AN338

CAPACITIVE TOUCH SENSE SOLUTION

RELEVANT DEVICES

The concepts and example code in this application note are applicable to the following device families:

C8051F30x, C8051F31x, C8051F320/1, C8051F33x, C8051F34x, C8051F35x, C8051F36x, C8051F41x, C8051F52x/53x, C8051T60x, C8051T61x, C8051F93x/92x.

This application note includes example code for the C8051F93x/92x and C8051F336–9 device families.

1. Introduction

Touch-sensitive switches are found in a variety of consumer products including home appliances, MP3 players, and cell phones. The technology behind these switches has been available for many years, but the demand for smaller products has made these switches a popular choice for manufacturers.

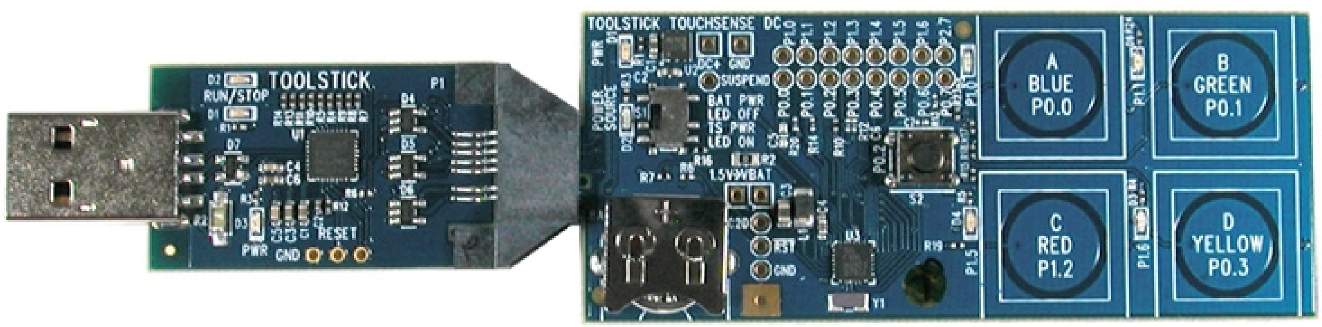

Figure 1. C805193x/92x Capacitive Touch Sense Development Platform

A touch-sensitive switch is a switch that is implemented as a trace on a printed circuit board. The architecture of the trace creates a capacitive element on the board. Touching this trace with your finger creates a change in capacitance, which is detectable using a variety of techniques. The Silicon Labs Capacitive Touch Sense solution uses the switch capacitor as part of a simple resistor-capacitor (RC) relaxation oscillator.

In addition to being smaller, more reliable, and more cost-effective than physical switches, touch sense switches are also easier to manufacture because they do not have any mechanical components.

The hardware peripherals necessary to add touch-sensitive switches are available on most Silicon Labs small-form-factor microcontrollers; therefore, it is easy to find the right microcontroller for the product. The relaxation oscillator solution has many advantages, including low BOM cost, small Flash requirements (< 400 bytes), little MCU overhead, low power requirements, and the fact that the MCU is not required to use a precise voltage source.

This application note covers the following topics:

- Creating a touch sensitive switch with a relaxation oscillator
- Adding a single touch sense switch to a design
- Adding multiple touch sense switches to a design
- The most effective layout patterns for a switch
- The effects on sensitivity of different types of materials typically used on top of the switches

Copyright © 2008 by Silicon Laboratories

2. Touch Sense Switch Implementation

The following section describes how a touch-sensitive switch is created using a relaxation oscillator. This section also describes how to add single and multiple switches to a system and the different firmware methods for detecting a switch event.

2.1. How a Relaxation Oscillator Works

A relaxation oscillator is a circuit whose output frequency is inversely proportional to the value of a specific capacitor, among other components. The circuit charges this capacitor to a certain voltage threshold and triggers an event once the threshold is reached, causing the capacitor to discharge. When the voltage across the capacitor reaches a lower threshold, another event is triggered and the capacitor begins to charge again. The rate at which the capacitor charges and discharges determines the oscillation frequency.

2.2. How to Build a Touch-Sensitive Switch Using a Relaxation Oscillator

When using a relaxation oscillator as part of a touch-sensitive switch circuit, the switch itself is the capacitor that is charged and discharged. The switch capacitor is created using a trace on a printed circuit board. Interleaving two copper traces with each other creates the two layers of a capacitor. One of these traces is connected to ground, and the other trace is connected to the charge and discharge circuitry.

The physical architecture of the switch determines its capacitance. The switch capacitor shown in Figure 2 provides an example of one option for implementing a switch using interlocking copper traces on a printed circuit board.

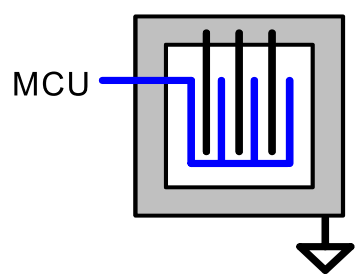

Figure 2. PCB Trace for a Touch-Sensitive Switch

2.2.1. Building a Relaxation Oscillator with a Standard Silicon Labs Comparator

For Silicon Labs microcontrollers that do not have a capacitive touch sense-enabled comparator, the relaxation oscillator is easily designed using a standard comparator and a few external resistors, as shown in Figure 3. The negative input to the comparator (CP0–) is the voltage across the switch capacitor, and the positive input (CP0+) switches between both the high and low event threshold depending on the switch state. The output of the comparator is a logic 1 when CP0+ is greater than CP0–, and 0 when the CP0– is greater than CP0+. The asynchronous output of the comparator (CP0A) is routed to an external pin and is part of the feedback that sets the required threshold.

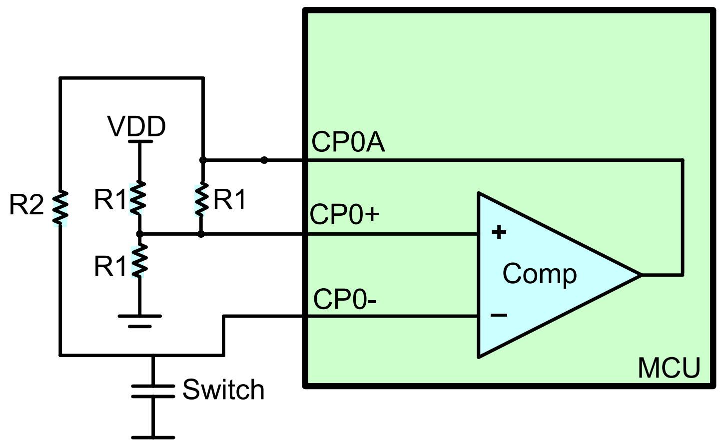

Figure 3. Relaxation Oscillator Schematic Using the MCU's On-Chip Comparator

When the voltage on CP0+ is greater than CP0–, the output of the comparator is 1 or VDD, which sets the CP0+ voltage to 2/3 VDD by using the resistor divider created by the three R1 resistors. Also, the switch capacitor is charged using the comparator output as the voltage source.

