# How to "DEBOUNCE" Mechanical Switches for Digital Logic Use

## Interface circuitry eliminates false pulses.

BY E. W. GRAY

MANY projects involving digital logic require the use of one or more mechanical switches. These can take the form of toggle switches, relays, pushbuttons, or keyboards. Two characteristics of these devices, switching noise and timing, require special consideration when connecting them to a digital logic system.

All mechanical switches, regardless of type, normally generate some electrical noise when the contacts transfer from one position to the other. This is due to bouncing of the contacts for several milliseconds after actuation. They actually make, break, and remake several times before finally coming to rest in the new position. This bounce period is called settling time. Digital logic elements, being much faster in their operation than mechanical switches, respond to each transition during the bounce period if the logic is connected directly to the switch. Thus false signals are produced. For this reason, "debouncing" circuits are used between the mechanical switch and the driven logic.

**Contact Bounce in Spdt Switches.** A single-pole, double-throw (form-C) switch and an idealized timing diagram of its action during transfer from normally closed to normally open are shown in Fig. 1. Initially, the movable arm (operating strap) is in contact with the normally closed contact. As transfer begins, the arm moves away from the normally closed side, opening the contact. The slightly springy, normally closed contact attempts to follow the arm and bounce occurs. This is called "break bounce;" the multiple openings and closings of the normally closed contact and the arm as the switch transfer is initiated.

As transfer continues, break bounce ceases. At this time the arm is not in contact with either side of the switch, but is "in transit" to the opposite side. Both normally closed and normally open contacts are now open. (Note: "Make-before-break," or shorting, switches are available; similar bounce conditions can occur, however.)

When the arm reaches the normally open contact, the two collide, and "make bounce" begins. The arm and the normally open contact close, open, and close again until the mechanical movement ceases. The switch transfer is then complete. In most switches, make bounce is much more severe than break bounce.

When the form-C switch is released from the transferred position, the reverse of the above actions occur. Break bounce takes place at the normally open contact and make bounce occurs at the normally closed contact.

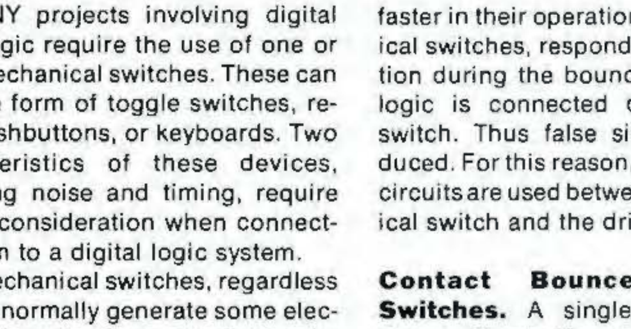


*Fig. 1. During the few milliseconds it takes a switch to operate, erroneous pulses can confuse digital logic.*

**Debouncing Spdt Switches.** To use the form-C switch successfully in digital logic, the debounce circuit must mask both break and make

