

**Introduction**

Capacitive touch sensors are commonly used for human control interfaces. Being a non-mechanical system, they do not wear out like mechanical switches and are relatively low cost.

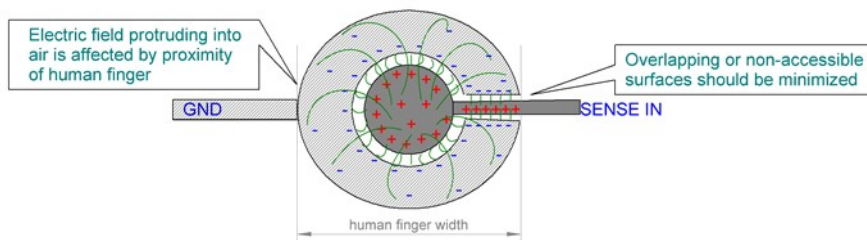
Capacitive touch systems do have their challenges however: the objective is to sense very small changes in capacitance and reliably give a definitive output, giving the user the equivalent to the “click” of a mechanical push-button.

Capacitive touch sensors are often required to be under layers of protective materials, so the effective change in the sensed capacitance could be quite small – on the order of a pF or so.

**Capacitive Interface Circuit**

The circuit in Figure 2 is used to sense changes in capacitance of an external touch pad interface. An inverter in GreenPAK drives an external feedback resistor to the input, causing an RC oscillation. The GreenPAK internally executes an algorithm based on changes in RC oscillation frequency to determine if a human finger touched the sensor.

For an effective sensor, the change in frequency must be non-negligible. Hence the resistor should be chosen such that the RC time constant is much larger than the internal propagation delays, but small enough for an adequately fast response.



**Figure 1. Circular Capacitive Touch Pad**

Figure 1 shows a common sensor layout for a non-contact touch sensor. Rounded geometries help distribute the electric fields evenly unlike angular shapes which can concentrate electric field at the corners. The conductive plates lie flat to allow a significant portion of the electric field to arch out into the air, maximizing the volume in which a human finger can effectively interfere.

During layout, minimize any stray capacitances (such as trace and ground capacitance from the sensor to the GreenPAK) as they only add to total capacitance and thus decrease sensitivity.

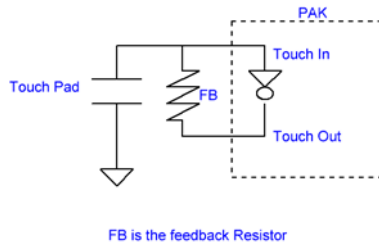


Figure 2. Capacitive Interface Circuit

### GreenPAK Design: Improving the Sensing Algorithm

In a previous application note, **AN-1077 Capacitive Touch Button Sensor**, the GreenPAK algorithm involved comparing the RC oscillation frequency with a reference RC. The issue with that design lies with guaranteeing performance over temperature, voltage, and part to part variation.

In this improved design, the GreenPAK compares the RC oscillation frequency with itself, not a reference RC. Using a differential instead of absolute measurement eliminates the need for a reference RC. Not only are fewer external components needed, we get more robust performance with less variables to account for.

The improved algorithm can be implemented with any chip in the GreenPAK4 family. We are using a SLG46140 in this example.

### Implementation

In order to compare the current value with the previous value, we use the FSM counter to count up one cycle and count down the next. If the frequency did not change, then the difference would be close to zero. If there was a change, then the difference would be non-zero (positive for touch on and negative for touch off).

