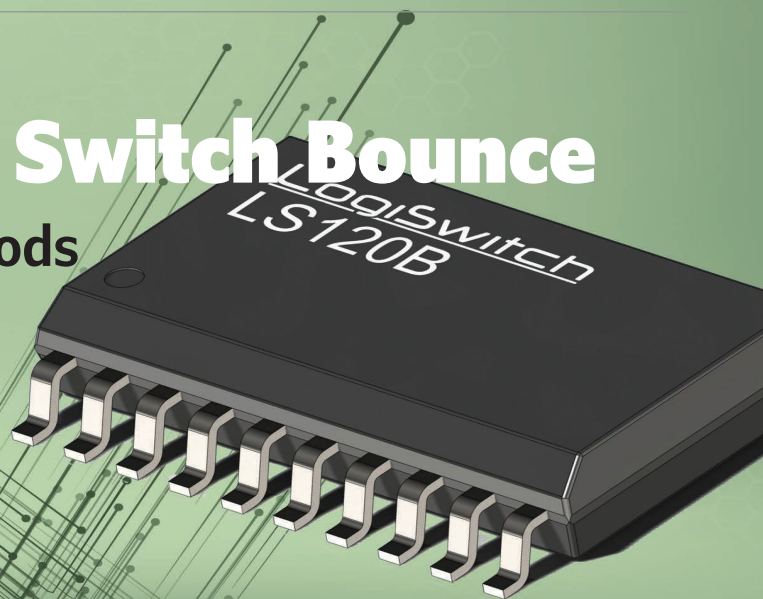


How to Eliminate Switch Bounce

5 Basic Methods

By
Michael H. Pelkey, Founder
and CEO, LogiSwitch



There are five basic methods for debouncing switches using single pole-single throw (SPST), or single pole-double throw (SPDT) switches. In this article, Michael describes the two methods for SPDT switches which require no output delay and the two best methods for interfacing with the simpler SPST switches.

The electric "snap switch" consists of one or two stationary electrodes called "Throws", and a movable electrode, called a "Pole". When the switch is activated, the movable pole is forcefully snapped onto the fixed surface of the stationary throw. Like snapping a rubber ball onto the floor, it naturally bounces. The greater the force of the snap, the greater the bounce.

Unfortunately, the spring in the switch that provides the "snap" action through mechanical hysteresis also adds to the force of the contacts coming together. If the switch is used to turn on and off electric lights or to start a motor at 50Hz or 60Hz, the bounce is not noticeable and causes no problems.

Problem is, such is not the case with high-speed, low current DC applications such as microprocessor and microcontroller-based digital applications. Switch bounce is an inherent characteristic of the electric switch, and almost always causes problems that must be dealt with by the digital circuit designer. To make life easier for electronics design engineers and hobbyists, LogiSwitch manufactures NoBounce switches and ICs incorporating integral debounce circuitry. At LogiSwitch, we believe that it is time for the field of digital electronics to have their own switches that work for them rather than to continue borrowing from the field for which they were originally designed that requires way too much extra effort just to make them usable. LogiSwitch

switches are made for high-speed digital applications.

LEGACY SWITCH HISTORY

Since the advent of the microprocessor and microcontrollers (MCUs) the propensity of the mechanical switch to "bounce" has proved to be a design problem for hardware and/or software designers that nearly always must be dealt with.

The Electric Snap Switch was designed and patented in 1884 by J. H. Holmes to ease the arcing and burning of contacts for Thomas Edison's new invention of five years earlier, the incandescent light bulb. At the time of this writing, it has been with us for the past 137 years, and still does a great job when used for the purpose for which it was originally intended: high-voltage, high-current 50Hz and 60Hz AC applications.

In this article, we'll demonstrate all five methods for debouncing switches, using single pole-single throw (SPST), or single pole-double throw (SPDT) switches. The guide includes the two methods for SPDT switches which require no output delay:

1. *The 2-pin debounce.* This is the simplest method, but it requires two processor pins.
2. *The cross-coupled NAND or S/R debounce.* This method is functionally equivalent to the 2-pin method, but it uses only one processor pin.

