CanaKit Programmer Error 27 Workaround Mark Spencer, WA8SME <u>mspencer@arrl.org</u> 860-381-5335

Introduction. Many readers of the <u>PIC Programming for Beginners</u> book that are using the CanaKit programmer supplied in the kit of parts purchased from the ARRL are having difficulty with an ERROR 27 when they attempt to program the PICs. Diligently following the directions in the book, they will connect the programmer and the programmer will be recognized.

Outp	put		_ 🗆 ×
Build	Version Control Find in Files	PICkit 2	
Initiali Found PICkit Targe PIC16 PICkit	izing PICkit 2 version 0.0.3.6 d PICkit 2 - Operating Syste t 2 Unit ID = CANAKIT.COM et power not detected - Pov 6F676 found (Rev 0x7) t 2 Ready	3 em Versi vering fro	on 2.32.0 om PICkit 2 (5.00V)

When the program is complied, built, and sent to the programmer, the return is an ERROR 27 message and the programming stops.

Output	
Build Version Control Find in Files PICkit 2	
Programming Target (10/29/2011 12:07:30 PM)	
Erasing Target	
Programming Program Memory (0x0 - 0xA4) Verifying Program Memory (0x0 - 0xA4)	
PK2Error0027: Failed verify (Address = 0x0 - Expected Value 0x0 - Value Read 0x3FFF PICkit 2 Ready)

The reason why this is happening is unknown. There appears to be a conflict between the CanaKit PICKit2 clone programmer and MPLab IDE. When you hit the programming icon in MPLab, a number of steps are taken by the software to accomplish the actual programming. Early in the programming sequence, the contents of a critical register called the OSCCAL value is retrieved from the PIC and is supposed to be stored temporarily while the rest of the program is sent to the PIC. After the main program is delivered to the PIC, the OSCCAL value is re-installed back onto the PIC. For some yet

undetermined reason, this is not happening with the CanaKit programmer. The OSCCAL value is lost and the programming is stopped with the ERROR 27 message returned. The OSCCAL value is actually gone and needs to be recovered before the PIC device can be used again as mentioned in the book.

The CanaKit works fine with another programming software package called PICkit 2 (not to be confused with the programmer called PICKit 2). The following steps outline a workaround to allow you to continue your studies of PIC programming while using the CanaKit programmer with PICkit 2 software.

<u>Using PICKit 2 Software to Program the PIC.</u> The work around takes advantage of the fact that the CanaKit appears to work fine with the PICKit 2 software package. This PICKit 2 software is a pretty functional programming package. While it does not facilitate the writing of programs as you are learning to do in the book (MPLab is still required for program development), the PICKit 2 software utilities allow you to recover a lost or corrupted OSCCAL value and also to program the PIC from a HEX file.

Step 1. Install the PICkit 2 software package. Go to the Microchip web page at this URL: <u>http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en023</u> <u>805</u>. On the right side of the page you will see this link: <u>PICkit 2 v2.61</u>. Download and install the PICkit 2 software accepting the defaults. When you run the software with your CanaKit programmer attached, you will see the software displayed like this:

PICkit 2 Pro	ogramm	er - CANAK	IT.COM						
File Device	Family	Programme	er Tools	View	Help				
-Midrange/St	andard Co	onfiguration -	5						10
Device:	PIC16F	676		<u>Confic</u>	juration: 00	000			
User IDs:	FF FF F	F FF		All Pro	ltect				
Checksum:	FFFF			OSCC	Invalid Va AL: 3FFF	lue	BandGap:	0000	
						-83 -83			
Reading d	evice:			ss 200	0.00		Mic	ROCH	HP
Program N	lemory.	EE Us	serlDs (Config	. Done.				
						VD	D PICkit 2		
Read)/rito	1 Marilu	1 5,000	1 0	lank Chaok		On	2.5	+
	white						/MULR	S.	-
Program M	emory								Į.
Enabled	Hex Or	nly 🗾	Source:	Read fro	m PIC16F67	6			
000	0000	0000	0000	0000	0000	0000	0000	0000	
008	0000	0000	0000	0000	0000	0000	0000	0000	
010	0000	0000	0000	0000	0000	0000	0000	0000	
018	0000	0000	0000	0000	0000	0000	0000	0000	
020	0000	0000	0000	0000	0000	0000	0000	0000	
028	0000	0000	0000	0000	0000	0000	0000	0000	
030	0000	0000	0000	0000	0000	0000	0000	0000	
038	0000	0000	0000	0000	0000	0000	0000	0000	
040	0000	0000	0000	0000	0000	0000	0000	0000	
048	0000	0000	0000	0000	0000	0000	0000	0000	
050	0000	0000	0000	0000	0000	0000	0000	0000	
058	0000	0000	0000	0000	0000	0000	0000	0000	-
							23		
Enabled	Hex Or	nly 💌					Au + ¹	to Import H Write Devi	lex ice
00 00 0	0 00 00 0 00 00	0 00 00 0	0 00 00 0 00 00	00 00 00 00	00 00 00 00 00 00	00 00	Be Ex	ead Devic port Hex F	e + File

Notice that the OSCCAL value is highlighted in red and indicates that there is an invalid value of 3FFF (if your PIC has been previously corrupted, if not, the value here will probably be valid with the value of 34XX...the XX are hex numbers that reflect the PICs OSCCAL value.

Step 2. Recovering the OSCCAL value. This step can take two directions. One if you were diligent in recording the OSCCAL value as suggested in the book before you do any programming of the device. The other direction is if you forgot to do that, or couldn't for some reason, and allows you to recover the unknown value to make the PIC usable again.

Direction one. Click on Tools/OSCCAL/Manual Set.

PICkit 2 Pro	ogrammer - C	ANAKIT.C	M		
File Device - Midrange/St Device:	e Family Prog tandard Configu PIC16F676	grammer	Tools View Help Image: Context state Image: Context state Image: Context state Image: Context state Image: Context state Image: Context state	Ctrl+P Ctrl+D	Sat Manually
User IDs: Checksum:	FFFF		Target VDD Source Display Unimplemented Config Bil	• ts •	Auto Regenerate
Reading d Program M Read	evice: 1emory EE Write	E User Verify [Calibrate VDD & Set Unit ID Use VPP First Program Entry Use LVP Program Entry Fast Programming		2.5 ÷
Program M	lemory Hex Only	▼ Sc	UART Tool Logic Tool		OCHIP
000	0000 00	00000	Check Communication Troubleshoot	10	
010 018	0000 00	00 0 000 0	Download PICkit 2 Operating Sys	00 tem)01	

This popup window will be displayed allowing you to manually enter the OSCCAL value, in this case 3454.

Set OSCCAL		_0×
OSCCAL value:	3454	(hex)
WARNING:		
Setting OSCCAI	L will erase	ALL
memory in part!		

All the OSCCAL values will begin with 34, flowed by two alpha-numeric characters that are the actual HEX values of the OSCCAL value. Don't worry about the WARNING, that ship has already sailed. Click on Set. Once the software has reinstalled the OSCCAL value, you will be returned to the main software window and the correct OSCCAL value will be displayed and you're ready to move on.

Direction two. If you have no idea what the OSCCAL value is for a particular device, you can allow the software to experimentally determine the value and install it. This system apparently is not perfect, because fequently you will find that the software generated OSCCAL value and the factory determined OSCCAL value (that you read off of a fresh device) are not always the same. There must be some range of acceptable values because I have used the software generated OSCCAL value, click on Tools/OSCCAL/Auto Regenerate.

ile Device Family Programmer	Tools View Help	
Midrange/Standard Configuration Device: PIC16F676	Enable Code Protect Ctrl+P Enable Data Protect Ctrl+D	<u>e=en023</u>
User IDs: FF FF FF FF	OSCCAL +	Set Manually
Checksum: 0043	Target VDD Source Display Unimplemented Config Bits	Auto Regenerate
rogramming Successful.	Calibrate VDD & Set Unit ID	CHIP

Then follow the prompts to allow the software to regenerate and install the OSCCAL value. Click on OK.

