10.1 The basic flip-flop: NAND latch

This circuit, the most fundamental of flip-flop or memory circuits, can be built with either NANDs or NORs. We will build the NAND form shown in Figure 10.1.

![Figure 10.1. A simple flip-flop: cross-coupled NAND latch](image)

Build this latch and record its operation using three channels of the oscilloscope to measure Set (pin 1), Reset (pin 5), and Q (pin 3); take a SCREENSHOT (2 pts) (Hint: set the scope is on “auto/roll” and slow enough such that you "catch" the output. Please push the buttons in this order: Reset, Set, Set, Reset.) Complete the logic table below, indicate which of the four combinations define the memory state, and include the table in your report (1 pt).

<table>
<thead>
<tr>
<th>Operation Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Note which combination of inputs, i.e., "S" and "R", defines a “memory”, or resilient, state.

10.2 Monitoring switch bounce

Bouncing of the contacts in a mechanical switch is hard to see because the bounces do not occur at exactly repeatable times after the switch is pushed (e.g., Figure 10.2).

![Figure 10.2: Switch bouncing at HIGH to LOW transition (pulled up through 10k)](image)

Use the “R-S” flip-flop (Figure 10.1) as a switch "debouncer" (Figure 10.3). Add pull-up resistors and as input use a "bouncy" SPDT pushbutton with the common terminal grounded (Figure 10.1). Show that your circuit performs correctly by measuring from Set, Reset, and the output Q. Trigger the oscilloscope in the "Normal" mode (hint: start with a time-base of 100 µs/cm and use "S" as the source) and take a SCREENSHOT (2 pts).
Why does the latch – a circuit designed to remember an input – work as a "debouncer" (1 pt)?

![Figure 10.3. NAND latch as switch debouncer](image)

**Figure 10.3. NAND latch as switch debouncer**

*Keep this circuit for the remainder of the exercises.*

### 10.3 D flip-flops

The R-S latch is instructive but rarely used in circuit design. A more complicated version, the clocked flip-flop, is much easier to work with.

The simplest of the clocked flip-flops, the D, simply saves at its output (O) what is present at its input (D, for "data") just before the previous clocking edge (Figure 10.4). The particular D flip-flop used below, the 74HC74, responds to a rising edge of the clock input.

![Figure 10.4. D flip-flop](image)

**Figure 10.4. D flip-flop.** The '74 DIP package includes *two* D flip-flops. Tie the inputs of the unused flip-flop high (through a 10KW resistor).

Dis-assert \(\overline{R}\) (active low Reset) and \(\overline{S}\) (active low Set, sometimes called Clear or Preset) by tying them high (as these are active LOW inputs) with 10 kΩ resistors to +5 V (pins 1, 2 & 4) to ease switching from HIGH to LOW. Ditto for input "D".

Clock the flip-flop with the “debounced” push button that you built.

- Confirm that the D flip-flop ignores information presented to its input (D) until the flip-flop is clocked through input CLK (for "clock"). Demonstrate this with a SCREENSHOT; use a slow time-base and show CLK, D, and Q (2 pts).

- Try asserting \(\overline{R}\). Demonstrate this with a SCREENSHOT; show at least \(\overline{R}\) and Q (1 pt). You can do this with a jumper wire; bounce is harmless here. Why (1 pt)? What
happens if you try to clock in a HIGH at D while asserting \( \overline{R} \). Demonstrate this with a SCREENSHOT as well (1 pt).

- Try asserting \( \overline{R} \) and \( \overline{S} \) at the same time, something you would never purposely do in a useful circuit, with jumpers (hint: ground them simultaneously through a single wire). What happens (hint: Look at both the \( Q \) and \( \overline{Q} \) outputs). Demonstrate this with a SCREENSHOT; show at least \( \overline{R} \), \( Q \) and \( \overline{Q} \) (1 pt). What determines what state the flip-flop rests in after you release both (1 pt)?

### 10.4 A toggle – D flip-flop with feedback

The addition of negative feedback to a D flip flop leads to a circuit (Figure 10.5) that would appear to have an oscillatory output. Yet the need for a clock breaks the feedback path. Build the circuit and try it!

![D flip-flop with inhibitory feedback](Figure 10.5)

- Clock the circuit manually using the "debounced" push-button. Document with a SCREENSHOT (1 pt).

- Clock the circuit with a square wave (0 to 5V) from the function generator. Measure CLK and Q on the oscilloscope and document with a SCREENSHOT (1 pt). What is the relation between \( f_{clock} \) and the frequency of the output, \( f_{OUT} \) (1 pt); document with a SCREENSHOT (1 pt).

- Increase the clock rate toward the function generator’s maximum and measure the flip-flop’s propagation delay. You will have to consider what CLK and output Q levels to use when you measure the time elapsed (hint: just what it is that is “propagating.”) Document with a SCREENSHOT (2 pts).

- At the high clock rate, measure the expected difference in the transition onset at \( Q \) versus \( \overline{Q} \). Document with a SCREENSHOT (1 pt) (hint: you may need to use the averaging feature on the oscilloscope). Why might the transitions occur at different instances (1 pt)?

### 10.5 Shift register

We consider a circuit (Figure 10.6) delays the a (possibly periodic) signal called “IN” connected to D of the first flip-flop, and synchronizes it to the faster CLK. Build the circuit (Figure 10.6). Use two function generators to periodic inputs that vary from 0 to +5 V, and choose \( f_{CLK} > 10 f_{IN} \).
Figure 10.6. Shift register

- One flip-flop: Synchronizer
  - Use the oscilloscope to watch CLK, IN, and QA; trigger the scope on IN and supply a SCREENSHOT (1 pt).
  - What accounts for the jitter (i.e., fluctuation in timing) in signal QA (1 pt)?

- Several flip-flops: Delay
  - Now watch the later outputs, i.e., QB, QC, or QD, along with CLK, IN, and QA; trigger the scope on IN and supply SCREENSHOTs for all cases (3 pt).
  - Make a composite timing diagram on the propagation for flip flop A to B to C to D (1 pt).

10.6 "Double-Barreled" one-shot. A shift register (Figure 10.6) plus NAND gates yield a "Double-Barreled" one-shot (Figure 10.7). Before we start, we will need a pair of read-out indicators. Build two of the n-channel MOSFET LED drivers (Figure 10.8). These will be used to display the output from the D flip-flop register in the following exercise.

Figure 10.7. Digitally-timed (synchronous) one-shot (double-barreled)

- Add NAND gates to the shift register circuit of figure 10.6 (Figure 10.7)

  *Hint:* use the debounced pushbutton to drive TRIG, and set the clock rate to one Hertz or less. Watch the two one-shot outputs, along with TRIG, with oscilloscope probes. Watch the buffered LEDs. You should see first one LED then the other wink low, in response to this low-to-high transition.
When you are satisfied that the circuit works, drive TRIG from one function generator, while CLK is driven by the other function generator, and choose $f_{CLK} > 10 f_{TRG}$ for clarity. Show a SCREENSHOT of TRIG, CLK, and outputs 1 and 2 (3 pts).

To ensure you understand what this circuit is doing, draw a timing diagram (to be included with your report), showing TRIG, clock, the four flip-flop outputs, and outputs 1 and 2 from the NAND gates (3 pts). What are the strengths and weaknesses of this circuit as a one-shot (pulse) circuit relative to an analog one-shot (1 pt)?