LAB3 GRP7 SESS202 REPORT

VHDL SOURCE CODE 1: TOP LEVEL STRUCTURAL

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


entity LogicalStep_Lab3_top is port (
   clkin_50            : in    std_logic;
        pb                              : in    std_logic_vector(3 downto 0);
        sw                      : in  std_logic_vector(7 downto 0); -- The switch inputs
   leds                : out std_logic_vector(7 downto 0);   -- for displaying the switch content
   seg7_data    : out std_logic_vector(6 downto 0); -- 7-bit outputs to a 7-segment
        seg7_char1  : out     std_logic;                                                    -- seg7
digi selectors
        seg7_char2  : out     std_logic                                                     -- seg7
digi selectors

);
end LogicalStep_Lab3_top;

architecture Energy_Monitor of LogicalStep_Lab3_top is



-- Components Used
------------------------------------------------------------------
component fourbitcomparator port(

        bitA0, bitA1, bitA2, bitA3, bitB0, bitB1, bitB2, bitB3 : in std_logic;

        AGTB    : out std_logic;
        AEQB    : out std_logic;
        ALTB    : out std_logic

        );
end component;


component SevenSegment port ( -- converts a 4-bit number to 7 bit-number to be use for the 7-
segment display
   hex                :  in  std_logic_vector(3 downto 0);   -- The 4 bit data to be displayed
   sevenseg    :  out std_logic_vector(6 downto 0)    -- 7-bit outputs to a 7-segment
   );
end component;


component segment7_mux port ( -- the mux to turn the 2 7-segment display on simultaneously
               clk     : in  std_logic :='0';
               DIN2    : in std_logic_vector(6 downto 0);
               DIN1    : in std_logic_vector(6 downto 0);
               DOUT    : out std_logic_vector(6 downto 0);
               DIG2    : out std_logic;
               DIG1    : out std_logic
               );
end component;

component EnergyControl is port (
        CurrentGreater, CurrentEqual, CurrentSmaller : in std_logic;
        DoorWindowOpenControl : in std_logic_vector(2 downto 0); -- our push buttons
        -- remember it is active low, so 0 means door open
        --Fdoor->pb2 window->pb1 Bdoor ->pb0
        --Fdoor-> led6 window->led5, Bdoor -> led4

        FurnaceON: out std_logic; -- 1 if on
        SystemAtTemp: out std_logic;-- 1 if on
        ACON: out std_logic;-- 1 if on
```

```
        BlowerON: out std_logic;-- 1 if on
        DoorWindowOpenOutput: out std_logic_vector(2 downto 0)-- 1 if on
        );
end component;
-----------------------------------------------------------------
signal Current_Temp : std_logic_vector(3 downto 0); -- represents the current Temp
signal Desired_Temp : std_logic_vector(3 downto 0); -- represents the desired temp
signal GT, EQ, LT : std_logic;
signal sevenseg_Current_Temp, sevenseg_Desired_Temp : std_logic_vector(6 downto 0); -- the signal
that goes directly into the sevenseg mux



begin
----- input from our switches for temperature -----------------
Current_Temp <= sw(7 downto 4); -- left side input
Desired_Temp <= sw(3 downto 0); -- right side input



---------------- comparing Current_Temp to Desired_Temp -----------------------------
        INST1: fourbitcomparator port
map(Current_Temp(0),Current_Temp(1),Current_Temp(2),Current_Temp(3),
        Desired_Temp(0),Desired_Temp(1),Desired_Temp(2),Desired_Temp(3),
        GT,EQ,LT); -- a bit inefficient in terms of amount if input, but does the job
                                -- could be cleaner


------------------DISPLAYING THE TEMPS--------------------
        INST2: SevenSegment port map(Current_Temp, sevenseg_Current_Temp);
        INST3: sevenSegment port map(Desired_Temp, sevenseg_Desired_Temp);
        INST4: segment7_mux port map (clkin_50       , sevenseg_Current_Temp,
sevenseg_Desired_Temp, seg7_data, seg7_char1, seg7_char2);
------------------END OF DISPLAYING-----------------------