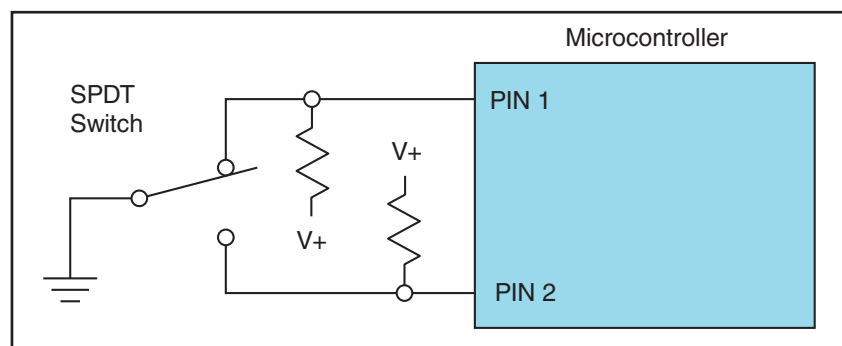


FIGURE 1

Simple 2-Pin debounce. Using a SPDT switch, pin 1 low indicates the switch is deactivated, and pin 2 low indicates it is activated. Switch closure is determined by finding pin 2 low, and switch release is determined by finding pin 1 low.

...and the two best methods for interfacing with the simpler SPST switches. Note that a delayed output is needed for the SPST switch:

1. Hardware SPST debounce.
2. Software SPST debounce.

Note that both the hardware and the software methods for debouncing the SPST switch simply delay the switch output until after the bounce time has ended.

METHOD 1

Simple 2-Pin Debounce: If you can spare two pins, this is the way to go. Using a SPDT switch, pin 1 low indicates the switch is deactivated, and pin 2 low indicates it is activated (**Figure 1**). In other words, switch closure is determined by finding pin 2 low, and switch release is determined by finding pin 1 low.

This rarely used, but highly effective debounce method performs every bit as well as the best cross-coupled NAND circuit when two pins are available for use with a single switch. Note that most MCUs provide programmable internal pull-up resistors for pins parameterized as inputs, in which case the circuit may be even further simplified by omitting the two resistors shown in Figure 1.

METHOD 2

Cross-Coupled NAND Debounce: The cross-coupled NAND gate method is ideal for projects where two processor pins are not available for use with one switch (**Figure 2**). Output Q is high when the switch is idle, and low when it is activated. The /Q output is the opposite: low when idle and high when activated. Note that only Method 1, 2-Pin Debounce, Method 2, Cross-Coupled NAND Debounce or Method 3, D-Type Flip-Flop Debounce provides immediate outputs with no delay on activation and release (see mini sidebar "Switch Bounce: The Waveforms").

For some industrial automation and robotics applications, the immediate attention of signaling from these debounce methods is needed. Note that the LogiSwitch integral debounce switches or LS100 series debouncer ICs incorporate a circuit like the circuitry of Method 2.

METHOD 3

D-Type Flip-Flop Debounce: Using the internal cross-coupled NAND gating of a D-Type flip-flop provides an excellent Method 3 debounce (**Figure 3**). Note the Q output is normally high (NH) and goes low when the switch is activated.

METHOD 4

Hardware Debounce for SPST Switches: Whenever a SPST switch is used for input into high-speed digital devices, the only thing you can do is to wait out the bounce interval with either hardware or software delays (again, see mini sidebar "Switch Bounce: The Waveforms").

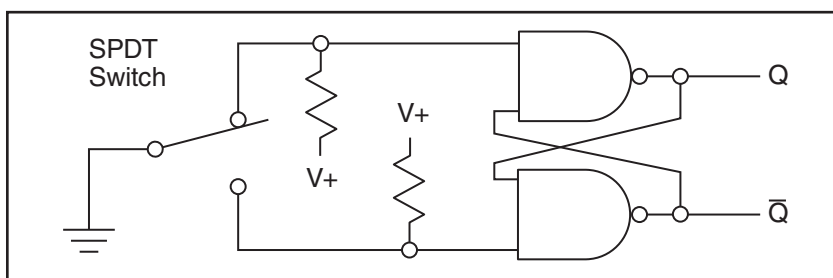


FIGURE 2 Cross-Coupled NAND debounce. Output Q is high when the switch is idle, and low when it is activated. The /Q output is the opposite: low when idle and high when activated.

