

# <u>AN564</u>

4

### Using the PWM

#### INTRODUCTION

The PIC16/17 family of RISC-like microcontrollers has been designed to provide advanced performance and a cost-effective solution for a variety of applications. This application report provides examples which illustrate the uses of Pulse Width Modulation (PWM) using the PIC17C42 Timer1 or Timer2 modules. These examples may be modified to suit the specific needs of your application.

- This Application Note describes the operation of the PWM. They include the following topics:
- 1. Simple PWM Operation
- 2. Variable Period / Variable Duty Cycle PWM
- 3 External Clock for Timer Timebase (ramifications/issues)

The listing file for the Variable Period / Variable Duty Cycle example can be found in Appendix A. The source files can be found on the Microchip BBS. On directions on how to access the Microchip BBS please refer to DS30128, which can also be found in the Microchip Embedded Control Handbook (Literature Number DS00092).



#### FIGURE 1 - TIMER1 AND TIMER2 BLOCK DIAGRAM WITH PWM MODE

© 1993 Microchip Technology Inc.

The control registers that are utilized by Timer1 and Timer2 are shown in Table 1. Shaded Boxes are control bits that are not used by the Timer1 nor Timer2 module.

Name	BANK	ADDR	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bi	t 1	bit O
INTSTA		0x06	PEIR	RTXIR	RTCIR	INTIR	PEIE	RTXIE	RTCI	E	INTIE
CPUSTA		0x07	-	-	STKAV	GLINTD	то	PD	-		-
TMR1	2	0x10		Timer1 Register							
TMR2	2	0x11		Timer2 Register							
PR1	2	0x14		Timer1 Period Register							
PR2	2	0x15			Timer	2 Period Regi	ster				
PIR	1	0x16	IRB	TM3IR	TM2IR	TM1IR	CA2IR (	A1IR	твмт	RI	BFL
PIE	1	0x17	IEB	ТМЗІЕ	TM2IE	TM1IE (	A2IE C	A1IE	TXIE	RC	E
PW1DCL	3	0x10		PWM1 Duty Cycle Low Register							
PW2DCL	3	0x11		PWM2 Duty Cycle Low Register							
PW1DCL	3	0x12		PWM1 Duty Cycle High Register							
PW2DCL	3	0x13		PWM2 Duty Cycle High Register							-
TCON1	3	0x16	A2ED1	CA2ED0	GA1ED1 (	CA1ED0	16/8 <sup>—</sup> TN	IR3C T	MR2C	TMF	1C
TCON2	3	0x17	CA20VF	CA10VF	PWM2ON F	WM10N C	A1/PR3 TN	R30N T	MR20N	TMR	10N

#### TABLE 1 - REGISTERS ASSOCIATED WITH TIMER3 AND CAPTURE

Care must be taken when loading values into the PWM registers. These registers are the duty cycle registers (PWxDCH:PWxDCL) and the period register (PRx). Figure 2 shows the proper update timing of these values.

#### FIGURE 2 - TIMING FOR UPDATING THE DUTY CYCLE REGISTERS AND PERIOD REGISTER



#### SIMPLE PWM OPERATION

Simple PWM operation is where the period of the PWM output remains constant, and only the duty cycle is modified. The PWM can operate in either of two modes:

- · Hi-resolution mode-the PWxDCL register is modified
- Standard resolution mode the PWxDCL register is not modified

When operating in the standard-resolution mode, only the PW-DCH register is ever modified. Since this takes only a single cycle, this can be done at any time. Also since the period is remaining constant this may be done without any PWM interrupt software overhead.

When operating in the high-resolution mode both the PWxDCH:PWxDCL register pair is modified. Since this is a multicycle update, care needs to be taken that the "new" PWM duty cycle value is not latched until the update is complete. If the duty cycle is latched before this update is complete, the duty cycle will display a "glitch". If the PWxDCH is written first, the maximum error is 3 Q-cycles (187.5 ns @ 16 MHz). If the PWxDCL is written first, the maximum error is also 3 Q-cycles (187.5 ns @ 16 MHz), with the PWxDCH delayed by one PWM period. This may be acceptable for some applications. If this is not acceptable for your application then a subroutine can be written to ensure that these duty cycle writes are not done when the timer will equal the period. One implementation of this subroutine (PWM\_UD) is used in the Variable Period / Variable Duty Cycle PWM example. This is discussed in the following section, with the listing in Appendix A.

Additional code examples can be found in application note AN539 in the Embedded Control Handbook.

#### VARIABLE PERIOD / VARIABLE DUTY CYCLE PWM

In a variable period / variable duty cycle PWM both the duty cycle of the PWM as well as the frequency (period) of the PWM are modified.

The PIC17C42's hardware double buffers the duty cycle registers, but the period registers are not double buffered. What this means is that you can modify the duty cycle registers, but the value will only be latched when the timer register equals period register. Since the period register is not buffered, as the period register is modified this becomes the "new" period. This means that care must be taken when modifying the period register resulting in a "glitch" to occur. This "glitch" occurs when the period register is modified with a value that is less than the present timer value. The timer does not have a match with the old period value, and continues to count until the timer register equals period register.

Figure 3, shows an example where the period (PR1) register = 7Fh. Then the period is modified to a smaller value (PR1 = 1Fh) without checking that the value in Timer1 (TMR1) register = 3Eh. Since the new period (PR1) value is less then the present timer (TMR1) value, a glitch has occurred.



#### FIGURE 3 - MODIFYING PERIOD REGISTER "GLITCH"

<sup>© 1993</sup> Microchip Technology Inc.

Care must be taken when writing a 10-bit duty cycle value. Since this requires two register writes, the Timer equals period could occur between these two writes, which would give a duty cycle that was not as expected. The cases are as follows:

- 1. If the duty cycle low (DCL) register is written, and then the Timer equals period. The old DCH register and the new DCL register becomes the duty cycle.
- If the duty cycle high (DCH) register is written, and then the Timer equals period. The new DCH register and the old DCL register becomes the duty cycle

At the following occurrence of the timer equaling the period, the second register written would be updated. The subroutine PWM\_UD (Appendix A) ensures that these duty cycle writes are not done when the timer will equal the period.

