

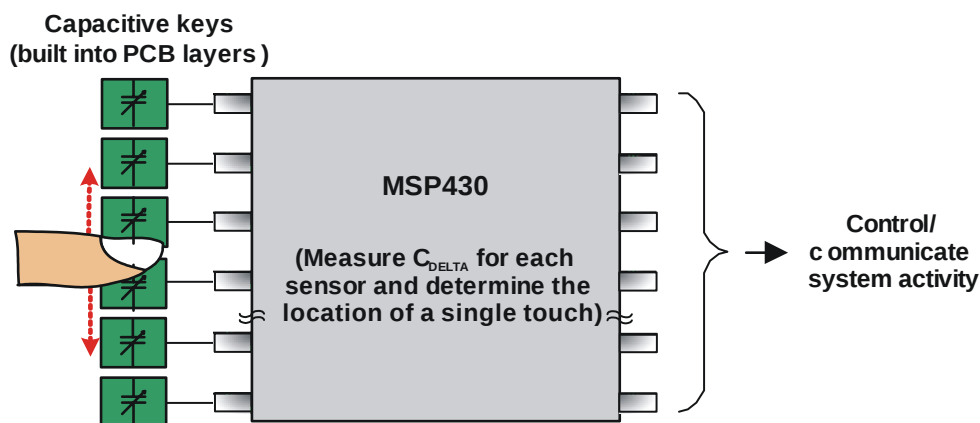
## ***PCB-Based Capacitive Touch Sensing With MSP430***

Zack Albus

*MSP430 WW Applications*

### **ABSTRACT**

This application report discusses the design of a single-touch capacitive sensor interface using the MSP430 microcontroller. With ultra-low power features and integrated peripherals, integrating a single-touch user interface into MSP430 applications can be readily accomplished. This application report provides an overview of the technology, details about system careabouts and details for different methodologies of capacitive touch sensing implementations using the MSP430 family. Figure 1 shows a graphical high level representation of the system described in detail in the remainder of the document.



**Figure 1. Capacitive Touch Sensor System Overview Using the MSP430**

## Contents

|                                                                         |                                                                   |           |
|-------------------------------------------------------------------------|-------------------------------------------------------------------|-----------|
| <b>1</b>                                                                | <b>Capacitive Touch Sensing Overview .....</b>                    | <b>3</b>  |
| <b>2</b>                                                                | <b>Sensor and Interface Construction .....</b>                    | <b>4</b>  |
| 2.1                                                                     | PCB Sensor Specifics .....                                        | 4         |
| 2.2                                                                     | Sensor Insulating Overlay .....                                   | 6         |
| <b>3</b>                                                                | <b>Measuring a Capacitive Touch Sensor Using the MSP430 .....</b> | <b>8</b>  |
| 3.1                                                                     | Oscillator-Based Capacitive Measurement .....                     | 8         |
| 3.2                                                                     | Resistor-Based Capacitive Measurement .....                       | 11        |
| <b>4</b>                                                                | <b>Software Implementation .....</b>                              | <b>15</b> |
| 4.1                                                                     | Tracking Sensor Base Capacitance .....                            | 15        |
| 4.2                                                                     | Implementing Button Function.....                                 | 17        |
| 4.3                                                                     | Implementing Slider Function.....                                 | 17        |
| 4.4                                                                     | Handling Slider Endpoints .....                                   | 19        |
| 4.5                                                                     | Sensor Multiplexing for Sliders.....                              | 20        |
| <b>5</b>                                                                | <b>Summary.....</b>                                               | <b>22</b> |
| <b>6</b>                                                                | <b>References.....</b>                                            | <b>22</b> |
| <b>Appendix A. Flexible Evaluation and Demonstration Hardware .....</b> |                                                                   | <b>23</b> |
| <b>Appendix B. Simple Demonstration Hardware.....</b>                   |                                                                   | <b>24</b> |

