Lab 6 – Arithmetic Unit

CS1050 Computer Organization and Digital Design

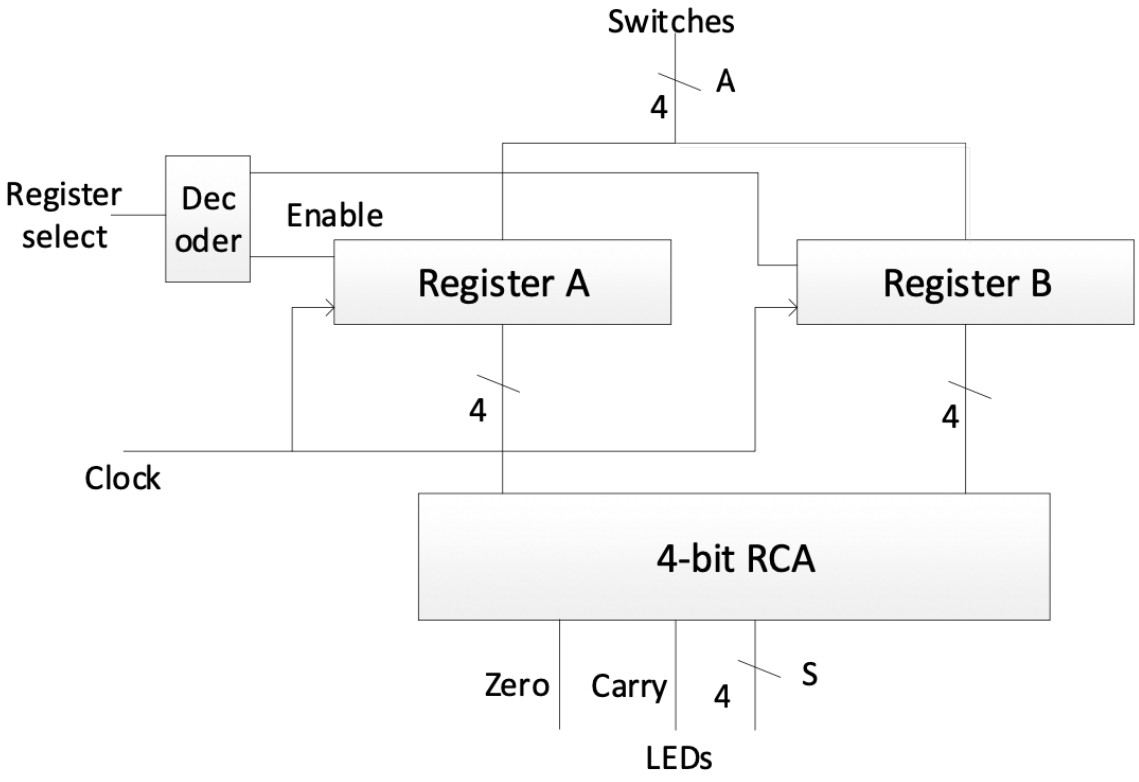
Name : **Dissanayake D.M.A.K.**
Index No. : **220135N**
Group : **44**

Lab Task:

Designing and Development of 4-bit register and 4-bit Arithmetic Unit run on Basys3 board.

Introduction

In the realm of microprocessors, registers play a pivotal role as repositories for sets of bits. Our mission in this lab is to engineer a 4-bit Arithmetic Unit capable of performing additions on numbers housed in two separate registers. To grasp the intricacies of this task, we'll begin with the creation of a 4-bit register utilizing D Flip Flops. This register, featuring an Enable input and a Clock input, will set the stage for the subsequent integration with our 4-bit RCA.

