

PRE-LAB

1. Draw a block diagram for a 4-bit register that has the following characteristics:
 - a. Using 4 DFFs (symbol for 1 DFF shown),
 - b. Rising/leading edge triggered,
 - c. Active low *reset* signal
 - d. No *preset* signal.

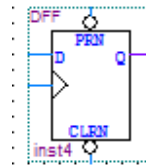


Fig.1. Symbol for 1 DFF

2. Note that the 4-bit register in #1 will transfer the input D to the output Q every clock cycle (when $reset = 1$). We want to design a new 4-bit register that can be controlled to transfer D to Q at specific clock cycles. This is done by using a *load* signal (logic shown below). Based on the 4-bit register from #1, design and draw a block diagram for the improved 4 bit register with an active high *load* signal.

Logic:

If $reset = 0$ then

$$Q^*[3..0] = 0000_2$$

If $reset = 1$ then

At active clock edge

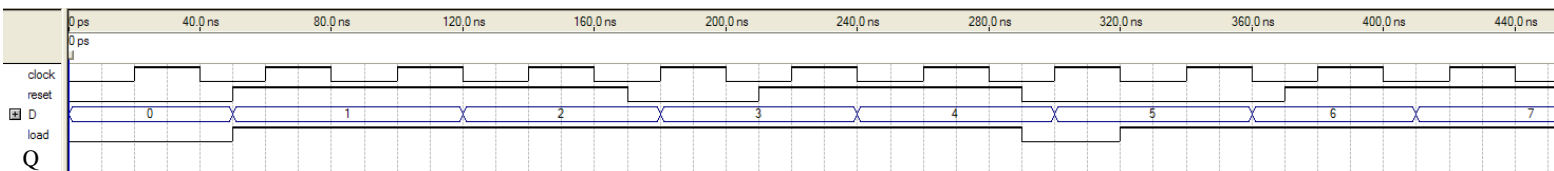
If $load = 1$ then

$$Q^*[3..0] = D[3..0]$$

If $load = 0$ then

$$Q^*[3..0] = Q[3..0] \text{ (No change)}$$

3. Complete the timing diagram for a 4-bit register (ignore propagation delay)



4. Design and draw a 4-bit up counter ($Count = 0_{16}$ to F_{16} and repeat) with an active high *enable* input signal and an active low *reset* input signal (logic shown below). Assume that the following components are available:
 - Any logic gates or building blocks we have covered.
 - 4-bit adders from lab 3.
 - 4-bit registers (from #1 or #2 above).

Logic:

If $reset = 0$ then

$$Count^* = 0$$

If $reset = 1$ then

At active clock edge

If $enable = 1$ then

$$Count^* = Count + 1$$

If $enable = 0$ then

$$Count^* = Count \text{ (No change)}$$