A software example of a variable period / variable duty cycle is shown in Appendix A. In this example the period is double buffered in software, and the new period value is loaded in the timer overflow interrupt service routine. When the new duty cycle needs to be loaded. The device connections are shown in Figure 4. This program has two PWM settings (period / duty cycle combinations) that are switched between depending on the level on pin RB0. A frequency generator was used to give a low frequency signal on the RB0 pin. Figure 5 shows an example of the input and output waveforms.

#### FIGURE 4 - APPLICATION HARDWARE SETUP



The program listing in Appendix A implements this example, Figure 8 is the flowchart of the program. This example may be modified to suit the particular needs of your application. The Table 3 is a summary of the requirements for this program (@ 16 MHz):

#### **TABLE 3 - PROGRAM REQUIREMENTS**

Code Size:	52 Words
RAM used:	11 Bytes
Interrupt Service Routine time	3.0 usec
Subroutine time	4.5 usec
	6.0 usec
Maximum PWM frequency:	200 KHz
PWM Accuracy:	62.5 nsec

#### **FIGURE 5 - EXAMPLE APPLICATION WAVEFORMS**



## EXTERNAL CLOCK FOR TIMER TIMEBASE

The counters used for the time base of the PWM outputs can be software selected to operate from an external clock source. This allows a lower frequency PWM to be achieved. Doing this brings up new issues that must be understood for the application.

One of these issues is clock synchronization. All external clocks must be synchronized to the internal operating speed of the microcontroller, as shown in figure 9. When this synchronization occurs the PWM output is not truly operating from the external clock, but actually the internal synchronized clock. This leads to a "jitter" of the output to the clock. This jitter is caused from the delta time between the external clock and the synchronized clock not being constant. The synchronization errors are:

Duty cycle error = +/- Tcy Period error = +/- Tcy If you needed to run the PWM at a low frequency, and also want to reduce the "jitter" from the use of an external asynchronous clock, a PWM output could be used as the synchronous clock source. When the clock is synchronized to the device the clock error is always constant, so there is no jitter. Figure 7 shows this example.

#### FIGURE 7 - PWM OUTPUT TO GENERATE A SYNCHRONOUS CLOCK



#### FIGURE 6 - EXTERNAL CLOCK SYNCHRONIZATION



Another use is where precise timing of updates need to be done, but not at the frequency of the PWM output. In this discussion, TMR1 is used as the time-base of a constant frequency PWM output. TMR1 uses the internal clock of the device and TMR2 uses the external clock input. TMR2 will get the clock input from the PWM2 output.

The PWM output is a constant frequency variable duty cycle output. The PW1DCH:PW1DCL register pair contain the variable duty cycle value of PWM1 output. The PW2DCH:PW2DCL register pair is set for a fixed duty cycle (50%) for the PWM2 output. The PWM outputs could be programmed to have a frequency of 20 KHz, so to reduce audible noise. The PWM2 signal is connected to the RB4/TCLK12, as shown in Figure 8. The PR2 register could be loaded with 14h (20), to give an interrupt every 1 KHz. This interrupt can then trigger tasks, such as updating the duty cycle of PWM1. This is useful in motor control as well as other applications where the update rate is less then the PWM frequency.

#### CONCLUSION

The PIC17C42's PWM features offer a high performance solution at a lower system cost then previously available. The versatility of PWMs make the PIC17C42 ideal for motor control applications (ses AN532) and many industrial control applications.

Author: Mark Palmer Logic Products Division



#### FIGURE 8 - SAMPLING SCHEME

#### APPENDIX A: LISTING FILE

MPASM B0.54

PAGE 1

LIST P=17C42, T=ON, C=120, N=0

	<pre>; fints is the Data of the FDM 's program that generates a ; variable PDM output. The PDM's period and duty cycle can ; be varied. The new period (NEW_PR1) and the new duty cycle ; (NEW_DC1 and NEW_DC1Q) are loaded by the user program. ; The peripheral interrupt routine loads the new period value ; (frequency) into the PR1 register. A subroutine (PDM_UD) ; is also used to ensure that the 10-bit duty cycle registers ; are updated in the same PDM cycle, i.e. the timer match does not ; occur between two duty cycle register writes. ; ; The duty cycle value gets latched on the overflow (Period match) ; of the timer. The period value gets modified as soon as the period ; register is changed. Therefore care must be taken in updating ; the period register. In cases where the period value is modified ; to a smaller value, we must ensure that the Timer counter is less ; then this value when the period register is updated (TMR1 &lt; new PR1).</pre>							
	; to 00H, and or ; the period val	nly cause lue. This	the overflow would give a	w i a v	vrong PWM output.			
	; ; In this examp ; is an asynchro ; port pin RB0. ; For a high lev ; PR1 = 7Fh, ; For a low leve ; PR1 = 1Fh, ;	le the even onous even vel the P PW1DCH = PW1DCH =	vent which cau ent. A low free PWM registers = 3Fh, and PWJ M registers a = 07h, and PWJ	use equ 1DC are 1DC	e the PWM to be updated ency signal was placed on re updated as follows: CL = 40h e updated as follows: CL = 80h			
	; Do the EQUate ;	table						
0020	NEW_DC1	EQU	0x20	;	New PWM1 duty cycle value			
0021	NEW_DC1Q	EQU	0x21	;				
0022	NEW_PR1	EQU	0x22	;	New PWM1 period value			
0025	PWM_WIN	EQU	0x25		Register for the PWM window cycle count			
0026	CALC_PR	EQU	0x26		Calculated period value			
0027	FLAG_REG	EQU	0.827	'	Register for flag bits			
001A	, DC1H	EOU	0x1A	;	PWM registers for high time			
001B	DC10H	EOU	0x1B		5			
001C	PR1H	EOU	0x1C					
	;	~ -						
001D	DC1L	EQU	0x1D	;	PWM registers for low time			
001E	DC10L	EOU	0x1E					
001F	PR1L	EQU	0x1F					
	;							
	;							
07FF	END_OF_PROG_MEM	EQU 0x07	/FF					
	;							
0004	ALUSTA	EQU	0x04					
0006	CPUSTA	EQU	0x06					
0007	INTSTA	EQU	0x07					
A000	W	EQU	0x0A					
	;							
0011	DDRB	EQU	0x11	;	Bank 0			
0012	PORTB	EQU	0x12					
	;							
0016	PIR	EQU	0x16	;	Bank 1			
0017	PIE	EQU	0x17					
	;							