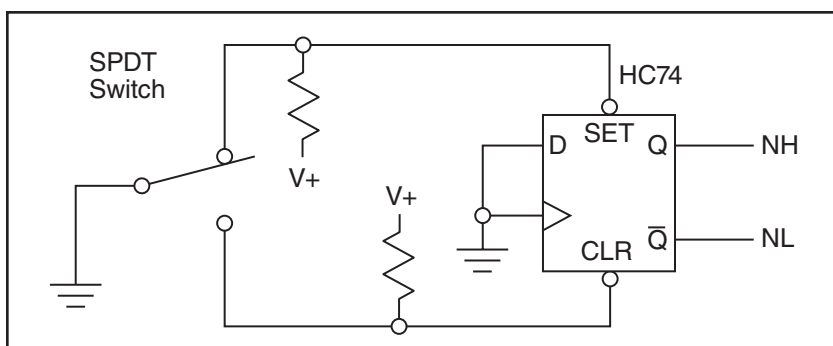


FIGURE 3 D-type Flip-Flop Debounce: Using the internal cross-coupled NAND gating of a D-Type flip-flop provides an excellent Method 3 debounce. Note the Q output is normally high (NH) and goes low when the switch is activated.

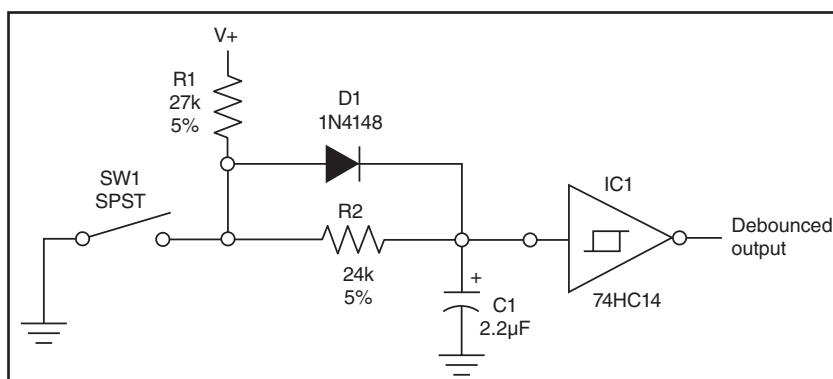


FIGURE 4 Hardware Debounce for SPST switches: This circuit charges the capacitor C1 through resistor R1 and diode D1 when the switch is released and discharges it through resistor R2 when it is activated. The 74HC14 Schmitt trigger provides the needed hysteresis for a snap-action output when the slow ramp of the RC circuit rises and falls. Hysteresis assures a single transition with no oscillation when the switch is activated or released.

For 100ms delay:	$R1 = 100 \times 750$	$R1 = 75,000\Omega$	
	$R2 = R1 \times 0.9$	$R2 = 75,000 \times 0.9$	$R2 = 67,500\Omega$
For 20ms delay:	$R1 = 20 \times 750$	$R1 = 15,000\Omega$	
	$R2 = R1 \times 0.9$	$R2 = 15,000 \times 0.9$	$R2 = 13,500\Omega$

TABLE 1 A simple way to calculate the delay time for the circuit shown in Figure 4 is $R1 = T \times 750$, and $R2 = R1 \times 0.9$, where T is the desired delay time in milliseconds, and R1 and R2 are resistance in ohms.

The Waveforms"). The individual application dictates the usefulness of this debounce method. Most applications can handle delays of up to 100ms with a barely noticeable delay. Method 4 is the hardware implementation for SPST switches (**Figure 4**).

SWITCH BOUNCE: THE WAVEFORMS

Note the immediate Q output in Waveform 1 (**Figure A**). Methods 1 and 2 change the output state with no delay both on actuation and release.

