NoBounce Switches with Handshake Overview

General Description

LogiSwitch provides a series of pushbutton and limit switches incorporating the same unique NoBounce™ adaptive debounce technology and handshake protocol as implemented in the LogiSwitch LS100 line of bounce-free integrated circuits.

All LogiSwitch switch products provide a selection of three different output formats:

1. Normally Low/Handshake (NL/HS) pin provides an active-high handshakeable output.
2. Normally High (NH) pin provides a low-impedance active-low output.
3. Toggle (TG) pin changes state each time the switch is actuated. (powers-up high)

16 mm Vandal-Resistant Switch with Green LED - LS16MM-G

LogiSwitch provides three styles of switches modified with our unique immediate output and handshake protocol. The line consists of:

a) High-end 16 mm, 19 mm, and 22 mm vandal resistant IP67 pushbutton switches with ring LEDs

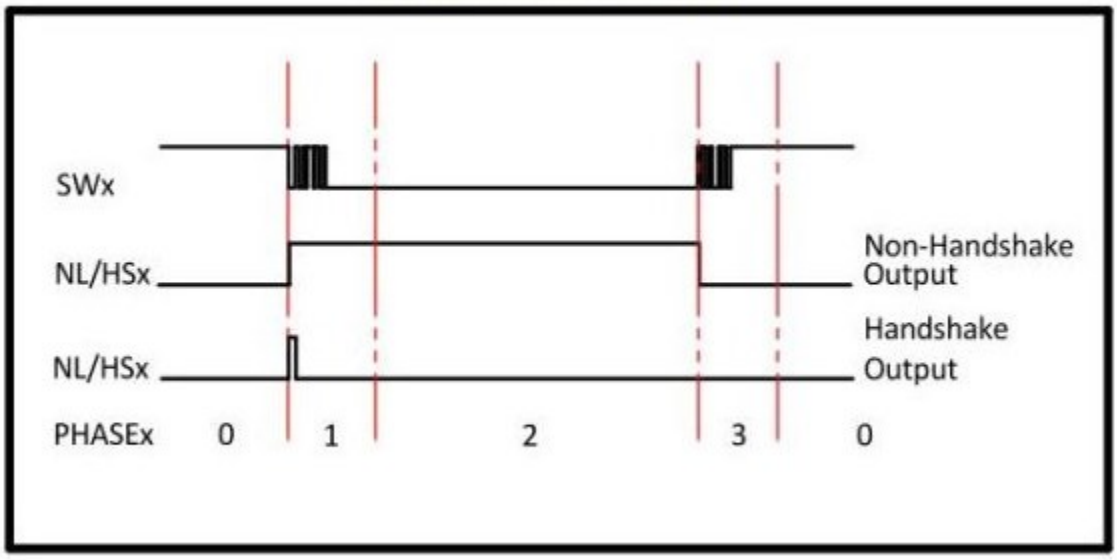
b) Simple low-cost 6mm pushbutton switches with red, white, or black caps

c) Low-cost subminiature limit switches in 5 lever styles

Pin Description

Pin	Name	Function
1	Vdd	+2.3 V to +5.5 V Supply Voltage
2	NH	Normally High Output
3	NL/HS	Normally Low/Handshake I/O
4	TG	Toggle Output
5	GND	Ground

Waveforms



Electrical Characteristics

Parameter	Symbol	Conditions	Min	Typ	Max	UNIT
Operating	Vcc		2.5		5.5	V
Supply	Icc	Vcc = 3.0V, All Inputs Open	1.0		1,550	µA
Input Pull-	Ipu	LS118	25	100	200	mA
Input Pull-	Ipu	LS119, LS120	25	120	200	mA
Debounce	tdb	Vcc = 2.5V - 5.5V		21		mA

A typical "polled" switch routine entails sitting in a loop wasting time until the switch is released. In the lion's share of applications, the length of time a switch is held active is irrelevant. The LogiSwitch handshake protocol is used to pass control of terminating the switch routine to the program, which is always perfectly relevant. Only the program knows when it no longer needs the switch service request.

All LogiSwitch devices and switches incorporate the handshake protocol on the NL/HS (Normally Low/Handshake) pins. The NL/HS line is configured as an open collector wired-OR line so both the LogiSwitch device and the host computer can drive the line low simultaneously to accomplish the handshake as needed.

Test for Switch - a Service Request in the Main Loop

If the NL/HS line is high, the switch has become active. If not, continue with the main loop until the next time through.

Okay, it is active. To acknowledge that we have received the request, we will send a pulse back to the LogiSwitch device.