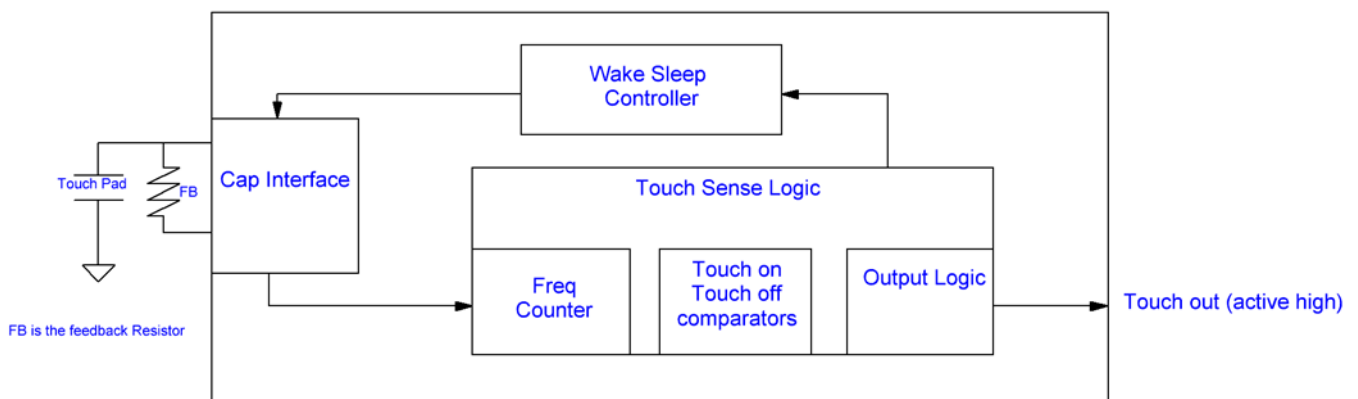


Figure 3. PAK Design Block Diagram

The initial finite state machine FSM counter value is set to 128 (half of 255, the digital comparator's max value). Therefore, the differential measurement would result in a counter value is greater than or less than 128.

After the differential measurement phase, the digital comparators check for a positive  $>128 + \alpha$  or negative  $<128 - \beta$  value, where  $\alpha$  and  $\beta$  are user-defined for hysteresis. If either comparator detects an event, the output will toggle to the corresponding state.

For example, during the count up phase, the sensor resonates at 100kHz. But during the count down phase, the user touches the sensor and the input frequency decreases to 80kHz. If the input to the digital comparators is greater than  $128 + \alpha$ , the output goes high, signifying a touch on event. Similarly, if the opposite happened and the input to the digital comparators is a value below  $128 - \beta$ , then DCMP's output (OUT+) becomes low, signifying a touch off event.

Steps:

1. Wake. FSM counts up for 5ms
2. Sleep. FSM keep for 45ms
3. Wake. FSM counts down for 5ms
4. Sleep. FSM keep for 45ms
5. Compared and Reset

Figures 4, 5 and 6 depicts the sequence of events for "no touch", "finger touch on" or "finger touch off (release)" detection. At the first wake cycle, the FSM Counter Data increases at each rising clock of the capacitor input, then proceeds to sleep. At the next wake cycle, the FSM Counter Data decreases at each rising clock of the capacitor input and then proceeds to sleep.

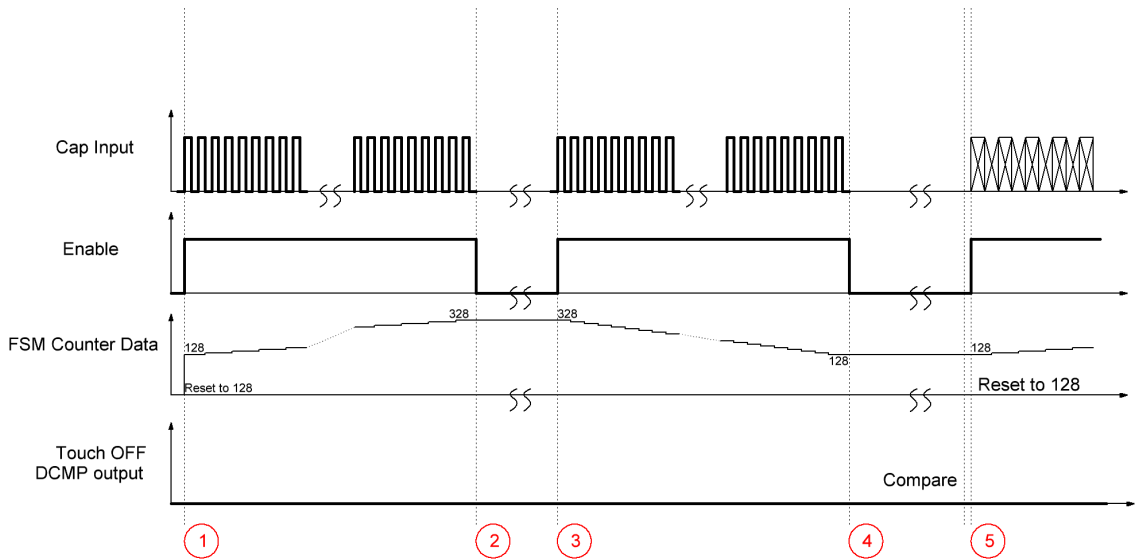
Before the next Wake cycle, the FSM counter data is compared to two digital values, the touch ON threshold and the touch OFF threshold. Once the comparison is complete, the counter values are reset to the initial value and the cycle repeats. If a Touch ON is detected, meaning the counter value is above the touch ON threshold, the wake/sleep is bypassed, and the part stays awake until a Touch OFF is detected. This prevents the output from being "stuck on" due to a missed Touch OFF event.

