EL323//El324 Digital System Laboratory II

LAB 10:Simple Processor without Control Circuit

Objective:

1. Can use the Quartus II and create the new project using VHDL
2. To know the assign pin of DE-0 in Quartus.
3. Design using VHDL
4. To know the Processor.

LAB 10.1 ALU4BITS VHDL CODE

1. Create new Project name “SimpleCPU”.
2. Create new entity for SimpleCPU in this below.

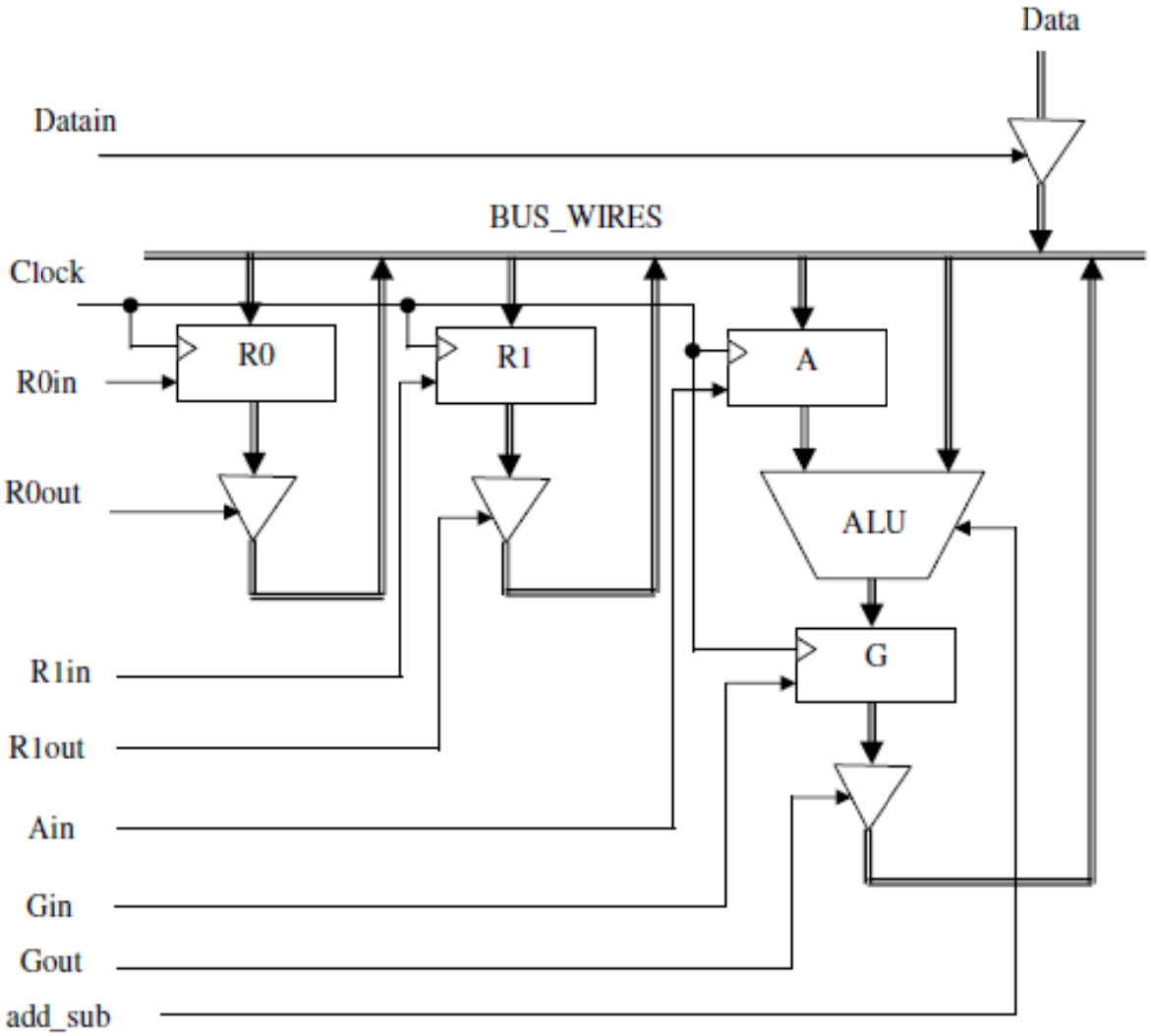
**** Note: GPIO0 (28-30) must using Vcc at 3.3 v for Logic ‘1’ and using Gnd for Logic ‘0’ ****
**** You must read the DE0-manual before connected. ****

Connect	**To**
Data[3..0]	Sw[3 downto 0]
Datain	Sw9
R1in	Sw8
R1out	Sw7
R0in	Sw6
R0out	Sw5
Ain	Sw4
Gin	GPIO0_D28
Gout	GPIO0_D29
Add_sub	GPIO0_D30
r_select	Button0
Clock	Button2
Dec_7	Hex0 and Hex1