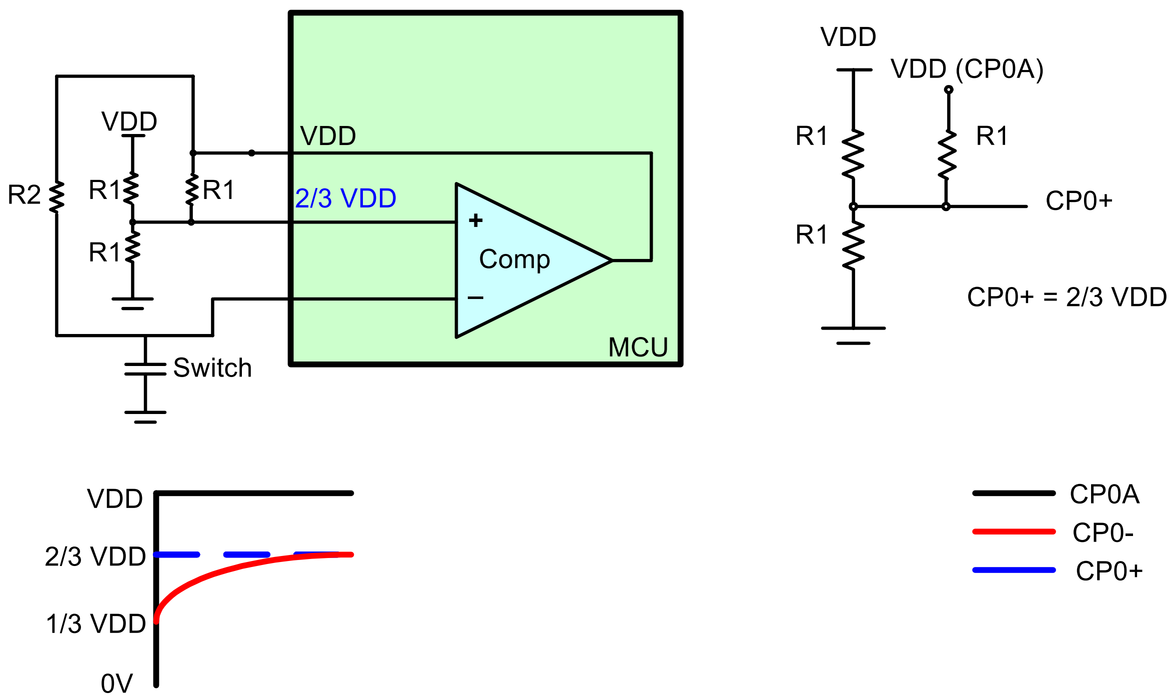

Figure 4. Comparator Output High and Charging the Switch Capacitor

When the CP0– voltage goes above 2/3 VDD, the comparator output changes to 0, and the switch capacitor begins to discharge until it reaches 1/3 VDD, as shown in Figure 5.

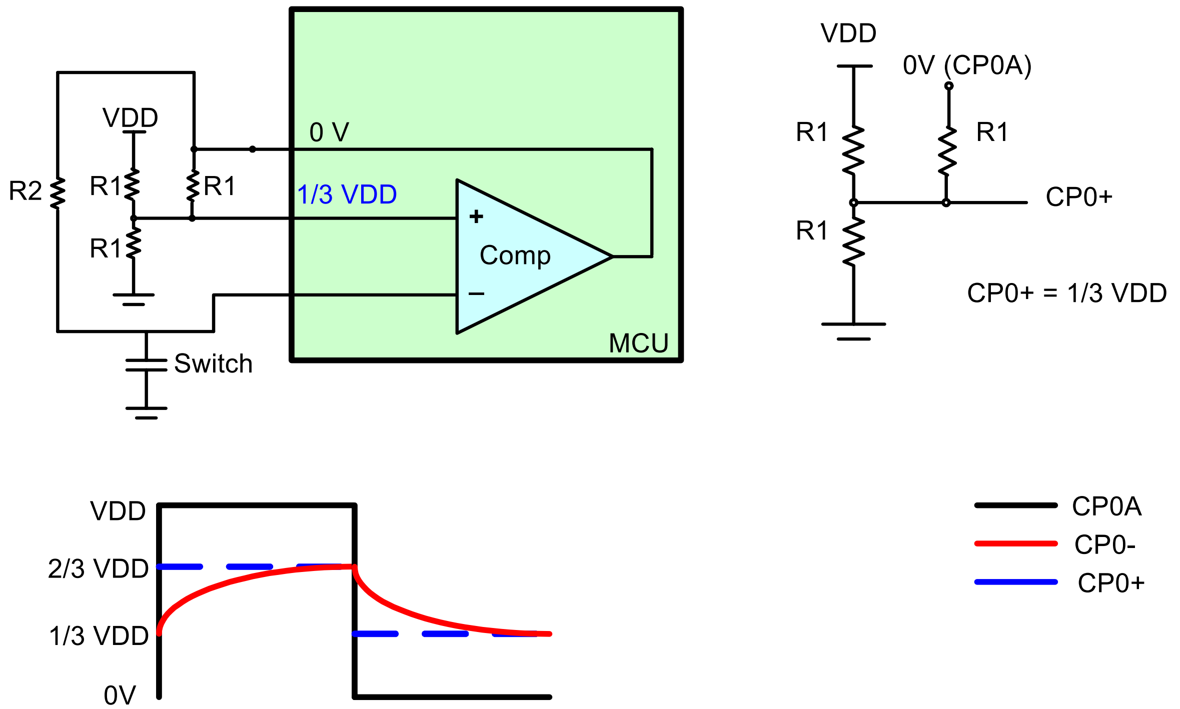

Figure 5. Comparator Output Low and Discharging the Switch Capacitor

Once the voltage on the capacitor discharges below 1/3 VDD, the comparator output changes to a 1 and restarts the charging cycle. The output waveform present on the asynchronous comparator output (CP0A) is the output frequency of the relaxation oscillator.

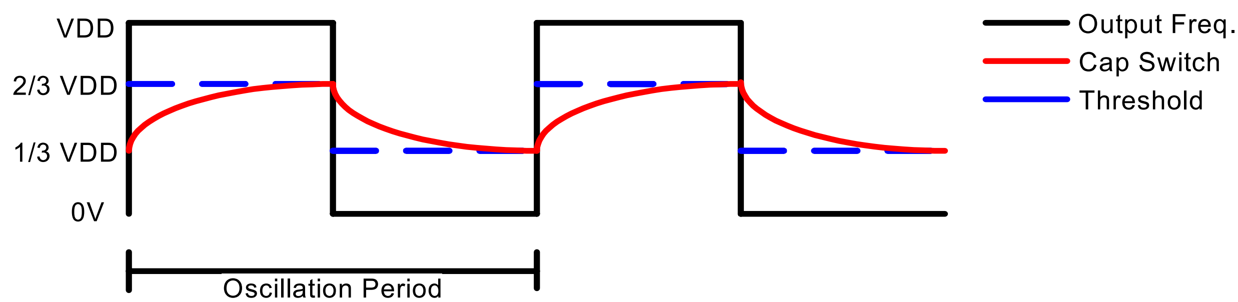

Figure 6. Relaxation Oscillator Output Frequency

The comparison between CP0+ and CP0– happens asynchronously to the system clock, and CP0A is set automatically by the comparator hardware. Once the relaxation oscillator is started, no firmware intervention is necessary to maintain it.

2.2.2. Building a Relaxation Oscillator with a Capacitive Touch Sense-Enabled Comparator

The C8051F93x/92x family of devices include two comparators that include touch sense functionality. These enhanced comparators have the necessary circuitry on-chip for a relaxation oscillator. The relaxation oscillator works in the same manner as described in “2.2.1. Building a Relaxation Oscillator with a Standard Silicon Labs Comparator”, with some advantages. When configured for touch sense, the output of the comparator is internally tied to the comparator switch multiplexer to provide the feedback to charge and discharge the switch capacitor. Also, the 2/3 VDD and 1/3 VDD thresholds are automatically configured internally, which saves a GPIO pin. The block diagram of a C8051F93x/92x comparator configured for touch sense is shown in Figure 7.

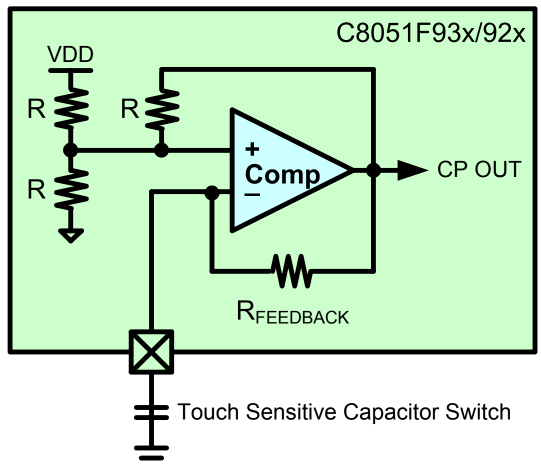

Figure 7. C8051F93x/92x Comparator configured for Capacitive Touch Sense

2.3. Touching the Switch Changes the Output Frequency

When a finger is placed on the touch-sensitive switch, it increases the dielectric of the switch capacitor. Increasing the dielectric increases the capacitance of the switch as it is directly proportional to the dielectric. This additional capacitance causes an increase in the charge and discharge timing, which leads to a decrease of the relaxation oscillation frequency. This change in frequency indicates the presence of the finger.