Regenerate (ISCCAL X
Regenerating will completel part.) the OSCCAL value y erase this
Are you sure continue?	you wish to
ОК	Cancel

Step 3. Using PICkit 2 software to download a HEX file into the PIC. When you use MPLab IDE to author and build a program for the PIC, the software actually creates a number of files without your direct intervention. The programs that you write in the MPLab editor window are not the actual files that are downloaded into the PIC, but the "human" language interpretation of the program. The MPLab compiler actually converts your writing into machine language that the PIC understands, and the program in machine language is actually what is installed onto the PIC. That machine language version of your program is in the HEX file (HEX is short for hexadecimal...base 16 numbering system) and has the extension .HEX. If you look in the directory where your program files are stored, you will find the associated HEX file for the program you are working on. It is this HEX file that you will install using PICkit 2 software. After you have created your programs as outlined in the book in MPLab, instead of using MPLab to install the program on the PIC, launch the PICkit 2 software.

=ile	Device	Family	Programme	er Tools	View	Help				
	Import He	ex				Itrl+I	<u>}</u>			
	Export He	ex			C	itrl+E	7F			
	1 C:\Fi	les\Ch 3	Program\Firs	t Program.	HEX C	trl+1				
	E×it				c	trl+Q		BandGap:	0000	
_	1		1	1	1			Un	2.5	-
Re Pro	ogram Me	Write emory -		Source:	None (Err	ontu/Frased		/MULR	2	
Re Pro	ogram Me Enabled	Write emory – Hex Or		Source:	None (En	ank Uneck hpty/Erased		2FFF	2555	
Re Pro	pgram Me Enabled	Write emory Hex Or 3FFF 3FFF	Verity	Source:	None (En	ank Check apty/Erased 3FFF 3FFF) 3FFF 3FFF	3FFF	3FFF 3FFF	-
	pgram Me Enabled	Write emory Hex Or 3FFF 3FFF 3FFF	Verity	Source: 3FFF 3FFF 3FFF	None (En 3FFF 3FFF 3FFF	ank Check opty/Erased 3FFF 3FFF 3FFF) 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF	3FFF 3FFF 3FFF	
	pgram Me Enabled 000 008 010 018	Write emory Hex Or 3FFF 3FFF 3FFF 3FFF	Verity	Source: 3FFF 3FFF 3FFF 3FFF	None (En 3FFF 3FFF 3FFF 3FFF	ank Check apty/Erased 3FFF 3FFF 3FFF 3FFF) 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF	
	ead ogram Mic Enabled 000 008 010 018 020	Write emory Hex Or 3FFF 3FFF 3FFF 3FFF 3FFF	Verity Aly SFFF 3FFF 3FFF 3FFF 3FFF	Source: 3FFF 3FFF 3FFF 3FFF 3FFF	None (En 3FFF 3FFF 3FFF 3FFF 3FFF	ank Check 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF 3FFF	
Re Pro ▼ 1 0 0 0 0 0 0 0 0 0	ad ogram Ma Enabled 000 008 010 018 0120 028	Write emory Hex Or 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	SFFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	Source: 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	None (En 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	ank Lheck 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF) 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	
Re Pro ○ () () () () () () () (ad pgram Ma Enabled 000 008 010 018 020 028 028 030	Write emory 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	Source: 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	None (En 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	ank Check 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F) 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	
Re Pro (((((((((((((ead pgram Ma Enabled D00 D08 D10 D18 D20 D28 D20 D28 D30 D38	Write emory 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	Source: 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	None (En 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	ank Check 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	
Re - Pro ✓ 1 (((((((((((((ead ogram Ma Enabled 000 008 010 018 020 028 020 028 020 028 020 028 020	Write emory Hex Or 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	Venty 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	Source: 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	None (En 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	ank Check 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F) 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	
Ree Pro	2000 Enabled 2000 200 200 200 202 220 228 230 238 230 238 230 238 240 248	Write emory 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	Venty 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	Source: 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	None (En 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	ank Check 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F) 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	
Ree Pro	230 2937 200 200	Write emory – 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	Venty 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	Source: 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF	None (En 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	ank Check 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3F	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FFF 3FF	