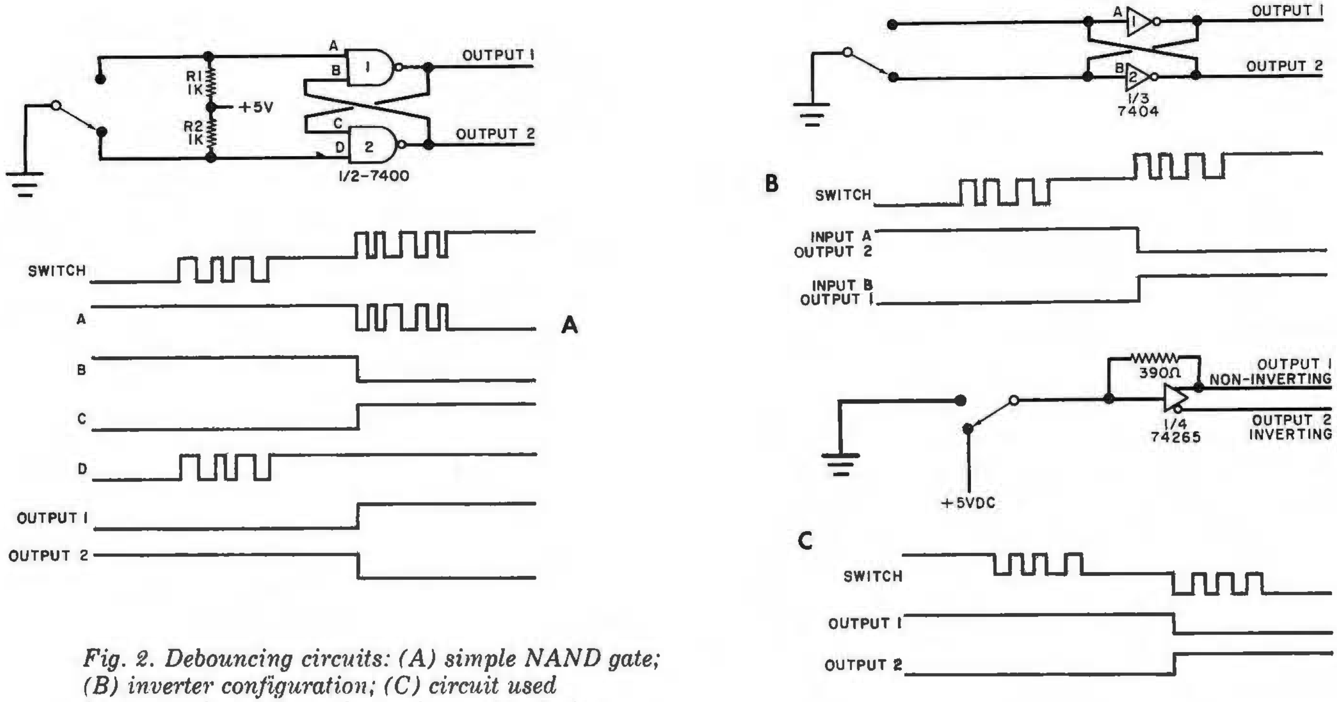


*Fig. 2. Debouncing circuits: (A) simple NAND gate; (B) inverter configuration; (C) circuit used in a recently announced complementary device.*

bounce and produce a single, noise-free transition at its output each time the switch is operated or released.

Three circuits to accomplish this are shown in Fig. 2. TTL devices and positive logic are used. (High voltage equals logic one; low voltage equals logic zero.)

In the circuit of Fig. 2A, two cross-coupled NAND gates are used to form a latching circuit. With the switch not operated, the ground (logic zero) applied to input D, gate 2, holds output 2 at logic one. Output 2 is applied to input B, gate 1. This high voltage level, and the high voltage level on input A, gate 1 (through R1 to +5 volts) cause output 1 to be logic zero.

When the switch is operated, input D, gate 2, changes in step with the break bounce. However, outputs 1 and 2 do not change state since the movable arm has not yet grounded input A, gate 1, and the logic zero on output 1 (fed back to input C, gate 2) maintains output 2 at logic one. Thus break bounce is ignored at the outputs.

As mechanical action continues, break bounce ceases and the arm makes its first contact with the normally open side. This applies ground (logic zero) to input A. With input A low, output 1 and input C switch to logic one. Inputs C and D are now both at logic one and output 2 goes to logic zero. Fed back to input B, output 2 now latches output 1 high and the circuit is stable.

This switching action between outputs 1 and 2 requires very little time: a maximum of 52 nanoseconds if 7400 NAND gates are used. Thus, the switching is complete long before the first bounce during make occurs. Input A will continue to follow the make bounce transitions but no output changes will occur. When the switch is released, the action is reversed, with output 2 reverting to logic one and output 1 to logic zero.

A somewhat simpler circuit is shown in Fig. 2B. The 7400 NAND gates are replaced by 7404 inverters. Pullup resistors are not required for this circuit. The switching operation is similar to that of Fig. 2A, except that one gate output will be short-circuited for about 37 nanoseconds each time the switch is activated. Device operation is not affected by the short-circuited output; the manufacturer's specification allows a single output in the logic one state to be grounded temporarily.

Note that switch bounce can be observed with an oscilloscope at inputs A and D of Fig. 2A. It cannot be observed at the inputs of the circuit of Fig. 2B due to the direct coupling of input and output.

