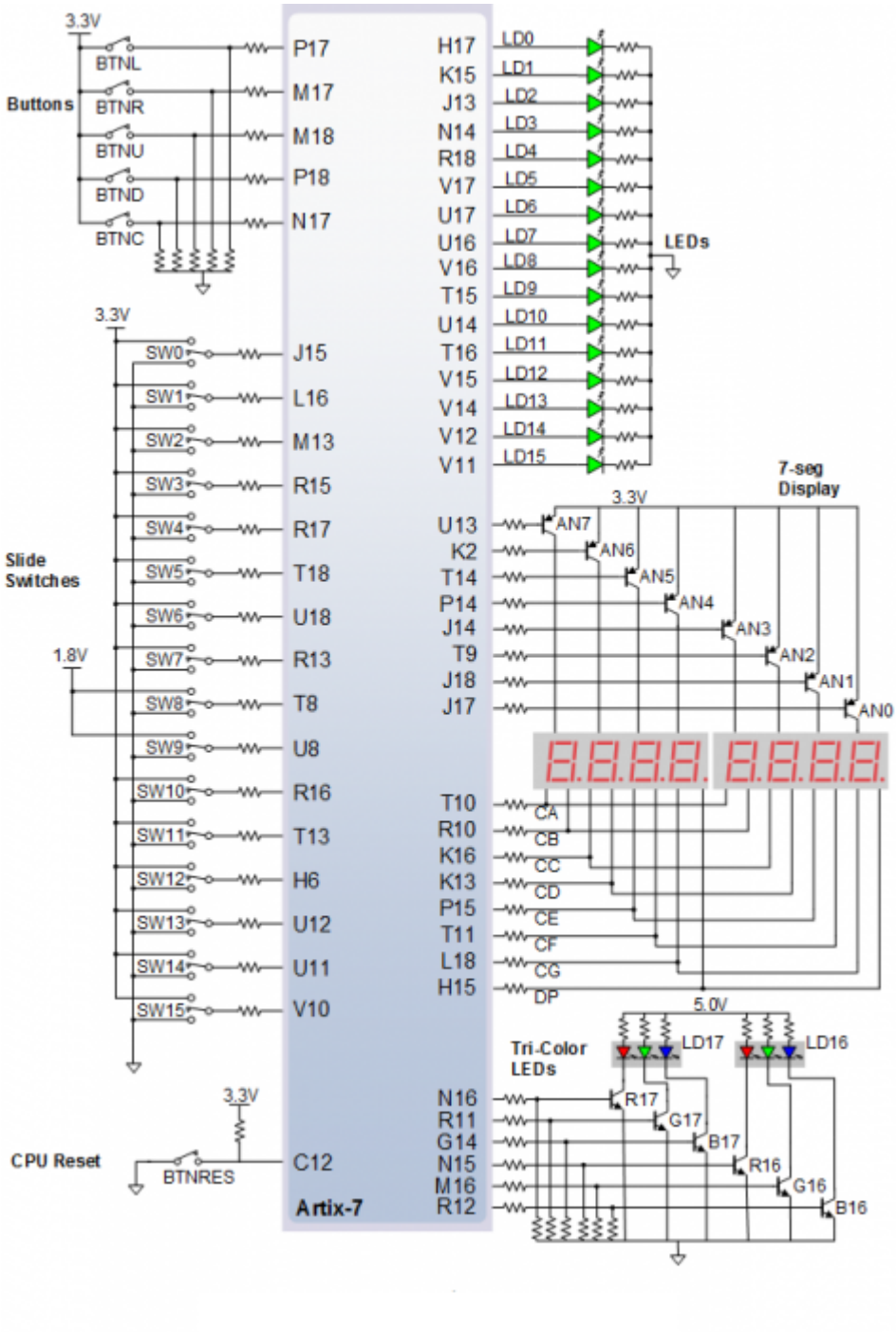
Labs 04-segment

1)

a) Obrázek s připojením 7segmentových displejů na desce Nexys A7



b) Pravdivostní tabulka

Hex	Inputs	A	B	C	D	E	F	G
0	0000	0	0	0	0	0	0	1
1	0001	1	0	0	1	1	1	1
2	0010	0	0	1	0	0	1	0
3	0011	0	0	0	0	1	1	0

