```
--------------------------------------------------------------------------------
      -
-- Company:
-- Engineer: David Paquette
--
-- Create Date:    17:27:40 11/19/2015
-- Design Name:
-- Module Name:    TemperatureSensorInterface - Behavioral
-- Project Name:
-- Target Devices:
-- Tool versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
--------------------------------------------------------------------------------
      -
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
--use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
USE ieee.numeric_std.ALL;
use ieee.math_real.all;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TemperatureSensorInterface is
generic(
           X_TMP_COL_WIDTH    : natural := 50;   -- = SZ_TH_WIDTH - width of a
                       TEMP column
           Y_TMP_COL_HEIGHT   : natural := 472;  -- = SZ_TH_HEIGHT - height of a
                       TEMP column
           X_TMP_H_LOC        : natural := 1050; -- X Location of the TEMP Column
           Y_TMP_V_LOC        : natural := 80;   -- Y Location of the TEMP Column
           INPUT_DATA_WIDTH   : natural := 12; -- Data width is 13 for the
                       ADT7420 Temperature Sensor and
                                               -- 12 for the XADC temperature
                                                       data and the Accelerometer
                                                       Temperature Sensor
           TMP_TYPE           : string := "XADC" -- Either "XADC" or "TEMP_ACC"
           );
      Port (clk_i: in std_logic;
                   rst_i : in std_logic;
                   temperatureCelcius : out integer range 0 to 100 );
end TemperatureSensorInterface;
```

```
architecture Behavioral of TemperatureSensorInterface is
      signal temperature : std_logic_vector(11 downto 0);
      signal reset : std_logic;
      signal clk : std_logic;


      -- Maximum temperature
      constant TEMP_MAX  : std_logic_vector (23 downto 0) := X"000500"; -- 80C *
            16

      -- Scale incoming XADC temperature data, according to the XADC datasheet
      constant XADC_TMP_SCALE : std_logic_vector(17 downto 0) := "111110111" &
            "111110011"; --503.975 (18bit)
      constant thirtyTwobuffer : std_logic_vector(30 downto 0):=(others=>'0');
      -- Convert Kelvin to Celsius
      constant XADC_TMP_OFFSET : std_logic_vector(30 downto 0) := thirtyTwobuffer
            +integer(round(273.15)*4096.0);

      -- Synchronize incoming temperature to the clock
      signal temp_sync0, temp_sync : std_logic_vector(temperature'range);

      -- signal storing the scaled XADC temperature data
      signal temp_xad_scaled : std_logic_vector(temp_sync'length
            +XADC_TMP_SCALE'length-1 downto 0); --12bit*18bit=30bit
      -- signal storing the offseted XADC temperature data
      signal temp_xad_offset : std_logic_vector(XADC_TMP_OFFSET'range); --31bit
      -- signal storing XADC temperature data converted to Celsius
      signal temp_xad_celsius : std_logic_vector(temp_xad_offset'length-8-1
            downto 0); --23bit
      -- Signal storing the FPGA temperature limited to between 0C and 80C * 16
      signal temp_xad_capped : std_logic_vector(temp_xad_celsius'high-1 downto
            0); --no sign bit

      signal temp : std_logic_vector(7 downto 0);
begin
      clk<=clk_i;
      reset<=rst_i;

      temperatureCelcius<=to_integer(unsigned(temp));

      Inst_FPGAMonitor: entity work.FPGAMonitor PORT MAP(
            CLK_I          => clk,
            RST_I          => reset,
            TEMP_O         => temperature
      );


process(clk)
begin
      if clk'EVENT and clk = '1' then
                  temp_sync0 <= temperature; --synchronize with pxl_clk domain
                  temp_sync <= temp_sync0;

                  --30b                             12b                       18b
                  temp_xad_scaled <= temp_sync * XADC_TMP_SCALE; -- ADC *
```

```
                        503.975 (fixed-point; decimal point at 9b)

                  temp_xad_offset <= '0' & temp_xad_scaled(29 downto 9) -
                        XADC_TMP_OFFSET; -- ADC * 503.975 - 273.15 * 4096

                  temp_xad_celsius <= temp_xad_offset(temp_xad_offset'high
                        downto 8); -- (ADC * 503.975 - 273.15) / 256;
                        1LSB=0.625C

                  if (temp_xad_celsius(temp_xad_celsius'high) = '1') then --if
                        negative, cap to 0
                        temp_xad_capped <= (others => '0');
                  elsif (temp_xad_celsius(temp_xad_celsius'high-1 downto 0) >
                        TEMP_MAX) then --if too big, cap to maximum scale /
                        0.0625
                        temp_xad_capped <= TEMP_MAX(temp_xad_capped'range);
                  else
                        temp_xad_capped <=
                              temp_xad_celsius(temp_xad_celsius'high-1 downto
                              0); --get rid of the sign bit
                  end if;

                  temp<=temp_xad_capped(11 downto 4); -- remove all data under
                        0C (decimals)
            end if;
end process;
end Behavioral;
```