PyroElectro.com - PyroEDU

Introduction To FPGA And CPLD – Lesson 7: Parallel Hardware

FORMULAS

The following formulas and information are meant to go with the online lesson found here:
http://www.pyroelectro.com/edu/fpga/parallel_hardware/

LESSON7 VHDL CODE

In lesson 7 we built a 4 bit timer-counter and a d-flip-flop in parallel. Both modules used the same clock and master reset signal, but other than that they are completely separate processes, both in the code to the right and in the CPLD configuration.

The timer-counter should look familiar to you as it is the same one that we used in lesson6.

Similarly, the d-flip-flop should look familiar because we've used it before as well. It's a simple push button input, red led output, using the clock to clock the data from input to data.

Lesson7.vhd

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity lesson7 is
   port(
   RESET:  in std_logic;      --PIN_44
   CLOCK:  in std_logic;      --PIN_37
   BUTTON: in std_logic;      --PIN_23
   LED:       out std_logic;     --PIN_22
   COUNT:  out std_logic_vector(3 downto 0) --PIN_18 to PIN_21
   );
end lesson7;

architecture rtl of lesson7 is

signal data: std_logic;
signal count_value: std_logic_vector(3 downto 0);

begin

--First Process
   DATA_REGISTER_0: process(RESET,CLOCK,BUTTON)
   begin
      if RESET = '1' then
         data <= '0';
      elsif rising_edge(CLOCK) then
         data <= BUTTON;
      end if;
   end process DATA_REGISTER_0;

   LED <= data;

--Second Process
   TIMER_0: process(RESET,CLOCK)
   begin
      if RESET = '1' then
         count_value <= "0000";
      elsif rising_edge(CLOCK) then
         count_value <= count_value + 1;
      end if;
   end process TIMER_0;

   COUNT <= count_value;
end rtl;
```

ADDITIONAL INFORMATION

If you have any questions about the formulas or information found in this document, please feel free to head on over to the forums and ask us some questions!
http://www.pyroelectro.com/forums/viewforum.php?f=26