Hex	Inputs	A	B	C	D	E	F	G
4	0100	1	1	0	1	1	0	0
5	0101	0	1	0	0	1	0	0
6	0110	0	1	0	0	0	0	0
7	0111	0	0	0	1	1	1	1
8	1000	0	0	0	0	0	0	0
9	1001	0	0	0	0	1	0	0
A	1010	0	0	0	1	0	0	0
b	1011	1	1	0	0	0	0	0
C	1100	0	1	1	0	0	0	1
d	1101	1	0	0	0	0	1	0
E	1110	0	1	1	0	0	0	0
F	1111	0	1	1	1	0	0	0

2)

a) Výpis architektury VHDL ze zdrojového souboru

```
    p_7seg_decoder : process(hex_i)
    begin
        case hex_i is
            when "0000" =>
                seg_o <= "0000001";     -- 0
            when "0001" =>
                seg_o <= "1001111";     -- 1
            when "0010" =>
                seg_o <= "0010010";     -- 2
            when "0011" =>
                seg_o <= "0000110";     -- 3
            when "0100" =>
                seg_o <= "1101100";     -- 4
            when "0101" =>
                seg_o <= "0100100";     -- 5
            when "0110" =>
                seg_o <= "0100000";     -- 6
            when "0111" =>
```

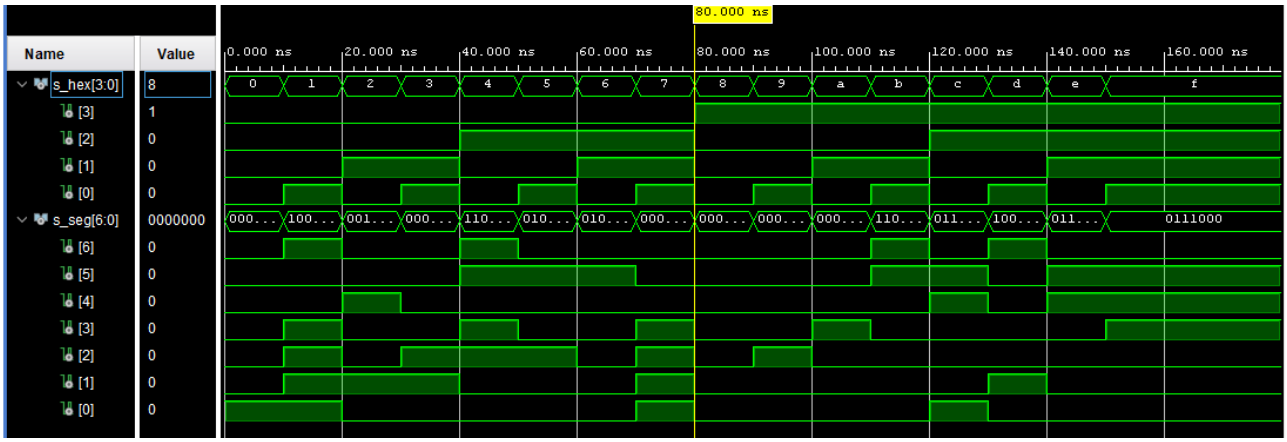
```
                seg_o <= "0001111";     -- 7
            when "1000" =>
                seg_o <= "0000000";     -- 8
            when "1001" =>
                seg_o <= "0000100";     -- 9
            when "1010" =>
                seg_o <= "0001000";     -- A
            when "1011" =>
                seg_o <= "1100000";     -- b
            when "1100" =>
                seg_o <= "0110001";     -- C
            when "1101" =>
                seg_o <= "1000010";     -- d
            when "1110" =>
                seg_o <= "0110000";     -- E
            when others =>
                seg_o <= "0111000";     -- F
        end case;
    end process p_7seg_decoder;
```