Both of the above circuits provide complementary outputs, and both require two input lines from the switch. If both ground and +5 volts are available at the switch, debouncing can be ac-

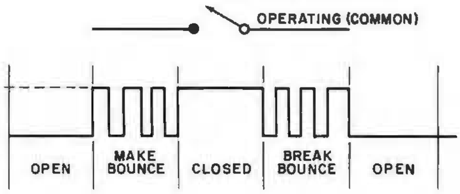
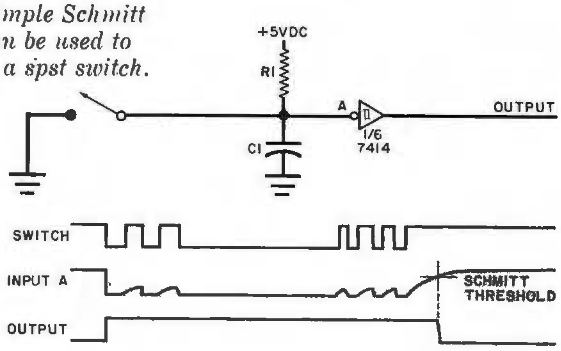


*Fig. 3. Bounce pattern of a single-pole-single-throw switch.*

*Fig. 4. Simple Schmitt trigger can be used to debounce a spst switch.*



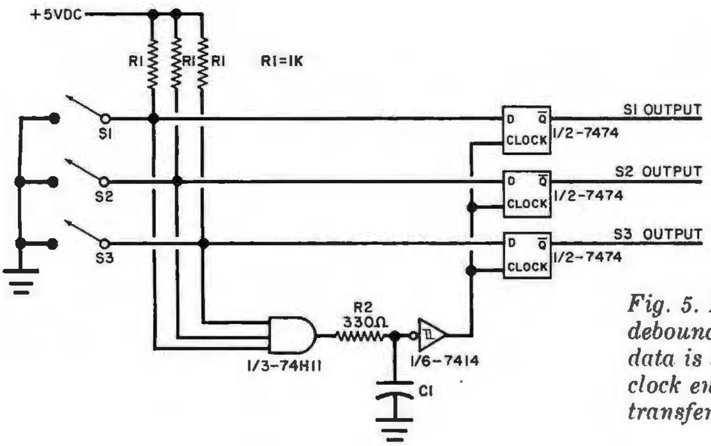


*Fig. 5. In keyboard debouncing, switch data is stored until clock enable transfers the data.*

complished with only a single input line as illustrated in Fig. 2C. The integrated circuit used is the 74265, a recently announced complementary output device. Here, the noninverting output is fed back to the input, providing stabilizing action much like the 7404 latch of Fig. 2B.

**Contact Bounce in Spst Switches.** A single-pole, single-throw (form-A) switch and its idealized timing diagram are shown in Fig. 3. This switch is equivalent to a single side of the form-C switch. When a spst switch is operated, break bounce occurs between the movable arm and the normally closed contact. No further action occurs until the switch is released. Following release, the operating strap remakes with the normally closed contact, bounces, and make bounce occurs. As in the form-C switch, make bounce is usually more severe than break bounce. The time required for make bounce to subside is an important factor in debouncing the form-A contact.

**Debouncing Spst Switches.** Since only a single output is available from the form-A switch, the latching method of debouncing previously described cannot be used. Instead, a delay circuit is normally used to mask the contact bounce. The debounce circuit must: (1) detect the switch transition; (2) delay response for a sufficient amount of time to allow all bounce to cease, and (3) produce an output defining the new switch position. A simple circuit to accomplish this is shown in Fig. 4.

The gate used is a Schmitt trigger device such as 7413 or 7414. With the switch contacts open as shown, input A will be approximately 5 volts, and the output will be logic zero. When the switch is operated, input A goes low and the output switches from low to high. Each time the switch bounces open, point A starts to return to the 5-volt level at a rate determined by the time constant of *R1* and *C1*. As long as input A does not reach its positive-going threshold voltage of about 1.7 volts, the output will remain at logic one. Therefore, the time constant of *R1* and *C1* should be sufficiently long to allow all bounce to subside before this threshold is reached. Usually about 5 to 10 milliseconds is satisfactory.

**Keyboard Debouncing.** Keyboards, having many switches, pose a special problem when debouncing is considered. Building a debounce circuit for each switch is impractical. Many keyboards include a strobe, or gating, contact which is activated whenever any key is operated. In such keyboards, this common switch should be debounced and its output used to test the state of the remaining switches. Where a common switch is not available, the individual switches can be logically "OR'ed" and the resultant output used as input to the debounce circuit. A typical circuit is shown in Fig. 5. Note that this is an application of the circuit of Fig. 4. Each time a switch is operated, the output of the debounce circuit clocks a flip-flop register which stores the switch information until another switch is operated.

Note that only Schmitt trigger gates should be used with RC networks as shown, since such networks on the input of an ordinary gate can cause oscillation at the gate's output. Series resistance such as *R2* in Fig. 5 should not exceed 330 ohms.