BYOC course

Homework exercise #1

An example project

Part1

1) First project – p4

a. Description of the p4 project

This example design has a counter 0-15 that is displayed on the LEDs of the Nexys2 board. The two rightmost switches determine the operation of the circuit. The rightmost one denoted switches0 forces the right hand side LED to show 0 when it is in "off" position. When this switch is in "on" position and switch1, which is the next switch on the left, is "off" the counter is incremented all the time automatically in a rate of ~1.5Hz (i.e., every 2/3 of a second). When switch1 is "on" the counter is incremented every time we press the rightmost pushbutton on the Nexys2 board. When the counter reaches 15, i.e., when the LED displays the letter F, its next state is 0 and the LED that is lit is the next LED to the left. After reaching the 4th LED, at F it will go back to the first LED on the right.

The following drawing describes the design:

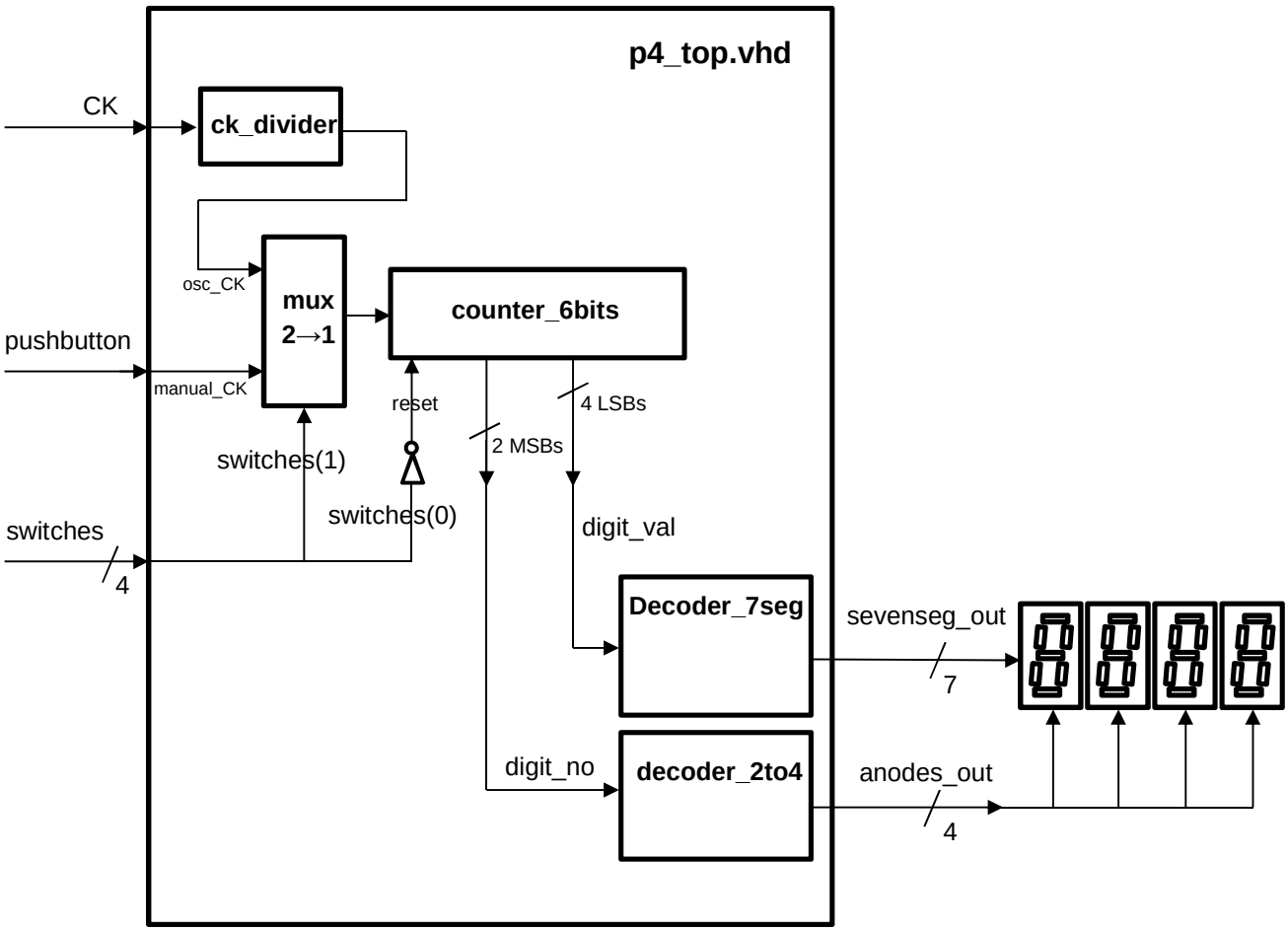

Fig. 3.1 – The project (p4_TOP.vhd)

2) The project files

1) Installing SW

In case you only work in the class (S2 class near the library), you do not need to install any SW. In case you want to work at home, on your own PC or laptop, you may install Xilinx Webpack (the main SW), Modelsim student edition and Notepad++. See the installation instructions in the "BYOC_doc_for_students" document.

2) Creating the directories

a. We start by deciding in which directory we want to keep our projects. Say it is c:\projects\p4
b. In this directory we will create 3 sub-directories:
   i. C:\projects\p4\Src - in which we will keep the project sources files (*.vhd, *.ucf)
   ii. C:\projects\ p4\Sim - in which Modelsim SW or Xilinx ISE ISIM will keep its project files
   iii. C:\projects\ p4\ISE - in which Xilinx ISE Design Suite will keep its project files (including the *.bit for programming)

3) Editing the files

We then edit our *.vhd files and *.ucf files in the Src directory. We can use any editor, e.g., the notepad++ editor that 'knows' VHDL syntax. You can use the Xilinx ISE editor or, if you installed the Modelsim Student Edition, you can use the Modelsim editor.
In our design example the files are:

a. **p4_top.vhd** - This is the top block including the entire design and its inputs and outputs
b. **ck_divider.vhd** - A counter used to divide the 50MHz external CK coming from the Nexys2 board.
c. **mux_2to1.vhd** - A mux used to select the divided CK or a manual CK to be fed to the display counter
d. **counter_6bits.vhd** - The display counter. The 4 LSBs are connected to the 7Seg decoder while the 2 MSBs determine which of the 4 LEDs is lit via the 2to4 decoder.
e. **decoder_7seg.vhd** - gets a 4 bit number and decodes which of the 7 segments should be lit to display the appropriate Hex digit on the LED.
f. **decoder_2to4.vhd** - used to select which digit is lit
g. **p4_top.ucf** - describes the connection of the p4_top input & outputs to the FPGA i/o pins. Used in implementation only.
h. **p4_TB.vhd** – for simulation only

You have these files in the course web site under Exercise HW1. The files you get have errors in them. You need to run compilation on the Modelsim (or Xilinx ISE) and fix all of the syntax errors. Then Run the simulation on the Modelsim (or Xilinx ISIM) and fix all of the logical errors. Then use the Xilinx ISE to create the BIT file, load it into the design and fix more errors if exist.
.