The percentage change in frequency due to the presence of a finger is dependent on many factors, and they are described in more detail in "3. Designing Touch-Sensitive Switches" on page 11.

2.4. How to Detect the Change in Frequency

There are two straightforward methods for detecting the decrease in the relaxation oscillator frequency caused by the presence of a finger. The first method is to count the number of cycles of the relaxation oscillation frequency over a fixed period of time. This method can be described as measuring the frequency. The second method is to count the number of MCU system clock cycles during a fixed number of cycles of the relaxation oscillator output frequency. This method can be described as measuring the period. Each of these methods has its own advantages, and both methods are described in “2.4.1. Measuring the Frequency” and “2.4.2. Measuring the Period”.

When using a standard microcontroller comparator, the asynchronous output of the comparator must be externally routed back to the MCU to implement either one of these counting methods. The counting of the relaxation oscillator edges is performed by an on-chip timer that can use an external clock source as its trigger. Figure 8 shows the comparator output (CP0A) routed back to the Timer input pin.

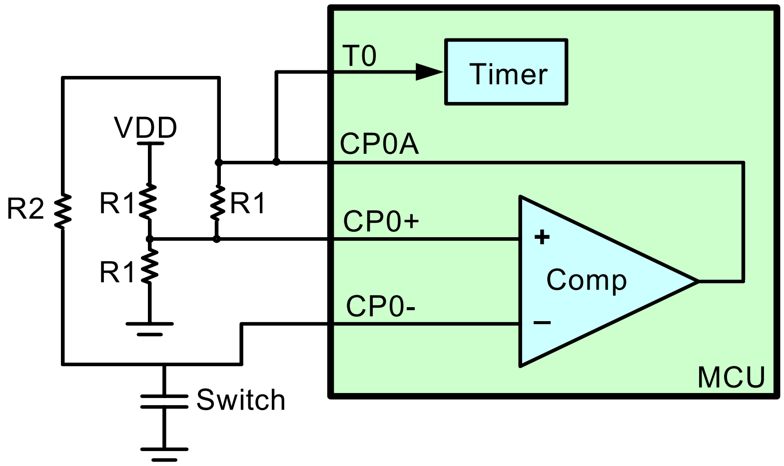

Figure 8. Asynchronous Comparator Output (CP0A) Routed Back to the Timer0 Input (T0)

The schematic shown in Figure 8 is the basic configuration necessary to implement one touch-sensitive switch. The solution requires four digital I/O pins and an on-chip comparator and timer. A second Timer is typically used to periodically check the counter value or count the system clock cycles, depending on the measurement method. The requirements for additional switches are described in “2.5. Multiple Switches”.

When using an enhanced comparator on the C805192x/93x devices, the comparator output is internally routed to a Timer as shown in Figure 9.

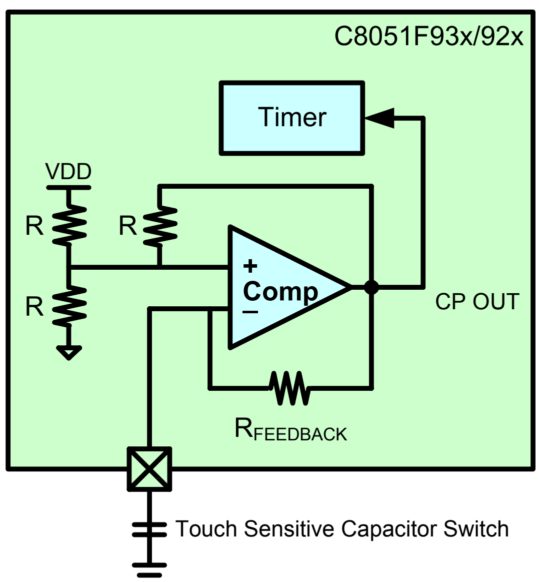

Figure 9. Comparator Output Connected to a Timer on C8051F93x/92x devices

This C8051F93x/F92x solution requires no external components and only an on-chip comparator and timer. Additional switches can be connected directly to spare MCU ports pins and the comparator multiplexer input is used to select between them.

2.4.1. Measuring the Frequency

Measuring the frequency is one of the two methods for detecting a switch event. For this method, a timer is used to count the number of relaxation oscillator cycles over a fixed period of time. If the number of measured cycles over the fixed period time is lower than a pre-calibrated threshold value due to the presence of a finger, the switch is considered pressed.

One of the examples provided with this application note targets a C8051F336DK development board, which includes one touch sensitive switch. In the example F338_CapTouchSense_MeasureFrequency.c, Timer0 is gated by the relaxation oscillator and Timer2 checks the counter value of Timer0 every 20 ms. Without a finger on the switch, Timer0 counts about 42400 relaxation oscillator cycles every 20 ms. With a finger on the switch, the Timer0 counts about 13000 relaxation oscillator cycles over the same time period. On the average sensitivity setting, the comparison threshold is set to 27700 cycles, and so this change is easily and reliably detected in firmware as a switch event.

2.4.2. Measuring the Period

Another method for detecting a switch event is to count the number of system clock cycles during a fixed number of relaxation oscillator cycles. If the switch is pressed, the relaxation oscillator frequency decreases and so more system clock cycles are measured over those same number of cycles.

Another example provided with this application note, F338_CapTouchSense_MeasurePeriod.c, measures the number of system clock cycles over 20 relaxation oscillator cycles. Timer0 is set to interrupt after 20 relaxation oscillator cycles and Timer2 is the free running timer counting system clock cycles. Without a finger on the switch, about 1800 system clock cycles occur every 20 relaxation oscillator cycles. With a finger on the switch, about 4100 cycles occur over the same measurement range. Similar to the method of measuring frequency, with an average sensitivity, the comparison threshold is set in the middle to ~3000 system clock cycles, so it is unambiguous to the firmware when the switch is and is not pressed.

The advantage of measuring the period is that a switch can be scanned very quickly compared to measuring the frequency. Even measuring across only a few relaxation oscillator cycles, the measurement is very stable and accurate.

2.5. Multiple Switches

The on-chip comparator input multiplexers available on Silicon Labs MCUs make it very easy to add multiple touch sensitive switches to a system. On the standard comparators, the negative input (CP0–) is firmware-configurable to use one of many GPIO pins as the source. For example, the C8051F41x devices support up to 12 different comparator negative inputs. Each of the CP0– pins can be directly connected to a different switch. Each additional switch requires only one additional port pin and one external resistor. The schematic in Figure 10 shows how to connect multiple switches to a system.

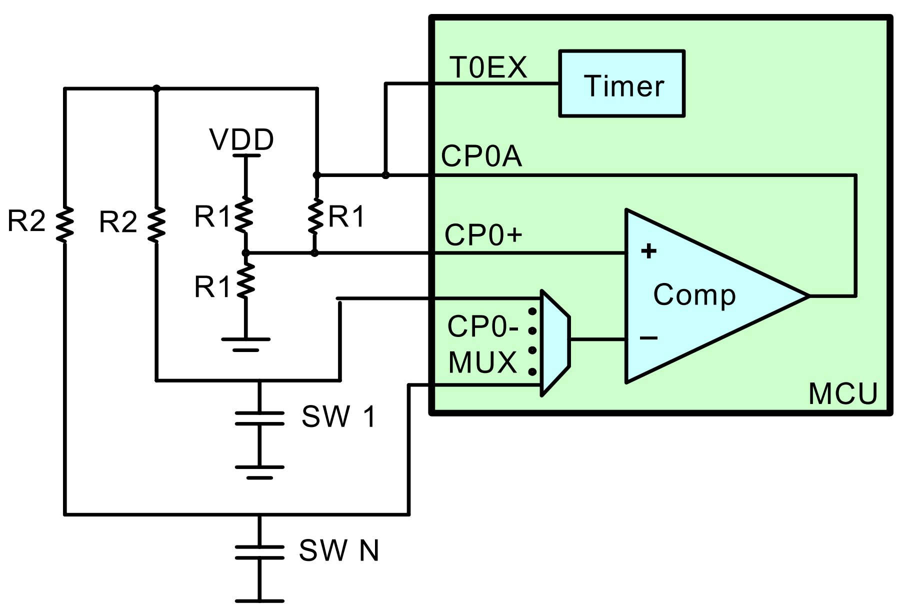

Figure 10. Multiple Touch Sensitive Switches Connected to a Standard Comparator MCU

The additional firmware overhead to monitor the extra switches is minimal. The only additional code required is the code necessary to configure the CP0– multiplexer after each measurement. With this implementation, only one switch is measured at any one time. As long as the switches are monitored often enough, the user will not experience any delay or lag in a system that uses multiple switches.

