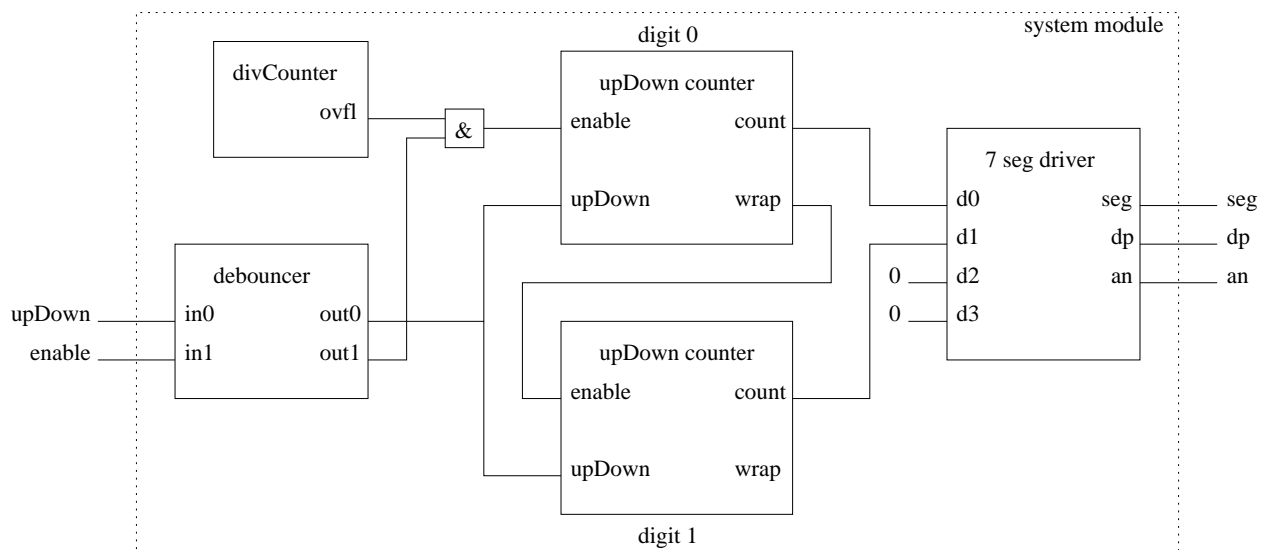# Decimal Counter

CS 240

25 points, due May 4, 2012 at 12:00 pm

In this assignment you'll be implementing this two digit decimal up/down counter in VHDL and using a Nexys 2 board to test your design:

Not shown: clk and reset, which go to each sub−module.

Input Signals

1. 50 MHz clock.

2. upDown, controlled by switch SW2.

3. enable, controlled by switch SW1.

4. Reset, active high, controlled by switch SW0.

Output Signals

1. seg, dp, and an, control signals for the four digit seven segment display.

Circuit Operation

1. When reset is brought high, the count should reset to 0. This signal has highest priority.

2. When enable is brought high, the circuit should count.

3. When enable is high and upDown is high the counter counts up. When enable is high and upDown is low the counter counts down.

   Only a decimal (digits 0000–0099) count should be displayed. The counter should count modulo 100. The decimal points should all be un-lit.

4. The count frequency should be 1 Hz.

VHDL Module Files

1. system.vhd — The top level module. The project's pin constraints appear here. Component declarations are provided. You will be adding signal declarations, instantiating components, and adding some dataflow VHDL. Overall, this is structural VHDL.

   Before generating the bitstream file, confirm the pin constraints in this file against the Pinout Report.

2. upDownCounter.vhd — The skeleton for the decimal up down counter. You will be implementing this module. The file's header comment contains a detailed description of the module's behavior.

3. divCounter.vhd — The skeleton for the 50 MHz frequency divider circuit. You will be implementing this module. The file's header comment contains a detailed description of the module's behavior.

4. switchDebouncer.vhd — The VHDL module for the input synchronizer/debouncer. This module should be used as-is. The module's behavior is described in the file's header comment.

5. simpleSevenSegDriver.vhd — The VHDL module for the four digit seven segment display. This module should be used as-is. The module's behavior is described in the file's header comment.

Assignment Turn-In

E-mail a copy of your **commented** VHDL source code (three `.vhd` files) and your `.bit` file to kelliher[at]goucher.edu.