

PROTON+ TIMER MACROS

The background of the image is a close-up photograph of an antique clock. The clock face is white with black Roman numerals. The hands are ornate and made of dark metal. The clock is surrounded by intricate, dark metalwork, possibly brass or iron, with a patina that suggests age. The lighting is dramatic, highlighting the textures and curves of the metal.

PROTON+ TIMER Macros.

Crownhill reserves the right to make changes to the products contained in this publication in order to improve design, performance or reliability. Neither Crownhill Associates Limited or the author shall be responsible for any claims attributable to errors omissions or other inaccuracies in the information or materials contained in this publication and in no event shall Crownhill Associates or the author be liable for direct indirect or special incidental or consequential damages arising out of the use of such information or material. Neither Crownhill or the author convey any license under any patent or other right, and make no representation that the circuits are free of patent infringement. Charts and schedules contained herein reflect representative operating parameters, and may vary depending upon a user's specific application.

All terms mentioned in this document that are known to be trademarks or service marks have been appropriately marked. Use of a term in this publication should not be regarded as affecting the validity of any trademark.

PICmicrotm is a trade name of Microchip Technologies Inc. www.microchip.com

PROTONtm is a trade name of Crownhill Associates Ltd. www.crownhill.co.uk

Web url's are correct at the time of publication

The PROTON+ compiler and documentation was written by Les Johnson.

All Manufacturer Trademarks Acknowledged

PROTON+ TIMER Macros.

Table Of Contents

Introduction	3
OPEN_TIMER0	4
OPEN_TIMER1	5
OPEN_TIMER2	6
OPEN_TIMER3	7
OPEN_TIMER4	8
CLOSE_TIMER0	9
CLOSE_TIMER1	9
CLOSE_TIMER2	9
CLOSE_TIMER3	9
CLOSE_TIMER4	9
READ_TIMER0	10
READ_TIMER1	10
READ_TIMER2	10
READ_TIMER3	10
READ_TIMER4	10
WRITE_TIMER0	11
WRITE_TIMER1	11
WRITE_TIMER2	11
WRITE_TIMER3	11
WRITE_TIMER4	11

PROTON+ TIMER Macros.

Introduction

Setting up the various timers on PICmicros can be a tedious task, with constant referrals to the, sometimes obscure, datasheets. Therefore I have created a series of macros to help alleviate some of the problems.

The macros are only for use with 18F devices, but could easily be converted for 16F types. They should also be used only with the PROTON+ Compiler version 3.1 onwards.

To use a macro for a particular timer, simply include the appropriate timer file in the main BASIC program.

The files available are: -

TIMER0_18.INC	Timer0 Macros
TIMER1_18.INC	Timer1 Macros
TIMER2_18.INC	Timer2 Macros
TIMER3_18.INC	Timer3 Macros
TIMER4_18.INC	Timer4 Macros

Copy all of these files into the compiler's INC folder and they will be available to all programs without specifying a location.

Of course, this doesn't mean that the datasheet is redundant, and the datasheet for the particular PICmicro you're using should always be at hand, but I hope it will help overcome some of the hurdles when dealing with timers.

PROTON+ TIMER Macros.

OPEN_TIMER0

Syntax

OPEN_TIMER0 *configs*

Overview

Configure Timer0

Operator

Configs is a bitmask that is created by performing a bitwise AND operation ('&') with a value from each of the categories listed below. These values are defined in the TIMER0_18.INC file.

Enable Timer0 Interrupt:

T0_INT_ON	Interrupt enabled
T0_INT_OFF	Interrupt disabled

Timer Width:

T0_8BIT	8-bit mode
T0_16BIT	16-bit mode

Clock Source:

T0_SOURCE_EXT	External clock source (I/O pin)
T0_SOURCE_INT	Internal clock source (TOSC)

External Clock Trigger (for T0_SOURCE_EXT):

T0_EDGE_FALL	External clock on falling edge
T0_EDGE_RISE	External clock on rising edge

Prescale Value:

T0_PS_1_1	1:1 prescale
T0_PS_1_2	1:2 prescale
T0_PS_1_4	1:4 prescale
T0_PS_1_8	1:8 prescale
T0_PS_1_16	1:16 prescale
T0_PS_1_32	1:32 prescale
T0_PS_1_64	1:64 prescale
T0_PS_1_128	1:128 prescale
T0_PS_1_256	1:256 prescale

Example

```
OPTIMISER_LEVEL = 6      ` Enable the optimiser
Device = 18F452          ` Choose a 16-bit core device
Include "TIMER0_18.INC" ` Include the macros into the program
OPEN_TIMER0 [T0_INT_OFF & T0_8BIT & T0_SOURCE_INT & T0_PS_1_32]
```

PROTON+ TIMER Macros.

