LIBRARY IEEE;

USE IEEE.STD\_LOGIC\_1164.ALL;

USE IEEE.NUMERIC\_STD.ALL;

ENTITY DISP\_7SEG IS

PORT ( SEG\_0 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_1 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_2 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_3 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_4 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_5 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_6 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SEG\_7 : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

CLK : IN STD\_LOGIC;

AN : OUT STD\_LOGIC\_VECTOR(7 DOWNTO 0);

CA : OUT STD\_LOGIC;

CB : OUT STD\_LOGIC;

CC : OUT STD\_LOGIC;

CD : OUT STD\_LOGIC;

CE : OUT STD\_LOGIC;

CF : OUT STD\_LOGIC;

CG : OUT STD\_LOGIC;

DP : OUT STD\_LOGIC

);

END DISP\_7SEG;

ARCHITECTURE DISP\_7SEG\_BEHAVE OF DISP\_7SEG IS

TYPE DATA\_ARRAY IS ARRAY(0 TO 7) OF STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SIGNAL ANODES : STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SIGNAL CATHODES : STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SIGNAL INDEX\_DISP : INTEGER RANGE 0 TO 7 := 0;

SIGNAL REFRESH : STD\_LOGIC;

SIGNAL DATA : DATA\_ARRAY;

FUNCTION INT\_TO\_SIG(INT : STD\_LOGIC\_VECTOR(7 DOWNTO 0))

RETURN STD\_LOGIC\_VECTOR IS

BEGIN

CASE INT IS

WHEN X"00" => return X"F4"; -- r

when X"01" => return X"C6"; -- u

when X"02" => return X"C4"; -- o

when others => return X"FD"; -- -

end case;

end;

begin

REFRESH\_DRIVER : process(CLK)

variable FREQ\_DIVIDER : unsigned(16 downto 0);

begin

if(rising\_edge(CLK)) then

FREQ\_DIVIDER := FREQ\_DIVIDER + 1;

end if;

REFRESH <= FREQ\_DIVIDER(FREQ\_DIVIDER'LEFT);

end process REFRESH\_DRIVER;

DISP\_DRIVER: process(REFRESH, DATA, INDEX\_DISP)

begin

if(rising\_edge(REFRESH)) THEN

if(INDEX\_DISP = 7) then

INDEX\_DISP <= 0;

else

INDEX\_DISP <= INDEX\_DISP + 1;

end if;

end if;

ANODES <= X"FF";

ANODES(INDEX\_DISP) <= '0';

CATHODES <= INT\_TO\_SIG(DATA(INDEX\_DISP));

end process DISP\_DRIVER;