**Case 1: No Touch**

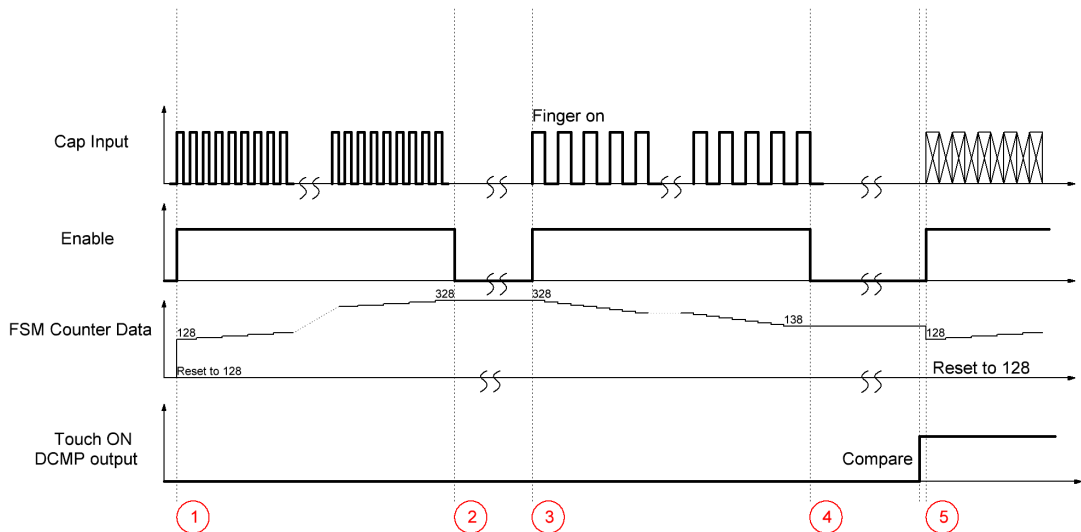
At the undisturbed frequency, the FSM counter Data at step 5 is equal to the initial value.

**Case 2: Finger Touch On**

Due to the presence of a Finger, the FSM Counter Data at step 5 is greater than the touch ON threshold. This is because the counter counts down fewer times than it counted up. The result is a positive difference between the starting and ending value.



**Figure 4. No Touch sequence of events (case 1)**

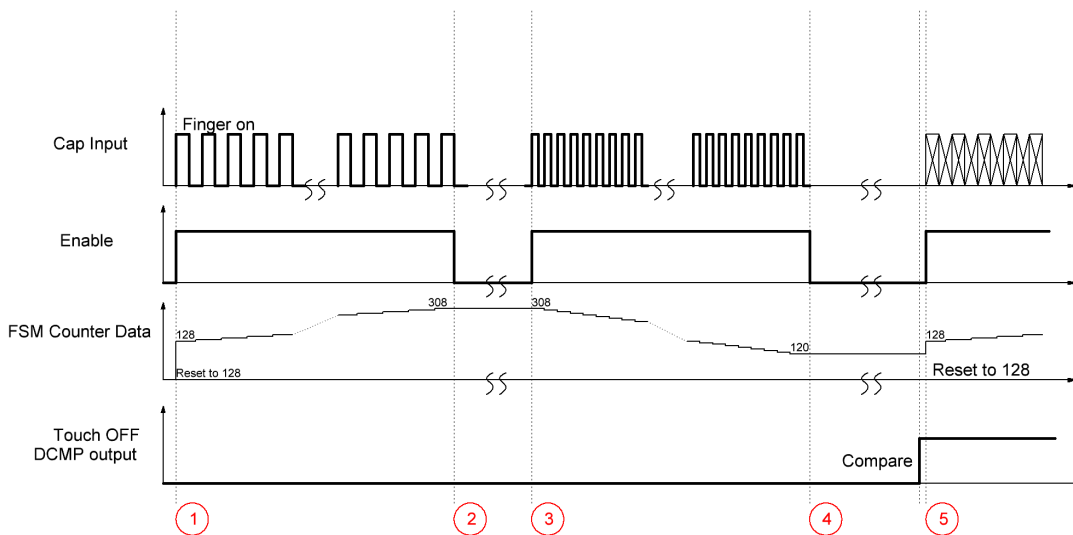


**Figure 5. Touch On sequence of events (case 2)**

**Case 3: Finger Touch Off (release)**

When touch is released, the FSM counter data at step 5 is less than the touch OFF threshold. This is because the counter counts down more times than it counted up. The result is a negative difference between the starting and ending value.