------------------CONTROLLING THE outputs using our Energy Control component----------------------
---------
        INST5: EnergyControl port map(GT,EQ,LT, pb(2 downto 0), leds(0), leds(1), leds(2),
leds(3), leds(6 downto 4) );

---------------

end Energy_Monitor;
```

VHDL SOURCE CODE 2: 1-BIT COMPARATOR

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


entity SingleBitComparator is port (

        bit1    : in std_logic;
        bit2    : in std_logic;

        gt      : out std_logic;
        et      : out std_logic;
        lt      : out std_logic

        );

end SingleBitComparator;

architecture Behavioural of SingleBitComparator is



Begin

gt <= bit1 and not bit2; --bit1 is 1 and bit2 is 0
et <= NOT (bit1 XOR bit2); --when bit1 and bit2 are the same values
lt <= not bit1 and bit2; --bit1 is 0 and bit2 is 1


end Behavioural;
```

VHDL SOURCE CODE 3: 4-BITS COMPARATOR

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


entity FourBitComparator is port (

	bitA0, bitA1, bitA2, bitA3, bitB0, bitB1, bitB2, bitB3 : in std_logic;

	AGTB    : out std_logic;
	AEQB    : out std_logic;
	ALTB    : out std_logic

	);

end FourBitComparator;

architecture Behavioural of FOurBitComparator is

component SingleBitComparator port(
	bit1    : in std_logic;
	bit2    : in std_logic;

	gt      : out std_logic;
	et      : out std_logic;
	lt      : out std_logic
);

end component;

signal A3LTB3, A3GTB3, A3EQB3, A2LTB2, A2GTB2, A2EQB2,A1LTB1, A1GTB1, A1EQB1, A0LTB0, A0GTB0,
A0EQB0 : std_logic;
signal ALTB_signal , AEQB_signal , AGTB_signal : std_logic;


Begin

	INST1: SingleBitComparator port map(bitA0, bitB0, A0GTB0,A0EQB0, A0LTB0);
	INST2: SingleBitComparator port map(bitA1, bitB1, A1GTB1,A1EQB1, A1LTB1);
	INST3: SingleBitComparator port map(bitA2, bitB2, A2GTB2,A2EQB2, A2LTB2);
	INST4: SingleBitComparator port map(bitA3, bitB3, A3GTB3,A3EQB3, A3LTB3);


	-- equal if all 4 bits compare to equal
	AEQB_signal <= A3EQB3 and A2EQB2 and A1EQB1 and A0EQB0;

	-- greater if all bits compared early are equal and this bit comparing is bigger
	AGTB_signal <= (A3GTB3) or (A3EQB3 and A2GTB2) or (A3EQB3 and A2EQB2 AND A1GTB1) OR
(A3EQB3 AND A2EQB2 AND A1EQB1 AND A0GTB0);

	--else....
	ALTB <= not(AEQB_signal OR AGTB_signal);


	-- just setting the other signal to the real output
	AEQB <= AEQB_signal;
	AGTB <= AGTB_signal;