Click on File/Import Hex and navigate to where your programs are being filed on your hard disk.

Import Hex File					<u>?×</u>
Look in:	🗀 Ch 3 Program	n	•	3 🕫 📂 🖽•	
My Recent Documents Desktop My Documents My Computer	First Program	HEX			
My Network	File name:	First Program		•	Open
Places	Files of type:	HEX files		•	Cancel

If you use the default folders for the program files in the book, you will find the HEX file for the program as shown here for the first program in chapter 3. Click on Open and the HEX version of your program will be imported into PICkt 2 and is ready for installation into the PIC memory.

PICkit 2 Pro	ogramme	er - CANAK	IT.COM					_	
ile Device	Family	Programme	er Tools	View	Help				
Midrange/St	andard Co	onfiguration -							
Device:	PIC16F	676		Config	uration: 00	144			
User IDs:	FF FF FF	FF		All Pro	ect				
Checksum:	0043			OSCC/	AL: 3454		BandGap:	0000	
lov filo eu	cacefully	vimnorte	d				N		
lex nie su	cessiun	ymporte	u.				MIC	ROCI	HIF
							D PICkit 2	-	
		6	1	1			l On	25	
Read	Write	Verify	Eras	e Bi	ank Check		/MCLR	12.0	-
Program M	emory					- 19 (B)			
Enabled	Hex Or	ily 💌	Source:	C:\Files	Ch 3 Progr	am\First F	Program.HE>	<	
000	0000	2805	वन्द्र	ननन्द्र	0008	1683	3040	0090	
008	1283	0185	0187	3007	0000	3000	008B	1683	
010	3000	0081	3000	0085	3000	0087	3000	0091	
018	1283	1685	2049	2049	3000	2035	205D	3020	
020	0084	0145	202C	2035	0445	OBA4	2822	1687	
028	2049	1287	2049	2827	3085	0725	3000	1803	
030	3E01	008A	0825	2083	0008	00A3	3008	00A2	
038	1285	207B	0CA3	1C03	1285	1803	1685	207B	
040	OBA2	283A	1685	205D	0008	2049	2049	2049	
048	2049	204D	204D	204D	2851	2067	2067	2067	
050	2067	2067	2067	0008	30C6	00A00	0000	2858	
058	2859	285A	OBAO	2859	0008	2054	2054	2054	-
							25. 31		
EEPROM [Data						Au	ito Import I	Hex
	11.000	luc 🔤					10.23	1.1.1. D.	

You will notice that the HEX values in the Program Memory area of PICkit 2 have changed from all 0000's to HEX numbers that reflect the program values. Finally, click on Write and the PICkit 2 software will do its thing and install the program onto the PIC.