On the C8051F93x/92x microcontrollers, the comparators are flexible so that both the positive and negative inputs for the comparators can be connected directly to touch sensitive switches. The comparator inputs and configuration are easily configurable in firmware and so adding multiple touch sensitive switches to an MCU in this family only requires a simple change to firmware. Up to 23 GPIO pins on the C8051F93x/92x devices can be connected to switches as shown in Figure 11.

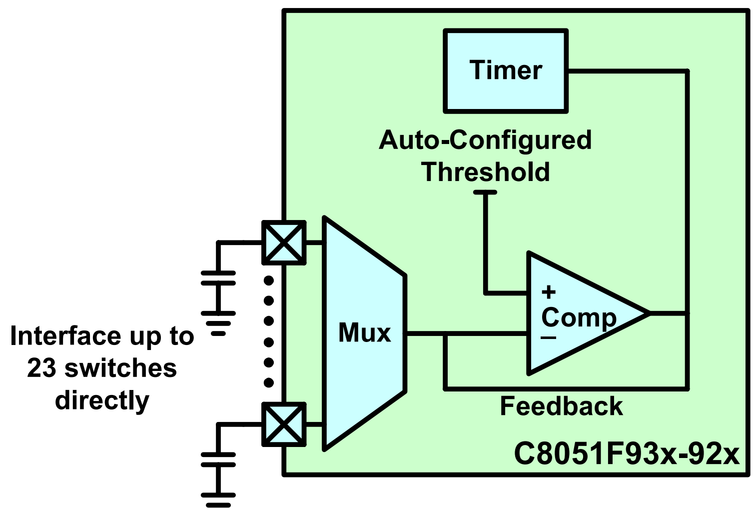

Figure 11. Interfacing Multiple Touch-Sensitive Switches to an Enhanced Comparator

Most Silicon Labs MCUs have multiple comparator peripherals. Each of the comparators can be used to monitor switches. Figure 12 shows the schematic for using multiple comparators. In this configuration, the inactive comparator is turned off so that its output does not affect the charge and discharge timing of the active switch. The comparators can share the same Timer to conserve resources. The CP0– and CP1– multiplexers can be used as shown in Figure 10 to maximize the use of each comparator.

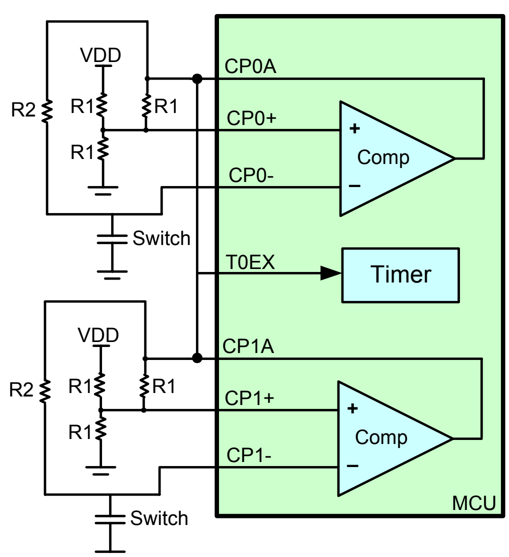

Figure 12. Using Multiple Comparators to Monitor Additional Switches

Table 1 shows the maximum number of touch-sensitive switches that Silicon Labs MCUs can support without using an external multiplexer.

Table 1. Maximum Number of Switches Directly Supported by Silicon Labs MCU Families

MCU Family	Switches Supported	MCU Family	Switches Supported
C8051F30x	4	C8051F36x	8
C8051F31x	8	C8051F41x	12
C8051F320/1	8	C8051F52x/53x	8
C8051F33x	8	C8051T60x	4
C8051F34x	10	C8051T61x	8
C8051F35x	8	C8051F93x/92x	23

When using an MCU with a standard comparator, the number of digital I/O pins required to directly interface to N switches is 3 + N. The 3 pins required for each system are CP0+, CP0A, and the Timer input. Each Touch sense switch requires one additional digital I/O port pin. When using an enhanced comparator, only one digital I/O pin is necessary per switch.

If the system is pin-limited, an external analog multiplexer can be used in addition to or instead of the on-chip comparator multiplexer. An external multiplexer requires fewer dedicated digital I/O pins at the cost of adding an additional component to the bill of materials. For example, connecting eight switches directly to the MCU requires 11 GPIO pins, but only 7 pins are required if an external multiplexer is used.

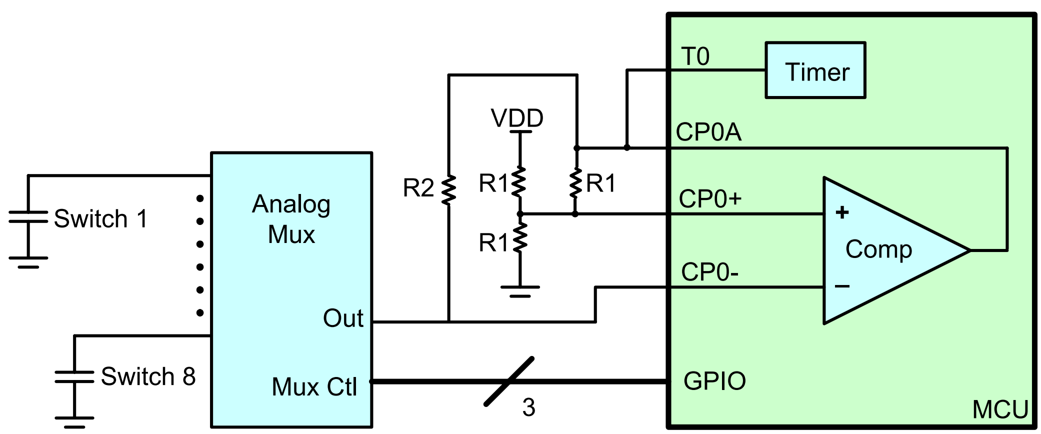

Figure 13. Using an External Analog Multiplexer to Connect the Switches to the MCU

2.6. Calibration

Both methods for detecting a switch event compare the Timer frequency counter against a comparison threshold. Properly calibrating this threshold is an important factor in the sensitivity of the switch. If the threshold is set too far from the idle value (the value when the switch is not pressed), a switch event will not occur unless the user presses their finger very heavily against the switch. If the threshold is set too close to the idle value, a switch event could occur before the user even physically makes contact with the PCB.

In all systems, an initial calibration is necessary to set the threshold values for all the switches. A periodic calibration might be necessary for systems operating in a dynamic environment.

2.6.1. Performing an Initial Calibration

The initial calibration is performed to set the comparison threshold for all of the switches. The calibration needs to be performed individually for each switch on the PCB because the appropriate threshold varies between switches. Even if each switch is the same size and shape, the different placement of the switches relative to the MCU will affect the idle oscillation frequency.

The initial calibration can be performed on a few prototypes, and the threshold values determined from these calibrations can be used in the production firmware. Alternately, the initial calibration can be performed on each system during the first power-up, and the threshold values can be written to the MCU Flash.

The typical procedure for the initial calibration when using the frequency measurement method is as follows:

1. Count the number of relaxation oscillator cycles over a fixed period of time (sample time) with no finger on the switch. This value is SWITCH_OPEN_COUNT.
2. Count the number of relaxation oscillator cycles over the same sample time with a finger on the switch. This value is SWITCH_CLOSED_COUNT.
3. Set the threshold value to (SWITCH_OPEN_COUNT – SWITCH_CLOSED_COUNT)/2.
4. Repeat steps 1–3 for each switch.

Keep in mind the following notes when performing the initial calibration:

1. The sample time used in steps 1 and 2 should not be so long that the counter overflows from too many relaxation oscillator cycles. Increasing the sample time does increase the resolution of the measurement, but the upper limit is the 16 bits of the Timer counter registers.
2. The threshold set in Step 3 does not have to be the midpoint between SWITCH_OPEN_COUNT and SWITCH_CLOSED_COUNT. To make the switch more sensitive, the threshold can be set closer to SWITCH_OPEN_COUNT.

