**קובץ מס 1**

-- Create Date: 06/24/2019 09:33:13 AM

-- Design Name:

-- Module Name: Calculator - 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;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

use IEEE.NUMERIC\_STD.ALL;

use IEEE.std\_logic\_unsigned.all;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity Calculator is

Port ( CLK : in STD\_LOGIC;

RESET : in STD\_LOGIC;

DATA\_IN\_A : in STD\_LOGIC\_VECTOR (17 downto 0);

DATA\_IN\_B : in STD\_LOGIC\_VECTOR (17 downto 0);

SEL : in STD\_LOGIC\_VECTOR (2 DOWNTO 0);

DATA\_OUT : out STD\_LOGIC\_VECTOR (36 downto 0));

end Calculator;

architecture Behavioral of Calculator is

COMPONENT xbip\_dsp48\_macro

PORT (

CLK : IN STD\_LOGIC;

SEL : IN STD\_LOGIC\_VECTOR(2 DOWNTO 0);

A : IN STD\_LOGIC\_VECTOR(17 DOWNTO 0);

B : IN STD\_LOGIC\_VECTOR(17 DOWNTO 0);

C : IN STD\_LOGIC\_VECTOR(47 DOWNTO 0);

P : OUT STD\_LOGIC\_VECTOR(47 DOWNTO 0)

);

END COMPONENT;

signal DSP\_SEL : STD\_LOGIC\_VECTOR(2 DOWNTO 0) := (others => '0');

signal A : STD\_LOGIC\_VECTOR(17 DOWNTO 0) := (others => '0');

signal B : STD\_LOGIC\_VECTOR(17 DOWNTO 0) := (others => '0');

signal C : STD\_LOGIC\_VECTOR(47 DOWNTO 0) := (others => '0');

signal P : STD\_LOGIC\_VECTOR(47 DOWNTO 0) := (others => '0');

COMPONENT blk\_mem\_gen\_0

PORT (

clka : IN STD\_LOGIC;

addra : IN STD\_LOGIC\_VECTOR(6 DOWNTO 0);

douta : OUT STD\_LOGIC\_VECTOR(47 DOWNTO 0)

);

END COMPONENT;

signal rom\_douta : STD\_LOGIC\_VECTOR(47 DOWNTO 0) := (others => '0');

signal rom\_addra : STD\_LOGIC\_VECTOR(6 DOWNTO 0) := (others => '0');

component clk\_wiz\_0

port

(clk\_100 : out std\_logic;

clk\_10 : out std\_logic;

clk\_300 : out std\_logic;

-- Status and control signals

reset : in std\_logic;

locked : out std\_logic;

clk\_in1 : in std\_logic

);

end component;

signal clk\_100 : std\_logic := '0';

signal clk\_10 : std\_logic := '0';

signal clk\_300 : std\_logic := '0';

signal pll\_locked : std\_logic := '0';

signal count : Integer range 0 to 20 := 0;

signal math\_sel : STD\_LOGIC\_VECTOR(2 DOWNTO 0) := SEL;

begin

DSP : xbip\_dsp48\_macro

PORT MAP (

CLK => clk\_300,

SEL => DSP\_SEL,

A => A,

B => B,

C => C,

P => P

);

PLL : clk\_wiz\_0

port map (

clk\_100 => clk\_100,

clk\_10 => clk\_10,

clk\_300 => clk\_300,

reset => RESET,

locked => pll\_locked,

clk\_in1 => CLK

);

ROM : blk\_mem\_gen\_0

PORT MAP (

clka => clk\_100,

addra => rom\_addra,

douta => rom\_douta

);

process (clk\_100)

begin

if (clk\_100'event and clk\_100 = '1') then

if ((RESET = '1') or (math\_sel /= "101")) then

rom\_addra <= (others => '0');

else

if rom\_addra = 100 then

rom\_addra <= rom\_addra;

else

rom\_addra <= rom\_addra + 1;

end if;

end if;

end if;

end process;

process (clk\_100)

variable sum : std\_logic\_vector(47 downto 0):= (others => '0');

begin

if (clk\_100'event and clk\_100 = '1') then

if ((RESET = '1') or (math\_sel /= SEL)) then

A <= (others => '0');

B <= (others => '0');

C <= (others => '0');

DSP\_SEL <= (others => '1');

