BILKENT UNIVERSITY

*Department of Computer Engineering*

*Bilkent 06800 Ankara Turkey*



**CS223 – Digital Design**

Melih Obut

21502659

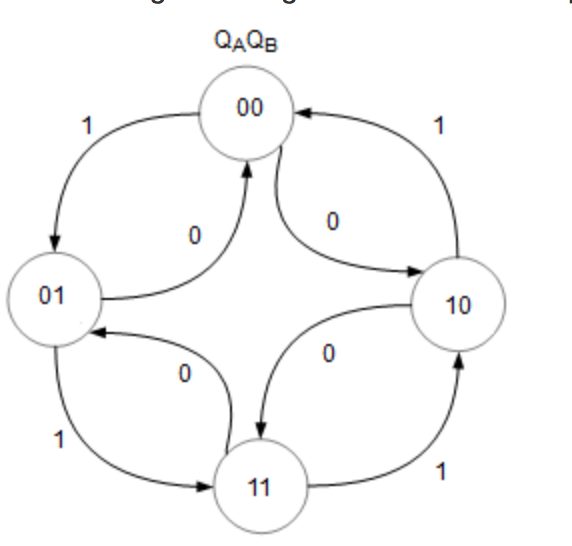
Section: 01

Lab: 04

17.11.2019

**2-The FSM for the driver which will drive the step motor**

State transition diagram

****

**3-The SystemVerilog code for the driver. Testbench for your SystemVerilog code.**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

 entity stepper is

port (

 clk : in std\_logic;

phase : out std\_logic\_vector (3 downto 0);

direction : in std\_logic;

rst : in std\_logic

);

end stepper;

 architecture st of stepper is

signal step\_count :std\_logic\_vector ( 2 downto 0);

begin

process (rst,clk)

begin if rst = '0' then

step\_count <= "000";

else if clk'event and clk = '1' then  --event on positive clock

if direction = '1' then  ---forward direction

step\_count <= step\_count +1 ;

 elsif direction = '0' then  ---reverse direction

step\_count <= step\_count -1 ;

end if;

end if;

end if;

-- end if;

end process;

 process (step\_count)

 begin

case ( step\_count) is

  when "000" => phase <= "0001";

when "001" => phase <= "0011";

when "010" => phase <= "0010";

when "011" => phase <= "0110";

when "100" => phase <= "0100";

when "101" => phase <= "1100";

when "110" => phase <= "1000";

when "111" => phase <= "1001";

when others => phase <= "0000";

end case;

end process;

end;