2.6.2. Performing a Periodic Calibration

In some systems, the conditions are so dynamic that the idle relaxation oscillator frequency will drift closer to the threshold frequency. If this happens, a switch event could occur accidentally, or, in the worst case, the switch would always be considered active. This drift in the relaxation oscillator could occur for the following reasons:

1. The material covering the PCB gets coated in some other material, such as oil or water, which changes the capacitance of the switch.
2. In a battery-operated system, the VDD supply drops greatly, which causes the capacitance charging time to increase.
3. The ambient temperature fluctuates, causing the system clock oscillator to drift, changing the measurement. The percentage tolerance of the on-chip oscillator for Silicon Labs MCUs is guaranteed across the operating temperature range of the device; so, this is not an issue for these microcontrollers.

One method to detect the oscillator drift is to keep a history of the measured frequency for the switches. If the measurements are constantly outside of a threshold, a recalibration should be performed. Another option is to recalibrate periodically without checking for the oscillator drift.

If a full recalibration is performed in the field, the device will only be able to measure the new idle output frequency of each switch and will not be able to measure the frequency when the switch is pressed, unless a user is present.

An alternative to performing a calibration in the field is to perform multiple calibrations during the prototyping phase under the different conditions for VDD and temperature and store all of the calibrated values in the MCU's Flash. The MCU can then use the different thresholds depending on the current conditions.

3. Designing Touch-Sensitive Switches

Designing a system that includes touch-sensitive switches requires taking into a account many different factors. For a satisfactory user experience, the switch behavior must be consistent and robust in different operating conditions. This section describes the various factors that influence switch behavior and provides design considerations for optimal switch layout and placement.

3.1. Influences on Switch Sensitivity

A touch-sensitive switch is not useful unless the system can reliably determine the state of the switch. Mechanical switches create an electrical connection that is unambiguous. If a mechanical switch is properly debounced, the on or off state of the switch can be exactly determined. With a touch sensitive switch, the switch state is less clearly defined.

In the implementation described in this application note, a switch event occurs when the presence of the user's finger on the switch causes a sufficient change in capacitance. It follows that maximizing the change in capacitance when a finger is present increases the reliability of detecting the switch event. The change in capacitance is mainly affected by the following parameters:

- The size, shape, and placement of the switch pattern on the PCB
- The type of material between the PCB trace and the user's finger
- The characteristics of the trace that connects the switch to the MCU

This section of application note compares the various options for the switch design and describes design rules for creating the most responsive touch-sensitive switch.

3.1.1. Testing Environment

In order to obtain the data in "3.1.2. Switch Capacitor PCB Pattern", tests were performed on a PCB with 12 different touch-sensitive switches. The switches were of different trace patterns and sizes. Figure 14 shows the test board.

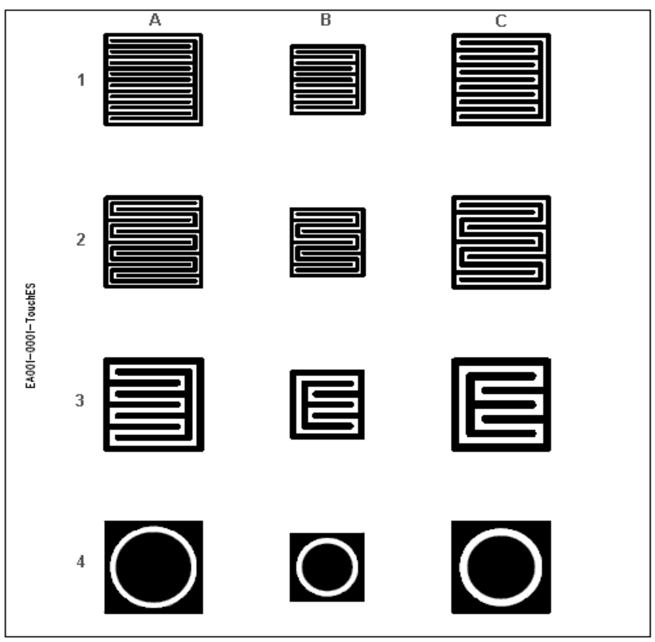

Figure 14. Test Board

The buttons are identified by column letters A through C and row numbers 1 through 4. The rest of this section and the appendices refer to the buttons by their column and row designation. The buttons in columns A and C are 20x20 mm, and the four buttons in column B are 15x15 mm.

The A and B column buttons have different outer dimensions, but in every row, the tracks and spaces between the tracks are identical. The B and C columns buttons have different outer dimensions, but their tracks and spaces between tracks are increased proportionally.

The goal of the tests was to determine how the switch geometry and materials affected the idle and pressed states of the switches. Another goal was to find switches with a higher idle capacitance because these switches should be less affected by parasitic capacitances on the board.

3.1.2. Switch Capacitor PCB Pattern

The two main aspects of the design that affect the switch capacitance are the size and shape of the switch, and the type and thickness of the material on top of the switch.

3.1.2.1. Effects of Switch Size and Shape

The measured idle capacitance of each button is provided in Table 2. These measurements include only the capacitance for the switch and exclude any parasitic capacitances.

Table 2. Idle Capacitance of the Switches in pF

	A	B	C
1	8.4	4.8	6.4
2	8.4	4.8	6.4
3	3.9	2.5	3.1
4	2.2	1.1	1.9

When comparing switches of the same size (rows A and C), it is clear that having more traces within the same area increases the idle capacitance on the switch. Comparing Row A to Row B indicates that the capacitance is directly proportional to the size of the switch.

3.1.2.2. Effects of Different Materials

In most products, the switches on the PCB are not directly exposed to the end user. They are typically covered by a layer of plastic or glass for aesthetic and protective reasons.

The test results in this section show the percentage change in capacitance from the idle state to the active state with a different material between the PCB and the user's finger. Five different materials of varying thickness were tested. The different materials along with their relative dielectric constant are provided in Table 3.

Table 3. Description of Materials Tested between the PCB and User's Finger

<table>
<tr><th>Material</th><th>Relative Dielectric Constant (?)</th><th>Thickness (mm)</th></tr>
<tr><td rowspan="3">Plexiglass</td><td rowspan="3">2.8</td><td>1.6</td></tr>
<tr><td>5.0</td></tr>
<tr><td>9.8</td></tr>
<tr><td rowspan="2">Glass</td><td rowspan="2">7.5</td><td>3.2</td></tr>
<tr><td>5.9</td></tr>
<tr><td rowspan="2">Mylar</td><td rowspan="2">3</td><td>0.35</td></tr>
<tr><td>0.7</td></tr>
<tr><td rowspan="2">ABS Plastic</td><td rowspan="2">2.3</td><td>2</td></tr>
<tr><td>4</td></tr>
<tr><td>FR4</td><td>4.5</td><td>1.6</td></tr>
</table>

Placing the material on top of the PCB has an effect on the idle capacitance on the switches. See "Appendix A—Effects of Different Materials on the Idle Capacitance of Each Switch" on page 19.

In order to reliably simulate the presence of a finger, a sheet of steel was used instead of a finger. The results from this artificial finger closely resemble the results from an actual finger. See "Appendix B—Percentage Change in Capacitance with an Artificial Finger Touching the Switch" on page 22.

The following points are the primary conclusions from the test data:

1. Across the different material types and thicknesses, the circular switches in row 4 created the greatest difference in capacitance between the idle and active states. The narrow track square switches in rows 1 and 2 had the smallest change in capacitance.
2. As expected, increasing the thickness of a certain type of material on the PCB lowers the change in capacitance from the idle to active state.
3. Material types with a higher dielectric constant create larger capacitors for the same switch type compared to materials with a lower dielectric constant.

For designers, these conclusions lead to the following design rules:

1. Use the circular pattern for the switches. Between the three circular switches in Row 4, switch 4C showed the greatest change in capacitance between the active and idle states, which makes the easiest and most reliable switch geometry.
2. Use the thinnest material possible to maximize the change in capacitance.
3. Use materials with a higher dielectric constant to increase the absolute capacitance of the touch switch. With a higher value of capacitance on the switch relative to other capacitances, such as the trace or other parasitic capacitances, the MCU will detect a higher change in overall capacitance between the idle and active states.