0010		TMR1	EOU	0x10	; Bai	nk 2
0011		TMD 2	FOIL	0x11		
0011		11/ICZ	100	0.10		
0012		IMR31	EQU	UXIZ		
0013		TMR3h	EQU	0x13		
0014		PR1	EQU	0x14		
0015		PR2	EOU	0x15		
0016		DP 3T.	FOIL	0x16		
0010		PROL	100	0.10		
0017		PR3n ;	EQU	UX1/		
0010		PW1DCL	EOU	0x10	; Bai	nk 3
0011		PW2DCT.	FOIL	0x11		
0012		DW1DCH	FOU	0x12		
0012		PWIDCH	EQU	012		
0013		PWZDCH	EQU	UX13		
0016		TCON1	EQU	0x16		
0017		TCON2	EQU	0x17		
			OPC	0.20000	• 0m	igin for the PESET weater
	_		ORG	0x0000	, 01.	IGIN IOF CHE RESET VECCOF
0000 C02	3		G010	START	; On	reset, go to the start of the program
			ODC	00.0.0.9		igin for the outernal DAG (TNT
			OKG	0x0008	; 01.	interrupt vector
0008 0070	-		COTO	EXT INT	Got	to the ext interrupt
0000 007	-		0010	DAT_INT	; 00	on RA0/INT routine
			ORG	0x0010	; Or	igin for the RTCC
			0103	0X0010		avenflow interment weater
					, (	overilow interrupt vector
0010 C071	2		GOTO	RTCCINT	; Got	to the RTCC overflow interrupt
				0 0010		
			ORG	0×0018	; 0r:	igin for the external
					; 1	RA1/RT interrupt vector
0018 C07	2		GOTO	RT_INT	; Got	to the ext. interrupt on
					; 1	RA1/RT routine
			ORG	0x0020	; Or	igin for the interrupt vector
			0100	040020		of any enabled peripheral
		;				
		; The interrupt	routine	for any peripher	al in	nterrupt, This routine
		; The interrupt	routine	for any peripher	al in	nterrupt, This routine
		; The interrupt ; only deals wi :	routine th Timer	for any peripher 1 interrupt.	al in	nterrupt, This routine
		<pre>; The interrupt ; only deals wi ;</pre>	routine th Timer	for any peripher 1 interrupt.	al in	nterrupt, This routine
		<pre>; The interrupt ; only deals wi ; ; Time required</pre>	routine th Timer	for any peripher 1 interrupt. ute interrupt rou	al in	. Not including
		<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat</pre>	routine th Timer to exec ency (ti	for any peripher 1 interrupt. ute interrupt rou me to enter into	al in tine the :	nterrupt, This routine . Not including interrupt routine)
		<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ;</pre>	routine th Timer to exec ency (ti	for any peripher 1 interrupt. ute interrupt rou me to enter into	al in tine the :	nterrupt, This routine . Not including interrupt routine)
		<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel -</pre>	routine th Timer to exec ency (ti only T1	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow	al in tine the :	nterrupt, This routine . Not including interrupt routine) = 12 cycles
		<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 -</pre>	routine th Timer to exec ency (ti only T1 Other	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow	al in tine the :	nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop
		<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; ;</pre>	routine th Timer to exec ency (ti only T1 Other	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow	al in tine the :	nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop
		<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 - ;</pre>	routine th Timer to exec ency (ti only T1 Other	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow	al in tine the :	nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop
0020 500		; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; casel - ; case2 - ; ;	routine th Timer to exec ency (ti only Tl Other	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow	tine the t	nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop
0020 880	1	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; case2 - ; ; PER_INT</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow	tine : ; Se	nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1
0020 B80 0021 941	1	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 - ; ; PER_INT</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4	tine the : ; Se ; D:	nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow?
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; case2 - ; ; PER_INT EERROR</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR	the : ; Se ; D: ; No	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow.</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; case2 - ; ; PER_INT ERROR</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interr	tine the : ; Se ; D: ; No upts	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; case2 - ; ; PER_INT ERROR</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interru be enabled, so o	tine the : ; Se ; D: ; No upts error	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru ; be enabled, so o	the : ; Se ; D: ; No upts error	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; case2 - ; ; PER_INT ERROR ;</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru : be enabled, so o	tine the ; Se ; D: ; No error	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should .</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO ; ; led Time	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru : be enabled, so o r1 overflow occur	itine the ; Se ; D: ; No upts error	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should . he period register</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO ; i led Time is PWM w	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru : be enabled, so o r1 overflow occur aveform will rema	it ine the : ; Se ; D: ; No upts error cs, th uin u	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should he period register ntil the PWM duty</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; casel - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; ; led Time is PWM w r period	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interrr : be enabled, so o r1 overflow occur aveform will rema is updated. Unti	sal in tine the : ; So ; Do ; No upts error cs, th in un .1 suc	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should . he period register ntil the PWM duty ch update, there is no</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; i led Time is PWM w r period from Tl	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interrupt : be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in	al in tine the : ; Se ; D: ; No error cs, th .1 suc iterro	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow? ot a Timerl overflow. should . he period register ntil the PWM duty ch update, there is no upts can be disabled).</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ;</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO ; led Time is PWM w r period from T1	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interrupts : be enabled, so o r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in	the start in the start in the start in the start in the start is start in the start is start in the start in	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled).</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO ; ied Time is PWM w r period from T1 CH >= PR	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interru be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty	x stine time the strength stre	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should  he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100%</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; i led Time is PWM w r period from Tl CH >= PR	<pre>for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru : No other interrun : be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty</pre>	stine the : ; SG ; D ; No upts error cs, the in wull such terror cyclo	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is l00% ;</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO ; iled Time is PWM w r period from T1 CH >= PR	<pre>for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru : No other interrun : be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty</pre>	stine the : ; Sc ; D ; N upts error cs, tl in uu terror cyclo	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should . he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO is PWM w r period CH >= PR Period	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interrupt be enabled, so o r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va	sal in tine : ; SG ; D: ; No upts error cs, tl Lin uu terrr cyclo ulue,	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new ; than th</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO ; led Time is PWM w r period from T1 CH >= PR Period e value	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interru be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va in the Timer1 reg	ral in tine the : ; Sf ; D: ; N upts error rs, tl tin u tterro cycle tlue, ; ;ister	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should  he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value</pre>