end Behavioural;
```

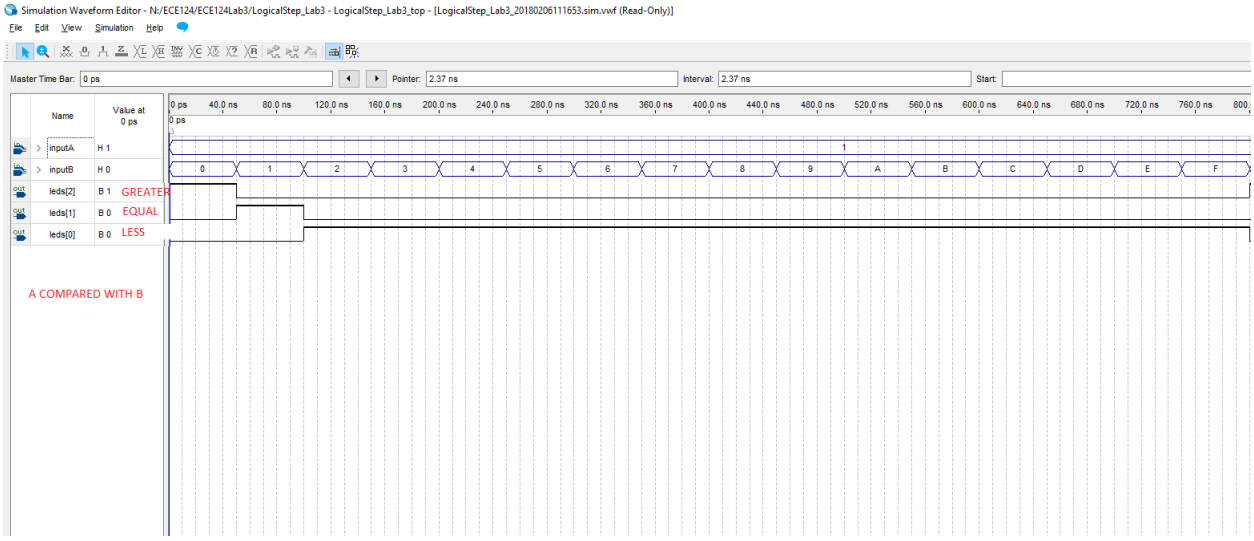
VHDL SOURCE CODE 4: ENERGY CONTROL

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


entity EnergyControl is port (
	CurrentGreater, CurrentEqual, CurrentSmaller : in std_logic;
	DoorWindowOpenControl : in std_logic_vector(2 downto 0); -- our push buttons
	-- remember it is active low, so 0 means door open
	FurnaceON: out std_logic; -- 1 if on
	SystemAtTemp: out std_logic;-- 1 if on
	ACON: out std_logic;-- 1 if on
	BlowerON: out std_logic;-- 1 if on
	DoorWindowOpenOutput: out std_logic_vector(2 downto 0)-- 1 if on

);
end EnergyControl;


architecture Behaviour of EnergyControl is

	signal isolatedSystem: std_logic; --- 1 if it is isolated

begin
	-- hooking up the PB to LEDS to show if anything is open
	DoorWindowOpenOutput(0) <= not DoorWindowOpenControl(0);

	DoorWindowOpenOutput(1) <= not DoorWindowOpenControl(1);
	DoorWindowOpenOutput(2) <= not DoorWindowOpenControl(2);

	-- check if everything is shut
	isolatedSystem <= DoorWindowOpenControl(0) AND DoorWindowOpenControl(1) AND
DoorWindowOpenControl(2);



	-- check if blower should be on
	BlowerON <= (CurrentGreater or CurrentSmaller) AND (isolatedSystem);

	-- check if AC should be on
	ACON <= (CurrentGreater)AND (isolatedSystem);

	-- check if Furnace should be on
	FurnaceON <= (CurrentSmaller) AND (isolatedSystem);

	--check if current temperature is equal
	SystemAtTemp <= (CurrentEqual);




end Behaviour;
```

