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

Кафедра автоматизації технологічних процесів та виробництв

Методичні вказівки для виконання лабораторної роботи №4 "Робота з аналого-цифровим перетворювачем на програмному симуляторі PIC Simulator IDE" з курсу "Проектування мікропроцесорних систем керування технологічними процесами"

Тернопіль 2017

Методичні вказівки для виконання лабораторної роботи №4 «Робота з аналого-цифровим перетворювачем 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 №4

# Робота з аналого-цифровим перетворювачем 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	Rate Tools Op	otions Help	U				
Program Location	C:\Pr	ogram Files\PIC	Simulator IDE	\demo	3.hex	2	_
Microcontroller	PIC16F877A	3					
Last Instruction		Next	Instruction -				
GOT	O 0x002A		DECR	SZ O	024,F		
Program Counter an	d Working Register	[	Instructions	Counter		262	
PC 002A			Clock Cycles	Counte	r 🗌	1536	
W Register			Real Time D	uration	3	84.00µ	s
Special Function Re	aisters (SFRs)		General	Purpos	e Regist	ers (GP	Rs
Address and Name	Hex Binar Value 7654	v Value 4 3 2 1 0	Addr.	Hex Value	Addr.	Hex Value	5
001h TMR0			020h	02	030h	00	
002h PCL	2A [ [		021h	00	031h	00	13
003h STATUS			022h	08	032h	00	
004h FSR			023h	00	033h	00	
005h PORTA			024h	OC	034h	00	
006h PORTB			025h	00	035h	00	
007h PORTC			026h	00	036h	00	
008h PORTD			027h	00	037h	00	
009h PORTE			028h	00	038h	00	
00Ah PCLATH			029h	00	039h	00	
00Bh INTCON			02Ah	00	03Ah	00	
00Ch PIR1			028h	00	03Bh	00	
And in case of the local division of the loc			02Ch	00	03Ch	00	
00Dh PIR2			02Dh	00	03Dh	00	
00Dh PIR2 00Eh TMR1L			Concernance of the local division of the loc				
00Dh PIR2 00Eh TMR1L 00Fh TMR1H			02Eh	00	03Eh	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 асемблер 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:\F	rogram Files\Pl	C Simulato	or IDE\rb0ir	nt.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	PRsìm					
	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	Ne>	t 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 (PC, W, STATUS), які використовуються при виконанні цієї програми, а також значення на виводах порту RB (RB0...RB7).

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 зчитує значення на аналоговому вході AN0 і відображає данні вимірювань на виводи порту PORTB як 8-бітове значення.

Текст програми з файлу adc.asm має наступний вигляд:

; Begin

R0L EQU 0x20 R0H EQU 0x21 R1L EQU 0x22 R1H EQU 0x23 R2L EOU 0x24 R2H EQU 0x25 R3L EQU 0x26 R3H EOU 0x27 R4L EQU 0x28 R4H EQU 0x29 R5L EQU 0x2A R5H EQU 0x2B ORG 0x0000 BCF PCLATH,3 BCF PCLATH,4 **GOTO L0003** ORG 0x0004 RETFIE

## L0003:

- ; 1: Symbol ad\_action = ADCON0.GO\_DONE 'set new name for A/D conversion start bit
- ; The address of 'ad\_action' is 0x1F,2
- ; 2: Symbol display = PORTB 'set new name for PORTB used to display the conversion result
  - The address of 'display' is 0x6 display EQU 0x6

; 3:

- ; 4: TRISB = %00000000 'set PORTB pins as outputs BSF STATUS,RP0 CLRF 0x06 BCF STATUS,RP0 ; 5: TRISA = %111111 'set PORTA pins as inputs BSF STATUS,RP0
  - MOVLW 0x3F
  - MOVWF 0x05

```
BCF STATUS, RP0
```

- ; 6: ADCON0 = 0xc0 'set A/D conversion clock to internal source
  - MOVLW 0xC0
  - MOVWF 0x1F

; 7: ADCON1 = 0 'set PORTA pins as analog inputs BSF STATUS, RP0 CLRF 0x1F BCF STATUS, RP0 ; 8: High ADCON0.ADON 'turn on A/D converter module BSF 0x1F,0 ; 9: ; 10: main: L0001 ; 11: Gosub getadresult 'go to conversion routine CALL L0002 ; 12: display = ADRESH 'display the result of the conversion MOVF 0x1E,W MOVWF 0x06 ; 13: Goto main 'repeat forever GOTO L0001 : 14: End L0004: GOTO L0004 : 15: ; 16: getadresult: 'conversion routine L0002: ; 17: High ad action 'start the conversion BSF 0x1F,2 ; 18: While ad action 'wait until conversion is completed L0005: BTFSS 0x1F,2 GOTO L0006 ; 19: Wend **GOTO L0005** L0006: MOVLW 0x1F ANDWF STATUS,F ; 20: Return RETURN ; End of program L0007: GOTO L0007 ; End of listing **END** 

# 3. Послідовність роботи з симулятором при виконанні програми

Переглянемо результати роботи цієї програми в PIC Simulator IDE. Для цього виконаємо наступне, вибравши модель МК PIC16F877:

1. Запустити PIC Simulator IDE;

- 2. Натиснути Options | Select Microcontroller;
- 3. Вибрати PIC16F877А, який містить АЦП, і натиснути кнопку Select;
- 4. Натиснути File | Load Program;
- 5. Вибрати файл adc.hex і натиснути кнопку Ореп;
- 6. Натиснути Tools | Microcontroller View (відкриється вікно Microcontroller View);
  - 7. Вибрати Rate | Extremely Fast simulation rate;
- 8. Натиснути Simulation | Start (почнеться робота програми);

9. Натиснути кнопку «А» перед виводом RA0/AN0. З'явиться білий прямокутник в середній частині панелі «Microcontroller View» з «панеллю прокрутки», повзунком якої

можна змінювати аналогове значення напруги на цьому виводі. Це значення відображається у вікні під написом «AN0» та одночасно перед виводом «AN0/RA0» (на рис. 2 це «435»).

