	280Eh 0000h 00CFh 0E03h 0183h 00CEh 2016h 0E4Eh 0083h 0E4Fh 0083h 0E4Fh 0009h 1683h 0185h	10100000001110 0000000000000 0000011001111 0000011000001 000000	GOTO 0X00E NOP MOVWF 0X04F SWAPF STATUS,W CLRF STATUS,W CLRF STATUS CALL 0X016 SWAPF 0X04E,W MOVWF STATUS SWAPF 0X04F,F SWAPF 0X04F,F SWAPF 0X04F,W RETFIE BSF STATUS,RP0 CLRF PORTA	•
086h TRISB	FF 00	018h 00	028h 00	Aimays	ынтор			_
TMR0 Prescaler		01Ah 00	02Ah 00					
		01Bh 00	02Bh 00 -					

Рис. 6 Вигляд екрану з виконуваною програмою «Обробка зовнішніх переривань на прикладі RB0/INT»

З вікна Program Memory Editor (рис. 7) 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).

S Progra	m Memory I	ditor	
Address	Hex Value	Binary Value	Instruction
0000h 0001h 0002h 0003h 0004h 0005h 0006h 0007h 0008h 0009h 0008h 0000h 0000h 0000h 0000h	118Ah 120Ah 280Eh 0000h 0EC3h 0183h 00CEh 2016h 0E4Eh 0083h 0ECFh 0E4Fh 0009h 1683h 0185h	01000110001010 01001000001010 1010000000	BCF PCLATH, 3 BCF PCLATH, 4 GOTO 0X00E NOP MOVWF 0X04F SWAPF STATUS, W CLRF STATUS MOVWF 0X04E CALL 0X016 SWAPF 0X04E, W MOVWF STATUS SWAPF 0X04F, F SWAPF 0X04F, F SWAPF 0X04F, W RETFIE BSF STATUS, RP0 CLRF PORTA
Always	On Top		Close

Рис. 7 Вигляд вікна Program Memory Editor з текстом виконуваної програми на асемблері та кодами

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 регістрів Adress and Name, яка розташована в лівій нижній частині основного вікна симулятора (виділені рожевим кольором) (рис. 2). Всі регістри возьмирозрядні.

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

Приклад 3	
Регістр	Вміст регістру
PORTĂ	1F
і т.д.	

4. Контрольні запитання

1. Система команд мікроконтролера.

2. Структура портів мікроконтролера.

3. Програмування портів мікроконтролера.

4. Види переривань мікроконтролера.

5. 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 PIC- 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.