0020 B80 0021 941 0022 C02	1 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new ; than th ; is load that the set of the set</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; ; led Time is PWM w r period from Tl CH >= PR Period e value ed that	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru- : No other interru- : No other interru- : No other interru- is updated. Unti interrupts (T1 into 1, then the duty register (PR1) va in the Timerl reg is less then the	ral in ttine tthe : SG ; DD ; NG error rs, tl suc tterror cyclo tterror cyclo ulue, ; jiste: TMR]	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? obt a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ;</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; led Time is PWM w r period from Tl CH >= PR Period e value e to cov	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interrupt be enabled, so of r1 overflow occur aveform will remains is updated. Unti interrupts (T1 interrupts (T1 interrupts) 1, then the duty register (PR1) va in the Timer1 register the	stal in titine the : ; SG ; D; ; No upts error cyclo ulterro cyclo ulterro cyclo ulterro TMRI must r TMRI	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will he peR1 value I E TMP1</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new ; than th ; is load ; continu ; cill</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO ; ied Time is PWM w r period from T1 CH >= PR Period e value ed that e to cou	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interru be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va in the Timer1 reg is less then the nt until it reach	sal in ttine ; SS ; D: ; Nu upts error ss, tl tin uu tterrr cycld ulue, ; jister TMR1 uest t	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should  he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will he PR1 value. I.E. TMR1 e the new PB1 walue</pre>
0020 B80 0021 941 0022 C02	1 5 2	<pre>; The interrupt ; only deals wi ; ; Time required ; interrupt lat ; ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is lo0% ; NOTE: The new ; is load ; is load ; is load ; NOTE: The new ; is load ; is load</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; ; led Time is PWM w r period from Tl CH >= PR Period e value ed that e to cou erflow a	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru- : No other interru- is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va in the Timerl reg is less then the nt until it reach t FFh and the cou	ral in tine ; Sq ; D ; No upts error cyclo tlue, ; jister TMR1 ues th mt to	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? obt a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will he PR1 value. I.E. TMR1 o the new PR1 value.</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new ; than th ; is load ; continu ; will ov ; Minimum</pre>	routine th Timer to exec ency (ti only T1 Other MOVLB BTFSS GOTO ; led Time is PWM w r period from T1 CH >= PR Period e value e to cou erflow a	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interrupt be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va in the Timerl reg is less then the nt until it reach t FFh and the cou ue is OAh, due to	stime since si	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should the period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will the PR1 value. I.E. TMR1 o the new PR1 value. e to load new values and</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is load ; ; NOTE: The new ; than th ; is load ; continu ; will ov ; Minimum ; execute</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO ; led Time is PWM w r period from T1 CH >= PR Period e value ed that e to cou erflow a pR1 val the per	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interru be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va in the Timerl reg is less then the nt until it reach t FFh and the cou ue is OAh, due to ipheral interrupt	<pre>sal in time the :: ; St ; D: ; Nu upts error ss, tl tin uutterr cycld llue, jistee TMR1 les tl unt td timt td timt td timt td ; time</pre>	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timer1 overflow? ot a Timer1 overflow. should  he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will he PR1 value. I.E. TMR1 o the new PR1 value. e to load new values and vice routine.</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new ; than th ; is load ; continu ; will ov ; Minimum ; execute ;</pre>	routine th Timer to exec ency (ti only Tl Other MOVLB BTFSS GOTO ; ied Time is PWM w r period from Tl CH >= PR Period e value ed that e to cou erflow a the per	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR : No other interru- : No other interru- : No other interru- : No other interru- is updated. Unti interrupts (T1 into- 1, then the duty register (PR1) va in the Timerl reg is less then the nt until it reach t FFh and the cou ue is OAh, due to ipheral interrupt	ral in tine ; Sr ; Di ; No upts error cyclo ulue, ; jister TMR1 mes th mut to ; time time t time	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will he PR1 value. I.E. TMR1 o the new PR1 value. e to load new values and vice routine.</pre>
0020 B80 0021 941 0022 C02	L 5 2	<pre>; The interrupt ; only deals wi ; interrupt lat ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is load ; ; NOTE: The new ; than th ; is load ; continu ; is load ; ; NOTE: The new ; than th ; is load ; continu ; will ov ; Minimum ; execute ; TIOVFL</pre>	routine th Timer to exec ency (ti Other MOVLB BTFSS GOTO ; led Time is PWM w r period from T1 CH >= PR Period e to cou erflow a the per BCF	for any peripher 1 interrupt. ute interrupt roume to enter into overflow 1 PIR,4 ERROR : No other interrupt : be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 int 1, then the duty register (PR1) va in the Timer1 reg is less then the the trimer1 reg is less then the turtil it reacht t FFh and the couv ue is OAh, due to ipheral interrupt PIR,4	<pre>sal in time the : ; SG ; D ; NG upts error cycld llue, pistee tmRl unt td timd ; serr ;</pre>	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PRI value value, the timer will he PRI value. I.E. TMR1 o the new PRI value. e to load new values and vice routine. Clear Overflow interrupt flag</pre>
0020 B80 0021 941 0022 C02 0022 C02	L 52 2	<pre>; The interrupt ; only deals wi ; Time required ; interrupt lat ; case1 - ; case2 - ; ; PER_INT ERROR ; ; Once the enab ; is loaded. Th ; cycle and / o ; S/W overhead ; ; NOTE: If PWID ; is 100% ; ; NOTE: The new ; than th ; is load ; continu ; will ov ; Minimum ; execute ; TIOVFL</pre>	routine th Timer to exec ency (ti Only T1 Other MOVLB BTFSS GOTO ; led Time is PWM w r period from T1 CH >= PR Period e value ed that e to cou erflow a PR1 val the per BCF MOVLB	for any peripher 1 interrupt. ute interrupt rou me to enter into overflow 1 PIR,4 ERROR No other interru be enabled, so of r1 overflow occur aveform will rema is updated. Unti interrupts (T1 in 1, then the duty register (PR1) va in the Timer1 reg is less then the nt until it reach t FFh and the cou ue is OAD, due to ipheral interrupt PIR,4 2	<pre>sal in time the : ; Sf ; D: ; Nu upts error cs, tl in uu l su terru cycl ulue, ; jiste: TMR1 tc o timu ; ; ;</pre>	<pre>nterrupt, This routine . Not including interrupt routine) = 12 cycles = Infinite Loop elect register Bank 1 id Timerl overflow? ot a Timerl overflow. should  he period register ntil the PWM duty ch update, there is no upts can be disabled). e of this PWM output must always be greater r (TMR1). If a PR1 value value, the timer will he PR1 value. I.E. TMR1 o the new PR1 value. e to load new values and vice routine. Clear Overflow interrupt flag Bank2</pre>