## Figures

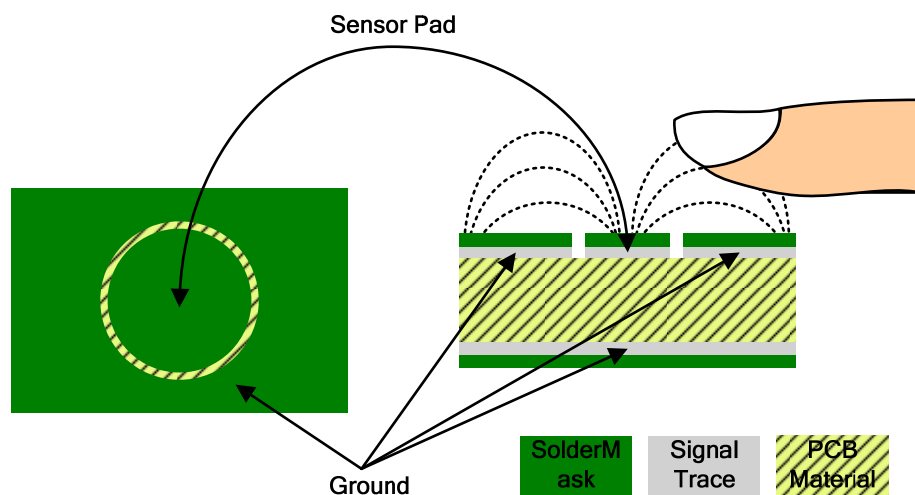
|                   |                                                                                                 |           |
|-------------------|-------------------------------------------------------------------------------------------------|-----------|
| <b>Figure 1.</b>  | <b>Capacitive Touch Sensor System Overview Using the MSP430 .....</b>                           | <b>1</b>  |
| <b>Figure 2.</b>  | <b>Open Capacitor Acting as a Sensor .....</b>                                                  | <b>3</b>  |
| <b>Figure 3.</b>  | <b>Plate Capacitor Basics.....</b>                                                              | <b>4</b>  |
| <b>Figure 4.</b>  | <b>Example Four-Sensor System for Button and Slider Function .....</b>                          | <b>5</b>  |
| <b>Figure 5.</b>  | <b>Pour Styles (Red = Top Signal Layer, Blue = Bottom Signal Layer = GND Pour) .....</b>        | <b>5</b>  |
| <b>Figure 6.</b>  | <b>Theoretical and Empirical Sensor Sensitivity vs Insulator Thickness.....</b>                 | <b>7</b>  |
| <b>Figure 7.</b>  | <b>Basic Capacitive-Dependent Relaxation Oscillator .....</b>                                   | <b>8</b>  |
| <b>Figure 8.</b>  | <b>Frequency Measurement Principle.....</b>                                                     | <b>9</b>  |
| <b>Figure 9.</b>  | <b>Multi-Sensor System Using Comparator_A+.....</b>                                             | <b>9</b>  |
| <b>Figure 10.</b> | <b>Current Consumption and Measurement Time for One Sensor .....</b>                            | <b>10</b> |
| <b>Figure 11.</b> | <b>Basic Resistive Discharge Capacitance Measurement .....</b>                                  | <b>11</b> |
| <b>Figure 12.</b> | <b>Measurement Methodology Using the GPIO Threshold and Timer_A .....</b>                       | <b>12</b> |
| <b>Figure 13.</b> | <b>Multi-Sensor Charge/Discharge Configuration .....</b>                                        | <b>13</b> |
| <b>Figure 14.</b> | <b>Single Measurement Cycle for Improved Noise Rejection .....</b>                              | <b>13</b> |
| <b>Figure 15.</b> | <b>Resistive Charge/Discharge <math>I_{cc}</math> and Measurement Time: Single Sensor .....</b> | <b>14</b> |
| <b>Figure 16.</b> | <b>Example Algorithm for Tracking Baseline Sensor Capacitance.....</b>                          | <b>16</b> |
| <b>Figure 17.</b> | <b>Button Press in a Four Key System .....</b>                                                  | <b>17</b> |
| <b>Figure 18.</b> | <b>Fundamental and Neighboring Sensor Response .....</b>                                        | <b>18</b> |
| <b>Figure 19.</b> | <b>Example Slider Position Determination Methodology.....</b>                                   | <b>18</b> |
| <b>Figure 20.</b> | <b>Representation of Count Measurement Results for Max Endpoint.....</b>                        | <b>19</b> |
| <b>Figure 21.</b> | <b>Example Slider Endpoint Handling.....</b>                                                    | <b>20</b> |
| <b>Figure 22.</b> | <b>Example Sensor Multiplexing of 12 Capacitive Sensor Elements.....</b>                        | <b>21</b> |
| <b>Figure 23.</b> | <b>Flexible 4-Button RO/RC System Block Diagram.....</b>                                        | <b>23</b> |
| <b>Figure 24.</b> | <b>Simple 4-Button RC System Block Diagram and Image.....</b>                                   | <b>24</b> |