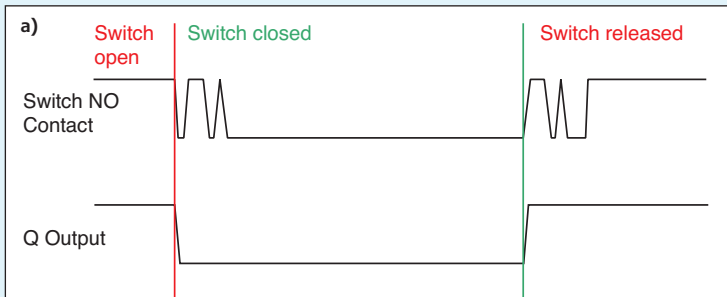


FIGURE A

Waveform 1. This applies to Cross-Coupled NAND and 2-Pin Debounce methods.

In Waveform 2 (**Figure B**), note the delayed Q output for these methods. The hardware debounce circuit of Method 3 and software debounce of Method 4 require delaying action until the bounce time is through.

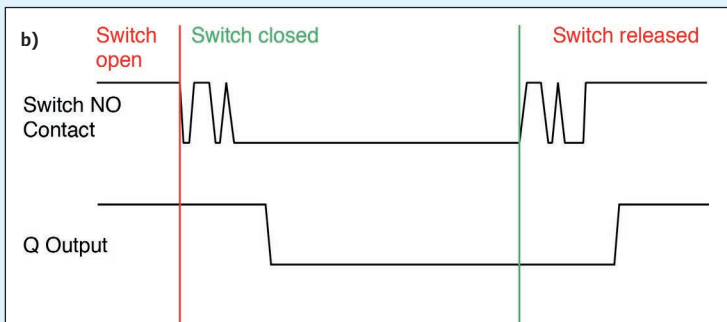


FIGURE B

Waveform 2. This applies to the Hardware Debounce and Software Delay methods.

Switch terminology: NO = Normally Open; NL = Normally Low; NH = Normally High

ABOUT THE AUTHOR

A serial inventor and entrepreneur, **Michael H. Pelkey**, founder and CEO of LogiSwitch, has a broad background in designing, using, and manufacturing electronic systems and equipment. Mike was an Automation Engineer at Jaxx Manufacturing where he designed and built assembly and metalworking machines. Also, Mike was the lead engineer for the first microprocessor-based product in the numerical control industry and developed the world's first networked cash register. Later, Mike founded Macrotech International, a major manufacturer of board-level computer products.

As shown in Figure 4, this circuit charges the capacitor C1 through resistor R1 and diode D1 when the switch is released and discharges it through resistor R2 when it is activated. The 74HC14 Schmitt trigger provides the needed hysteresis for a snap-action output when the slow ramp of the RC circuit rises and falls. Hysteresis assures a single transition with no oscillation when the switch is activated or released.

Note: The values shown Figure 4 provide delays of approximately 36ms on activation and release with a 5V supply. The resistance of R1 and R2 may be raised or lowered for longer or shorter delay times.

Using a 2.2 μ F capacitor for C1, a very simple way to calculate the delay time for the circuit shown in Figure 4 is $R1 = T \times 750$, and $R2 = R1 \times 0.9$, where T is the desired delay time in milliseconds, and R1 & R2 are resistance in ohms (**Table 1**).

METHOD 5

Software Delay Debounce: There are numerous examples online for software debounce routines for every computer language known to mankind. They all amount to simply waiting out the estimated bounce period. Here's how they work:


For Switch Activated:

1. Loop until switch activated.
2. Delay 20 ms to 100 ms.
3. Retest to be sure it is still activated. (If not, go to 1).
4. Continue with the program. Switch is activated.

For Switch Released:

1. Loop until switch released.
2. Delay 20 ms to 100 ms.
3. Retest to be sure it is still released. (If not, go to 1).
4. Continue with the program. Switch is released.

You must provide a longer delay period than any switch you use will ever bounce. Contact bounce time varies among switch types and even among switches of the same type and models. Delay time also is said to change over time with the same switch. Typical delay times used are 20ms to 100ms, even when the actual bounce time may be as little as a few milliseconds.

For compute-bound applications where time is literally of the essence, such as in high-speed automated industrial machinery or robotics applications, your choice of debounce method should be Method 1: 2-Pin Debounce, or Method 2: Cross-Coupled NAND Debounce. 

For detailed article references and additional resources go to: www.circuitcellar.com/article-materials.

RESOURCES

LogiSwitch | www.logiswitch.com