0026 B801		MOVLB	1	; Bank 0
0027 8C17		BCF	PIE, 4	; Disable T1 interrupt
			-	; (until transition on PORTB0)
0028 B800		MOVLB	0	; Bank 0
0029 3F12		BTG	PORTB 7	Transition PortB 7 pin (H->L
0023 0005		PETRIE	ronib, ,	; Return from Interrupt
00211 0005		101110		, Recard from incertape
	, Thia ia tho	atort of	the program	
	, into is the	Start Or	the program.	
0000 0406	/ 	DOF	CDUCERN 4	· Dischle AII interments wis the
0028 0400	START	DOF	CPUSIA,4	, Disable All interrupts via the
				(GLINTE) bit
				, (GLINID) DIC.
	MAIN			· Dlago Main program horo
0000 8800	MAIN	MOTA	2	; Colort mariator Dark 2
0020 8803		GLDE	5 TECONT2 0	, Step the timera Gingle Centure
002D 2617		MONTH	1CON2,0	, Stop the timers, Single Capture
002E B070		MOVEW	TCON1	$\tau$ m1 (9 bi+) m2 (9 bi+)
002F 0110		MOVWE	ICONI	, II (0-DIC), IZ (0-DIC),
				, and is full off the internal
				, system clock. limers uses
0020 0000		MOVIT	0x0D	· period register
0030 8000		MOVLW	DEM HIN	, Load the PWM window Cycle value
0031 0125		MOVWF	PWM_WIN	i
0000	i		0	
0032 B800		MOVLB	0	; Select register Bank U
0033 2811		SEIF	DDRB, I	, Port B is an input
0034 2912		CLRF	PORTB, 1	; Set output values to 0 (for
0035 8F11		BCF	DDRB, 7	; PORTB/ is an output. Used to
0036 2927		CLRF	FLAG_REG, 1	; Clear the Flag registers
	,			
	; Load registe	ers with 1	the PWM values that	we will switch between. One set
	; for the time	e PORTBO :	is high and another	set for when low.
	;			
	; For a high 1	level the	PWM registers are	updated as follows:
	; For a high 1 ; PR1 = 7F1	level the h, PW1DCH	PWM registers are = 3Fh, and PW1DCL =	updated as follows: = 40h
	; For a high ; ; PR1 = 7F1 ; At 16Mhz	level the h, PW1DCH this give	PWM registers are = 3Fh, and PW1DCL es a period of 31.7	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us
	; For a high ; ; PR1 = 7F1 ; At 16Mhz ; For a low le	level the h, PW1DCH this give evel the D	PWM registers are = 3Fh, and PW1DCL es a period of 31.7 PWM registers are up	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows:
	; For a high ; ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1	level the h, PW1DCH this give evel the D h, PW1DCH	PWM registers are = 3Fh, and PW1DCL es a period of 31.7 PWM registers are up = 07h, and PW1DCL	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h
	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1 ; At 16Mhz	level the h, PW1DCH this give evel the 1 h, PW1DCH this give	PWM registers are = 3Fh, and PW1DCL es a period of 31.7 PWM registers are up = 07h, and PW1DCL es a period of 7.75	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us
	; For a high 1 ; PR1 = 7F1 ; At 16Mhz ; For a low 1¢ ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this give evel the D h, PW1DCH this give	PWM registers are = 3Fh, and PW1DCL es a period of 31.7 PWM registers are u = 07h, and PW1DCL es a period of 7.75	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us
0037 B803	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low 16 ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this give evel the D h, PW1DCH this give MOVLB	PWM registers are a = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3
0037 B803 0038 B03F	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this give evel the 1 h, PW1DCH this give MOVLB MOVLB	PWM registers are a = 3Fh, and PW1DCL as a period of 31.7 PWM registers are up = 07h, and PW1DCL as a period of 7.75 3 0x3F	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is
0037 B803 0038 B03F 0039 4A1A	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low le ; PRl = 1Fl ; At 16Mhz ;	level the h, PW1DCH this give evel the D h, PW1DCH this give MOVLB MOVLB MOVLW MOVPF	PWM registers are a = 3Fh, and PWIDCL es a period of 31.7 PWM registers are u = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period
0037 B803 0038 B03F 0039 4A1A 003A B040	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this give evel the D h, PW1DCH this give MOVLB MOVLB MOVLW MOVPF MOVLW	PWM registers are a = 3Fh, and PWlDCL es a period of 31.7 PWM registers are u = 07h, and PWlDCL es a period of 7.75 3 0x3F W, DC1H 0x40	updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ;
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B	; For a high 1 ; PR1 = 7F1 ; At 16Mhz ; For a low 1 ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this give evel the 1 h, PW1DCH this give MOVLB MOVLB MOVLB MOVLW MOVPF MOVLW MOVPF	PWM registers are a = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low 1 ; PRl = 1Fl ; At 16Mhz ;	level the h, PW1DCH this give evel the 1 h, PW1DCH this give MOVLB MOVLB MOVLW MOVPF MOVLW	PWM registers are a = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is</pre>
0037 B803 0038 B03F 0039 4A1A 0038 B040 003B 4A1B 003C B007 003D 4A1D	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low ld ; PRl = 1Fl ; At 16Mhz ;	level the h, PWlDCH this give evel the J h, PWlDCH this give MOVLB MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF	PWM registers are of = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low 1¢ ; PR1 = 1F1 ; At 16Mhz ;	level the h, PWlDCH this giv. evel the J h, PWlDCH this giv. MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW	PWM registers are a = 3Fh, and PWIDCL as a period of 31.7 PWM registers are u = 07h, and PWIDCL as a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ;</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this give evel the J h, PW1DCH this give MOVLB MOVLB MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF	PWM registers are a = 3Fh, and PWIDCL as a period of 31.7 PWM registers are up = 07h, and PWIDCL as a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10</pre>
0037 B803 0038 B03F 0039 4A1A 0038 B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low l ; PRl = 1Fl ; At 16Mhz ;	level the h, PW1DCH this give evel the J h, PW1DCH this give MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF	PWM registers are a = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low ld ; PRl = 1Fl ; At 16Mhz ;	level the h, PWlDCH this give evel the J h, PWlDCH this give MOVLB MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB	PWM registers are r = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1H 0x07 W, DC1L 0x80 W, DC1QL 2	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low 14 ; PR1 = 1F1 ; At 16Mhz ;	level the h, PWlDCH this giv. evel the J h, PWlDCH this giv. MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB	PWM registers are s = 3Fh, and PWIDCL es a period of 31.7 PWM registers are u = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x70 W, DC1L 0x80 W, DC1QL 2 0x7F	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ;</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low ld ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this givv evel the J h, PW1DCH this givv MOVLB MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB	PWM registers are s = 3Fh, and PWIDCL es a period of 31.7 PWM registers are u = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50%</pre>