OPEN_TIMER1

Syntax

OPEN_TIMER1 *configs*

Overview

Configure Timer1

Operator

Configs is a bitmask that is created by performing a bitwise AND operation ('&') with a value from each of the categories listed below. These values are defined in the TIMER1_18.INC file.

Timer Interrupt:

T1_INT_ON	Interrupt enabled
T1_INT_OFF	Interrupt disabled

Timer Width:

T1_8BIT_RW	8-bit read-write mode
T1_16BIT_RW	16-bit read-write mode

Clock Source:

T1_SOURCE_EXT	External clock source (I/O pin)
T1_SOURCE_INT	Internal clock source (TOSC)

Prescaler:

T1_PS_1_1	1:1 prescale
T1_PS_1_2	1:2 prescale
T1_PS_1_4	1:4 prescale
T1_PS_1_8	1:8 prescale

Oscillator Use:

T1_OSC1EN_ON	Enable Timer1 oscillator
T1_OSC1EN_OFF	Disable Timer1 oscillator

Synchronize Clock Input:

T1_SYNC_EXT_ON	Sync external clock input
T1_SYNC_EXT_OFF	Don't sync external clock input

Example

```
OPTIMISER_LEVEL = 6           ` Enable the optimiser
Device = 18F452               ` Choose a 16-bit core device
Include "TIMER1_18.INC"      ` Include the macros into the program
OPEN_TIMER1 [T1_INT_OFF & T1_8BIT_RW & T1_SOURCE_EXT & T1_PS_1_1 &
                T1_OSC1EN_OFF & T1_SYNC_EXT_OFF & T1_SOURCE_CCP]
```

PROTON+ TIMER Macros.

OPEN_TIMER2

Syntax

OPEN_TIMER2 *configs*

Overview

Configure Timer2

Operator

Configs is a bitmask that is created by performing a bitwise AND operation ('&') with a value from each of the categories listed below. These values are defined in the TIMER2_18.INC file.

Timer Interrupt:

T2_INT_ON	Interrupt enabled
T2_INT_OFF	Interrupt disabled

Prescaler:

T2_PS_1_1	1:1 prescale
T2_PS_1_4	1:4 prescale
T2_PS_1_16	1:16 prescale

Postscale Value:

T2_POST_1_1	1:1 postscale
T2_POST_1_2	1:2 postscale
::	
T2_POST_1_15	1:15 postscale
T2_POST_1_16	1:16 postscale

Example

```
OPTIMISER_LEVEL = 6      ` Enable the optimiser
Device = 18F452          ` Choose a 16-bit core device
Include "TIMER2_18.INC" ` Include the macros into the program
OPEN_TIMER2 [T2_INT_OFF & T2_PS_1_1 & T2_POST_1_8]
```

PROTON+ TIMER Macros.

OPEN_TIMER3

Syntax

OPEN_TIMER3 *configs*

Overview

Configure Timer3

Operator

Configs is a bitmask that is created by performing a bitwise AND operation ('&') with a value from each of the categories listed below. These values are defined in the TIMER3_18.INC file.

Timer Interrupt:

T3_INT_ON	Interrupt enabled
T3_INT_OFF	Interrupt disabled

Timer Width:

T3_8BIT_RW	8-bit mode
T3_16BIT_RW	16-bit mode

Clock Source:

T3_SOURCE_EXT	External clock source (I/O pin)
T3_SOURCE_INT	Internal clock source (TOSC)

Prescale Value:

T3_PS_1_1	1:1 prescale
T3_PS_1_2	1:2 prescale
T3_PS_1_4	1:4 prescale
T3_PS_1_8	1:8 prescale

Synchronize Clock Input:

T3_SYNC_EXT_ON	Sync external clock input
T3_SYNC_EXT_OFF	Don't sync external clock input

Use With CCP:

T1_SOURCE_CCP	Timer1 source for both CCP's
T3_SOURCE_CCP	Timer3 source for both CCP's
T1_CCP1_T3_CCP2	Timer1 source for CCP1 and Timer3 source for CCP2

Example

```
OPTIMISER_LEVEL = 6           ` Enable the optimiser
Device = 18F8720             ` Choose a 16-bit core device with a timer3
Include "TIMER3_18.INC"      ` Include the macros into the program
OPEN_TIMER3 [T3_INT_OFF & T3_8BIT_RW & T3_SOURCE_EXT & T3_PS_1_1 &
                T3_OSC1EN_OFF & T3_SYNC_EXT_OFF & T3_SOURCE_CCP]
```

PROTON+ TIMER Macros.

