ESKİŞEHİR TECHNICAL UNIVERSITY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

EEM 334 – Digital Systems II

LAB 6 – SEQUENTIAL DESIGN II

SHIFT REGISTER

1. PURPOSE

In this laboratory, you will design a bidirectional shift register as shown Figure 1.

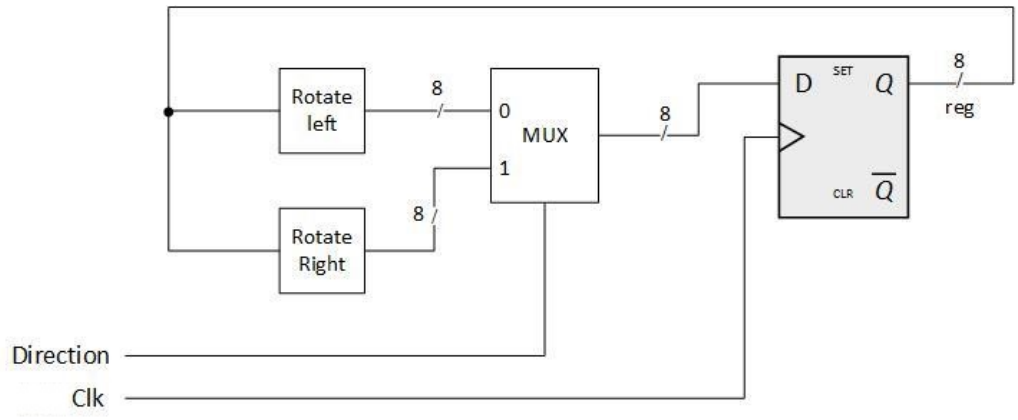

Figure 1: Shift register block diagram

2. BACKGROUND

Shift registers produce a discrete delay of a digital signal or waveform. A waveform synchronized to a clock, a repeating square wave, is delayed by ***n*** discrete clock times, where ***n*** is the number of shift register stages. Thus, a four stage shift register delays "data in" by four clocks to "data out". The stages in a shift register are delay stages, typically D-type flip-flops.

Basic shift registers are classified by structure according to the following types:

- Serial-in/serial-out
- Parallel-in/serial-out
- Serial-in/parallel-out
- Universal parallel-in/parallel-out
- Ring counter

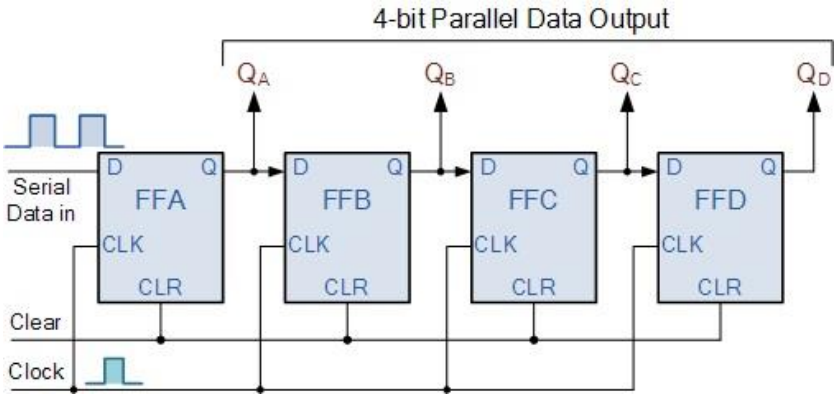

Figure 3: 4-bit serial-in to parallel-out shift register

3. PROCEDURE

You have to design a 8-bit shift register given in Figure 1 with the following properties:

- Four inputs, which are *clock, reset, enable* and *direction*.
- Input *direction* determines the direction of the rotation. It can be clockwise or counterclockwise.
- The initial value of shift register is "00000001" upon *reset*.
- When *enable* is set, rotation will begin. If it is disabled, register will remain at its current value.

Hint: In your shift register, you are free to use *rol* and *ror* operators from *IEEE.numeric_std* library. If you do not want to use these operators, you may use concatenation operator as well.