## Tables

|                 |                                                |           |
|-----------------|------------------------------------------------|-----------|
| <b>Table 1.</b> | <b>Dielectrics for Example Materials .....</b> | <b>6</b>  |
| <b>Table 2.</b> | <b>64-Position Slider Key .....</b>            | <b>18</b> |

## 1 Capacitive Touch Sensing Overview

The fundamental element required in the capacitive touch sensing application described is the variable capacitor itself. This capacitor should be easy to construct as well as sensitive to human touch in order to enable this as an alternative to mechanical buttons and switches. Such a touch sensitive sensor element can be constructed by “opening up” a capacitor structure so that the electric field can be interfered with by a conductive foreign object, in this case, a finger. Figure 2 shows the top and cross-sectional views of such a capacitive sensor as implemented in the printed circuit board itself.



**Figure 2. Open Capacitor Acting as a Sensor**

As shown, a PCB-based capacitor is formed between the center copper pad and the ground pour surrounding it. The electric field is allowed to leak into the area above the capacitor. The interaction of this sensor pad and the surrounding ground pour (also the ground plane underneath) create a baseline capacitance that can be measured. The base capacitance of such a sensor is in the range of ~10 pF for a finger-sized sensor. When a conductor, e.g., a finger, comes into the area above the open capacitor, the electric field is interfered with causing the resulting capacitance to change. The coupling of the conductive finger into the capacitive sensor increases the capacitance of the structure beyond the baseline capacitance, the capacitance of the sensor with no touch. By continuously measuring the capacitance of the sensor(s) in the system and comparing each result to a predetermined baseline capacitance, the system microcontroller can determine not only on/off button functions for each sensor element but also “amount” of press used for more complex interfaces such as positional sliders.

The sensitivity of this sensor is dependent on the gap between the surrounding ground and the sensor plate. A gap of around 0.5 mm is recommended. In addition, PCB thickness plays into the overall sensitivity as well: when it is very thin as in the case of a flexible PCB, this increases the tight coupling between the sensor and the ground plate beneath it and decreases its sensitivity. A standard FR4 PCB with 1-mm to 1.5-mm thickness is ideal.

The sensor pad size of around 10-mm diameter is typically used. This size is similar to the surface area of a human finger when pressed down. Such a sensor using the above careabouts typically has ~5 pF to 10 pF of capacitance untouched.

The highlighted ground plane underneath the sensor aids in shielding it from potential interference generated by other electronics in the system. It also helps to maintain a more constant baseline capacitance needed as a reference for each measurement.

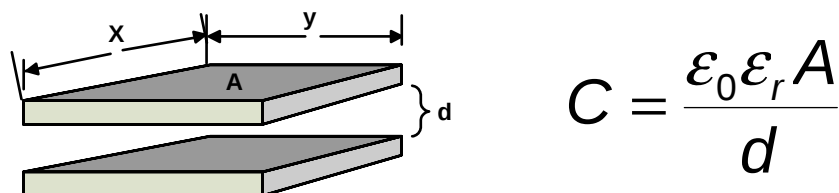
The base capacitance of such a design is affected by stray capacitances on the PCB as well as potentially other environmental effects such as temperature and humidity. Therefore, the detection system needs to constantly monitor and track this variation for correct comparison to touch events.

## 2 Sensor and Interface Construction

The complete interface consists of the actual PCB-based capacitive sensors themselves, as well as some type of insulator between the sensors and the user.

### 2.1 PCB Sensor Specifics

At a high level, the capacitive sensor dependencies can be visualized by understanding the basics of a plate capacitor. Figure 3 represents the key elements.



**Figure 3. Plate Capacitor Basics**

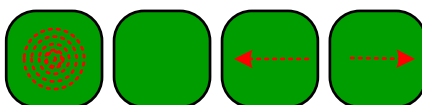
First the base capacitance must be accounted for. The term “base capacitance” refers to the measurement result of an “untouched” or uninfluenced sensor element. For simplicity, the base capacitor can be assumed to be constructed from the sensor pad on the topside of the PCB and the ground pour on the bottom side of the PCB. These are the top and bottom plates in Figure 3.

