Latches and Flip-Flops

You will learn how latches and flip-flops work. You will also learn how to use a few more of the simulator's features.

1.0 Overview

Latches and flip-flops are the primitive storage devices in sequential circuits. In this experiment you will study their functional and temporal behavior and develop some insights about sequential circuit operation in general.

2.0 Preparation

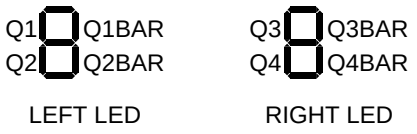
You must be thoroughly familiar with the material in Sec. 7.2 of Wakerly.

3.0 Design Specification

Use the schematic editor to create the circuit shown in Figure 6. It consists of a parallel connection of four storage elements: an SR latch, a clocked SR latch, a D latch and a D master-slave flip-flop. The schematics for these "macros" are shown in Fig. 2. The S, R, D, and C inputs of these devices should be connected, respectively, to the D0, D1, D2, and D3 lines of the PC parallel port. The true and complemented outputs Qi, QiBAR (i = 1, 2, 3, 4) should be connected to the indicated segments of the left and right LED displays; this makes it easy to quickly determine the state of each device.

What you need to do is to analyze each of these devices using the simulator. First, you will determine the device's functional behavior by performing unit-delay simulation. Next, you will implement the circuit, extract its actual delays, and perform a timing simulation to derive some of the devices' temporal parameters. In the lab, you will confirm

FIGURE 6. Top-level schematic of latch/flip-flop circuit



the simulation results by stimulating each of the devices from the PC parallel port and observing its output on the LED indicators.

4.0 Notes

- Make sure you add the KEEP attribute to all nets; this will insure that their driving gates are not collapsed (optimized away) by the synthesis tools, and makes it possible to back-annotate the estimated delays necessary for timing simulation.
- Set the simulation precision to 100ps (0.1ns). This setting controls the minimum resolvable time in the simulator; it also controls the number of significant digits displayed in the device delay table (accessed through the simulator's **Device->Edit Timing Specification**).
- Note that the simulator maps your gates to gates in the **simprims** library. These gates have types whose labels start with an "x_" prefix; the extracted delays are attached to

FIGURE 7. Latch and flip-flop macros.

these gates, not to the gates on the schematic. Note, also, that the simulator inserts 0-delay inverters to replace inversion bubbles.

- Name your nets and gates as shown in Figure 6 and Figure 7. This will facilitate analysis and debugging (especially figuring out what delays are for which gate.)
- Keep the **Transport Delay box** in the simulator's **Preferences** checked. If unchecked, the simulator uses an inertial delay model that filters narrow pulses; this makes it harder to trace through the simulator's waveforms to determine event causality.
- You will find it convenient in this lab to use the keyboard to apply stimulators. Just associate the c, d, r, and s keys on the keyboard with the Clock, D, RESET, and SET signals of the top-level schematic. Every time you press one of these keys, the value applied to the corresponding signal toggles. This allows you to control the spacing of events at the simulator's precision.
- Save your simulation state for each of the simulation runs in a file (these files have a .des extension). This makes it easy to reproduce an earlier simulation quickly.

5.0 Deliverables

5.1 Pre-Lab

1. Hardcopy of schematics and UCF file.
2. Hardcopy of unit-delay simulation results for the three latches and the flip-flop. For the SR latch, produce a simulation trace similar to that in Figure 7-6 on page 535 of Wakerly. For the clocked SR latch, produce a simulation trace similar to that in Fig-

ure 7-11 on page 538 of Wakerly. For the D latch, produce a simulation trace similar to that in Figure 7-14 on page 539 of Wakerly.Finally, for the D master-slave flip-flop, produce a simulation trace similar to that in Figure 7-17 on page 541 of Wakerly. In all cases, make sure you induce the conditions that lead to oscillation.

3. Hardcopy of the timing simulation results for the same scenarios in 2. Annotate the simulation traces with "measurements" that show the separations of important events, including the period of oscillation in cases when the device becomes unstable.

5.2 In-Lab

Demonstrate the operation of the three latches and the flip-flop to your lab GSI.

5.3 Post-Lab

Prepare your lab report as described in the *EECS270 Laboratory Overview* handout. Make sure you complete and include all parts of the report including the *Cover Sheet*, the *Design Narrative* section, and the *Design Documentation* section. Include as part of your design documentation all corrected pre-lab requirements. In your design narrative, point out any anomalies you encountered, and try to explain them as best you can. In the *Post-Lab Questions* section answer the following questions (assume unit delays in what follows):

1. What is the minimum required pulse widths for setting and resetting the SR latch? Provide a simulation trace to support your answer; the trace should show correct latching at the required minimum pulse widths, and oscillation at a a slightly narrower pulse width. What is the period of oscillation?
2. Assuming that the D input rises sufficiently early before the Clock signal rises, what is the minimum required clock pulse width to insure that the D latch is set? Provide a simulation trace to support your answer; the trace should show correct latching at the required minimum pulse widths, and oscillation at a slightly narrower pulse width. What is the period of oscillation?
3. What is the setup time of the D flip-flop? Provide a simulation trace to support your answer.