PICKIC Z PI	ogramme	er - CANAk	(IT.COM						
File Device	Family	Programm	er Tools	View	Help				
Midrange/St	andard Co	onfiguration	2						
Device:	PIC16F	676		Confia	uration: 0	044			
				All Prot	ect				
User IDs:	FF FF FI	r FF							
Checksum:	0043			OSCC/	AL: 3454		BandGap:	0000	
				e -			D PICkit 2 0n	25	
Read Program M	Write emory	Verify	Eras	e Bla	ank Check	ram\First P	/MCLR	×	
Read Program M ✓ Enabled	Write emory Hex Or	Verify	Eras Source:	e Bl	ank Check	ram\First P	/MCLR rogram.HE	X 0090	
Read Program M ✓ Enabled 000 008	Write emory - Hex Or 0000 1283	Verify	Source: 3FFF 0187	e Bi C:\Files 3FFF 3007	Ch 3 Prog	ram\First P	/MCLR rogram.HE> 3040 008B	× 0090	
Read Program M Enabled 000 008 010	Write emory	Verify	Eras Source: 3FFF 0187 3000	e Bi. C:\Files 3FFF 3007 0085	ank Check \Ch 3 Prog 0008 0099 3000	ram\First P	/MCLR rogram.HE 3040 008B 3000	× 0090 1683 0091	
Read Program M ✓ Enabled 000 008 010 018	Write emory Hex Or 0000 1283 3000 1283	Verify 2805 0185 0081 1685	Eras Source: 3FFF 0187 3000 2049	e Bi C:\Files 3FFF 3007 0085 2049	ank Check \Ch 3 Prog 0008 0099 3000 3000	ram\First P 1683 3000 0087 2035	7MCLR rogram.HE 3040 008B 3000 205D	× 0090 1683 0091 3020	
Read Program M ✓ Enabled 000 008 010 018 020	Write emory Hex Or 0000 1283 3000 1283 00A4	Verify 2805 0185 0081 1685 01A5	Eras Source: 3FFF 0187 3000 2049 202C	e Bi C:\Files 3FFF 3007 0085 2049 2035	Ank Check \Ch 3 Prog 0008 0099 3000 3000 0AA5	ram\First P 1683 3000 0087 2035 0BA4	7MCLR 70gram.HE 3040 008B 3000 205D 2822	× 0090 1683 0091 3020 1687	
Read Program M ✓ Enabled 000 008 010 018 020 028	Write emory	Verify 2805 0185 0081 1685 01A5 1287	Eras Source: 3FFF 0187 3000 2049 2020 2020 2049	e Bi 3FFF 3007 0085 2049 2035 2827	Ank Check Ch 3 Prog 0008 0099 3000 3000 0AA5 3085	ram\First P 1683 3000 0087 2035 0BA4 0725	7MCLR rogram.HE 3040 008B 3000 205D 2822 3000	× 0090 1683 0091 3020 1687 1803	
Read Program M ✓ Enabled 000 008 010 018 020 028 030	Write emory	Verify 2805 0185 0081 1685 01A5 1287 008A	Eras Source: 3FFF 0187 3000 2049 202C 2049 0825	e Bi 3FFF 3007 0085 2049 2035 2827 2083	Ank Check Ch 3 Prog 0008 0099 3000 3000 0AA5 3085 0008	ram\First P 1683 3000 0087 2035 0BA4 0725 00A3	7MCLR rogram.HE 3040 008B 3000 205D 2822 3000 3008	× 0090 1683 0091 3020 1687 1803 00A2	
Read Program M ✓ Enabled 000 008 010 018 020 028 030 038	Write emory Hex Or 1283 3000 1283 00A4 2049 3E01 1285	Verify 2805 0185 0081 1685 01A5 1287 008A 207B	Eras Source: 3FFF 0187 3000 2049 2020 2049 0825 00A3	e Bi 3FFF 3007 0085 2049 2035 2827 2083 1C03	Ank Check Ch 3 Prog 0008 0099 3000 3000 0AA5 3085 0008 1285	ram\First P 1683 3000 0087 2035 0BA4 0725 00A3 1803	7MCLR 10gram.HE 3040 008B 3000 205D 2822 3000 3008 1685	× 0090 1683 0091 3020 1687 1803 00A2 207B	
Read Program M ✓ Enabled 000 008 010 018 020 028 030 038 040	Write emory Hex Or 0000 1283 3000 1283 00A4 2049 3E01 1285 0BA2	Verify 2805 0185 0081 1685 01A5 1287 008A 207B 283A	Eras Source: 3FFF 0187 3000 2049 202C 2049 0825 0CA3 1685	e Bi 3FFF 3007 0085 2049 2035 2827 2083 1C03 205D	Ank Check Ch 3 Prog 0008 0099 3000 3000 0AA5 3085 0008 1285 0008	ram\First P 1683 3000 0087 2035 0BA4 0725 00A3 1803 2049	7MCLR 70gram.HE 3040 008B 3000 205D 2822 3000 3008 1685 2049	× 0090 1683 0091 3020 1687 1803 00A2 207B 2049	
Read Program M ✓ Enabled 000 008 010 018 020 028 030 038 040 048	Write emory - Hex Or 1283 3000 1283 00A4 2049 3E01 1285 0BA2 2049	Verify 2805 0185 0081 1685 01A5 1287 008A 207B 283A 204D	Eras Source: 3FFF 0187 3000 2049 2020 2049 0825 0CA3 1685 204D	e Bi 3FFF 3007 0085 2049 2035 2827 2083 1C03 205D 204D	Ank Check Ch 3 Prog 0008 0099 3000 3000 0AA5 3085 0008 1285 0008 2851	ram\First P 1683 3000 0087 2035 0BA4 0725 00A3 1803 2049 2067	7MCLR rogram.HE 3040 008B 3000 205D 2822 3000 3008 1685 2049 2067	× 0090 1683 0091 3020 1687 1803 00A2 207B 2049 2067	
Read Program M ✓ Enabled 000 008 010 018 020 028 030 038 040 048 050	Write emory - Hex Or 1283 3000 1283 00A4 2049 3E01 1285 0BA2 2049 2067	Verify 2805 0185 0185 0081 1685 01A5 1287 008A 207B 283A 204D 2067	Eras Source: 3FFF 0187 3000 2049 202C 2049 0825 0CA3 1685 204D 2067	e Bi 3FFF 3007 0085 2049 2035 2827 2083 1C03 205D 204D 0008	Ank Check Ch 3 Prog 0008 0099 3000 3000 0AA5 3085 0008 1285 0008 2851 3006	ram\First P 1683 3000 0087 2035 0BA4 0725 00A3 1803 2049 2067 00A0	7MCLR rogram.HE 3040 008B 3000 205D 2822 3000 3008 1685 2049 2067 0000	× 0090 1683 0091 3020 1687 1803 00A2 207B 2049 2067 2858	

