Connecting External Peripherals to the FPGA board

Objectives:

- Connecting an LED to the FPGA board and write a VHDL model to control it
- Connecting a push button to the FPGA board
  - in pull-up resistor mode
  - in pull-down resistor mode

  and write a VHDL model to read input from it.

Introduction:

Elbert V2 – Spartan 3A FPGA development board has five GPIO ports, these are P1, P2, P4, P5, P6 to which external devices can be connected. Pins of the ports can be configured as input to read a digital input, or they can be configured as a digital output. In this activity, several examples of connecting external devices to FPGA board are presented.

User Constraint File:

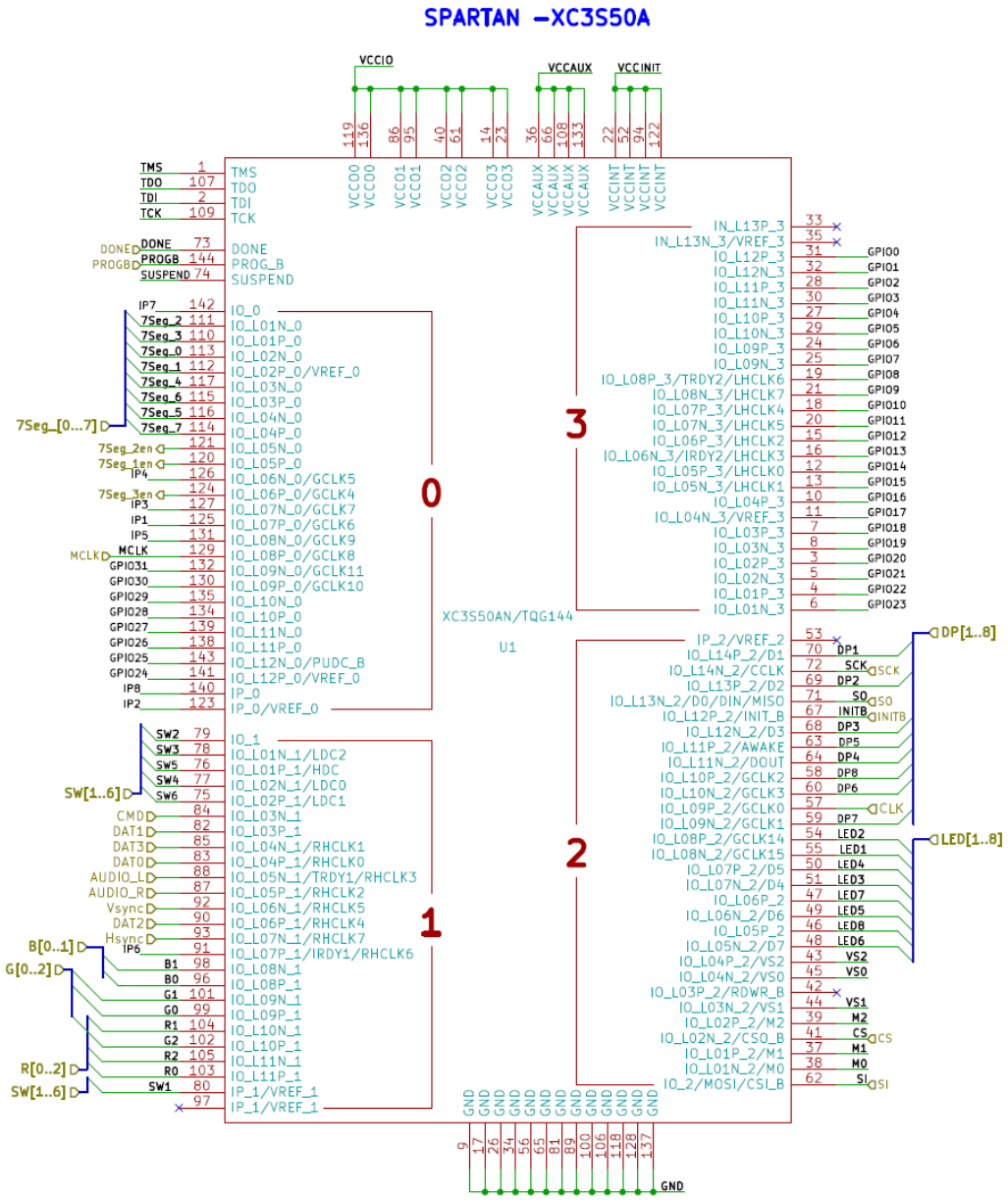
The user constraint file for GPIO ports is listed below. Any pin of any port can be an input or output pin. An internal pull-up or pulldown resistor can be selected for each pin when the pin is used as input. Pull-up resistor is not needed when the pin is used as output.

```
##############################################################################
# GPIO
##############################################################################

# HEADER P1

    #NET "gpio_P1[0]"        LOC = P31  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[1]"        LOC = P32  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[2]"        LOC = P28  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[3]"        LOC = P30  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[4]"        LOC = P27 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[5]"        LOC = P29 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[6]"        LOC = P24 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P1[7]"        LOC = P25 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;

# HEADER P6

   #NET "gpio_P6[8]"         LOC = P19  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
   #NET "gpio_P6[9]"         LOC = P21  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
   #NET "gpio_P6[10]"        LOC = P18  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
   #NET "gpio_P6[11]"        LOC = P20  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
   #NET "gpio_P6[12]"        LOC = P15  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
   #NET "gpio_P6[13]"        LOC = P16  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P6[14]"        LOC = P12 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
    #NET "gpio_P6[15]"        LOC = P13 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;

# HEADER P2

    #NET "gpio_P2[16]"        LOC = P10  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP;
```

```
#NET "gpio_P2[17]"        LOC = P11  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP;
#NET "gpio_P2[18]"        LOC = P7   | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P2[19]"        LOC = P8   | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P2[20]"        LOC = P3   | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P2[21]"        LOC = P5   | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P2[22]"        LOC = P4   | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P2[23]"        LOC = P6   | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;

# HEADER P4

#NET "gpio_P4[24]"       LOC = P141  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[25]"       LOC = P143  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[26]"       LOC = P138  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[27]"       LOC = P139  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[28]"       LOC = P134  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[29]"       LOC = P135  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[30]"       LOC = P130 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P4[31]"       LOC = P132 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;


# HEADER P5

#NET "gpio_P5[1]"        LOC = P125  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[2]"        LOC = P123  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[3]"        LOC = P127  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[4]"        LOC = P126  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[5]"        LOC = P131  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[6]"        LOC = P91  | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[7]"        LOC = P142 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
#NET "gpio_P5[8]"        LOC = P140 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP ;
```

