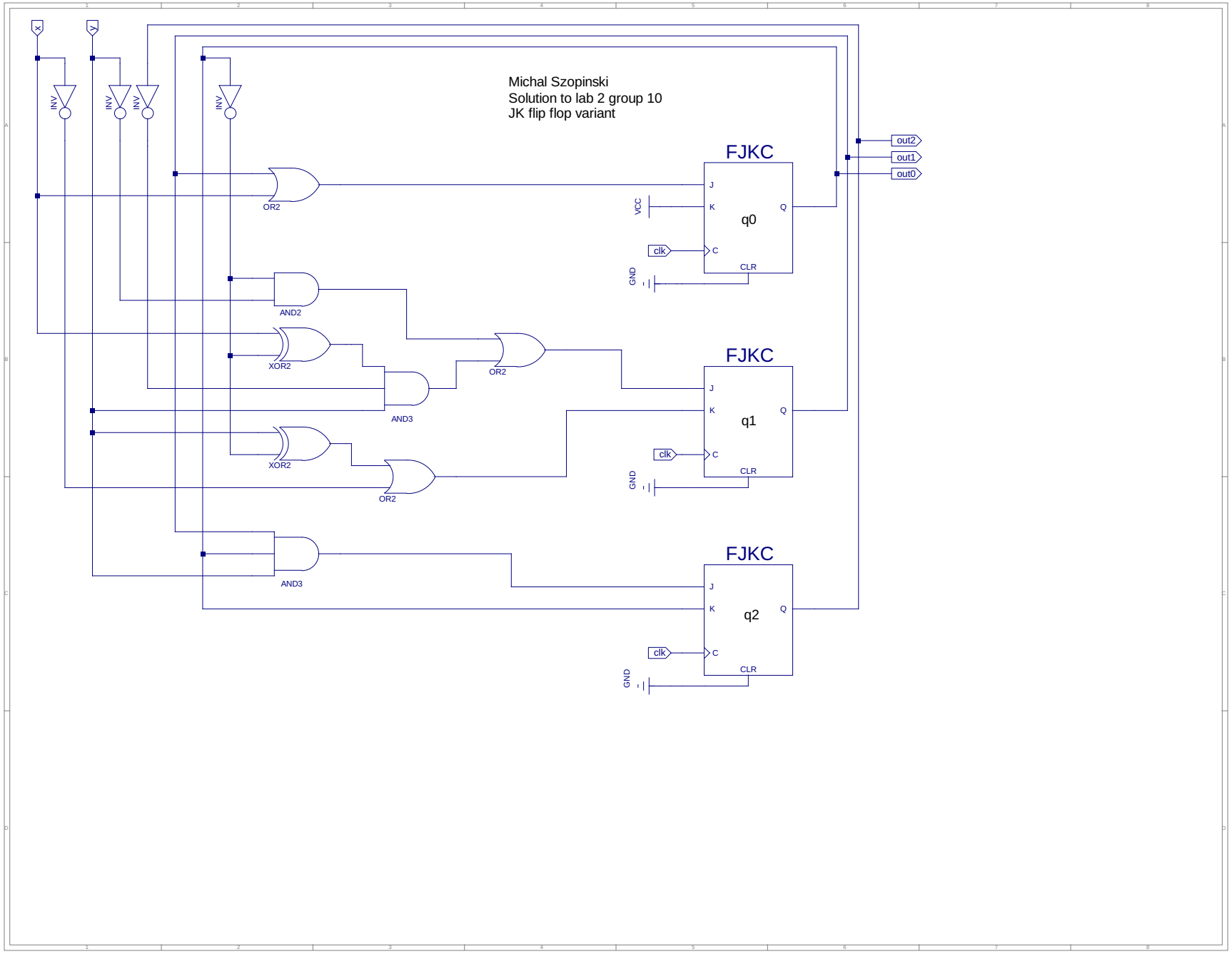
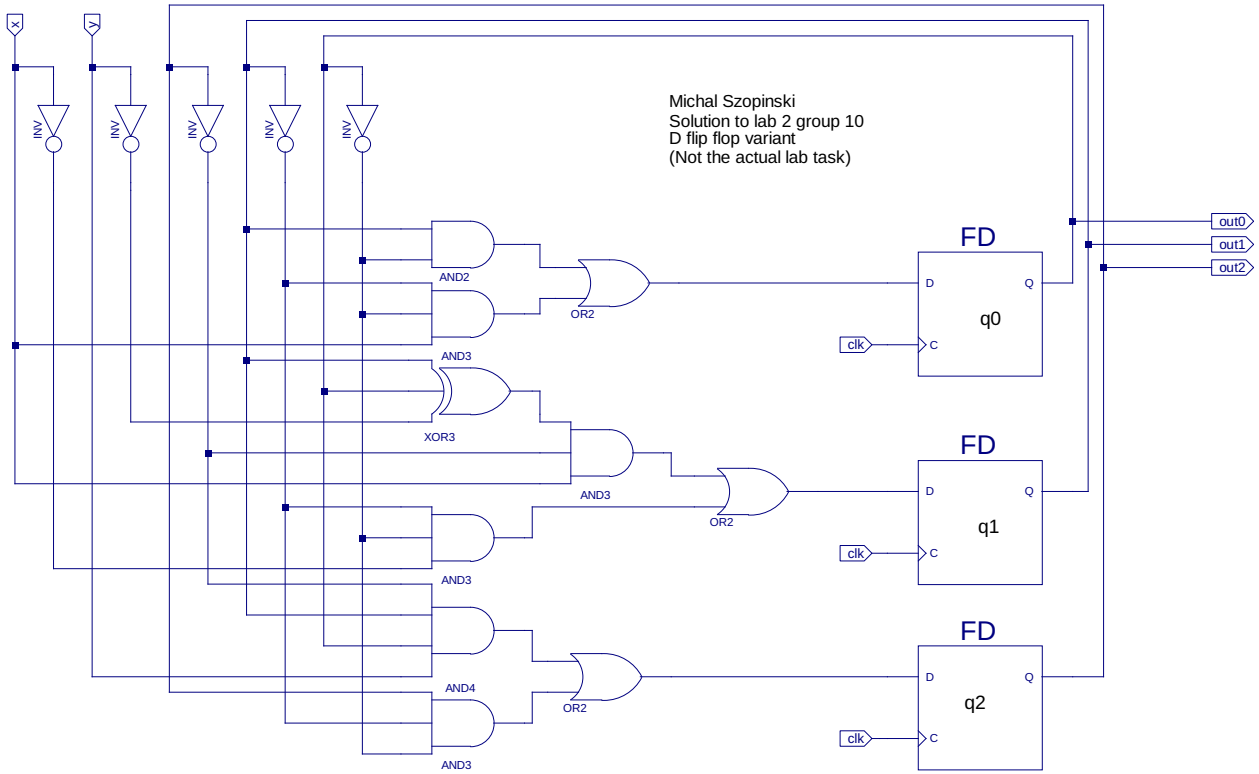