0037 B803 0038 B03F 0039 4A1A 0038 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low l ; PRl = 1Fl ; At 16Mhz ;	level the h, PW1DCH this giv, evel the J h, PW1DCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB	<pre>PWM registers are r = 3Fh, and PWIDCL : es a period of 31.7 PWM registers are up = 07h, and PWIDCL : es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High)</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low ld ; PRl = 1Fl ; At 16Mhz ;	level the h, PWIDCH this give evel the J h, PWIDCH this give MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB	PWM registers are a = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1H 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High)</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low 1d ; PR1 = 1F1 ; At 16Mhz ;	level the h, PWlDCH this giv, evel the J h, PWlDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLW MOVPF	<pre>PWM registers are s = 3Fh, and PWIDCL as a period of 31.7 PWM registers are u = 07h, and PWIDCL ss a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1L ; of full scale 0x1F W, PR1L</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C	; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low 16 ; PR1 = 1F1 ; At 16Mhz ;	level the h, PW1DCH this givv evel the J h, PW1DCH this givv MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLW MOVPF	<pre>PWM registers are s = 3Fh, and PWIDCL es a period of 31.7 PWM registers are u = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x40 W, DC1QH 0x07 W, DC1LL 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low)</pre>
0037 B803 0038 B03F 0039 4A1A 0038 B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low l ; PRl = 1Fl ; At 16Mhz ; ;	level the h, PW1DCH this giv, evel the J h, PW1DCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLB MOVLB	<pre>PWM registers are r = 3Fh, and PWIDCL : es a period of 31.7 PWM registers are up = 07h, and PWIDCL : es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low)</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low ld ; PR1 = 1F1 ; At 16Mhz ; ; ; ;</pre>	level the h, PWIDCH this giv, evel the J h, PWIDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLB MOVLB	<pre>PWM registers are s = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1H 0x40 W, DC1L 0x80 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low)</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1 ; At 16Mhz ; ; ; ; ; Default PWM</pre>	level the h, PW1DCH this giv, evel the J h, PW1DCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB	<pre>PWM registers are s = 3Fh, and PWIDCL as a period of 31.7 PWM registers are u = 07h, and PWIDCL ss a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L anould be set, and times </pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low)</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PR1 = 7Fi ; At 16Mhz ; For a low ld ; PR1 = 1Fi ; At 16Mhz ; ; ; ; ; ; ; Default PWM ; and the infe</pre>	level the h, PW1DCH this giv. evel the J h, PW1DCH this giv. MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB MOVPF MOVLB MOVLB MOVPF MOVLB MOVLB MOVPF MOVLB	<pre>PWM registers are s = 3Fh, and PWIDCL es a period of 31.7 PWM registers are u = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x40 W, DC1QL 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L hould be set, and thabled.</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low)</pre>
0037 B803 0038 B03F 0039 4A1A 0038 B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low le ; PRl = 1Fl ; At 16Mhz ; ; ; ; Default PWM; and the inte;</pre>	level the h, PW1DCH this giv, evel the J h, PW1DCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB MOVLB	<pre>PWM registers are r = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PRIL ; of full scale 0x1F W, PRIL pould be set, and the pabled.</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low ld ; PRl = 1Fl ; At 16Mhz ; ; ; ; ; Default PWM ; and the inte;</pre>	level the h, PWIDCH this giv, evel the J h, PWIDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLW MOVPF MOVLB	<pre>PWM registers are r = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1H 0x40 W, DC1L 0x80 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L hould be set, and the habled. 0xF0</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started ; Load the Period register</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low le ; PR1 = 1F1 ; At 16Mhz ; ; ; ; ; Default PWM ; and the inte;</pre>	level the h, PWIDCH this giv, evel the J h, PWIDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLW MOVPF Values sl errupts en MOVLW	<pre>PWM registers are s = 3Fh, and PWIDCL = as a period of 31.7 PWM registers are u = 07h, and PWIDCL = oxar 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L = nould be set, and t1 nabled. 0xF0 PR1</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started ; Load the Period register ;</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low le ; PRl = 1Fl ; At 16Mhz ; ; ; ; ; ; Default PWM ; and the inte;</pre>	level the h, PW1DCH this giv. evel the J h, PW1DCH this giv. MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLB MOVLB MOVLB MOVPF Values SB errupts en MOVLW MOVWF	<pre>PWM registers are s = 3Fh, and PWIDCL es a period of 31.7 PWM registers are u = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x40 W, DC1QL 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L hould be set, and the habled. 0xF0 PR1 3</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started ; Load the Period register ; ; Select register Bank 3</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low le ; PRl = 1Fl ; At 16Mhz ; ; ; ; Default PWM ; and the inte;</pre>	level the h, PWIDCH this giv, evel the J h, PWIDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLB MOVLW MOVPF values sl errupts er MOVLW	<pre>PWM registers are of = 3Fh, and PWIDCL : es a period of 31.7 PWM registers are up = 07h, and PWIDCL : es a period of 7.75</pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started ; Select register Bank 3 ; Load the Period register ; ; Select register Bank 3 ; Load the TL duty cycle register</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PRl = 7Fl ; At 16Mhz ; For a low le ; PRl = 1Fl ; At 16Mhz ; ; ; ; ; ; ; Default PWM ; and the inte;</pre>	level the h, PWIDCH this giv, evel the J h, PWIDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF Values sl errupts en MOVLW MOVLW MOVLW MOVLW	PWM registers are of = 3Fh, and PWIDCL es a period of 31.7 PWM registers are up = 07h, and PWIDCL es a period of 7.75 3 0x3F W, DC1H 0x40 W, DC1QH 0x07 W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L hould be set, and the habled. 0xF0 PR1 3 0xC0 PW1DCH	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started ; Load the Period register ; ; Select register Bank 3 ; Load the TI duty cycle register</pre>
0037 B803 0038 B03F 0039 4A1A 003A B040 003B 4A1B 003C B007 003D 4A1D 003E B080 003F 4A1E 0040 B802 0041 B07F 0042 4A1C 0043 B01F 0044 4A1F	<pre>; For a high : ; PR1 = 7F1 ; At 16Mhz ; For a low ld ; PR1 = 1F1 ; At 16Mhz ; ; ; ; ; ; Default PWM ; and the inte ;</pre>	level the h, PWIDCH this giv, evel the J h, PWIDCH this giv, MOVLB MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF MOVLW MOVPF Values sl errupts en MOVLW MOVLW MOVLB MOVLW MOVLB MOVLW MOVLB	<pre>PWM registers are s = 3Fh, and PWIDCL = as a period of 31.7 PWM registers are u = 07h, and PWIDCL = 07h, and PWIDCL = 07h, and PWIDCL = 0x1F W, DC1L 0x80 W, DC1QL 2 0x7F W, DC1L 0x80 W, DC1QL 2 0x7F W, PR1H ; of full scale 0x1F W, PR1L = nould be set, and t1 nabled. 0xF0 PR1 3 0xC0 PWIDCH PWIDCL </pre>	<pre>updated as follows: = 40h 5 us, and a duty cycle of 16.625 us pdated as follows: = 80h us, and a duty cycle of 6.00 us ; Bank 3 ; The Duty Cycle initial value is ; 50% of the initial period ; ; Duty Cycle low = 01 ; The Duty Cycle initial value is ; 25% of the initial period ; ; Duty Cycle low = 10 ; Bank 2 ; ; The initial period value is 50% (for High) ; The initial period value is ; of full scale (for Low) he timer should be started ; Load the Period register ; ; Select register Bank 3 ; Load the T1 duty cycle register ; ; affectively loaded with 0</pre>