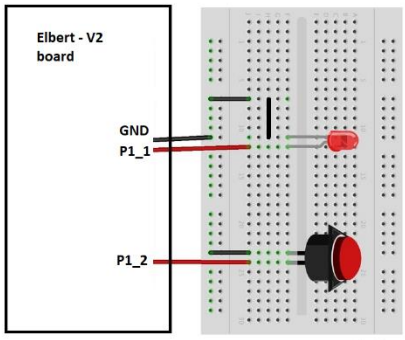
The FPGA pin layout is shown on the next page. The LOC in the above user constraint file refers to the pin number of each GPIO. The electrical connections to the LEDs and switches of the development board are also shown on the schematic.

FPGA PIN Layout



Activity 1: Connecting an LED and a Push Button to FPGA

Connect and LED and a push-button to FPGA board as shown in the Figure. Create a project with a module that has the push-button as input and the LED as output. In the User Constraint file tie the LED to pin 31 and push-button to pin 32 of FPGA.



Create a project with a module that has the push-button as input and the LED as output.

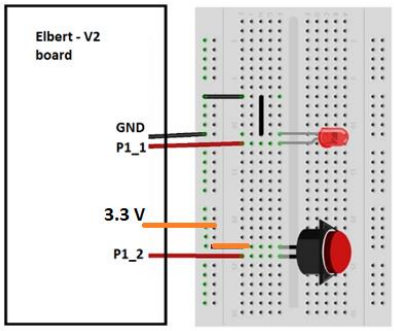
```
----------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;



entity ExtPeripheral is
    Port ( PButton : in  STD_LOGIC;
           LED : out  STD_LOGIC);
end ExtPeripheral;

architecture Behavioral of ExtPeripheral is

begin

LED <= PButton,

end Behavioral;
```

In the User Constraint file tie the LED to pin 31 and push-button to pin 32 of FPGA, use a pull-up resistor for the push button, as shown here.

```
##############################################################################
# Switches
##############################################################################
   NET "PButton"  LOC = P32 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLUP;
##############################################################################
# LEDs
##############################################################################
   NET "LED"   LOC = P31 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST ;
##############################################################################
```

Then compile and run the VHDL model on the board. The LED must be on when the push button is not pressed. When the button is pressed the LED must be off.

Now change the wiring of the button as shown below.



Modify the UCF to use a pull-down resistor.

```
################################################################################
# Switches
################################################################################
   NET "PButton"  LOC = P32 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST | PULLDOWN;
################################################################################
# LEDs
################################################################################
   NET "LED"   LOC = P31 | IOSTANDARD = LVTTL | DRIVE = 8 | SLEW = FAST ;
################################################################################
```

Then compile and run the VHDL model on the board. The LED must be off when the push button is not pressed. When the button is pressed the LED must be on.