The PCB itself makes up the “d” in the equation. As mentioned earlier, as d gets smaller (such is the case with flex PCBs), the baseline capacitance increases resulting in reduced sensitivity. The permittivity of free space ( $\epsilon_0$ ) and the material ( $\epsilon_r$ ) define the dielectric constant of the PCB insulator and will affect the ultimate base value.

The area of the sensor, “A”, is typically limited to the size of the interacting finger. Usually this is designed to be somewhere between a child’s small fingers and an adult’s larger fingers for a good compromise, but is ultimately application-dependent. Keep in mind that any sensor area that extends outside of the overlap of the finger is essentially wasted, as it does not contribute to the changing capacitance desired.

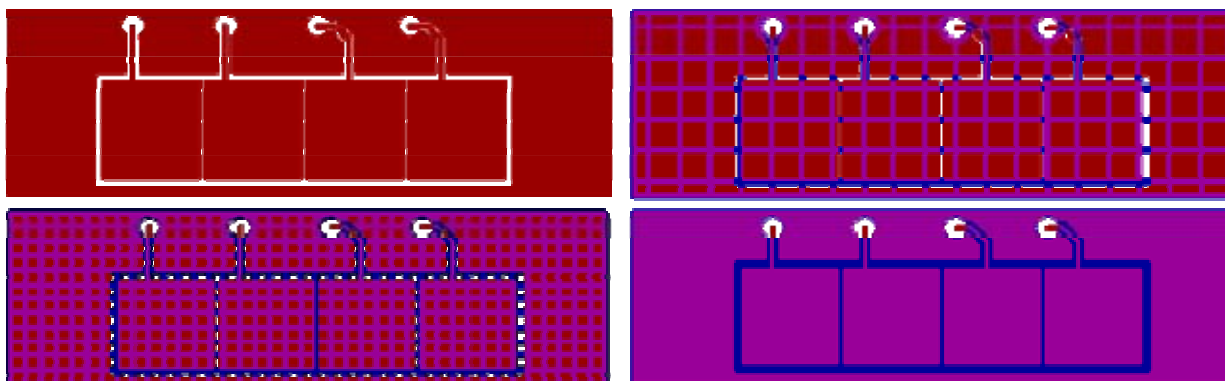
The design careabouts for the capacitive sensor are simple in theory: minimize the base capacitance of the sensor while maximizing the potential for user interaction. The more ideal that each of these are made results in increased capacitive change between a touched and untouched sensor, the key to good sensitivity and robust design. Of course, these two goals works against each other: as the area gets larger to match to the full size of the interacting finger the base capacitance also increases, as it is proportional to  $A$ . For a given sensor construction, the maximum change that can be created by a finger press is essentially fixed. As the base capacitance for the same sensor increases, the percent change in measured capacitance goes down, resulting in lower sensitivity and overall performance of the sensor interface.

One solution to help in this effort, given a fixed  $A$  for the sensor, is to manage the ground pour underneath the sensor pad. Figure 4 shows the simple orientation of a four key sensor interface for single-touch functions such as buttons or sliders. This four key sensor is used as an example throughout the application report.



**Figure 4. Example Four-Sensor System for Button and Slider Function**

Figure 5 shows the actual PCB layout of such a sensor key pad, but with four different approaches to implementing the ground pour.



**Figure 5. Pour Styles (Red = Top Signal Layer, Blue = Bottom Signal Layer = GND Pour)**

The upper left image shows only a top signal layer: four sensor pads surrounded by a top layer ground pour; no bottom layer is implemented. The upper right section shows the same board design; except now a bottom layer 25% hatched ground pour is implemented. The lower left version is with a 50% pour and the lower right with a 100% filled ground pour, each below the sensors constructed in the top layer.

At least some bottom layer ground pour is recommended beneath each sensor in order to isolate the sensor elements from noise and external variation that could affect the sensor based capacitance. While the obvious choice might be a 100% fill as shown in the lower left implementation providing maximum noise isolation, it also maximizes the area of the lower plate in the capacitor constructed between the sensor and ground pour. This increases the base capacitance through an increased area,  $A$ . To get the benefits of noise isolation as well as minimized base capacitance, a fill in the order of 50% to 75% is typical.

