LAB no. 12.

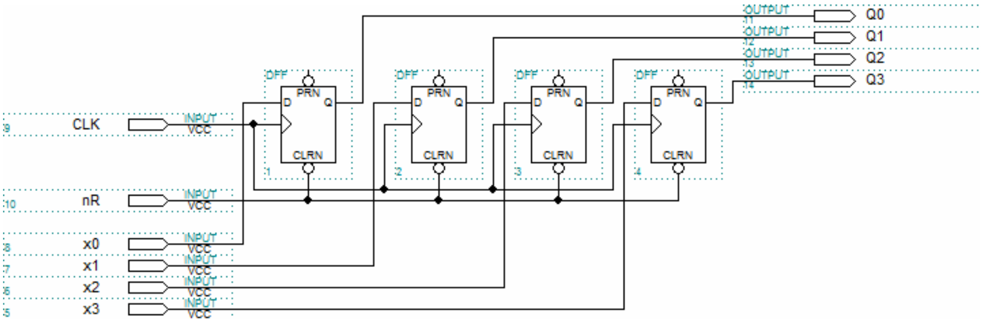
REGISTERS

Registers are sequential logic circuits that store and/or shift binary sequences. Registers can be classified in:

- memory registers (with parallel load) - latch
- shift registers (with serial load)
- combined registers (with parallel and serial load)
- universal registers.

1. Memory registers – latches (MR)

Latches are used for temporary storage of binary numbers in digital systems. They are realized with D flip-flops driven by a common clock signal. Data is stored simultaneously in all cells, on the clock active edge or level. In the following figure a schematic of such a register is presented:



The binary number $N_b=x_3x_2x_1x_0$, existing at the moment t_n at the D_k register's inputs, is stored in its cells and at the moment t_{n+1} the same number appear at its outputs. The process can be described like this:

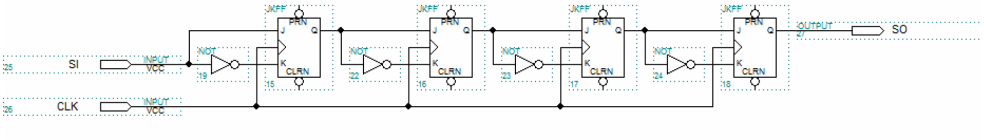
t_n: $D_k=x_k$ ⇒ t_{n+1}: $Q_k=D_k=x_k$, where $x_k=0$ or 1, and $k=0,1,...N-1$

Thus, the simultaneous load of the n bits has been realized (parallel load) MR is also called parallel load register or latch. It can be realized with any number of cells, but usually they have 3, 8, 16 cells.

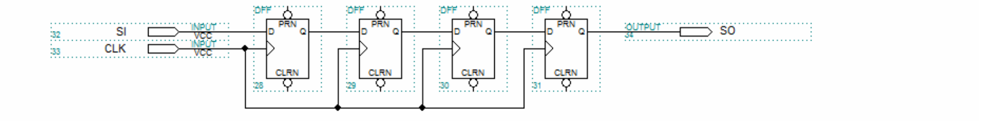
2. Shift registers (SR)

SR's are SLC that shift their content to the left or to the right on every clock pulse. They store a cell's content in the next cell or in the previous one. The first cell will store the value existing at the serial input and the content of the last cell will be lost. It can be realized with any type of flip-flops connected in cascade. The schematics of such circuit are presented below:

SR with JKff



SR with Dff



The output of the k FF is connected at the data input of the $k+1$ FF ($Q_k=D_k$ or $Q_k = S_k(J_k)$, $\overline{Q}_k = R_k(K_k)$, all cells having the same clock signal. The only data input is SI (Serial Input) and the only output is “SO” (Serial Output). Considering that at moment t_n the cells outputs are

$$Q_0(n)=x_0,\ Q_1(n)= x_1,\ ...Q_{N-2}(n)= x_{N-2},\ Q_{n-1}(n)= x_{N-1},$$

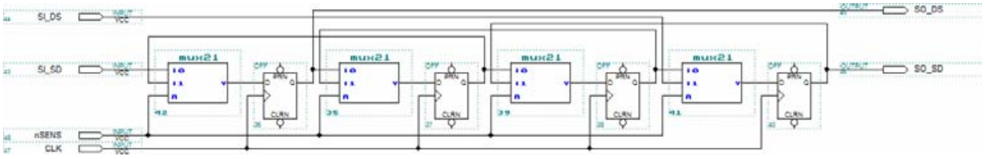
and $SI(n)= x_{IN}$

is the input at the same moment t_n, then at the moment t_{n+1}, after the active edge of the clock signal, the register’s outputs will be:

$$Q_0(n+1)=SI(n)=x_{IN},\ Q_1(n+1)=Q_0(n)= x_0,\ ...Q_{N-2}(n+1)=Q_{N-3}(n)= x_{N-3},$$
$$Q_{N-1}(n+1)=Q_{N-2}(n)= x_{N-2}$$