3.1.3 Effect of the Layer of the Board

The layer of the board on which the switch's traces are located also have an effect on the capacitance change. For the tests in "3.1.2. Switch Capacitor PCB Pattern", all of the switches were located on the top layer of the PCB. For aesthetic, routing, or other reasons, it is reasonable to put the switches on a different layer. Moving the switches farther from the top layer reduces the change in capacitance and makes it more difficult to recognize a switch event.

3.1.3. Trace Length

Another important factor in the switch's effectiveness is the capacitance of the trace that connects the switch to the MCU. The trace capacitance can be the largest source of parasitic capacitance for the switches, and too much parasitic capacitance can make the switches unusable. If the parasitic capacitance is too high, pressing a finger on the switch will not cause a sufficiently large change in the overall capacitance for the MCU to detect a switch event. The measurements in Appendix B show that a switch's capacitance typically ranges from 2 to 15 pF depending on the switch geometry and covering material. Appendix B also shows that the maximum change in capacitance with a finger present is about 3 pF (button 4C with a 0.35 mm layer of Mylar).

When designing the system, a safe baseline to use is that the MCU can detect a 0.5% change in overall capacitance. Care must be taken to minimize the parasitic trace capacitance so that the typical change in capacitance with a finger present is greater than 0.5% of the overall capacitance.

3.2. Effects of VDD Supply

Another design consideration is the voltage source for the MCU. The VDD voltage or VIO voltage for the MCU determines the logic high voltage for the asynchronous comparator output (CP0A) and also the CP0+ threshold voltages (1/3 VDD and 2/3 VDD). These voltages have direct influence on the charge and discharge profile for the switch capacitor, which, in turn, determines the comparison threshold.

In systems that have a stable voltage source from a voltage regulator or other non-varying supply, the minor changes in VDD during operation will have a negligible effect on the switch operation.

For systems that have a variable voltage supply, such as those that directly power the MCU from a battery without any regulation, the drop in VDD over time can affect the switch operation. The relaxation oscillator solution does have built-in insensitivity to a voltage drop because the range between the 1/3 VDD and 2/3 VDD is proportionally reduced with the CP0A charging and discharging voltage. However, the most reliable solution is to compensate the comparison thresholds based on the VDD voltage. This can be accomplished in two ways:

1. Store comparison threshold profiles for the different ranges of VDD. The VDD for the MCU can be monitored using the on-chip ADC and the absolute on-chip voltage reference.
2. Perform the periodic calibration described in "2.6.2. Performing a Periodic Calibration" on page 10.

3.3. Design Considerations

Based on the test results and limiting the design to commonly-used materials, such as glass, Plexiglass, and ABS plastic, the most effective combination of switch geometry and material is the circular switch 4C with a 1.6 mm layer of Plexiglass. Even though the circular switches had the lowest idle capacitance, it was still high enough to be greater than the parasitic capacitances and so the switch behavior was robust.

The exact dimensions of the circular switch are provided in Figure 15.

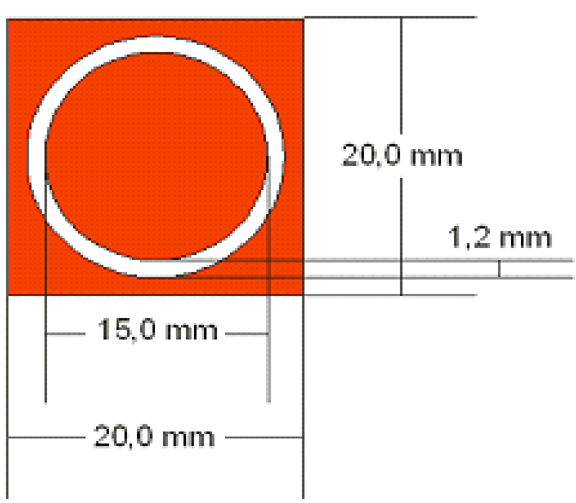

Figure 15. Dimensions of the Most Optimal Switch Geometry

To prevent switches from coupling with each other, the distance between the two adjacent switches should be at least 10 mm. If the distance is smaller than 10 mm, the detection is still possible, but a more sophisticated detection algorithm must be used.

If the switches are covered with a front panel, special care must be taken to assure stable, close contact between the front panel and the switches to avoid any gap between them, as the change in capacitance caused by the gap can appear as a switch active event.

For systems with multiple switches, Figure 16 shows an optimal layout. Red tracks are routed on the top layer. Keys are also placed on the top layer. Blue tracks are the ones routed on the bottom layer. This arrangement reduces the track's parasitic capacitance.

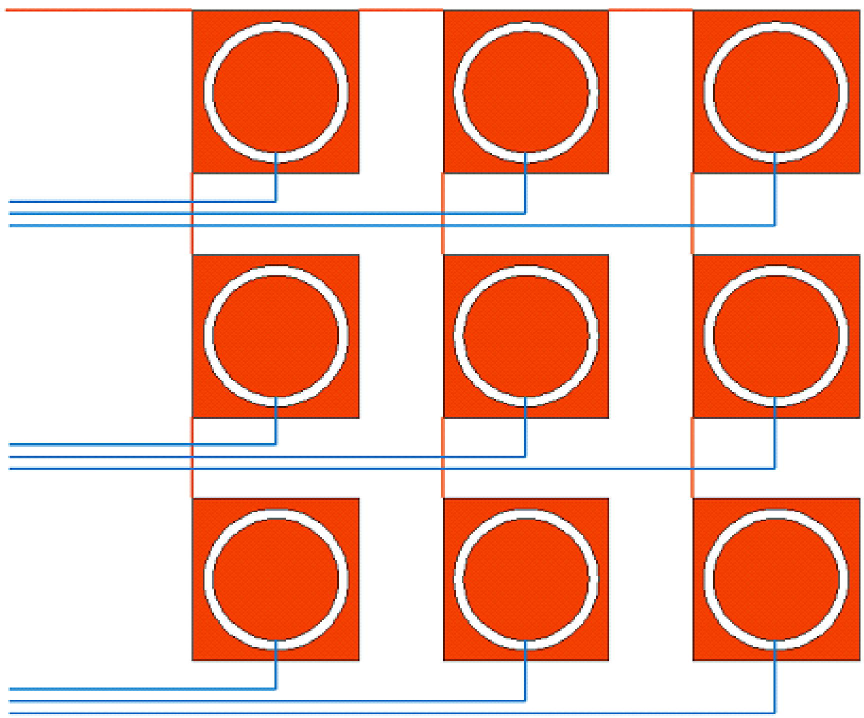

Figure 16. Optimal Keyboard Layout

In order to reduce the parasitic capacitance of the routing traces, the following recommendations should be followed:

- Use tracks no wider than 0.3 mm.
- Avoid routing signal tracks parallel to ground.
- Keep the distance between the signal tracks greater than 1 mm.
- Avoid routing the signal tracks over the ground plane.
- Avoid routing the signal tracks close to high-frequency or high-slew-rate circuits.

4. Example Code

Some of the examples included with this application note are written for the C8051F338-TB, which is available in the C8051F336-DK. This target board includes a single, touch-sensitive switch. Also included with the application note are examples for the ToolStick TouchSense DC and the C8051F930-TB.

This section highlights the various parts of one of the firmware examples, F338_TouchSense_MeasureFrequency.c, and describes how it works. The program is a simple example that performs an initial calibration for the comparison threshold and stores the result in Flash. When the touch-sensitive switch is pressed, an LED on the target board is turned on. When the switch is not pressed, the LED is turned off.

4.1. Example Program Flow Chart

The flow chart for the program is shown in Figure 17.

