CS1050-Computer organization and digital design

Lab 6 – Arithmetic Unit

Student Name: De Silva A.D.D.T.

Index Number: 220098C

Group Number: 33

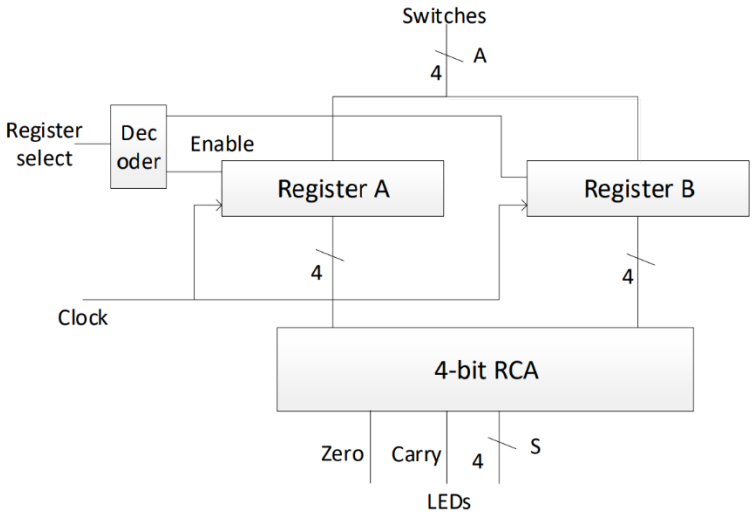
---

1. Introduction

1.1. Introduce the lab task

In this lab, first we developed a 4-bit register using D flip-flops. Then we developed a 4-bit arithmetic unit that can add 2 numbers stored in two different registers. The slow clock and the ripple carry adder (RCA) that we developed in previous labs were used to develop this arithmetic unit. Finally, we verified the functionality of arithmetic unit using simulation.

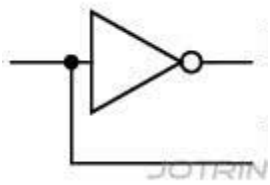
The high-level diagram of AU is as shown below.



1.2. Implementation of decoding

In this circuit which register to load is determined by the input to the Enable pin of a register which is controlled via Register select. This process is similar to the process of a 1-to-2 decoder. As my opinion implementing a 1-to-2 decoder is not necessary for this lab. We can do that process only using Boolean expressions of a 1-to-2 decoder.

RegSel	Enable_A	Enable_A
0	1	0
1	0	1



Boolean expressions: -

Enable_A= not (RegSel)

Enable_B= RegSel

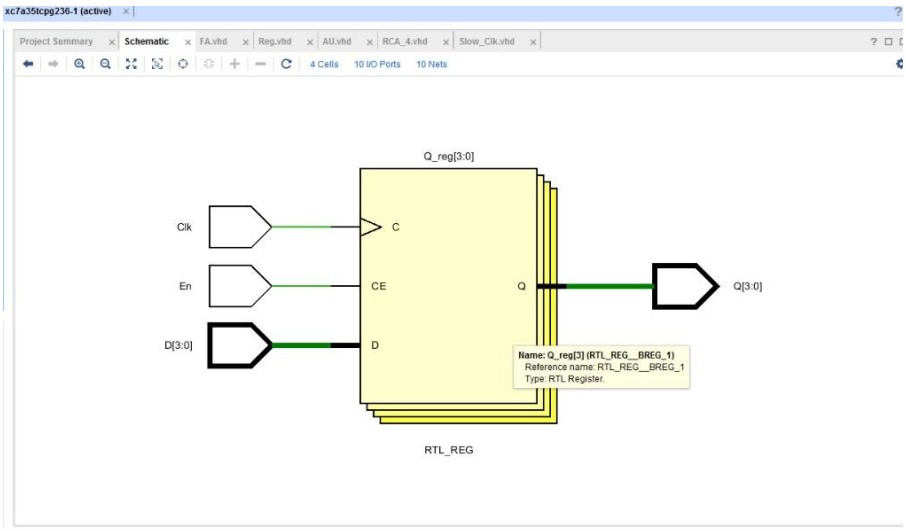
2. 4-bit register

2.1. Design source file

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer: DE SILVA A D D T
-- Create Date: 03/15/2024 01:40:07 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 -- respond when clock rises
        if En = '1' then -- Enable should be set
            Q <= D;
        end if;
    end if;
end process;