On every pulse clock the SR move its content from one cell to the next one (from LSB to MSB). By connecting the input of the k^{th} cell to the input of the $k+1^{th}$ cell shift to left register can be realized. The displacement sense counts when both directions are used, otherwise the order of the outputs has to be changed.

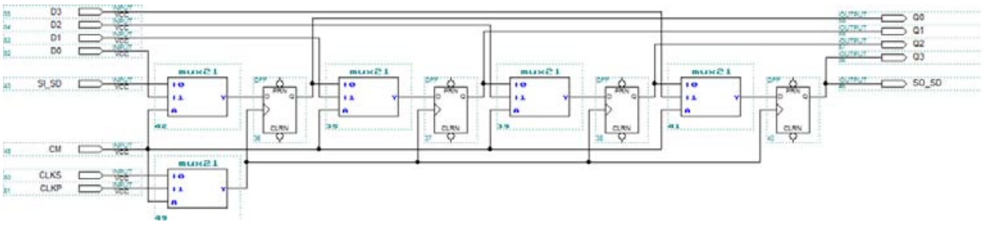
In practice we can find integrated registers having both shifting directions: bidirectional or reversible SR. Such register is shown in the following figure. The shift direction is established by the logic state at the input *nSENS*:



As we can see, 2:1 multiplexers are inserted between the flip-flops. They realize the connections $Q_k = D_{k+1}$ for the left-right sense or $D_k = Q_{k+1}$ for the right-left sense. For *nSENS*= logic 0, the MUX_k connects $Y=I_0$, realizing the connection $D_{k+1}=Q_k$, and the register will be left-right shifter. For *nSENS* = logic 1, MUX_k will connect $Y=I_1$, realizing the connection $D_k=Q_{k+1}$, and the register will be right-left shifter.

3. Combined register (CR)

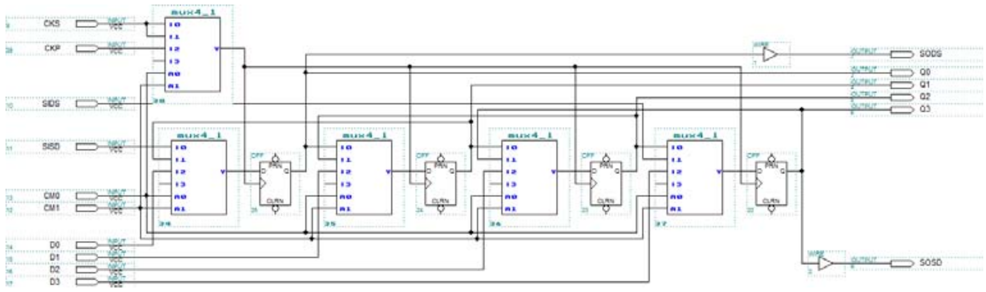
In many applications it is useful to have both parallel and serial inputs and outputs: serial to parallel converter and parallel to serial. The following figure shows such a register:



The input *CM* establishes the working mode: CM=0 makes it a shift register and CM=1 makes a memory register. The shift register works on CLK_S and the memory register works on CLK_P. It has separate clock inputs because there are applications where different clock signals are needed.

4. Universal register(UR)

The universal register has all the functions presented above: left-right or right-left shift, parallel inputs and outputs. In order to realize all these functions 4:1 multiplexers like in the figure below:



The following table descibes how it works:

$\overline{CM_0}$	$\overline{CM_1}$	
0	0	SR left-right
1	0	SR right-left
0	1	MR

1	1	Not used

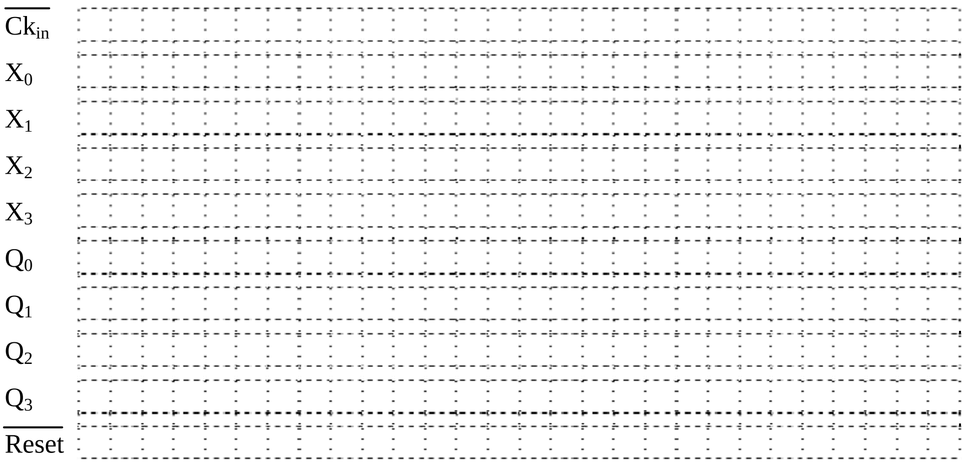
5. Lab works

Complete the following Lab sheet following the indications on it.

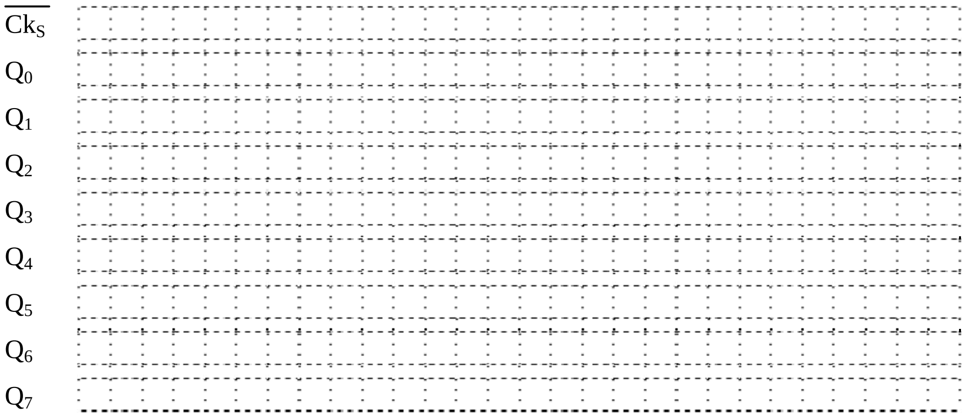
LAB no. 12.

LAB SHEET

1. Input in MaxPlusII a 4 bits memory register and simulate the circuit by applying at the input 1111, 0000, 1001, 0011, 1010 and 0101. Change the values at the input for every clock. Draw the resulted waveforms on the grid below and write down the delay times and the logic values on the waveforms. Compare the values at the input with the ones at the output.

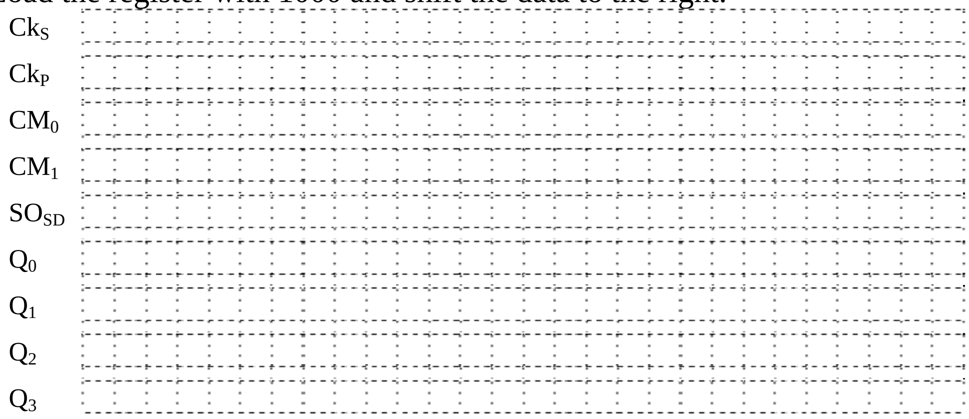


2. Input in MaxPlusII the 8 bit shift register with D flip-flops. Simulate it applying at the input the sequence 10011001 (one bit per clock pulse). Draw the resulted waveforms on the grid below. Write down the delay times and the logic values on the waveforms. Compare the outputs and observe what happens.

