

Experiment 7: Hardware Timers

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME
141	Nov. 2, 2014		H

Contents

1	Objectives	1
2	Parts List	1
3	Background	1
3.1	Peripheral Clock (PCLK)	1
3.2	How Timers Work	1
3.3	Important Notes	2
4	Tasks	2
5	Resources	2

1 Objectives

- Using hardware timers

2 Parts List

- LPC1769 LPCXpresso board
- USB A-Type to Mini-B cable
- Breadboard
- Seven-segment display
- 330-Ohm Resistors
- Jumper wires

3 Background

There are four hardware timers in LPC1769. They are identical and can work independently with different settings.

3.1 Peripheral Clock (PCLK)

Timers rely on *peripheral clocks* (PCLK), which in turn are derived from the *core clock* (CCLK).

Each timer, among other devices, has its own peripheral clock. There are four possible frequency configurations for each peripheral clock, which are set using a pair of bits.

01	$PCLK = CCLK$
10	$PCLK = CCLK / 2$
00	$PCLK = CCLK / 4$
11	$PCLK = CCLK / 8$

These pairs of bits belong to the PCLKSEL0 and PCLKSEL1 registers, which control the PCLK frequency for all peripherals.

Exercise

Refer to Chapter 4 in the [LPC1769 manual](#) to identify the two bits needed to set the PCLK frequency for the timer that you want to use.

3.2 How Timers Work

By default, the *Timer Counter* (TC) register is incremented every PCLK cycles. When the value of TC reaches the value in *Match Register 0* (MR0), an action can be taken. Therefore, setting MR0 specifies the timer's period.

The action to be taken every time TC reaches MR0 can be set through the *Match Control Register* (MCR) to one (or more) of the following:

1. Reset TC

2. Stop TC
3. Generate an interrupt

Note

There is one of each of these registers dedicated to each timer. The name of each register is prefixed with the timer's name, e.g. T0TC, T1TC, and so on.

To choose, enable or disable the above actions you need to set/clear the 3 least significant bits:

MCR bit	Bit value = 1	Bit value = 0
0	Enable timer interrupt	Disable timer interrupt
1	TC reset	Disable this feature
2	TC stop	Disable this feature

3.3 Important Notes

- For TC registers to start counting, the least significant bit of the *Timer Control Register* (TCR) must be set. This bit is known as *Counter Enable*.
- If you choose to enable the timer interrupt, remember to enable the the NVIC and to clear the interrupt bit in the ISR. To clear the MR0 interrupt flag, set the least significant bit in the *Interrupt Register* (IR).
- A common misconception is to assume that register MR0 can be used with timer 0 only, register MR1 with timer 1 only, and so on. Any of the MRx registers can be used with any of the four timers. Moreover, multiple MRx registers can be used with the same timer.
- As usual, all the registers in this experiment are fields of some structures. Refer to the LPC176xx.h header file to find the required name and field to access the required register.

4 Tasks

Use a hardware timer and timer interrupts to update the seven-segment display to count from 00 to 59.

5 Resources

[lpc1769-manual]

NXP Semiconductors. *UM10360 LPC176x/5x User manual*. Rev. 3.1. 2 April 2014.
http://www.nxp.com/documents/user_manual/UM10360.pdf