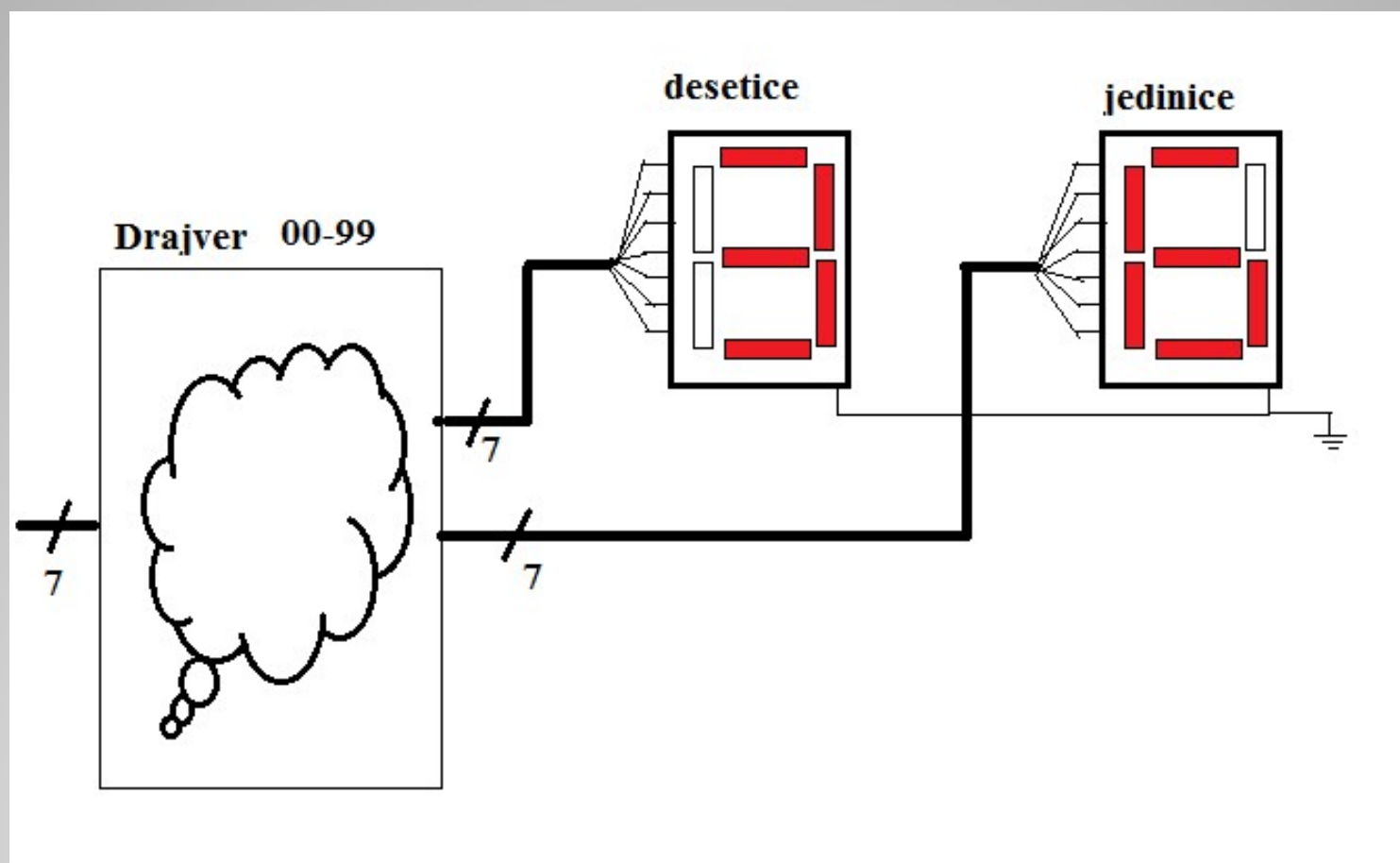


Realizovati drajver za dva sedmosegmentna displeja korišćenjem strukturnog opisa. Najpre je potrebno kreirati par entitet- arhitektura za opis drajvera za jedan sedmosegmentni element koji treba da ispisuje cifre u opsegu od 0 – 9. Nakon toga kreirati opis bloka koji će da ulazni 7-bitni broj (da bi predstavili brojeve u osegu od 0 do 99 potrebno nam je 7 bitova) da rastavi na dva cetvorobitna broja (cifra desetica i jedinica) koji će da se vode na odgovarajuće displeje. Displeji su sa zadedničkom anodom (da bi se upalio segment potrebno je da mu se dovede logička jedinica). Za nedozvoljena stanja ulaza, izlaz staviti u stanje visoke impedanse Z.



OPIS modula za drajvovanje jednog displeja

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Hex7segDriver is
    Port ( bcd : in  STD_LOGIC_VECTOR (3 downto 0);
          segY : out STD_LOGIC_VECTOR (6 downto 0));
end Hex7segDriver;

architecture Behavioral of Hex7segDriver is

begin
    seq: process(bcd)
    begin
        case bcd is
            when "0000" => segY<="1111110";
            when "0001" => segY<="0110000";
            when "0010" => segY<="1101101";
            when "0011" => segY<="1111001";
            when "0100" => segY<="0110011";
            when "0101" => segY<="1011011";
            when "0110" => segY<="1011111";
            when "0111" => segY<="1110000";
            when "1000" => segY<="1111111";
            when "1001" => segY<="1111011";
            when others => segY<="ZZZZZZZ";
        end case;
    end process seq;
end Behavioral;
```

Opis konvertora (razdvaja ulazni 7-bitni broj na dva cetvorobitna broja koja predstavljaju desetivce i jedinice)

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity convertor is
    Port ( Ulaz7bit : in  STD_LOGIC_VECTOR (6 downto 0);
          Izlaz10s : out STD_LOGIC_VECTOR (3 downto 0);
          Izlaz1s : out  STD_LOGIC_VECTOR (3 downto 0));
end convertor;

architecture Behavioral of convertor is

begin
    process(Ulaz7bit)
        variable ulaz: unsigned(6 downto 0);
        variable desetice: unsigned(6 downto 0);
        variable jedinice: unsigned (3 downto 0);
    begin
        ulaz:=unsigned(Ulaz7bit);
        desetice:=ulaz / "0001010";
        jedinice:=ulaz REM "1010";
        for i in 0 to 3 loop
            izlaz10s(i)<=std_logic(desetice(i));
        end loop;
        izlaz1s<=std_logic_vector(jedinice);
    end process;
end Behavioral;
```