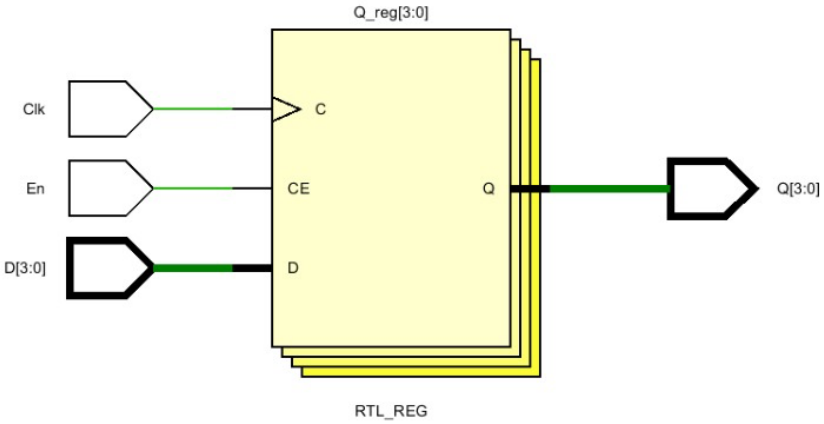


1)4-bit Register

Design Source File

```
----------------------------------------------------------------------------
-------
-- Company:
-- Engineer:
--
-- Create Date: 03/12/2024 02:07:04 PM
-- Design Name:
-- Module Name: Reg - 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;

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

entity Reg is
    Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
           En : in STD_LOGIC;
           Clk : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (3 downto 0));
end Reg;

architecture Behavioral of Reg is

begin

process (Clk) begin
    if (rising_edge(Clk)) then
        if En = '1' then
            Q <= D;
        end if;
    end if;
end process;

end Behavioral;
```

Elaborated design schematic



2)4-bit Arithmetic Unit

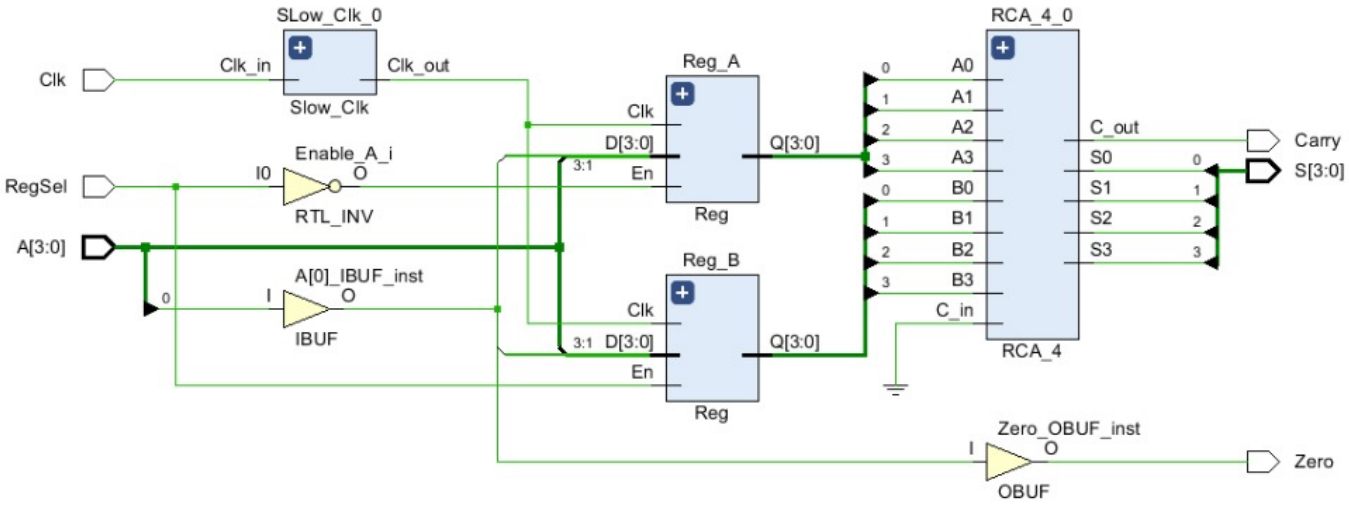
Design Source File

```
----------------------------------------------------------------------------------
-------
-- Company:
-- Engineer:
--
-- Create Date: 03/12/2024 02:11:40 PM
-- Design Name:
-- Module Name: AU - 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;

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

```
entity AU is
    Port ( A : in STD_LOGIC_VECTOR (3 downto 0);
           RegSel : in STD_LOGIC;
           Clk : in STD_LOGIC;
           S : out STD_LOGIC_VECTOR (3 downto 0);
           Zero : out STD_LOGIC;
           Carry : out STD_LOGIC);
end AU;

architecture Behavioral of AU is

component Reg
    Port ( D : in STD_LOGIC_VECTOR (3 downto 0);
           En : in STD_LOGIC;
           Clk : in STD_LOGIC;
           Q : out STD_LOGIC_VECTOR (3 downto 0));
end component;

component RCA_4
    Port ( A0 : in STD_LOGIC;
       A1 : in STD_LOGIC;
       A2 : in STD_LOGIC;
       A3 : in STD_LOGIC;
       B0 : in STD_LOGIC;
       B1 : in STD_LOGIC;
       B2 : in STD_LOGIC;
       B3 : in STD_LOGIC;
       C_in : in STD_LOGIC;
       S0 : out STD_LOGIC;
       S1 : out STD_LOGIC;
       S2 : out STD_LOGIC;
       S3 : out STD_LOGIC;
       C_out : out STD_LOGIC);
end component;

component Slow_Clk
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
end component;

signal Enable_A : STD_LOGIC;
signal Enable_B : STD_LOGIC;
signal A_out : STD_LOGIC_VECTOR (3 downto 0);
signal B_out : STD_LOGIC_VECTOR (3 downto 0);
signal Clk_Slow : STD_LOGIC;

begin
    --Logic of 1 to 2 decoder
    Enable_A <= not RegSel;
    Enable_B <= RegSel;