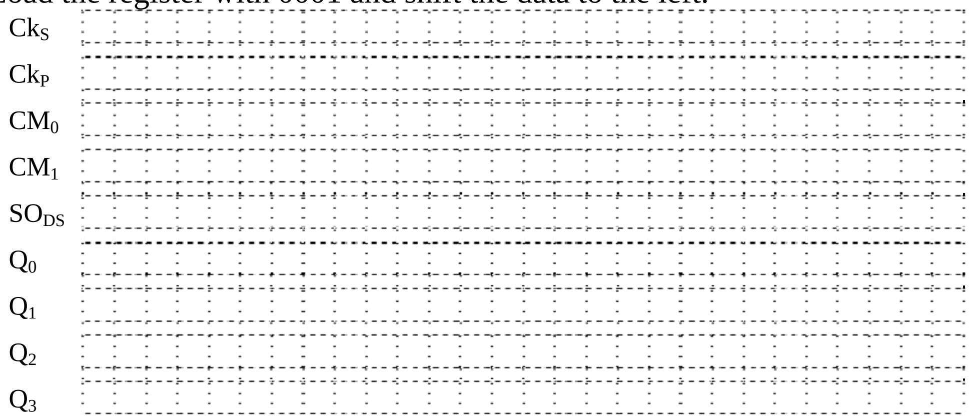


3. Input in MaxplusII the universal register. Simulate the circuit and draw the resulted waveforms on the grids below. Write down the delay times and the logic values on the waveforms.

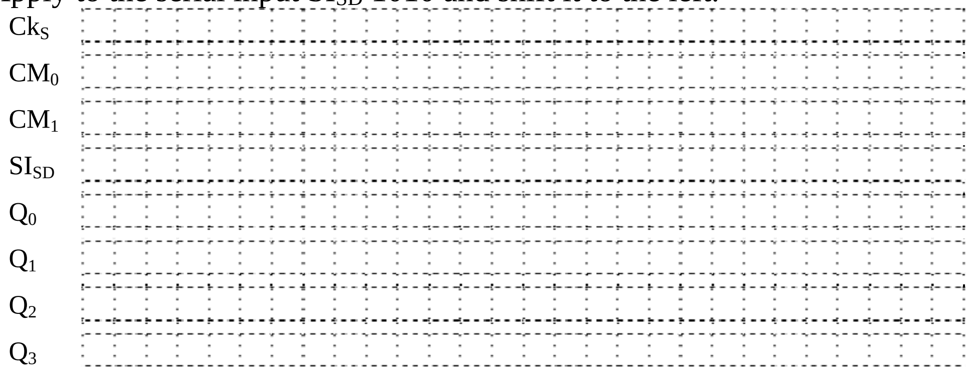
Load the register with 1000 and shift the data to the right.



Load the register with 0001 and shift the data to the left.

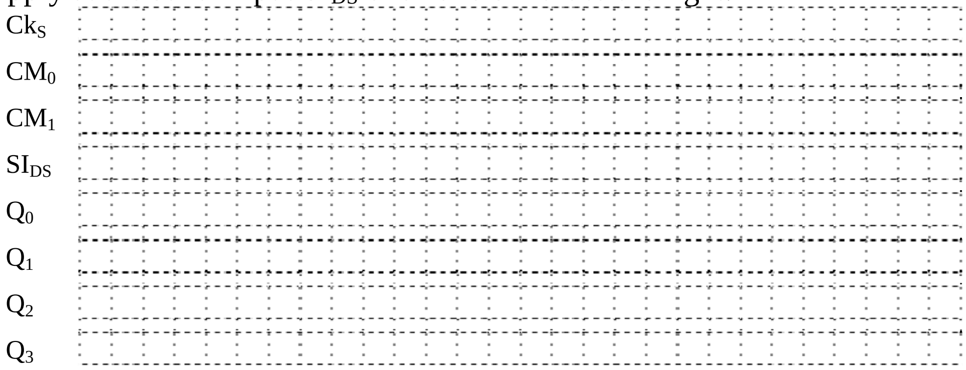


Apply to the serial input SI_{SD} 1010 and shift it to the left.



LAB no. 12.

Apply at the serial input SI_{DS} 0101 and shift it to the right.


Ck_S
CM_0
CM_1
SI_{DS}
Q_0
Q_1
Q_2
Q_3