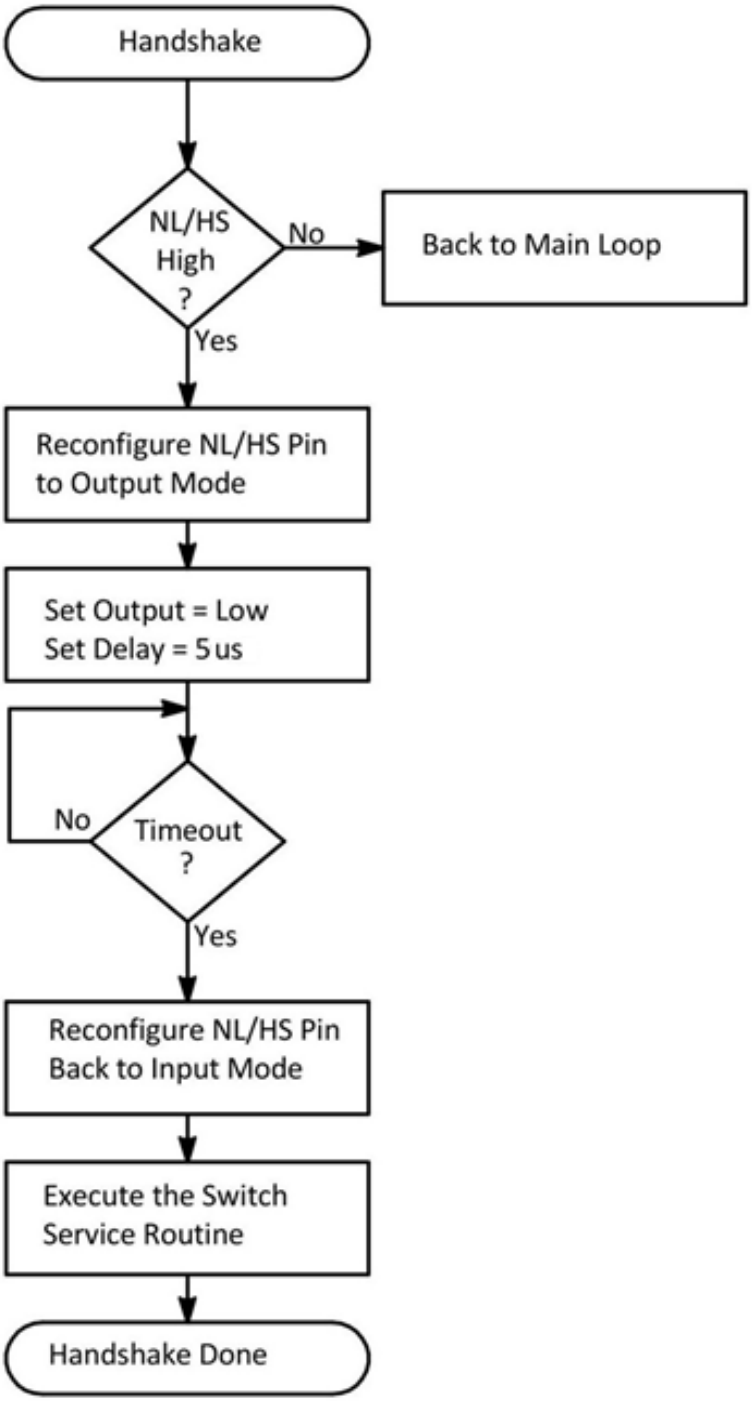
First, we set the NL/HS pin to output mode, then make it low.

Now we set a 5 µs delay to allow time for the LogiSwitch device to recognize our acknowledgement.

Has the delay timed out yet?

Timeout done. Now we want to go back to input mode. The LogiSwitch device has already seen the acknowledge pulse and has latched the line out low to end the cycle without the need for release of the switch. Note that another cycle will not be initialized until the switch has been released and its output debounced.

Now we are all done with this cycle. We can execute our switch service routine and go back to executing code in the main loop. The LogiSwitch device will determine when the switch has been released and debounced, so the next switch cycle may be initiated.



Software – Implementing the Handshake with an Arduino

The following few lines of code for an Arduino Uno demonstrate the simplicity of a host computer interface using the LogiSwitch handshake.

```
// ********************************************************************************
// This code snippet for Arduino Uno demonstrates the single-pin handshake
// protocol of the LS1xx Series Switch Debouncer ICs for Embedded Processors
// ********************************************************************************
int NL_HS = 8;                                  // Define the pin(s)
void setup () {                                 // Start with REQ_ACK configured to Input Mode
   pinMode (NL_HS, INPUT);
}
//*********************************************************************************
void loop() {
// Place this code at the appropriate place in the main loop...
   pinMode (NL_HS, INPUT);                      // Set NL/HS pin to input mode
   if (digitalRead (NL_HS) == HIGH)             // Request from switch?
  {
     pinMode (NL_HS, OUTPUT);                   // Yes, respond with handshake
     digitalWrite (NL_HS,LOW);                  // Acknowledge with a 5 µs low pulse
     delayMicroseconds(5);
     pinMode (NL_HS,INPUT);                     // Back to input mode
     delayMicroseconds(5);
// Place switch service routine or function call here...
   }
```