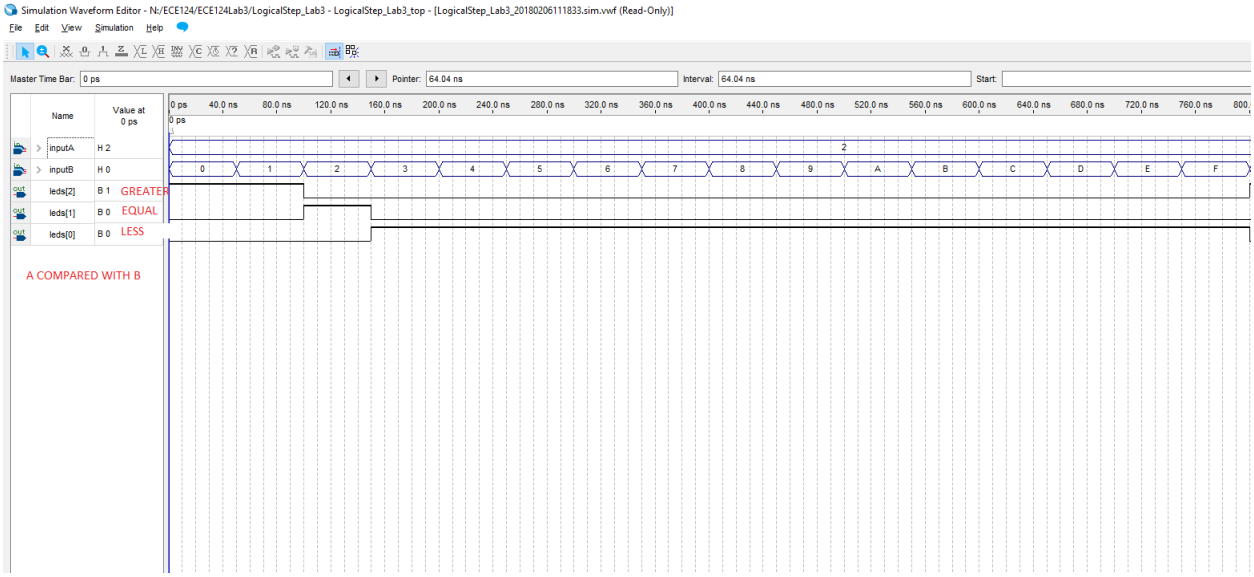
TABLE1: COMPARATOR TRUTH TABLE

Comparison Inputs from 1-Bit Comparators												4-Bit Comparator Outputs		
A3<B3	A3=B3	A3>B3	A2<B2	A2=B2	A2>B2	A1<B1	A1=B1	A1>B1	A0<B0	A0=B0	A0>B0	A<B	A=B	A>B
0	0	1	X	X	X	X	X	X	X	X	X	0	0	1
1	0	0	X	X	X	X	X	X	X	X	X	1	0	0
0	1	0	0	0	1	X	X	X	X	X	X	0	0	1
0	1	0	1	0	0	X	X	X	X	X	X	1	0	0
0	1	0	0	1	0	0	0	1	X	X	X	0	0	1
0	1	0	0	1	0	1	0	0	X	X	X	1	0	0
0	1	0	0	1	0	0	1	0	0	0	1	0	0	1
0	1	0	0	1	0	0	1	0	1	0	0	1	0	0
0	1	0	0	1	0	0	1	0	0	1	0	0	1	0