Michal Szopinski
Solution to lab 2 group 10
JK flip flop variant
x
y
INV
INV
INV
INV
OR2
AND2
XOR2
OR2
AND3
XOR2
OR2
AND3
FJKC
J
K
Q
q0
VCC
clk
C
CLR
GND
FJKC
J
K
Q
q1
clk
C
CLR
GND
FJKC
J
K
Q
q2
clk
C
CLR
GND
out2
out1
out0


x
y
INV
INV
INV
INV
INV
Michal Szopinski
Solution to lab 2 group 10
D flip flop variant
(Not the actual lab task)
AND2
OR2
AND3
XOR3
AND3
OR2
AND3
AND4
OR2
AND3
FD
D
Q
q0
clk
C
FD
D
Q
q1
clk
C
FD
D
Q
q2
clk
C
out0
out1
out2

```vhdl
-- Vhdl test bench created from schematic /home/mszopinski/Desktop/labko2d/schem.sch - Sat Nov 30 18:43:44 2019
--
-- Notes:
-- 1) This testbench template has been automatically generated using types
-- std_logic and std_logic_vector for the ports of the unit under test.
-- Xilinx recommends that these types always be used for the top-level
-- I/O of a design in order to guarantee that the testbench will bind
-- correctly to the timing (post-route) simulation model.
-- 2) To use this template as your testbench, change the filename to any
-- name of your choice with the extension .vhd, and use the "Source->Add"
-- menu in Project Navigator to import the testbench. Then
-- edit the user defined section below, adding code to generate the
-- stimulus for your design.
--
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;
LIBRARY UNISIM;
USE UNISIM.Vcomponents.ALL;
ENTITY schem_schem_sch_tb IS
END schem_schem_sch_tb;
ARCHITECTURE behavioral OF schem_schem_sch_tb IS

   COMPONENT schem
   PORT( x  :   IN  STD_LOGIC;
         y :   IN  STD_LOGIC;
         out2  :   OUT STD_LOGIC;
         out1  :   OUT STD_LOGIC;
         out0  :   OUT STD_LOGIC;
         clk   :   IN  STD_LOGIC);
   END COMPONENT;

   SIGNAL x :   STD_LOGIC;
   SIGNAL y :   STD_LOGIC;
   SIGNAL out2  :   STD_LOGIC;
   SIGNAL out1  :   STD_LOGIC;
   SIGNAL out0  :   STD_LOGIC;
   SIGNAL clk   :   STD_LOGIC;

BEGIN

   UUT: schem PORT MAP(
        x => x,
        y => y,
        out2 => out2,
        out1 => out1,
        out0 => out0,
        clk => clk
   );

-- *** Test Bench - User Defined Section ***
   tb : PROCESS

    procedure flashClock is
    begin
        for i in 1 to 30 loop
            clk <= not clk;
            wait for 1 ns;
        end loop;

        wait for 10 ns;
    end procedure;

   BEGIN
        -- set initial input
        clk <= '1';
        x <= '0';
        y <= '0';

        -- mod 3 down
        x <= '0';
        y <= '1';
        flashClock;

        -- mod 4 down
        x <= '1';
        y <= '0';
        flashClock;

        -- mod 6 up
        x <= '1';
        y <= '1';
        flashClock;

      WAIT; -- will wait forever
   END PROCESS;
-- *** End Test Bench - User Defined Section ***

END;

```