DATA\_OUT <= (others => '0');

count <= 0;

math\_sel <= SEL;

sum := (others => '0');

else

case (math\_sel) is

when "000" => -- +

DSP\_SEL <= "000"; -- C+A

C(17 DOWNTO 0) <= DATA\_IN\_A;

A <= DATA\_IN\_B;

DATA\_OUT <= P(36 downto 0);

when "001" => -- -

DSP\_SEL <= "001"; -- C-A

C(17 DOWNTO 0) <= DATA\_IN\_A;

A <= DATA\_IN\_B;

DATA\_OUT <= P(36 downto 0);

when "010" => -- \*

DSP\_SEL <= "010"; -- A\*B

A <= DATA\_IN\_A;

B <= DATA\_IN\_B;

DATA\_OUT <= P(36 downto 0);

when "011" => -- /

DSP\_SEL <= "011"; -- C-(A\*B)

C(17 DOWNTO 0) <= DATA\_IN\_A;

A <= DATA\_IN\_B;

if count = 0 then

if (A = 0) then

B <= (16 => '1', others => '0');

else

count <= count + 1;

end if;

elsif count = 17 then

B(17 - count) <= not P(47);

count <= count + 1;

elsif count = 18 then

DATA\_OUT (36 downto 18) <= (others => '0');

DATA\_OUT (17 downto 0) <= B;

else

B(17 - count) <= not P(47);

B(17 - count - 1) <= '1';

count <= count + 1;

end if;

when "100" => -- square root

DSP\_SEL <= "011"; -- C-(A\*B)

C(17 DOWNTO 0) <= DATA\_IN\_A;

if count = 0 then

if (A = 0) then

B <= (16 => '1', others => '0');

A <= (16 => '1', others => '0');

else

count <= count + 1;

end if;

elsif count = 17 then

B(17 - count) <= not P(47);

A(17 - count) <= not P(47);

count <= count + 1;

elsif count = 18 then

DATA\_OUT (36 downto 18) <= (others => '0');

DATA\_OUT (17 downto 0) <= B;

else

B(17 - count) <= not P(36);

B(17 - count - 1) <= '1';

A(17 - count) <= not P(36);

A(17 - count - 1) <= '1';

count <= count + 1;

end if;

when "101" => -- average

if rom\_addra = 100 then

DSP\_SEL <= "011"; -- C-(A\*B)

C <= sum;

A <= std\_logic\_vector(to\_unsigned(300, DATA\_IN\_A'length)); --300 (100\*3);

if count = 0 then

B <= (16 => '1', others => '0');

count <= count + 1;

elsif count = 17 then

B(17 - count) <= not P(47);

count <= count + 1;

elsif count = 18 then

DATA\_OUT (36 downto 18) <= (others => '0');

DATA\_OUT (17 downto 0) <= B;

else

B(17 - count) <= not P(47);

B(17 - count - 1) <= '1';

count <= count + 1;

end if;

else

DSP\_SEL <= "100"; -- C+P

C <= rom\_douta;

sum := P;

end if;

-- when "110" => -- Standard deviation

-- <statement>;

-- when "111" => -- The root equation

-- <statement>;

when others => --

A <= (others => '0');

B <= (others => '0');

C <= (others => '0');

DSP\_SEL <= (others => '0');

DATA\_OUT <= (others => '0');

end case;

end if;

end if;

end process;

end Behavioral;