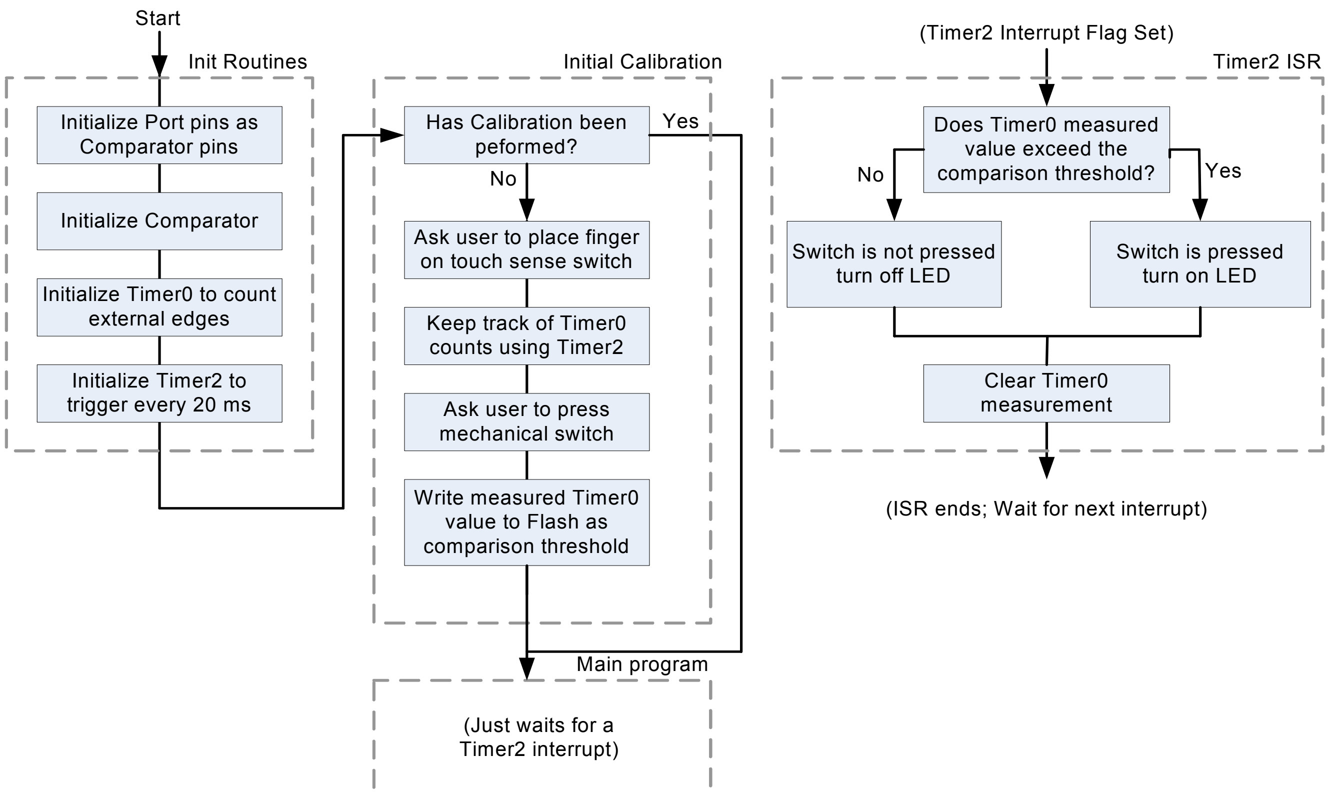

Figure 17. F338_TouchSense_MeasureFrequency.c Flow Chart

The initialization routines configure the port pins, comparator, and timers. As soon as the port pins and the comparator are initialized, the relaxation oscillator is functional. Timer0 is the timer that is gated by CP0A, and it counts the relaxation oscillator edges. Timer2 is the timer that is used to periodically interrupt the MCU and compares the Timer0 counter value to the comparison threshold.

The initial calibration is performed only if the comparison threshold is not set by a previous execution of the program. To perform the initial calibration, the firmware first stores the Timer0 value measured in the Timer2 ISR without a finger on the switch. Then, the user places their finger on the touch sensitive switch. With their finger still on the touch sensitive switch, the user presses the mechanical switch on target board. The Timer0 counts with the finger on the switch is averaged with the initial measurement, and this average is set as the comparison threshold.

The Timer2 interrupt service routine is the only regular overhead required by the solution. This function compares the Timer0 measured value to the comparison threshold. If the measurement is greater than the comparison threshold, the LED is turned off. If the measurement is less than the threshold, the LED is turned on.

4.2. Example Program Features

The F338_TouchSense_MeasureFrequency.c program requires about 400 bytes of code space to initialize the peripherals, perform the calibration and check the status of the switch.

The Timer2 ISR is the only function that needs to run periodically. The Timer2 ISR requires only 40 system clock cycles every 2 ms to make the Timer0 comparison and set the LED state. With the system clock running at 24.5 MHz, this configuration requires less than 0.01% of the system's available cycles. The rest of the system cycles are available for the other functions of the microcontroller.

In the example program, the microcontroller enters a low-power, idle mode while it is waiting for the Timer2 ISR to occur. In idle mode, the C8051F336–9 family of microcontrollers require only 4.4 mA (Typ) when running at 3.0 V and 24.5 MHz.

In the example, the switch is sampled every 20 ms. To increase the responsiveness of the switch, the sample time is configurable using the #define T2_OVERFLOW_RATE. If the sample time is increased to maximize the power savings, then care must be taken that the Timer0 counter does not overflow between samples.

5. Schematic of C8051F338-TB

The Touch sense portion of the C8051F338-TB schematic is shown in Figure 18. The full schematic for the C8051F338-TB is available in the C8051F336-DK User’s Guide.

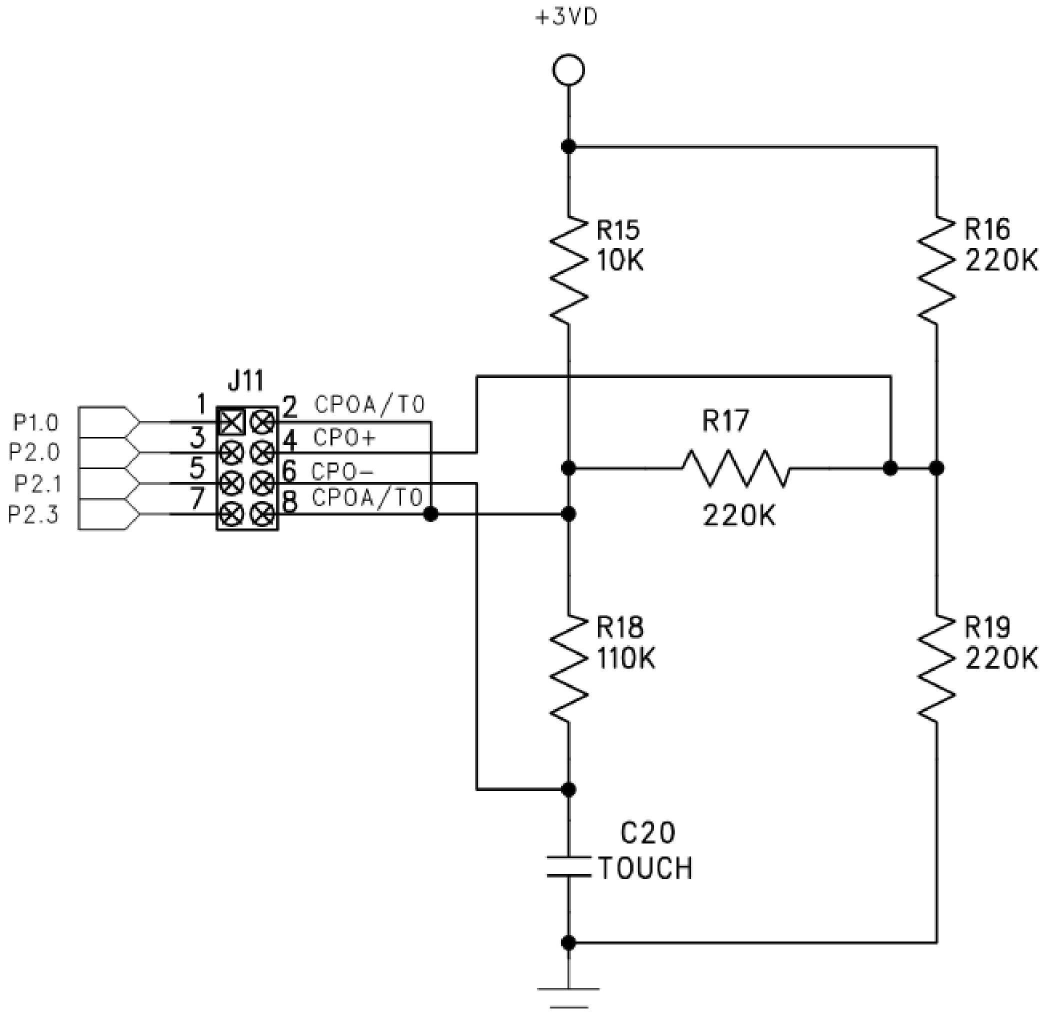

Figure 18. C8051F338 Touch Sense Schematic

Appendix A—Effects of Different Materials on the Idle Capacitance of Each Switch

The values in the following tables are the idle capacitances of the switches with different materials. These values exclude any parasitic capacitance. All values shown are in units of pF.