    SLow_Clk_0 : Slow_Clk
    port MAP (Clk, Clk_Slow);

    Reg_A : Reg
    port MAP (
        A,Enable_A,Clk_Slow,A_out
    );

    Reg_B : Reg
    port MAP (
```

```
        A,Enable_B,Clk_SLow,B_out
    );

    RCA_4_0 : RCA_4
    port MAP (
        A0 => A_out(0),
        A1 => A_out(1),
        A2 => A_out(2),
        A3 => A_out(3),
        B0 => B_out(0),
        B1 => B_out(1),
        B2 => B_out(2),
        B3 => B_out(3),
        C_in => '0',
        s0 => S(0),
        s1 => S(1),
        s2 => S(2),
        s3 => S(3),
        C_out => Carry
    );

    --logic for zero
    Zero <= (A_out(3) xnor B_out(3)) and (A_out(2) xnor B_out(2)) and
(A_out(1) xnor B_out(1)) and (A_out(0) xnor B_out(0));

end Behavioral;
```

Elaborated design schematic



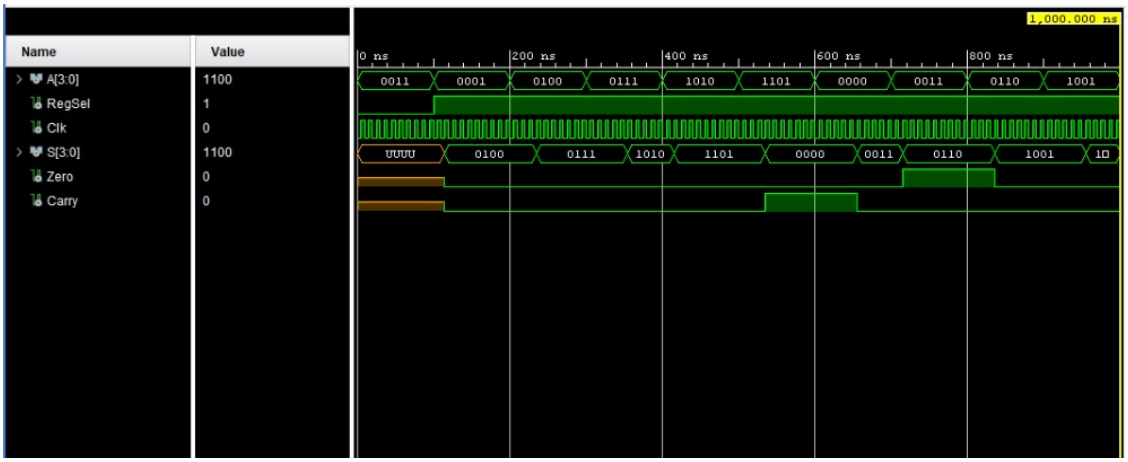
Simulation source file

