Exercise in Embedded Computing: Clocking and Power Management

Sivan Toledo, Tel-Aviv University

November 21, 2010

1. Implement a program that uses a timer to blink an LED once per second using the default clock settings of the board, which are a 12 MHz CCLK and a 3 MHz PCLK (you should already have such a program).

   (a) Now increase PCLK to the same frequency as CCLK by setting VPBDIV to 1. Does the LED blink faster?

   (b) Change PCLK back to CCLK/4 and increase CCLK to 60 MHz by configuring the PLL appropriately and choosing it as the CCLK clock source. How fast does the LED blink?

   (c) Turn PCLK back up to CCLK; the LED should blink even faster.

2. Implement an interrupt-driven LED blinker (you should already have the program).

   (a) What happens if you put the processor into Power-Down mode in the main loop?

   (b) What happens if you put the processor into Idle mode in the main loop?

   (c) Use PCONP to turn off all the peripherals that your program does not need. Test that it still blinks.

   (d) Count the invocations of the ISR using a static variable, and put the processor into power-down mode after 4 seconds. What do you observe?

   (e) Configure P0.14 as an external interrupt pin, and configure the external interrupt to wake up the processor. You do not need to implement an ISR for the interrupt; we just want the button switches to wake up the processor. You can disable the interrupt in the VIC. Pressing the switch should start up blinking again.

   (f) Configure the PLL to run the processor at 60 MHz. The initial blinking should be faster. How fast does the program blink when it wakes up from power-down mode? Explain your observation.