Table 4. Plexiglass 1.6 mm

	A	B	C
1	10.4	6.1	8.1
2	10.4	6.0	8.1
3	5.1	3.2	4.0
4	3.2	2.2	2.7

Table 5. Plexiglass 5.0 mm

	A	B	C
1	10.4	5.9	7.9
2	10.1	5.6	7.4
3	5.1	2.7	3.6
4	2.9	1.9	2.5

Table 6. Plexiglass 9.8 mm

	A	B	C
1	10.4	5.9	7.9
2	9.7	5.5	7.4
3	5.0	2.6	3.6
4	3.0	2.0	2.6

Table 7. Glass 3.2 mm

	A	B	C
1	14.0	7.3	9.9
2	11.0	6.0	8.5
3	6.1	3.1	4.4
4	4.0	2.7	3.5

Table 8. Glass 5.9 mm

	A	B	C
1	12.4	7.3	9.9
2	10.7	5.9	8.4
3	6.1	3.6	5.0
4	4.7	3.3	4.2

Table 9. Mylar 0.35 mm

	A	B	C
1	9.9	5.8	7.2
2	10.0	5.9	7.6
3	4.6	2.6	3.2
4	2.5	1.6	2.2

Table 10. Mylar 0.7 mm

	A	B	C
1	10.3	6.1	7.8
2	10.0	6.2	8.1
3	4.8	2.5	3.2
4	2.3	1.4	2.0

Table 11. ABS Plastic 2.0 mm

	A	B	C
1	9.4	5.5	7.4
2	9.3	5.2	7.1
3	4.6	2.7	3.4
4	2.8	1.9	2.4

Table 12. ABS Plastic 4.0 mm

	A	B	C
1	9.5	5.5	7.5
2	9.4	5.4	7.4
3	4.8	2.6	3.5
4	2.8	1.8	2.5

Table 13. FR4 1.6 mm

	A	B	C
1	10.9	6.3	9.2
2	11.1	6.3	9.7
3	6.1	3.3	4.7
4	3.3	2.4	3.2

Appendix B—Percentage Change in Capacitance with an Artificial Finger Touching the Switch

The values in the following tables indicate the measured capacitance and the percentage change in capacitance from idle with the artificial finger touching the switch. These values are derived using the idle capacitance of the switch and exclude any parasitic capacitance. All values shown are in units of pF.

Table 14. Plexiglass 1.6 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		**A**	**B**	**C**
1	10.9	6.4	8.6	**1**	4.6%	5.3%	5.6%
2	10.9	6.4	8.6	**2**	4.6%	5.9%	5.2%
3	5.6	3.5	4.6	**3**	9.3%	9.2%	12.2%
4	4.5	2.8	4.0	**4**	30.2%	21.3%	31.7%

Table 15. Plexiglass 5.0 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		**A**	**B**	**C**
1	10.6	6.0	8.0	**1**	1.9%	2.0%	2.3%
2	10.3	5.7	7.6	**2**	1.8%	2.0%	2.0%
3	5.3	2.8	3.7	**3**	3.0%	2.5%	3.2%
4	3.2	2.0	2.8	**4**	9.0%	5.5%	8.1%

Table 16. Plexiglass 9.8 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		A	B	C
1	10.5	5.9	8.0	**1**	0.9%	1.2%	1.2%
2	9.8	5.5	7.5	**2**	0.9%	1.2%	1.1%
3	5.0	2.7	3.6	**3**	1.4%	1.4%	1.4%
4	3.1	2.0	2.7	**4**	3.3%	2.2%	2.7%

SILICON LABS

Table 17. Glass 3.2 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		**A**	**B**	**C**
1	14.4	7.6	10.4	**1**	3.0%	4.1%	4.8%
2	11.3	6.2	8.9	**2**	3.2%	4.0%	4.0%
3	6.5	3.3	4.8	**3**	6.2%	6.0%	8.0%
4	5.3	3.2	4.6	**4**	23.8%	15.8%	23.3%

Table 18. Glass 5.9 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		**A**	**B**	**C**
1	12.7	7.5	10.1	**1**	2.0%	2.2%	2.8%
2	10.9	6.1	8.7	**2**	1.9%	2.6%	2.5%
3	6.3	3.7	5.2	**3**	3.4%	2.6%	3.4%
4	5.2	3.4	4.6	**4**	9.4%	4.9%	8.2%

Table 19. Mylar 0.35 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		**A**	**B**	**C**
1	11.9	6.9	8.8	**1**	17.5%	16.1%	19.1%
2	11.6	6.9	9.6	**2**	13.9%	15.1%	21.1%
3	6.6	4.0	5.1	**3**	29.9%	36.4%	37.8%
4	5.0	3.9	5.4	**4**	50.4%	58.3%	58.7%

Table 20. Mylar 0.7 mm

	Measured Capacitance (pF)			Percentage Change			
	A	**B**	**C**		**A**	**B**	**C**
1	11.0	6.6	8.7	**1**	6.6%	7.5%	10.7%
2	10.9	6.8	9.2	**2**	8.3%	9.0%	11.8%
3	6.0	3.2	4.7	**3**	20.2%	21.6%	31.0%
4	4.2	2.8	4.3	**4**	45.5%	49.4%	53.9%

Table 21. ABS Plastic 2.0 mm

	Measured Capacitance (pF)			Percentage Change			
	A	B	C		A	B	C
1	9.7	5.7	7.7	**1**	3.4%	4.7%	4.7%
2	9.7	5.5	7.5	**2**	3.9%	5.1%	5.0%
3	4.9	2.9	3.7	**3**	7.0%	7.2%	9.2%
4	3.6	2.3	3.1	**4**	23.4%	17.1%	22.3%

Table 22. ABS Plastic 4.0 mm

	Measured Capacitance (pF)			Percentage Change			
	A	B	C		A	B	C
1	9.7	5.7	7.7	**1**	2.0%	2.5%	2.7%
2	9.7	5.6	7.6	**2**	2.2%	2.6%	2.8%
3	5.0	2.7	3.6	**3**	3.6%	3.3%	4.1%
4	3.1	2.0	2.8	**4**	11.4%	6.9%	11.4%

Table 23. FR4 1.6 mm

	Measured Capacitance (pF)			Percentage Change			
	A	B	C		A	B	C
1	11.0	6.3	9.4	**1**	0.8%	1.1%	2.5%
2	11.1	6.3	9.8	**2**	0.6%	0.8%	1.1%
3	6.2	3.4	5.0	**3**	2.9%	2.1%	6.6%
4	4.1	2.7	4.3	**4**	18.6%	11.8%	26.1%

DOCUMENT CHANGE LIST

Revision 0.1 to Revision 0.2

- Added C8051F93x/92x content.
- Updated figures.

CONTACT INFORMATION

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, Texas 78701
Tel: 1+(512) 416-8500
Fax: 1+(512) 416-9669
Toll Free: 1+(877) 444-3032

Email: mcuapps@silabs.com
Internet: www.silabs.com

The information in this document is believed to be accurate in all respects at the time of publication but is subject to change without notice. Silicon Laboratories assumes no responsibility for errors and omissions, and disclaims responsibility for any consequences resulting from the use of information included herein. Additionally, Silicon Laboratories assumes no responsibility for the functioning of undescribed features or parameters. Silicon Laboratories reserves the right to make changes without further notice. Silicon Laboratories makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Silicon Laboratories assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Silicon Laboratories products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Silicon Laboratories product could create a situation where personal injury or death may occur. Should Buyer purchase or use Silicon Laboratories products for any such unintended or unauthorized application, Buyer shall indemnify and hold Silicon Laboratories harmless against all claims and damages.

Silicon Laboratories and Silicon Labs are trademarks of Silicon Laboratories Inc.

Other products or brandnames mentioned herein are trademarks or registered trademarks of their respective holders.