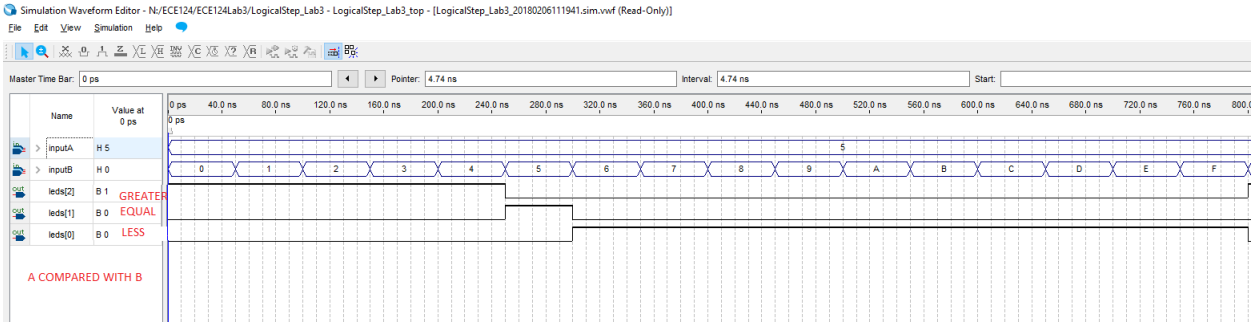
GRAPH1: SIMULATION1 where HEXA = 1



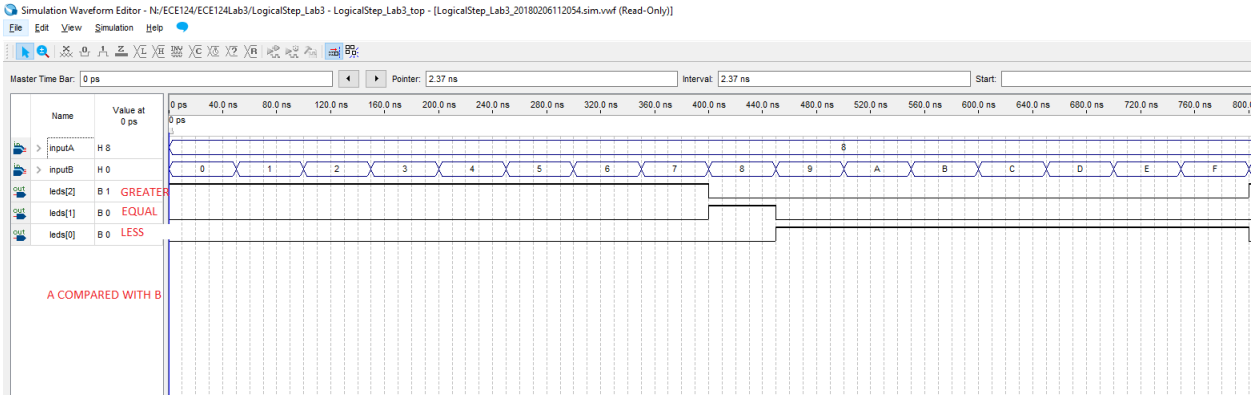
GRAPH2: SIMULATION2 where HEXA = 2



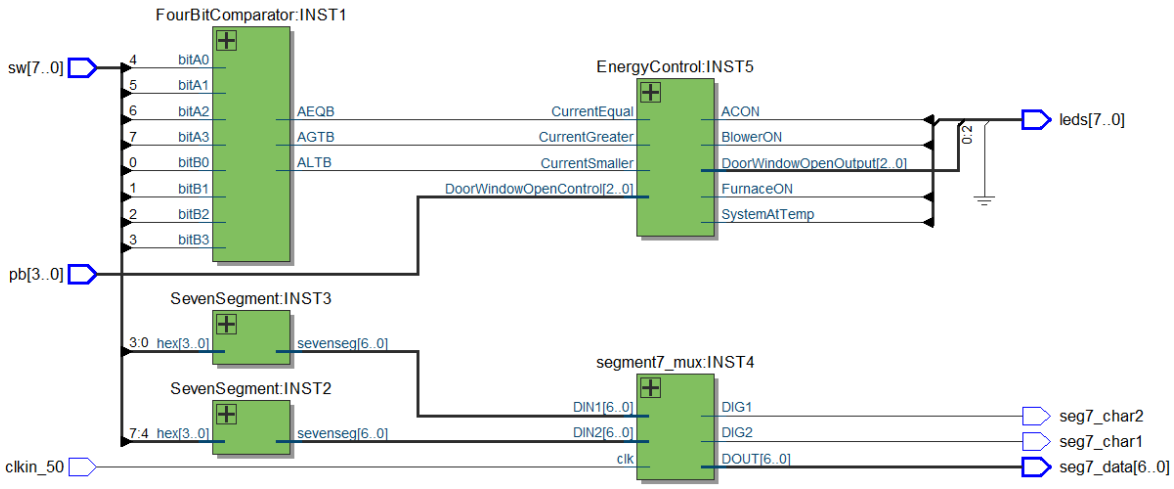
GRAPH3: SIMULATION3 where HEXA = 5



GRAPH4: SIMULATION4 where HEXA = 8



GRAPH5: RTL OF TOP LEVEL DESGIN



GRAPH6: TOTAL DESIGN LOGIC ELEMETNS

Flow Status	Successful - Sun Mar 04 14:56:21 2018
Quartus Prime Version	15.1.0 Build 185 10/21/2015 SJ Standard Edition
Revision Name	LogicalStep_Lab3_top
Top-level Entity Name	LogicalStep_Lab3_top
Family	MAX 10
Device	10M08SAE144C8G
Timing Models	Final
Total logic elements	43 / 8,064 (< 1 %)
Total combinational functions	43 / 8,064 (< 1 %)
Dedicated logic registers	11 / 8,064 (< 1 %)
Total registers	11
Total pins	30 / 101 (30 %)
Total virtual pins	0
Total memory bits	0 / 387,072 (0 %)
Embedded Multiplier 9-bit elements	0 / 48 (0 %)
Total PLLs	0 / 1 (0 %)
UFM blocks	0 / 1 (0 %)
ADC blocks	0 / 1 (0 %)