Lab 2

Pragaash Mohan

September 30, 2020

Part V

Main VHDL

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

entity part_5 is
        port
        (
                SW                  :         in std_logic_vector(8 downto 0);
                LEDR          :         out std_logic_vector(8 downto 0);
                HEX0, HEX1, HEX3, HEX5          :         out std_logic_vector(0 to 6)
        );
end part_5;

architecture Behavioral of part_5 is

        component Bdc7Seg port
        (
                c                 : in std_logic_vector(3 downto 0);
                disp          : out std_logic_vector(0 to 6)
        );
        end component;

        component Cir_B port
        (
                c                 : in std_logic;
                disp          : out std_logic_vector(0 to 6)
        );
        end component;

        signal a                  : std_logic_vector(3 downto 0);
```

```
signal b                : std_logic_vector(3 downto 0);
signal t0               : std_logic_vector(4 downto 0);
signal c0               : std_logic;
signal z0               : std_logic_vector(3 downto 0);
signal s0               : std_logic_vector(3 downto 0);
signal st               : std_logic_vector(4 downto 0);
signal s1               : std_logic;

begin
a <= SW(7 downto 4); --input
b <= SW(3 downto 0); --input
c0 <= SW(8); --carry

LEDR(7 downto 4) <= a;
LEDR(3 downto 0) <= b;
LEDR(8) <= c0;


process (a, b, c0)

begin
        t0 <= ('0' & a) + ('0' & b) + c0; --adders
        if t0 > 9 then --mux
                z0 <= "1010";
                s1 <= '1';
        else
                z0 <= "0000";
                s1 <= '0';
        end if;
        st <= t0 - z0;
        s0 <= st(3 downto 0);
end process;

--Display instances:
DISP3 : Bdc7Seg port map --Value of A
( a, HEX5 );

DISP2 : Bdc7Seg port map --Value of B
( b, HEX3 );

DISP1 : Cir_B port map --Value of S1
( s1, HEX1 );

DISP0 : Bdc7seg port map --Value of S0
```

```
        ( s0, HEX0 );

end Behavioral;
```

Bdc7seg.vhd

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

entity Bdc7seg is
        port
        (
                c               : in std_logic_vector(3 downto 0);
                disp        : out std_logic_vector(0 to 6)
        );
end Bdc7seg;

architecture Behavioral of Bdc7seg is
begin
        disp(0) <= (c(2) and not c(1) and not c(0)) or (not c(3) and not c(2) and no
        disp(1) <= (c(2) and not c(1) and c(0)) or (c(2) and c(1) and not c(0));
        disp(2) <= not c(2) and c(1) and not c(0);
        disp(3) <= (c(2) and not c(1) and not c(0)) or (not c(3) and not c(2) and no
        disp(4) <= (c(2) and not c(1) and not c(0)) or c(0);
        disp(5) <= (not c(3) and not c(2) and not c(1) and c(0)) or (c(1) and c(0))
        disp(6) <= (not c(3) and not c(2) and not c(1)) or (c(2) and c(1) and c(0))
end Behavioral;
```

Cir$_B$*.vhd*

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

entity Cir_B is
        port
        (
                c               : in std_logic;
                disp        : out std_logic_vector(0 to 6)
        );
end Cir_b;

architecture Behavioral of Cir_B is

        begin

        disp(0) <= c;
        disp(1) <= '0';
        disp(2) <= '0';
        disp(3) <= c;
        disp(4) <= c;
        disp(5) <= c;
        disp(6) <= '1';


end Behavioral;
```

Part VI

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

entity part_6 is
        port
        (
                SW                  : in std_logic_vector(9 downto 0);
                LEDR          : out std_logic_vector(9 downto 0);
                HEX0, HEX1 : out std_logic_vector(0 to 6)
        );
end part_6;

architecture Behavioral of part_6 is

        component Bdc7seg port
        (
                c                  : in std_logic_vector(3 downto 0);
                disp          : out std_logic_vector(0 to 6)
        );
        end component;

        signal v         : std_logic_vector(5 downto 0);
        signal d0 : std_logic_vector(5 downto 0);
        signal d1 : std_logic_vector(3 downto 0);

        begin
        v <= SW(5 downto 0);
        LEDR(5 downto 0) <= v;

        process (v)
        begin --if/else-instances:
                if v > "111011" then
                        d0 <= v - "111100";
                        d1 <= "0110";
                elsif v > "110001" then
                        d0 <= v - "110010";
                        d1 <= "0101";
                elsif v > "100111" then
                        d0 <= v - "101000";
                        d1 <= "0100";
                elsif v > "011101" then
```

```
                        d0 <= v - "011110";
                        d1 <= "0011";
                elsif v > "010011" then
                        d0 <= v - "010100";
                        d1 <= "0010";
                elsif v > "001001" then
                        d0 <= v - "001010";
                        d1 <= "0001";
                else
                        d0 <= v;
                        d1 <= "0000";
                end if;
        end process;

        --Display instances:
        DISP1 : Bdc7seg port map
        ( d1, HEX1 );

        DISP0 : Bdc7seg port map
        ( d0(3 downto 0), HEX0 );

end Behavioral;
```