ECE 4623/5623
Review
Decoders, Encoders, Latches, Flip-Flops

1. Decoders/Encoders
   (a) Do example of 2-to-4 decoder.
   (b) What is an encoder? Do example with BCD to 7-segment encoder.
   (c) Explain BCD.
   (d) Explain 7-segment display. Use Basys3 datasheet.
   (e) Explain binary to BCD double-dabble algorithm.
      i. Start with n-bit binary number.
      ii. The size of the shift register required to store the original binary value and the BCD result is $n + 4\times\text{ceilling}(\frac{n}{3})$.
      iii. Begin by constructing the shift register with the binary value aligned to the far right side.
      iv. Shift left n times.
      v. For each shift examine each 4-bit BCD result register that represents a 10's place. If the value is greater than 4, then add 3.
      vi. Once this has been done n times the binary to BCD conversion is complete.
      vii. Why do we add 3? What would happen if we didn't add 3?
   (f) Most combinational circuits will be realized in the FPGA using look-up-tables rather than actual logic gates.
   (g) LUTs are commonly used by your brain to do things like multiplication.

2. SR Latch (Fig. 1)
   (a) NOR gate

A	B	OR(A,B)	NOR(A,B)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

   (b) SR latch example with 2 NOR gates.
   (c) In the hold state where S=R=0, $Q_{next} = Q_{curr}$.
   (d) In the reset state $Q_{next} = 0$.
   (e) In the set state $Q_{next} = 1$.
   (f) In both the set and reset states Q_{curr} does not influence the value of Q_{next}.

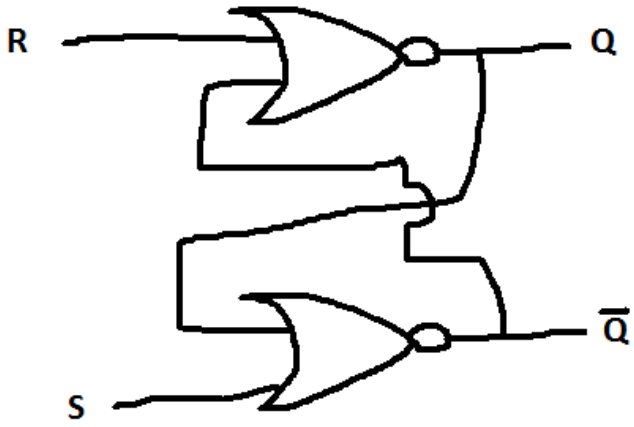

Figure 1: SR latch with NOR gates.

Table 1: SR Latch truth table.

S	R	Q_{curr}	$\bar{Q}_{next}$ = NOR(S,Q_{curr})	Q_{next} = NOR (R,$\bar{Q}_{next}$)	Description
0	0	0	1	0	Hold
0	0	1	0	1	Hold
0	1	0	1	0	Reset
0	1	1	0	0	Reset
1	0	0	0	1	Set
1	0	1	0	1	Set
1	1				Forbidden
1	1				Forbidden

(g) The R=S=1 input combination is forbidden because Q and $\bar{Q}$ would be forced to the same logic value.

3. Gated SR Latch (Fig. 2)

(a) If we AND the S and R inputs with an EN (enable) signal, then the latch will only work when EN=1. This accomplished by adding 2 AND gates to the circuit. The input to the first AND gate is R and EN. The input to the second AND gate is S and EN.

(b) It works the same as the SR latch, but only when EN=1.

(c) If EN=0, then it is always in the hold state.

(d) If the EN signal is a CLK, then the latch only works when the CLK=1.

(e) We could go one step further and place a pulse detector on the EN signal. In this case, the latch only works during the brief time that a rising or falling edge is detected. Usually, edge triggered latches are called flip-flops.

4. D latch/flip-flop (Fig. 3)

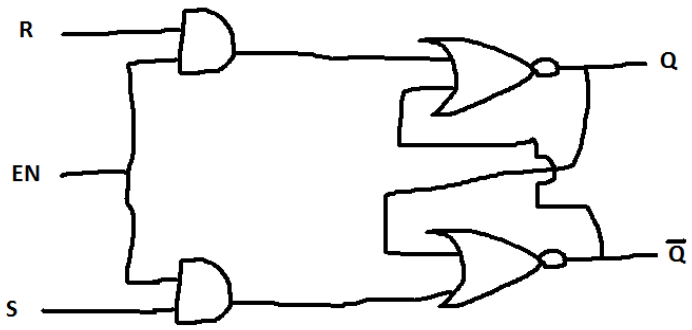

Figure 2: Gated SR latch.

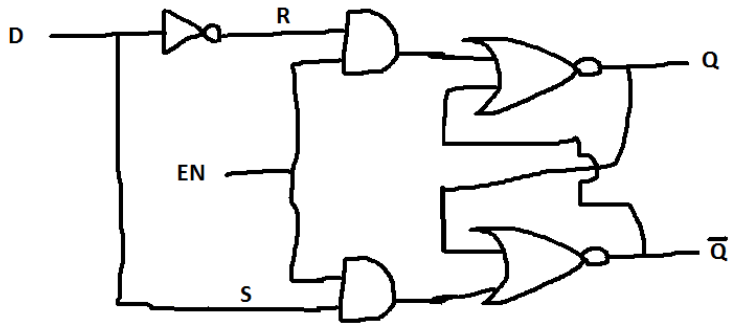

Figure 3: Gated D latch.

(a) The D-latch is obtained by combining the R and S inputs of the gated RS latch and adding an inverter on the R input.

(b) With this approach, the forbidden R=S=1 state is no longer a possibility.

(c) The hold state is only possible when EN = 0. Othewise we are in the set or reset state because S and R are forced to be either (0,1) or (1,0).

(d) The D-latch can also be either edge or level triggered depending on the source of the EN input.

(e) The Basys3 uses the D latch/flip-flop.

(f) Each CLB in the Artix7 has 8 D-latches. All 8 can be edge triggered, but only 4 can be level triggered. Look at diagram.

5. Latches and Flip-Flops Continued

(a) There are several other types of latches and flip-flops. You can read about them and use what you learn to complete HW#1.

(b) In general, a latch/flip-flop is used to store one bit of information.

(c) In general, latches are level triggered and flip-flops are edge triggered. This distinction is not always clear.