A “smart” wake sleep was implemented where a touch on would keep the oscillator circuit running constantly until there is a touch off event. This essentially increases the sensitivity to touch off and less to touch on. If there is a touch off event, then the system will resume the normal wake sleep cycle.



**Figure 6. Touch Off sequence of events (case 3)**

**Design considerations: Wake Sleep and Resolution**

Capacitive touch sensors are active components which need to constantly monitor its input. This can consume a significant amount of current if it’s running all the time. To achieve a longer battery life, PAK needs to turn on and off the capacitive touch interface in a wake sleep fashion. For this design, we used the setup in Figure 7 to gate the oscillator circuit at 50ms per cycle and a 5ms wake.

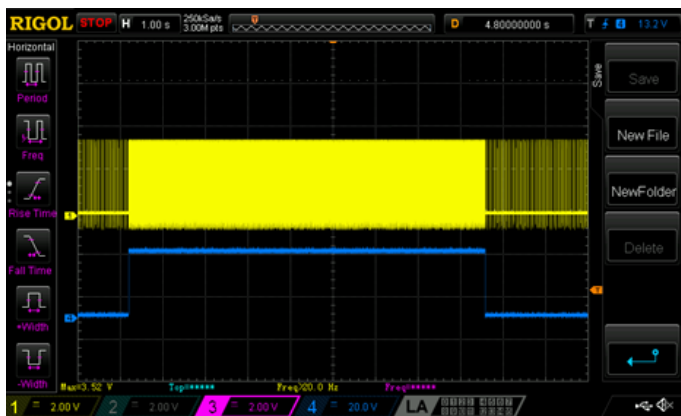
The wake sleep periods can be changed as pleased, but it is important to keep the resolution in mind. A longer wake means a greater amount of resolution at the cost of current. To calculate resolution, multiply the wake period by the input frequency. For example, if the input frequency is 100kHz and wake period is 2ms, then the total resolution is 200 counts per sample.

$$Resolution\ per\ Sample = Freq * Wake\ Period$$



Channel 1 (yellow/top line)- PIN#3 (Feedback to Touch)

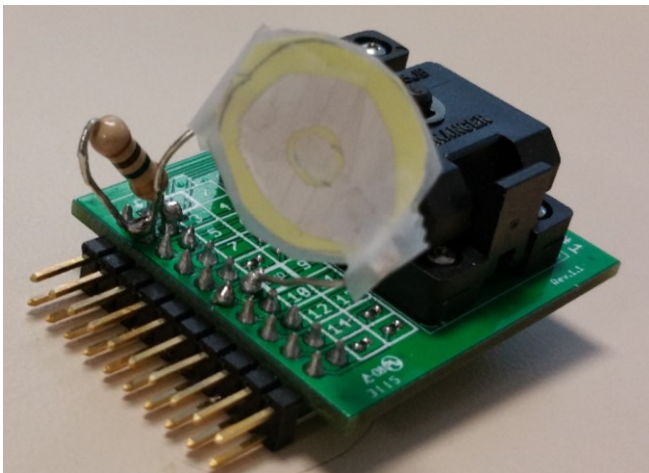
Channel 2 (dark blue/2<sup>nd</sup> line): PIN#7 (Touch Out)



**Figure 8. Waveform Capture of GreenPAK outputs Touch Out, and internal inputs Wake Sleep**

## Conclusion

An improved capacitive touch sensing algorithm using relative thresholds instead of absolute can be implemented with any GreenPAK4 device. The solution minimizes external circuitry while achieving more reliable performance that is less sensitive to component and environmental variations.



**Figure 9. Prototype of Touch Sensor**

## IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01 Jan 2024)

### Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,  
Koto-ku, Tokyo 135-0061, Japan  
[www.renesas.com](http://www.renesas.com)

### Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

### Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit [www.renesas.com/contact-us/](http://www.renesas.com/contact-us/).