end Behavioral;
```

2.2. Elaborated design schematic



3. Arithmetic Unit

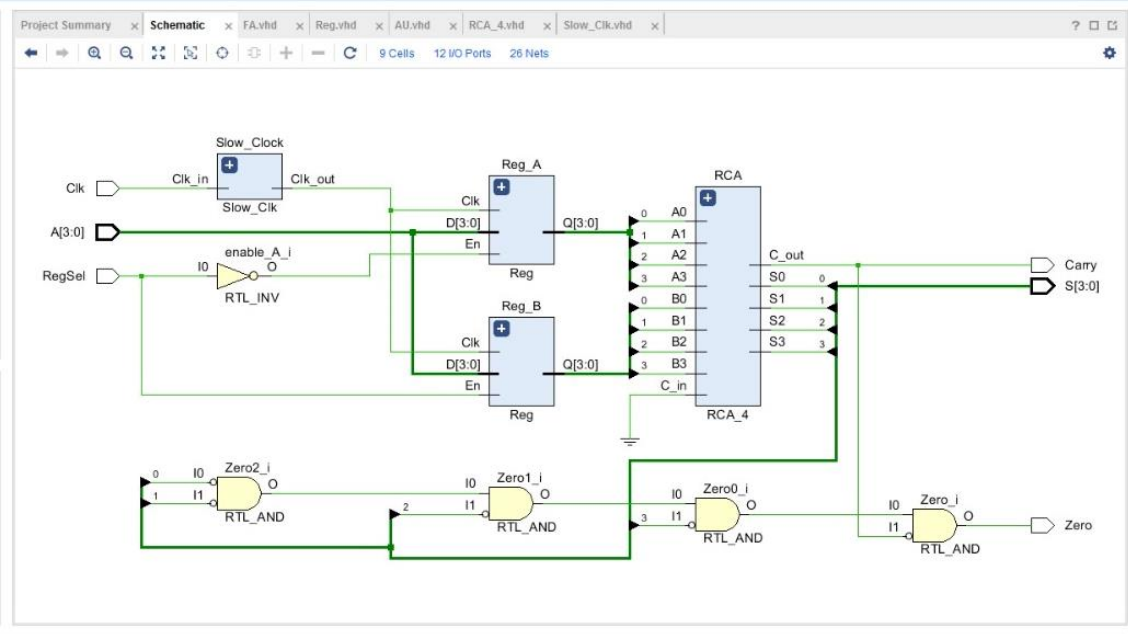
3.1. Design source file

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer: DE SILVA A D D T
--
-- Create Date: 03/15/2024 01:33:52 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 Slow_Clk
    Port ( Clk_in : in STD_LOGIC;
           Clk_out : out STD_LOGIC);
end component;

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;

signal A_output,B_output,S_output: std_logic_vector(3 downto 0);
signal Clk_slow,enable_A,enable_B,carry_out: std_logic;

begin

Slow_Clock: Slow_Clk

port map(
Clk_in=>Clk,
Clk_out=>Clk_slow
);

Reg_A: Reg
```

```
port map(
    D=>A,
    En=>enable_A,
    Clk=>Clk_slow,
    Q=>A_output);

Reg_B: Reg
port map(
    D=>A,
    En=>enable_B,
    Clk=>Clk_slow,
    Q=>B_output);


RCA:RCA_4
port map(
       A0 =>A_output(0),
       A1=>A_output(1),
       A2 =>A_output(2),
       A3 =>A_output(3),
       B0 =>B_output(0),
       B1 =>B_output(1),
       B2 =>B_output(2),
       B3 =>B_output(3),
       C_in =>'0',
       S0 =>S_output(0),
       S1 =>S_output(1),
       S2 =>S_output(2),
       S3 =>S_output(3),
       C_out=>carry_out
);
Zero<=NOT(S_output(0)) AND NOT(S_output(1)) AND NOT(S_output(2)) AND
NOT(S_output(3)) AND NOT(carry_out);
S<=S_output;
carry<=carry_out;
enable_A<=not RegSel;
enable_B<=RegSel;

end Behavioral;
```