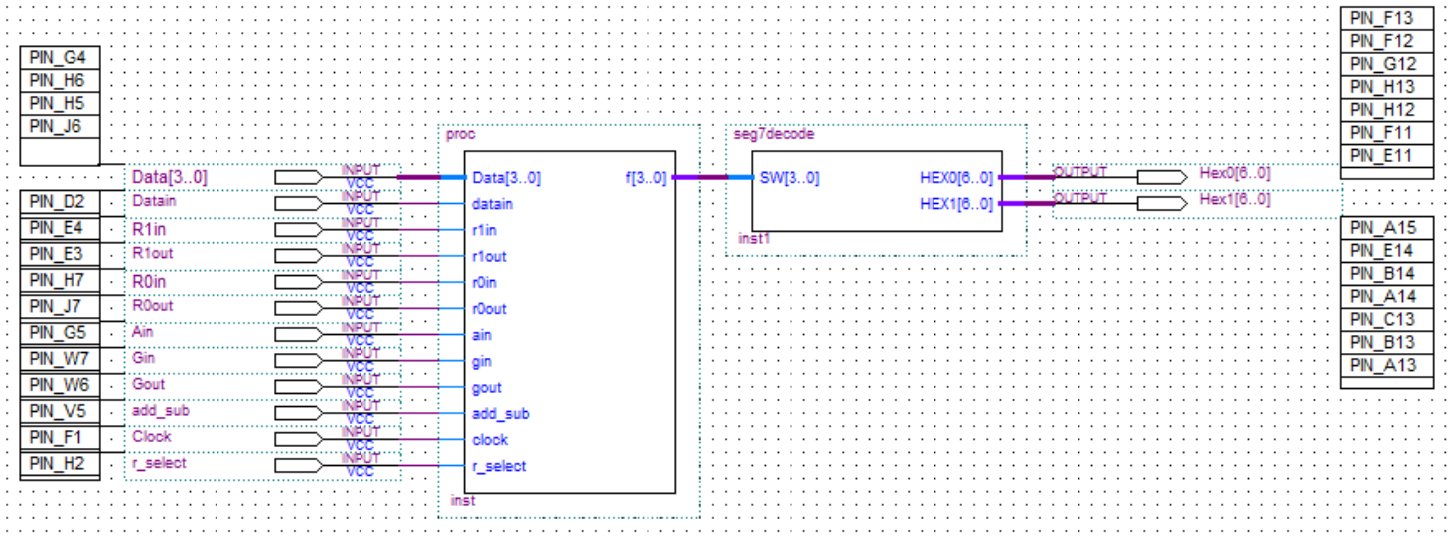
The defined of Switch and button to show the 7segdisplay.

Connector	**Value**	**Description**
Add_sub	0	A + Bus_wire
	1	A – Bus_wire
r_select	0	r1
	1	r0

The description of add_sub and r_select.



3. Use the DE0 User Manual to define the DE0 pin.
4. Compile the code and program to DE0.



CODE of "Proc"

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
entity proc is
        port (  Data : in std_logic_vector (3 downto 0);
                        datain : in std_logic;
                        r1in, r1out : in std_logic;
                        r0in, r0out : in std_logic;
                        ain : in std_logic;
                        gin, gout : in std_logic;
                        add_sub : in std_logic;
                        clock : in std_logic;
                        r_select : in std_logic;
                        f : out std_logic_vector (3 downto 0));
end proc;
architecture behavior of proc is
        signal bus_wires, sum, r0, r1, a, g: std_logic_vector (3 downto 0);
        begin

                registers:
                process
                begin
                        wait until clock'event and clock = '1';
                        if r1in = '1' then r1 <= bus_wires; end if;
                        if r0in = '1' then _______________; end if;
                        if ain = '1' then _______________; end if;
                        if gin = '1' then _______________;
                end process;

                mainbus:
                bus_wires <=    r1 when (r1out = '1' ) else
                                r0 when (_______________) else
                                g when (_______________) else
                                data when (_______________) else
                                "0000";

                alu:
                with add_sub select
                sum <= a + bus_wires when '0',
                        _______________;

                display_register:
                with r_select select
                f <=    r0 when '1',
                        r1 when others;

end behavior;
```

Result

			Button2	Sw3..0	Sw9	Sw8	Sw7	Sw6	Sw5	Sw4	GPIO0 _D28	GPIO0 _D29	GPIO0 _D30	Button0
RTL	R0	R1	Clock	Data	Datain	R1in	R1out	R0in	R0out	Ain	Gin	Gout	Add_sub	r_select
R0 <- 10	10	0	1 pulse	10	1	0	0	1	0	0	0	0	0	1
R1 <- 4	10	4	1 pulse	4										
A <- R1	10	4	1 pulse	x										
G <- A+R0	10	4	1 pulse	x										
R0 <- G	14	4	1 pulse	x										
R1 <- R0	14	14	1 pulse	x										
R1 <- 6	14	6	1 pulse	6										
A <- R0	14	6	1 pulse	x										
G <- A – R1	14	6	1 pulse	x										
R1 <- G	14	8	1 pulse	x										

Conclusion

……

……

……

……

……

……