b) Výpis stimulačního procesu VHDL ze souboru testbench

```
p_stimulus : process
begin

      -- Report a note at the beginning of stimulus process
       report "Stimulus process started" severity note;


       s_hex <= "0000"; wait for 10 ns;

       s_hex <= "0001"; wait for 10 ns;

       s_hex <= "0010"; wait for 10 ns;

       s_hex <= "0011"; wait for 10 ns;

       s_hex <= "0100"; wait for 10 ns;

       s_hex <= "0101"; wait for 10 ns;

       s_hex <= "0110"; wait for 10 ns;

       s_hex <= "0111"; wait for 10 ns;

       s_hex <= "1000"; wait for 10 ns;

       s_hex <= "1001"; wait for 10 ns;
```

```
        s_hex <= "1010"; wait for 10 ns;

        s_hex <= "1011"; wait for 10 ns;

        s_hex <= "1100"; wait for 10 ns;

        s_hex <= "1101"; wait for 10 ns;

        s_hex <= "1110"; wait for 10 ns;

        s_hex <= "1111"; wait for 10 ns;



        -- Report a note at the end of stimulus process
        report "Stimulus process finished" severity note;
        wait;
    end process p_stimulus;
```

c) Screenshot se simulovanými časovými průběhy



d) Výpis kódu VHDL ze zdrojového souboru instancí 7segmentového modulu

```
hex2seg : entity work.hex_7seg
        port map(
            hex_i    => SW,
            seg_o(6) => CA,
            seg_o(5) => CB,
            seg_o(4) => CC,
            seg_o(3) => CD,
            seg_o(2) => CE,
            seg_o(1) => CF,
            seg_o(0) => CG
        );
```

3)

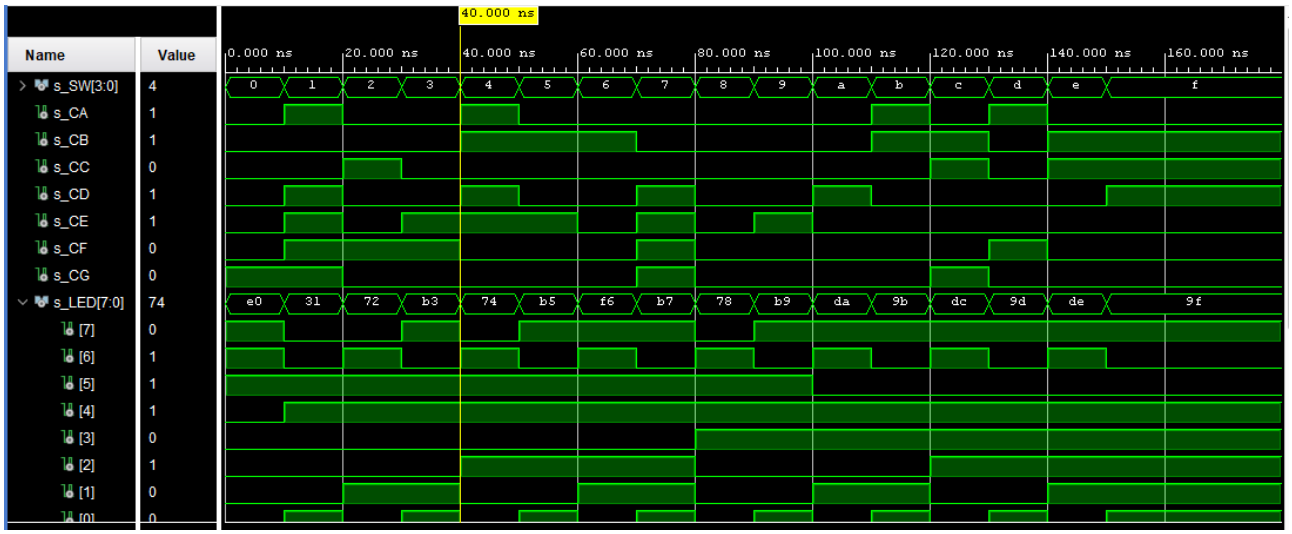
a) Pravdivostní tabulka a výpis VHDL kódu pro LED diody (7:4)