004B B031 004C 0117 004D 8307 004E B801 004F B010 0050 0117 0051 8C06 0052 B800		MOVLW MOVWF BSF MOVLB MOVLW MOVWF BCF MOVLB	0x31 TCON2 INTSTA,3 1 0x10 PIE CPUSTA,4 0	<pre>;** Enable PWM1 and PWM2 outputs ;** and turn on Timer1. ; Turn on Peripheral Interrupts ; Select register Bank 1 ; Enable Timer1 overflow ; Interrupts (when GLINTD = 0) ; Enable ALL interrupts ; Bank 0</pre>
	; ; Only need ; Else loop	to update waiting f	PWM values on the or level to change.	first occurance of a new level on
0053 8827 0054 9012 0055 C05F 0056 9827 0057 C054 0058 8027	; HIGH1ST HIGHCYC ;	BCF BTFSS GOTO BTFSC GOTO BSF	FLAG_REG, 0 PORTB, 0 LOW1ST FLAG_REG, 0 HIGHCYC FLAG_REG, 0	<pre>; First time in loop (this cycle) = True ; Is PortB0 low ; PORTB0 = L ; Is this the First High time(this cycle)? ; Loop looking for low signal on PortB0 ; Set First time in loop(this cycle)=False</pre>
	; Here is wh	ere we up	date the PWM values	(period and Duty cycle) for high
0059 B803 005A 5A20 005B 5B21 005C 5C22 005D E06B 005E C054	;	MOVLB MOVPF MOVPF MOVPF CALL GOTO	3 DC1H, NEW_DC1 DC1QH, NEW_DC1Q PR1H, NEW_PR1 PWM1_UD HIGHCYC	; Bank 3 ; ; ; ; ; Loop looking for low signal on
005F 8827 0060 9812 0061 C053 0062 9827 0063 C060 0064 8027	; LOWIST LOWCYC ; Here is wh	BCF BTFSC GOTO BTFSC GOTO BSF ere we up	FLAG_REG, 0 PORTB, 0 HIGHIST FLAG_REG, 0 LOWCYC FLAG_REG, 0 date the PWM values	<pre>; First time in loop (this ; Is PortB0 high ; PORTB0 = H ; Is this the First Low time ; Loop looking for high signal on PortB0 ; First time in loop (this ; (period and Duty cycle) for low</pre>
0065 B803 0066 5D20 0067 5E21 0068 5F22 0069 E06B 006A C060	;	MOVLB MOVPF MOVPF MOVPF CALL GOTO	3 DC1L, NEW_DC1 DC1QL, NEW_DC1Q PR1L, NEW_PR1 PWM1_UD LOWCYC	; Bank 3 ; ; ; ; ; Loop looking for high signal
	; ; This code ; ; are update; ; is at leas ; If not a "; ; cycle regi ; will latch	segment e d at the t PWM_WIN glitch" c ster is l the 2nd	ensure that all PWM same time. This is ((0Dh) cycles befor could occur in the P atched for this PWM duty cycle register	values (period and duty cycle) done by ensuring that the Timer e the PR1 value (PR1 - PWM_WIN > WM waveform. When only the 1st duty cycle, and the following PWM period
006B 8406 006C 8802 006D 6A10 006E 0414 006F 3025 0070 C06B 0071 B803 0072 6A20 0073 0112 0074 6A21	PWM1_UD	BSF MOVLB MOVFP SUBWF CPFSLT GOTO MOVLB MOVFP MOVWF MOVFP	CPUSTA, 4 2 TMR1, W PR1, 0 PWM_WIN PWM1_UD 3 NEW_DC1, W PW1DCH NEW_DC1Q, W	<pre>; Disable Global Interrupts ; Bank 2 ; Load W reg. with Timerl value ; PR1 - TMR1 -&gt; W reg. ; Check if Timerl is about to ; Overflow would have occurred ; Overflow would have occurred ; DWM updates, Delay a few ; Bank 3 ; Your New PWM MSB ; Loaded in duty cycle buffer ; Your New PWM LSB</pre>
0075 0110 0076 B801 0077 8C16		MOVWF MOVLB BCF	PWIDCL 1 PIR, 4	; Loaded in duty cycle buffer ; Back to Bank 1 ; Clear T1 Overflow interrupt