## 2.2 Sensor Insulating Overlay

In this type of application, an insulating layer, typically plastic, overlays the sensors on the PCB from the user. Therefore, the finger does not make physical contact with the sensor plate itself. Maintaining an insulated separation between sensor and the user is critical in maximizing the usefulness of the capacitive touch interface.

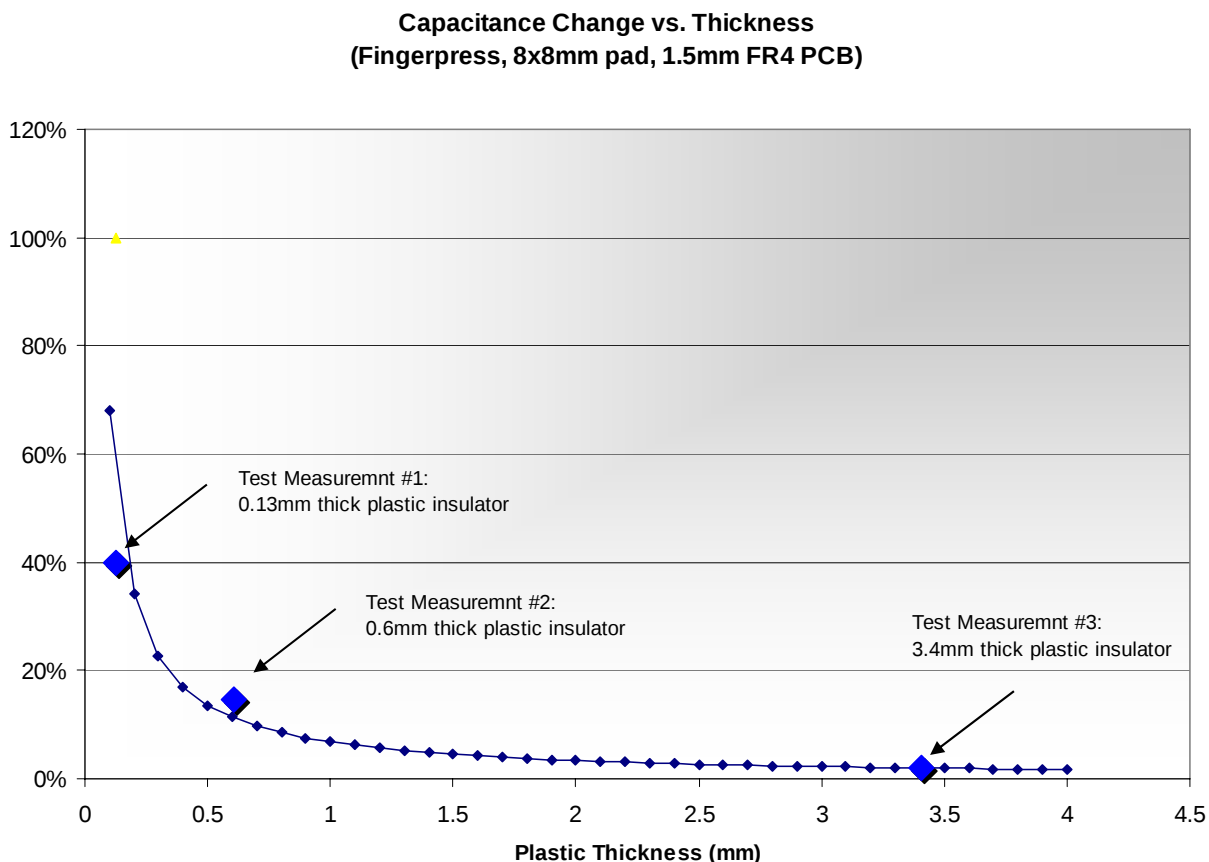
The plate capacitor concept from Figure 3 used to describe the base capacitance of the sensor can also be used to visualize the interaction of the finger in changing the capacitance. In this case, the sensor is now the bottom plate of the capacitor and the user's finger is the top conductive plate. The insulator between the two is at a minimum the solder mask covering the sensors or most likely the plastic housing the given sensor system is contained within. It becomes clear that as the area of interaction increases to the full size of the finger, an increasing  $A$ , the change in capacitance is maximized. Also, as the  $d$  of the insulator increases, the change in capacitance goes down in an inversely proportional relationship. A key factor that cannot be ignored is the actual material used as the user interface insulator. The dielectric constant of this material as well as its thickness play a very large role in dictating the sensitivity and usability of a given capacitive touch sensor. Table 1 highlights the dielectric constants ( $\epsilon_0 \times \epsilon_r$ ) of some example materials, plastics generally being in the 2-3 range.