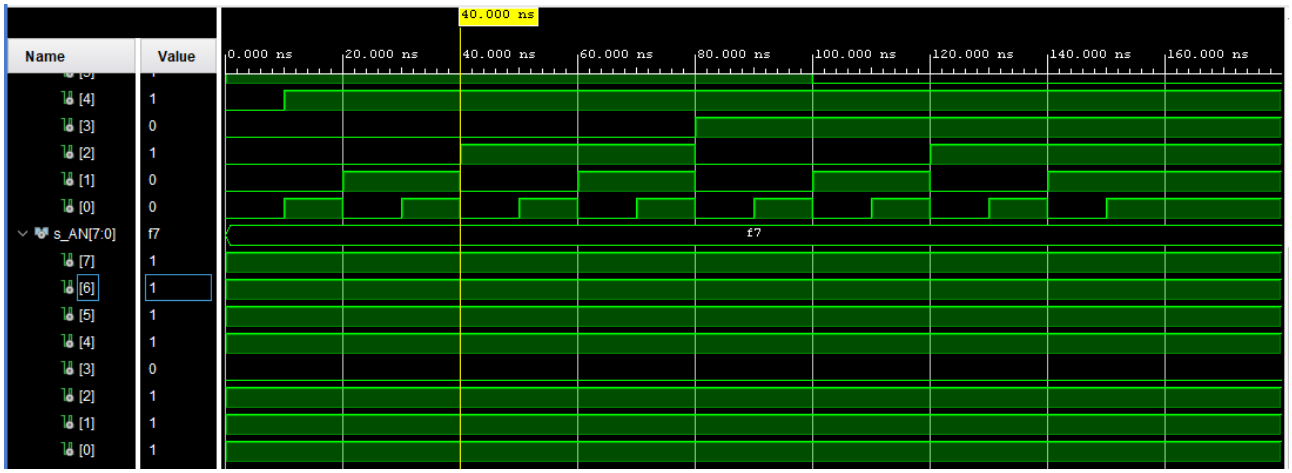
Hex	Inputs	LED4	LED5	LED6	LED7
0	0000	1	0	0	0
1	0001	0	0	1	1
2	0010	0	0	0	1
3	0011	0	0	1	0
4	0100	0	0	0	1
5	0101	0	0	1	0
6	0110	0	0	0	0
7	0111	0	0	1	0
8	1000	0	0	0	1
9	1001	0	0	1	0
A	1010	0	1	0	0
b	1011	0	1	1	0
C	1100	0	1	0	0
d	1101	0	1	1	0
E	1110	0	1	0	0
F	1111	0	1	1	0

```
-- Connect one common anode to 3.3V
   AN <= b"1111_0111";

   -- Display input value on LEDs
   LED(3 downto 0) <= SW;

   -- Turn LED(4) on if input value is equal to 0, ie "0000"
          LED(4) <=   '0' when (SW = "0000") else
```

```
                    '1' ;

    -- Turn LED(5) on if input value is greater than "1001"
           LED(5) <=   '0' when (SW > "1001") else
                       '1';

    -- Turn LED(6) on if input value is odd, ie 1, 3, 5, ...
           LED(6) <=   '0' when (SW = "0001") else
                       '0' when (SW = "0011") else
                       '0' when (SW = "0101") else
                       '0' when (SW = "0111") else
                       '0' when (SW = "1001") else
                       '0' when (SW = "1011") else
                       '0' when (SW = "1101") else
                       '0' when (SW = "1111") else
                       '1';


    -- Turn LED(7) on if input value is a power of two, ie 1, 2, 4, or 8
           LED(7) <=   '0' when (SW = "0001") else
                       '0' when (SW = "0010") else
                       '0' when (SW = "0100") else
                       '0' when (SW = "1000")else
                       '1' ;
```

b) Screenshoty se simulovanými časovými průběhy




40.000 ns
Name
Value
0.000 ns
20.000 ns
40.000 ns
60.000 ns
80.000 ns
100.000 ns
120.000 ns
140.000 ns
160.000 ns
[4] 1
[3] 0
[2] 1
[1] 0
[0] 0
s_AN[7:0] f7
f7
[7] 1
[6] 1
[5] 1
[4] 1
[3] 0
[2] 1
[1] 1
[0] 1