3.2. Elaborated design schematic



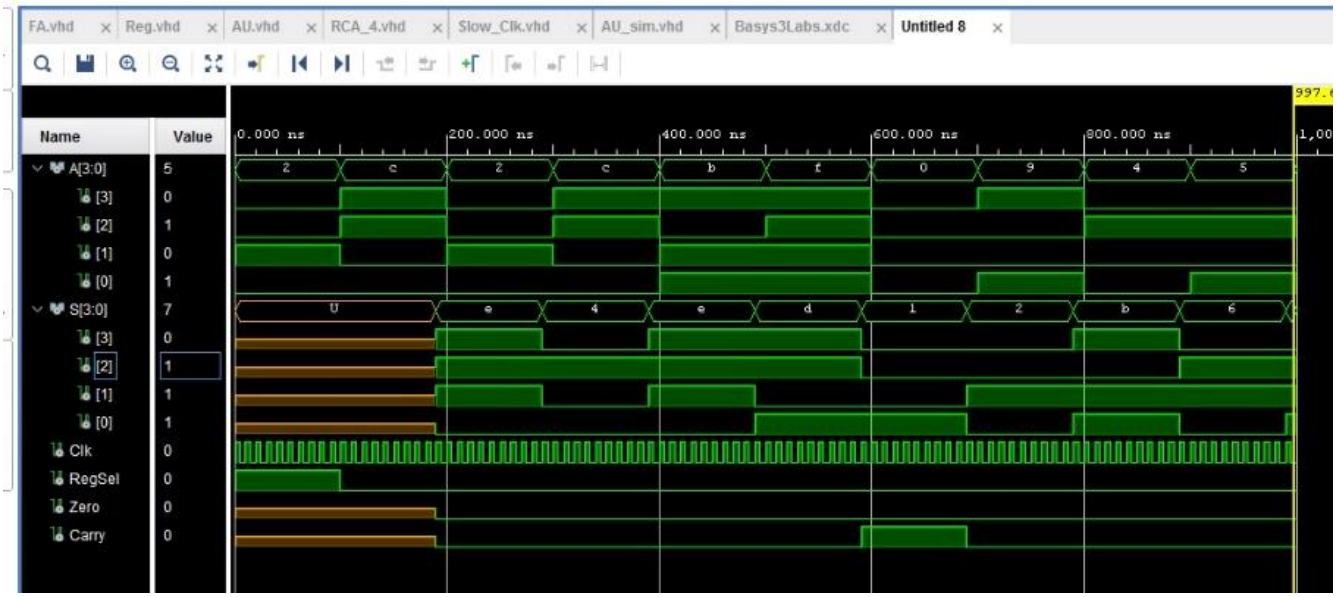
3.3. Simulation source file

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer: DE SILVA A D D T
--
-- Create Date: 03/15/2024 03:32:25 PM
-- Design Name:
-- Module Name: AU_sim - 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;
```

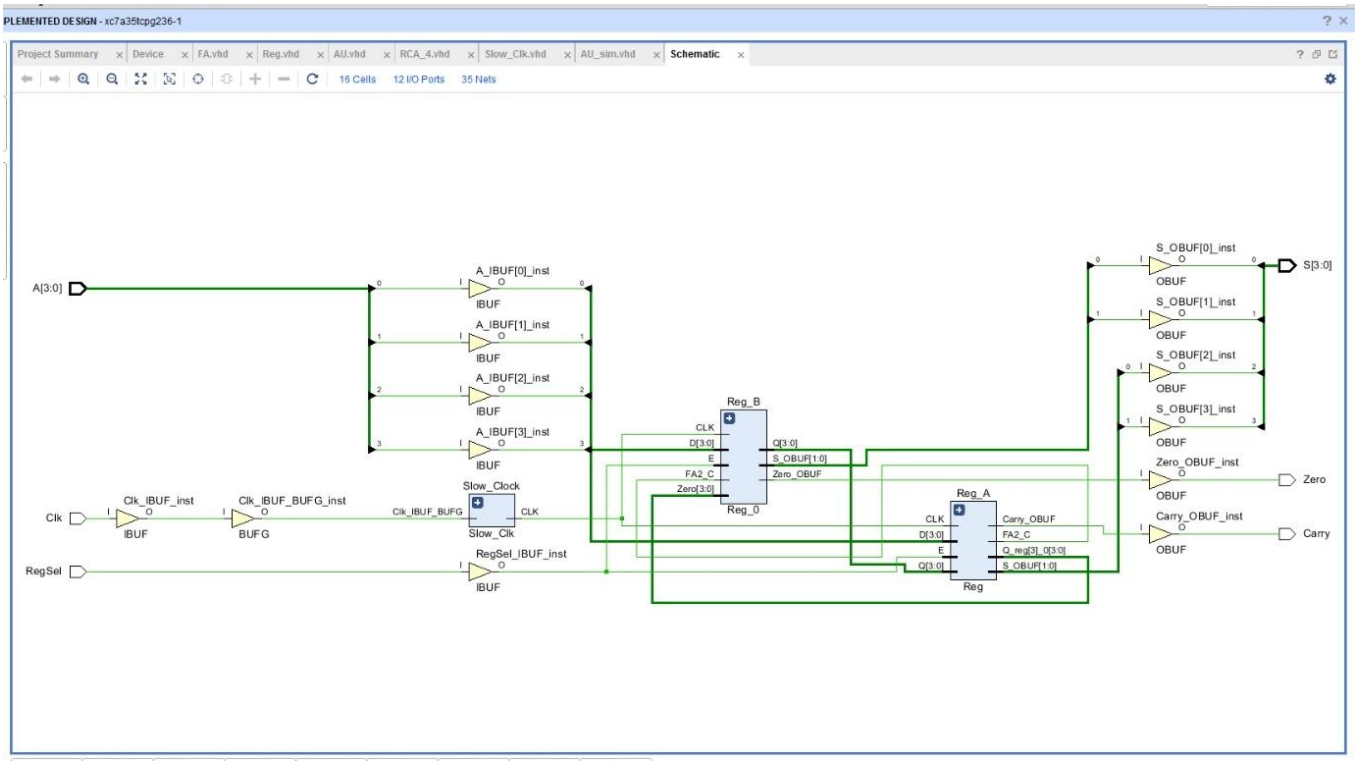
```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

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

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

architecture Behavioral of AU_sim 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,S: STD_LOGIC_VECTOR (3 downto 0);
signal Clk,RegSel,Zero,Carry: STD_LOGIC:='0';

begin

UUT: AU
    Port map(
           A =>A,
           RegSel=>RegSel,
           Clk=>Clk,
           S =>S,
           Zero =>Zero,
           Carry=>Carry );
process
begin

Clk<=not(Clk);
wait for 5 ns;
end process;
```

```
process
begin

A<="0010";
RegSel<='1';
wait for 100 ns;

A<="1100";
RegSel<='0';
wait for 100 ns;

A<="0010";
RegSel<='0';
wait for 100 ns;

A<="1100";
wait for 100 ns;

A<="1011";
wait for 100 ns;

A<="1111";
wait for 100 ns;

A<="0000";
wait for 100 ns;

A<="1001";
wait for 100 ns;

A<="0100";
wait for 100 ns;

A<="0101";
wait;

end process;

end Behavioral;
```

3.4. Timing diagram



3.5. Implemented design schematic



3.6. Constraints file (XDC file - without commented lines)

```
## This file is a general .xdc for the Basys3 rev B board
## To use it in a project:
## - uncomment the lines corresponding to used pins
## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project

## 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}]

```

4. Conclusion

Function of 4-bit-register is to store 4-bit binary number. This 4-bit AU calculates the sum of two numbers stored in 2 different registers using a 4-bit-RCA.

The clock speed is deliberately reduced to 0.5 Hz to ensure the visibility of LED outputs using slow_clock.

The zero flag is used to check the result of an arithmetic operation, including bitwise logical instructions.