OPEN_TIMER4

Syntax

OPEN_TIMER4 *configs*

Overview

Configure Timer4

Operator

Configs is a bitmask that is created by performing a bitwise AND operation ('&') with a value from each of the categories listed below. These values are defined in the TIMER4_18.INC file.

Timer Interrupt:

T4_INT_ON	Interrupt enabled
T4_INT_OFF	Interrupt disabled

Prescale Value:

T4_PS_1_1	1:1 prescale
T4_PS_1_4	1:4 prescale
T4_PS_1_16	1:16 prescale

Postscale Value:

T4_POST_1_1	1:1 postscale
T4_POST_1_2	1:2 postscale
::	
T4_POST_1_15	1:15 postscale
T4_POST_1_16	1:16 postscale

Example

```
OPTIMISER_LEVEL = 6           ` Enable the optimiser
Device = 18F8720              ` Choose a 16-bit core device with a timer4
Include "TIMER4_18.INC"      ` Include the macros into the program
OPEN_TIMER4 [T4_INT_OFF & T4_PS_1_1 & T4_POST_1_8]
```

PROTON+ TIMER Macros.

CLOSE_TIMER0

CLOSE_TIMER1

CLOSE_TIMER2

CLOSE_TIMER3

CLOSE_TIMER4

Overview

Disable the specified timer.

Example

```
OPTIMISER_LEVEL = 6           ` Enable the optimiser
Device = 18F452                ` Choose a 16-bit core device
Include "TIMER0_18.INC"        ` Include the Timer0 macros into the program
Include "TIMER1_18.INC"        ` Include the Timer1 macros into the program
Include "TIMER2_18.INC"        ` Include the Timer2 macros into the program
```

CLOSE_TIMER0

CLOSE_TIMER1

CLOSE_TIMER2

Notes

These macros disable the appropriate interrupt and specified timer.

PROTON+ TIMER Macros.

READ_TIMER0

READ_TIMER1

READ_TIMER2

READ_TIMER3

READ_TIMER4

Syntax

Variable = **READ_TIMER0** *Timer_Size*

Variable = **READ_TIMER1**

Variable = **READ_TIMER2**

Variable = **READ_TIMER3** *Timer_Size*

Variable = **READ_TIMER4**

Overview

Read the value of the specified timer.

Operators

Variable Can be a **BYTE**, **WORD** or **DWORD** variable that will contain the value of the specified timer.

Timer_Size Can be the constant value **8** or **16** depending on which timer is being used. Value 8 will read the low byte of a 16-bit timer, and 16 will read the full timer. Not all timers are 16-bit types, therefore the *Timer_Size* parameter is not required for Timers 1, 2 and 4

Example

```
OPTIMISER_LEVEL = 6           ` Enable the optimiser
Device = 18F452              ` Choose a 16-bit core device
Include "TIMER1_18.INC"      ` Include the Timer1 macros into the program
```

```
Dim WRD1 as Word
```

```
OPEN_TIMER1 [T1_INT_OFF & T1_16BIT_RW & T1_SOURCE_INT & T1_PS_1_1 &
              T1_OSC1EN_OFF & T1_SYNC_EXT_OFF]
```

```
WRD1 = READ_TIMER1 ` Read Timer1 into variable WRD1
```

PROTON+ TIMER Macros.

WRITE_TIMER0

WRITE_TIMER1

WRITE_TIMER2

WRITE_TIMER3

WRITE_TIMER4

Syntax

WRITE_TIMER0 *Timer_Size* , *Variable*

WRITE_TIMER1 *Variable*

WRITE_TIMER2 *Variable*

WRITE_TIMER3 *Timer_Size* , *Variable*

WRITE_TIMER4 *Variable*

Overview

Write a value into the specified timer.

Operators

Timer_Size Can be the constant value **8** or **16** depending on which timer is being used. Value 8 will write only to the low byte of a 16-bit timer, and 16 will write to the full timer. Not all timers are 16-bit types, therefore the *Timer_Size* parameter is not required for Timers 1, 2 and 4.

Variable Can be a **BYTE**, **WORD** or **DWORD** variable, or a user constant value that will be written to the specified timer.

Example

```
OPTIMISER_LEVEL = 6           ` Enable the optimiser
Device = 18F452              ` Choose a 16-bit core device
Include "TIMER1_18.INC"      ` Include the Timer1 macros into the program
```

```
Dim WRD1 as Word
```

```
OPEN_TIMER1 [T1_INT_OFF & T1_16BIT_RW & T1_SOURCE_INT & T1_PS_1_1 &
               T1_OSC1EN_OFF & T1_SYNC_EXT_OFF]
```

```
WRITE_TIMER1 WRD1           ` Write variable WRD1 into Timer1
```