2000 C 2000 C

Now that the program is in the PIC, it will start running on the PIC. Here is where you remove the PIC from the CanaKit programmer and install it in your circuit for testing.

Conclusion.

If you are getting ERROR 27 messages when you are using the CanaKit programmer with MPLab IDE, there is an unidentified conflict that is being worked. There is a workaround that will allow you to continue with your studies of PIC programming while using the CanaKit.

- 1. Install the PICkit 2 software package from the Microchip web site. The CanaKit programmer appears to work will with the PICkit 2 software package. PICkit 2 software allows you to install program HEX files onto the PIC.
- 2. Using PICkit 2, recover the OSCCAL value that was lost or corrupted during the attempt to program using the CanaKit with MPLab. The OSCCAL value can be restored using the known factory determined OSCCAL value or it can be auto regenerated using the PICkit 2 utility. (The

loss of the OSCCAL value appears to only be happening when using the CanaKit programmer with MPLab, it does not happen when using the PICkt 2 software.)

- 3. After you have developed the program using MPLab, build the program, but stop short of using MPLab to install the program onto the PIC.
- 4. Launch the PICkit 2 software. Click on File/Import and navigate to the folder where your program files are stored. Click on the appropriate file with the .HEX extension to import the hex version of your program into PICkit 2.
- 5. Click on Write to install the program. The OSCCAL value should not be lost using this procedure so set 2 above will not be needed further.
- 6. Remove the PIC form the CanaKit programmer and install it into your circuit for testing.