library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_unsigned.all;

use IEEE.std\_logic\_arith.all;

entity health\_display is

port (

health: in std\_logic\_vector(6 downto 0);

LED1, LED2: out std\_logic\_vector(0 to 6));

end health\_display;

architecture behavior of health\_display is

begin

run: process(health)

begin

if health >= "1011010" then

LED1 <= "0000100"; --any value of 90

elsif (health >= "1010000") then

LED1 <= "0000000"; --any value of 80

elsif (health >= "1000110") then

LED1 <= "0001111"; --any value of 70

elsif (health >= "0111100") then

LED1 <= "0100000"; --any value of 60

elsif (health >= "0110010") then

LED1 <= "0100100"; --any value of 50

elsif (health >= "0101000") then

LED1 <= "1001100"; --any value of 40

elsif (health >= "0011110") then

LED1 <= "0000110"; --any value of 30

elsif (health >= "0010100") then

LED1 <= "0010010"; --any value of 20

elsif (health >= "0001010") then

LED1 <= "1001111"; --any value of 10

elsif (health >= "0000001") then

LED1 <= "0000001"; --any value of 0

else

LED1 <= "0111000";

end if; --determines tens value of health display

if (health = "1100011" or health = "1011001" or health = "1001111" or health = "1000101" or health = "0111011" or health = "0110001" or health = "0100111" or health = "0011101" or health = "0010010" or health = "0001001") then

LED2 <= "0000100"; --any value of 9

elsif (health = "1100010" or health = "1011000" or health = "1001110" or health = "1000100" or health = "0111010" or health = "0110000" or health = "0100110" or health = "0011100" or health = "0010010" or health = "0001000") then

LED2 <= "0000000"; --any value of 8

elsif (health = "1100001" or health = "1010111" or health = "1001101" or health = "1000011" or health = "0111001" or health = "0101111" or health = "0100101" or health = "0011011" or health = "0010001" or health = "0000111") then

LED2 <= "0001111"; --any value of 7

elsif (health = "1100000" or health = "1010110" or health = "1001100" or health = "1000010" or health = "0111000" or health = "0101110" or health = "0100100" or health = "0011010" or health = "0010000" or health = "0000110") then

LED2 <= "0100000"; --any value of 6

elsif (health = "1011111" or health = "1010101" or health = "1001011" or health = "1000001" or health = "0110111" or health = "0101101" or health = "0100011" or health = "0011001" or health = "0001111" or health = "0000101") then

LED2 <= "0100100"; --any value of 5

elsif (health = "1011110" or health = "1010100" or health = "1001010" or health = "1000000" or health = "0110110" or health = "0101100" or health = "0100010" or health = "0011000" or health = "0001110" or health = "0000100") then

LED2 <= "1001100"; --any value of 4

elsif (health = "1011101" or health = "1010011" or health = "1001001" or health = "0111111" or health = "0110101" or health = "0101011" or health = "0100001" or health = "0010111" or health = "0001101" or health = "0000011") then

LED2 <= "0000110"; --any value of 3

elsif (health = "1011100" or health = "1010010" or health = "1001000" or health = "0111110" or health = "0110100" or health = "0101010" or health = "0100000" or health = "0010110" or health = "0001100" or health = "0000010") then

LED2 <= "0010010"; --any value of 2

elsif (health = "1011011" or health = "1010001" or health = "1000111" or health = "0111101" or health = "0110011" or health = "0101001" or health = "0011111" or health = "0010101" or health = "0001011" or health = "0000001") then

LED2 <= "1001111"; --any value of 1

elsif (health = "1011010" or health = "1010000" or health = "1000110" or health = "0111100" or health = "0110010" or health = "0101000" or health = "0011110" or health = "0010100" or health = "0001010") then

LED2 <= "0000001"; --any value of 0

else

LED2 <= "0111000";

end if; --determines ones value of health display

end process run;

end behavior;

library ieee;

use ieee.std\_logic\_1164.all;

use ieee.std\_logic\_unsigned.all;

use IEEE.std\_logic\_arith.all;

ENTITY Project is

PORT (pushbutton, pushbutton1, p1s1, p1s2, p1, p2s1, p2s2, p2 : in std\_logic;

P1LED1, P1LED2 : out std\_logic\_vector(0 to 6);

P2LED1, P2LED2 : out std\_logic\_vector(0 to 6));

END Project;

ARCHITECTURE behavior of Project is

signal health1: std\_logic\_vector(6 downto 0);

signal attack1: std\_logic\_vector(6 downto 0);

signal health2: std\_logic\_vector(6 downto 0);

signal attack2: std\_logic\_vector(6 downto 0);

signal healthp1: std\_logic\_vector(6 downto 0);

signal healthp2: std\_logic\_vector(6 downto 0);

--define signals here

COMPONENT health\_display

PORT (health: in std\_logic\_vector(6 downto 0);

LED1, LED2: out std\_logic\_vector(0 to 6));

end COMPONENT;

begin

health1 <= "1001011";

health2 <= "1000000";

BGN: process(pushbutton)

begin

if (pushbutton = '0') then

healthp1 <= health1;

healthp2 <= health2;

end if;

end process;

BTL: process(pushbutton1)

begin

if (pushbutton1 ='0' AND pushbutton = '1') then

healthp1 <= health1 - "0001000";

healthp2 <= health2 - "0001000";

end if;

end process;

BGN1: health\_display PORT MAP (healthp1, P1LED1, P1LED2);

BGN2: health\_display PORT MAP (healthp2, P2LED1, P2LED2);

end behavior;