0078 8417 0079 8C06 007A 8800 007B 0002	;	BSF BCF MOVLB RETURN	PIE, 4 CPUSTA, 4 O	<pre>; Enable T1 int ; Enable Global Interrupts ; Bank 0 ;** this does not need to be ;** as a subroutine.</pre>
	; Other Interru	pt routi	nes. (Not utilized	in this example)
007C 0005	EXT_INT	RETFIE		; RAO/INT interrupt routine ; (NOT used in this program)
007D 0005	RTCCINT	RETFIE		; RTCC overflow interrupt routine ; (NOT used in this program)
007E 0005	RT_INT	RETFIE		<pre>; RA1/RT interrupt routine ; (NOT used in this program) ;</pre>
007F C02B	SRESET ; ; ; When the exec	GOTO uted add	START ress is NOT in the	; If program became lost, goto ; START and reinitalize. program range, the
	<pre>; 16-bit addres ; this location ; shut down from ;</pre>	s should you cou m the in	contain all l's (a ld branch to a rout valid program execu	CALL 0x1FFF). At ine to recover or tion.
07FF C07F		ORG GOTO END	END_OF_PROG_MEM SRESET	; ; The program has lost it's mind, ; do a system reset

Errors : 0 Warnings : 0

<sup>© 1993</sup> Microchip Technology Inc.

NOTES:

## WORLDWIDE SALES & SERVICE

#### AMERICAS

#### **Corporate Office**

Microchip Technology Inc. 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 602 786-7200 Fax: 602 786-7277 Technical Support: 602 786-7627 Web: http://www.mchip.com/microhip

#### Atlanta

Microchip Technology Inc. 500 Sugar Mill Road, Suite 200B Atlanta, GA 30350 Tel: 770 640-0034 Fax: 770 640-0307 Boston Microchip Technology Inc. 5 Mount Royal Avenue Marlborough, MA 01752 Tel: 508 480-9990 Fax: 508 480-8575 Chicago Microchip Technology Inc. 333 Pierce Road, Suite 180 Itasca, IL 60143 Tel: 708 285-0071 Fax: 708 285-0075 Dallas Microchip Technology Inc. 14651 Dallas Parkway, Suite 816 Dallas, TX 75240-8809 Tel: 214 991-7177 Fax: 214 991-8588 Dayton Microchip Technology Inc. 35 Rockridge Road Englewood, OH 45322 Tel: 513 832-2543 Fax: 513 832-2841 Los Angeles Microchip Technology Inc. 18201 Von Karman, Suite 455 Irvine, CA 92715 Tel: 714 263-1888 Fax: 714 263-1338 **New York** Microchip Technology Inc. 150 Motor Parkway, Suite 416 Hauppauge, NY 11788 Tel: 516 273-5305 Fax: 516 273-5335

#### **AMERICAS** (continued)

San Jose

Microchip Technology Inc. 2107 North First Street, Suite 590 San Jose, CA 95131 Tel: 408 436-7950 Fax: 408 436-7955

#### ASIA/PACIFIC

Hong Kong Microchip Technology Unit No. 3002-3004, Tower 1 Metroplaza 223 Hing Fong Road Kwai Fong, N.T. Hong Kong Tel: 852 2 401 1200 Fax: 852 2 401 3431 Korea Microchip Technology 168-1, Youngbo Bldg. 3 Floor Samsung-Dong, Kangnam-Ku, Seoul, Korea Tel: 82 2 554 7200 Fax: 82 2 558 5934 Singapore Microchip Technology 200 Middle Road #10-03 Prime Centre Singapore 188980 Tel: 65 334 8870 Fax: 65 334 8850 Taiwan Microchip Technology 10F-1C 207 Tung Hua North Road

### Taipei, Taiwan, ROC Tel: 886 2 717 7175 Fax: 886 2 545 0139

#### EUROPE

United Kingdom Arizona Microchip Technology Ltd. Unit 6, The Courtyard Meadow Bank, Furlong Road Bourne End, Buckinghamshire SL8 5AJ Tel: 44 0 1628 851077 Fax: 44 0 1628 850259 France Arizona Microchip Technology SARL 2 Rue du Buisson aux Fraises

91300 Massy - France Tel: 33 1 69 53 63 20 Fax: 33 1 69 30 90 79

#### Germany

Arizona Microchip Technology GmbH Gustav-Heinemann-Ring 125 D-81739 Muenchen, Germany Tel: 49 89 627 144 0 Fax: 49 89 627 144 44 Italy

Arizona Microchip Technology SRL Centro Direzionale Colleoni Palazzo Pegaso Ingresso No. 2 Via Paracelso 23, 20041 Agrate Brianza (MI) Italy Tel: 39 039 689 9939 Fax: 39 039 689 9883

#### JAPAN

Microchip Technology Intl. Inc. Benex S-1 6F 3-18-20, Shin Yokohama Kohoku-Ku, Yokohama Kanagawa 222 Japan Tel: 81 45 471 6166 Fax: 81 45 471 6122

9/22/95

All rights reserved. © 1995, Microchip Technology Incorporated, USA.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.