# COMPUTER SYSTEMS I — LAB 3A
## Memory and a 1-bit counter

## 1 A starting bit of memory

To get started on this lab, use our existing chips and their gates to create for yourself an *S-R latch*. You can follow our description from class by using actual push-buttons for the $S$ and $R$ inputs.

Specifically, on the *ETS-7000*, you will notice that, in the lower-left corner of the device, there are two push-buttons, $P_A$ and $P_B$. Near them, there are four outputs on the board: $A$, $\bar{A}$, $B$, and $\bar{B}$. When button $P_A$ is **not** being depressed, then $A = 0$ and $\bar{A} = 1$; when $P_A$ **is** depressed, then $A = 1$ and $\bar{A} = 0$. $P_B$, $B$, and $\bar{B}$ behave analogously. Thus, these two buttons can serve as $S$ and $R$ for your latch.

## 2 Enhancing that bit of memory

You will then recall that we observed a limitation with our *S-R latch*. We wanted there to be an input, $D$, whose value, 0 or 1, would be adopted into the memory element at some moment. Specifically, we wanted $Q$, the output of the memory element, to take on whatever value of $D$ had when another input, $C$ last has the value 1. That is, when $C = 1$, $Q = D$; when $C = 0$, $Q$ does not change—whatever **that** means!

Figure out how to make your *S-R latch* take on the value of $D$ when $C = 1$, but not when $C = 0$. To perform this task, you will likely need to stop using your push-buttons as $S$ and $R$, but you will have to use one of them as $C$. Once you've built it, you now have something called a *D latch*.

## 3 A one-bit counter

Now that you have a memory element that can take on a value only when you press a particular button $C$, let's build a *1-bit counter*. That is, let's build a circuit that counts from 0 to 1 and then "rolls over" back to 0 again. It's like an odometer with a single, binary digit.

To do so, use your *D latch* to store the *current counter value*. That is, its output, $Q$, shows the current value of the counter. The input to the *D latch*—$D$—should be the **next** value of the counter. Thus, if the value of the counter is 0, you should be able to press the button (or "tick the clock") and advance the counter to 1. Press the button again, and the counter "rolls over" to 0.

**A problem:** It's a great idea! But it won't work. Your task is to figure out **why is doesn't work**. When you do, come and talk to me, and we'll figure out what needs to happen to fix it! Only then may you move on to part B of the lab.

**This assignment is due Friday, October 1, at the beginning of your lab section**