**קובץ מס 2**

**Company:**

**-- Engineer:**

**--**

**-- Create Date: 06/24/2019 11:47:08 AM**

**-- Design Name:**

**-- Module Name: Calculator\_TB - 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;**

**-- Uncomment the following library declaration if using**

**-- arithmetic functions with Signed or Unsigned values**

**use IEEE.NUMERIC\_STD.ALL;**

**use IEEE.std\_logic\_unsigned.all;**

**-- Uncomment the following library declaration if instantiating**

**-- any Xilinx leaf cells in this code.**

**--library UNISIM;**

**--use UNISIM.VComponents.all;**

**entity Calculator\_TB is**

**-- Port ( );**

**end Calculator\_TB;**

**architecture Behavioral of Calculator\_TB is**

**component Calculator is**

**Port ( CLK : in STD\_LOGIC;**

**RESET : in STD\_LOGIC;**

**DATA\_IN\_A : in STD\_LOGIC\_VECTOR (17 downto 0);**

**DATA\_IN\_B : in STD\_LOGIC\_VECTOR (17 downto 0);**

**SEL : in STD\_LOGIC\_VECTOR (2 DOWNTO 0);**

**DATA\_OUT : out STD\_LOGIC\_VECTOR (36 downto 0));**

**end component;**

**signal CLK : STD\_LOGIC := '0';**

**signal RESET : STD\_LOGIC;**

**signal DATA\_IN\_A : STD\_LOGIC\_VECTOR (17 downto 0) := (others => '0');**

**signal DATA\_IN\_B : STD\_LOGIC\_VECTOR (17 downto 0) := (others => '0');**

**signal SEL : STD\_LOGIC\_VECTOR (2 DOWNTO 0) := (others => '0');**

**signal DATA\_OUT : STD\_LOGIC\_VECTOR (36 downto 0) := (others => '0');**

**begin**

**uut : Calculator**

**PORT MAP (**

**CLK => CLK,**

**RESET => RESET,**

**DATA\_IN\_A => DATA\_IN\_A,**

**DATA\_IN\_B => DATA\_IN\_B,**

**SEL => SEL,**

**DATA\_OUT => DATA\_OUT**

**);**

**CLK <= not (CLK) after 50 ns;**

**RESET <= '1', '0' after 100 ns;**

**DATA\_IN\_A <= std\_logic\_vector(to\_unsigned(64, DATA\_IN\_A'length)); --64**

**DATA\_IN\_B <= std\_logic\_vector(to\_unsigned(4, DATA\_IN\_B'length)); --4**

**SEL <= "000", "001" after 10 us, "010" after 15 us, "011" after 20 us, "100" after 50 us, "101" after 80 us;**

**end Behavioral;**

**קובץ מס 3**

**Company:**

**-- Engineer:**

**--**

**-- Create Date: 06/24/2019 01:03:03 PM**

**-- Design Name:**

**-- Module Name: DSP\_TB - 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;**

**-- Uncomment the following library declaration if using**

**-- arithmetic functions with Signed or Unsigned values**

**use IEEE.NUMERIC\_STD.ALL;**

**use IEEE.std\_logic\_unsigned.all;**

**-- Uncomment the following library declaration if instantiating**

**-- any Xilinx leaf cells in this code.**

**--library UNISIM;**

**--use UNISIM.VComponents.all;**

**entity DSP\_TB is**

**-- Port ( );**

**end DSP\_TB;**

**architecture Behavioral of DSP\_TB is**

**COMPONENT xbip\_dsp48\_macro**

**PORT (**

**CLK : IN STD\_LOGIC;**

**SEL : IN STD\_LOGIC\_VECTOR(2 DOWNTO 0);**

**A : IN STD\_LOGIC\_VECTOR(17 DOWNTO 0);**

**B : IN STD\_LOGIC\_VECTOR(17 DOWNTO 0);**

**C : IN STD\_LOGIC\_VECTOR(47 DOWNTO 0);**

**P : OUT STD\_LOGIC\_VECTOR(47 DOWNTO 0)**

**);**

**END COMPONENT;**

**signal CLK : STD\_LOGIC := '0';**

**signal DSP\_SEL : STD\_LOGIC\_VECTOR(2 DOWNTO 0) := (others => '0');**

**signal A : STD\_LOGIC\_VECTOR(17 DOWNTO 0) := (others => '0');**

**signal B : STD\_LOGIC\_VECTOR(17 DOWNTO 0) := (others => '0');**

**signal C : STD\_LOGIC\_VECTOR(47 DOWNTO 0) := (others => '0');**

**signal P : STD\_LOGIC\_VECTOR(47 DOWNTO 0) := (others => '0');**

**begin**

**DSP : xbip\_dsp48\_macro**

**PORT MAP (**

**CLK => CLK,**

**SEL => DSP\_SEL,**

**A => A,**

**B => B,**

**C => C,**

**P => P**

**);**

**CLK <= not (CLK) after 5 ns;**

**A <= std\_logic\_vector(to\_unsigned(3 , A'length)); --3**

**B <= std\_logic\_vector(to\_unsigned(2 , B'length)); --2**

**C <= std\_logic\_vector(to\_unsigned(12, C'length)); --12**

**DSP\_SEL <= "000", "001" after 200 ns, "010" after 400 ns, "011" after 600 ns, "100" after 800 ns, "101" after 1000 ns;**

**end Behavioral;**