```
----------------------------------------------------------------------------------
-------
-- Company:
-- Engineer:
--
-- Create Date: 03/12/2024 03:47:22 PM
-- Design Name:
-- Module Name: TB_AU - Behavioral
-- Project Name:
```

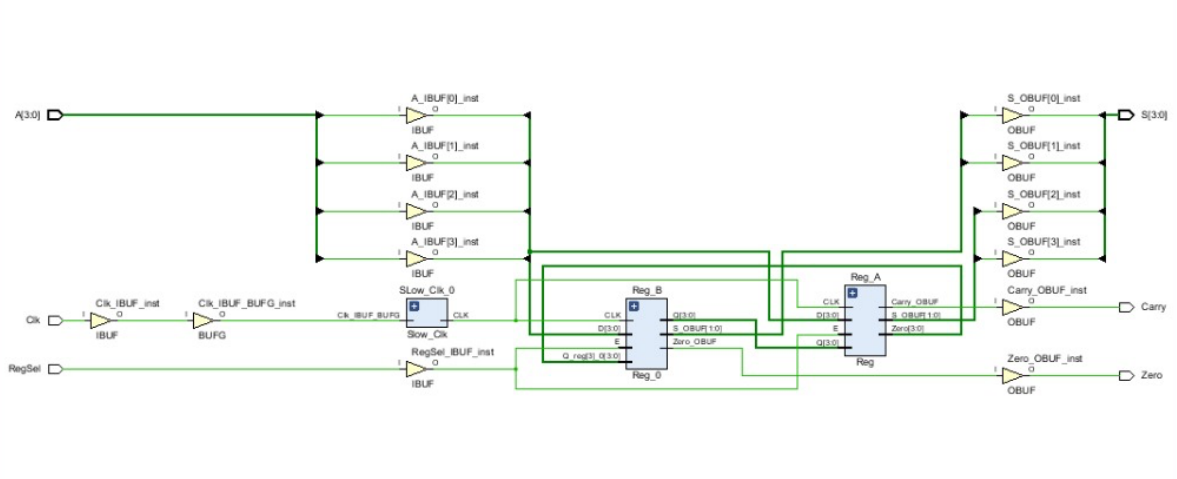
```
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------
-------


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;

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

entity TB_AU is
--  Port ( );
end TB_AU;

architecture Behavioral of TB_AU is
component AU
    Port ( A : in STD_LOGIC_VECTOR (3 downto 0);
           RegSel : in STD_LOGIC;
           Clk : in STD_LOGIC;
           S : out STD_LOGIC_VECTOR (3 downto 0);
           Zero : out STD_LOGIC;
           Carry : out STD_LOGIC);
end component;

signal A : STD_LOGIC_VECTOR (3 downto 0);
signal RegSel : STD_LOGIC;
signal Clk :  STD_LOGIC;
signal S : STD_LOGIC_VECTOR (3 downto 0);
signal Zero : STD_LOGIC;
signal Carry : STD_LOGIC;

begin

UUT : AU
PORT MAP (A, RegSel, Clk, S, Zero, Carry);

Clk_process: process
begin
Clk <= '0'; wait for 5ns;
Clk <= '1'; wait for 5ns;
end process;

process
begin
    RegSel <= '0';
```

```
        A <= "0011";
        wait for 100ns;
        RegSel <= '1';
        A <= "0001"; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;
        RegSel <= '1'; A <= S; wait for 100ns;

        wait ;

    end process;


    end Behavioral;
```

Timing Diagram



Implemented design schematic



Constraints File

```
## Clock signal
set_property PACKAGE_PIN W5 [get_ports Clk]
    set_property IOSTANDARD LVCMOS33 [get_ports Clk]
    create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5}
[get_ports Clk]

## Switches
set_property PACKAGE_PIN V17 [get_ports {A[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {A[0]}]
set_property PACKAGE_PIN V16 [get_ports {A[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {A[1]}]
set_property PACKAGE_PIN W16 [get_ports {A[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {A[2]}]
set_property PACKAGE_PIN W17 [get_ports {A[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {A[3]}]

set_property PACKAGE_PIN R2 [get_ports {RegSel}]
    set_property IOSTANDARD LVCMOS33 [get_ports {RegSel}]

## LEDs
set_property PACKAGE_PIN U16 [get_ports {S[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {S[0]}]
set_property PACKAGE_PIN E19 [get_ports {S[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {S[1]}]
set_property PACKAGE_PIN U19 [get_ports {S[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {S[2]}]
set_property PACKAGE_PIN V19 [get_ports {S[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {S[3]}]

set_property PACKAGE_PIN P1 [get_ports {Carry}]
    set_property IOSTANDARD LVCMOS33 [get_ports {Carry}]
set_property PACKAGE_PIN L1 [get_ports {Zero}]
    set_property IOSTANDARD LVCMOS33 [get_ports {Zero}]
```

Conclusion

In conclusion, the 4-bit Arithmetic Unit (AU) crafted in this lab serves as a digital powerhouse capable of adding numbers stored in distinct registers. Similar to a counter's versatility, this AU exhibits flexibility by accommodating two registers, enabling users to input 4-bit binary values via switches. Functionality is orchestrated through clock synchronization, and the outcome of the addition process is vividly displayed on LEDs. Much like a counter measuring occurrences, this AU extends its utility to compute and showcase the sum of registered numbers, underscoring its significance in diverse electronic applications. The seamless integration of theoretical concepts with practical implementation in this lab not only enhances our understanding of digital design but also equips us with valuable skills for future endeavors in microprocessor architecture.

-End-