Цифровий код, що відповідає цьому значенню, відображується на лініях порту RB0... RB7 («ON» відповідає логічній «1», «OFF»- логічному «0»);

10. Натиснути кнопку «ОК» на полі прокрутки і подивитися, як зміниться стан виводів порту PORTB.

Останні три кроки можна повторити кілька разів і подивитися на результати.

Вигляд екрану з виконуваною програмою показано на рис. 6.

S Prog	iram Memor <mark>y</mark> I		- 18 C	- • ×	S PIC Simulator IDI						186	- [	= 💌
Address	: Hex Value	Binary Value	Instruction		File Simulation Rate	Tools Op	itions Help						
0000h	118Ah	01000110001010	BCF PCLATH, 3	<u> </u>	Program Location		C:\Program Files	VPIC S	imulator ID	E\adc.l	hex		_
0001h	120Ah 2805h	101001000001010	BCF PCLATH,4		Microcontroller	PIC16F877A	Clock Freg	Jency	4.0	) MHz	_		
0003h	0000h	000000000000000000	NOP		Lest heterstien			and here					
0004h	0009h	00000000001001	RETFIE		Last Instruction	0.010	N	extins	ruction		45		
0005h	1583h	01011010000011	BSF STATUS,R	.P0	GUIU	UNTR			MU	VLW U	XIF		
0007h	1283h	01001010000011	BCF STATUS, R	PO	Program Counter and	W Register -		In	structions	Counter		52172	_
0008h	1683h 303Eh	010110100000011	BSF STATUS,R	.P0	PC 001B			Clo	nck Cucles	Counte	r 🗌	360432	,
000Ah	0085h	00000010000101	MOVWE PORTA		V/ Depister					). antina	0	100.00	
000Bh	1283h	01001010000011	BCF STATUS, R	.P0	w negister				earrime L	uration	1 30	108.00	μs
000Ch	009Fh	00000010011111	MOVLW UXCU MOVWF ADCONÛ		- Special Function Reg	isters (SFRs)			Genera	l Purpos	e Regis	ters (GP	'Rs)
000Eh	1683h	01011010000011	BSF_STATUS,R	.P0		Hex	Binary Value			Hex		Hex	
OUOFh	019Fh	00000110011111	CLRF ADCONU	-	Address and Name	Value 76	5543210		Addr.	Value	Addr.	Value	
- Alua	ue On Ton			Close	001h TMR0			-	020h	00	030h	00	
Aivia	ys on rop			0,000	002h PCL	1B			021h	00	031h	00	
E	Microcontrol	ler View - PIC16F877A		- • ×	003h STATUS	18			022h	00	032h	00	
	_			OFF	004h FSR	00			023h	00	033h	00	
7	435	ANO/RAO 2 39	RB6/PGC	ON	005h PORTA	00			024h	00	034h	00	
4		AN1/RA1 3 38	RB5	ON	UU6h PURTB	60			025h	00	035h		
1		AN3/Vref+/RA3 5 36	RB3/PGM	ON					025h	00	0360		
	OFF TOCK		32	ON	009h PORTE				027h	00	038h	00	
1	A U 475 A D	ANO	81 80/INT	OFF	ODAN PCLATH				020h	00	039h	00	
1	X O Y	WR/AN6/RI 435	16		00Bh INTCON	02			02Ah	00	03Ah	00	
-	7 0	VCS/AN7/RL	iS D7/PSP7	OFF T	00Ch PIR1	40			02Bh	00	03Bh	00	
			D6/PSP6	OFF T	00Dh PIR2	00			02Ch	00	03Ch	00	
			D5/PSP5	OFF I	00Eh TMR1L	00			02Dh	00	03Dh	00	
	OFF T10	SO/T1CKI/RC0 15 26	RC7/RX/DT	OFF T	00Fh TMR1H	00			02Eh	00	03Eh	00	
1	OFF T10	DSI/CCP2/RC1 16 25	RC6/TX/CK	OFF I	010h T1CON	00		•	02Fh	00	03Fh	00	-
i i	OFF	SCK/SCL/RC3 18 23	RC4/SDI/SDA	OFF T									
1	OFF	PSP0/RD0 19 22	RD3/PSP3	OFF I									
_	OIT		102/1312										
E	Always On Top			Close									

Рис. 6 Вигляд екрану з виконуваною програмою «Робота з аналого-цифровим перетворювачем»

З вікна "Program Memory Editor" 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
П		

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
PORTB	6C

і т. д.

### 4. Контрольні запитання

1. Послідовність роботи АЦП мікроконтролера.

- 2. Аналогові та цифрові виводи 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.