**Table 1. Dielectrics for Example Materials**

| <b><i>Material</i></b> | <b><i>Dielectric Constant</i></b> |
|------------------------|-----------------------------------|
| <b>Vacuum</b>          | <b>1 (by definition)</b>          |
| <b>Air</b>             | <b>1.00054</b>                    |
| <b>Polyethylene</b>    | <b>2.25</b>                       |
| <b>Paper</b>           | <b>3.5</b>                        |
| <b>Pyrex glass</b>     | <b>4.7</b>                        |
| <b>Rubber</b>          | <b>7</b>                          |
| <b>Silicon</b>         | <b>11.68</b>                      |

In addition to the insulator itself, the connection between the insulator and the sensor is critical. Given the low dielectric constant of air, any gap in the coupling of the insulating overlay and the sensor results in a rather poor capacitive change. Assuring that the connection between the two elements is as good as possible is important to maintaining uniform sensitivity to touch. Adhesives are commonly used to achieve this but, in addition to being thin and very tactile, they must also respond to environmental changes with minimum shift in thickness or adhesiveness. Nonconductive adhesive films such as 467MP and 468MP from 3M have resilient properties for such an application.

To illustrate the impact of insulator thickness on sensor sensitivity, Figure 6 plots the theoretical relationship as the  $d$  of the user interface insulator increases. Plotted along the continuous curve are three empirical data points for specific tests using the specified thickness of plastic.



**Figure 6. Theoretical and Empirical Sensor Sensitivity vs Insulator Thickness**

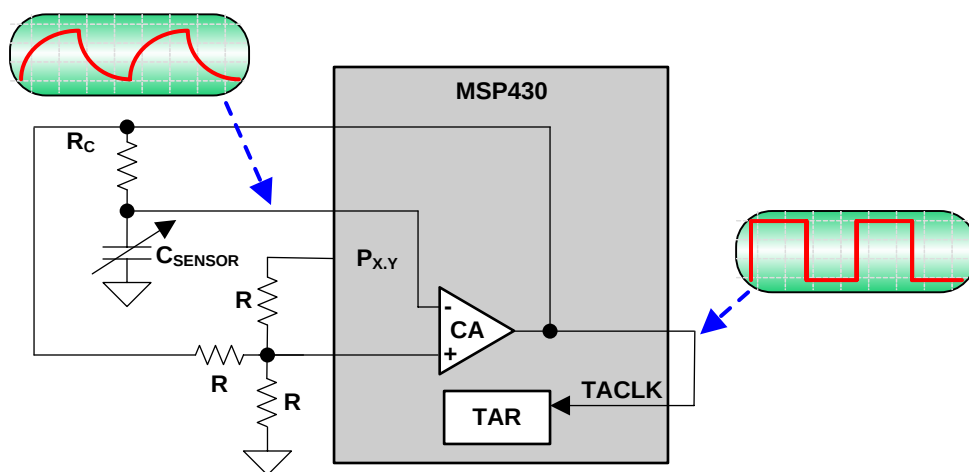
By understanding the physical and electrical tradeoffs for both the PCB sensor and the user interface insulator, performance expectations and feasibility of the system can be balanced to meet the given application's requirements. Ultimately, mechanical and material constraints drive the limits of a given implementation.

### 3 Measuring a Capacitive Touch Sensor Using the MSP430

Now that the sensor concepts and construction are defined, two different methods used to measure the capacitive touch sensor can be described. Implementations and performance results are detailed in each case along with a comparison and discussion of tradeoffs for each method.

#### 3.1 Oscillator-Based Capacitive Measurement

The first methodology used to measure a capacitive touch sensor described is using an oscillator. Fundamentally, a simple relaxation oscillator can be created using the MSP430's on-chip comparator and the capacitive sensor as the tuning element. Any change in capacitance of the sensor corresponds to a change in frequency which can be measured using the internal Timer\_A hardware of the MSP430. Figure 7 shows the implementation of such a system using an MSP430 with the Comparator\_A peripheral.

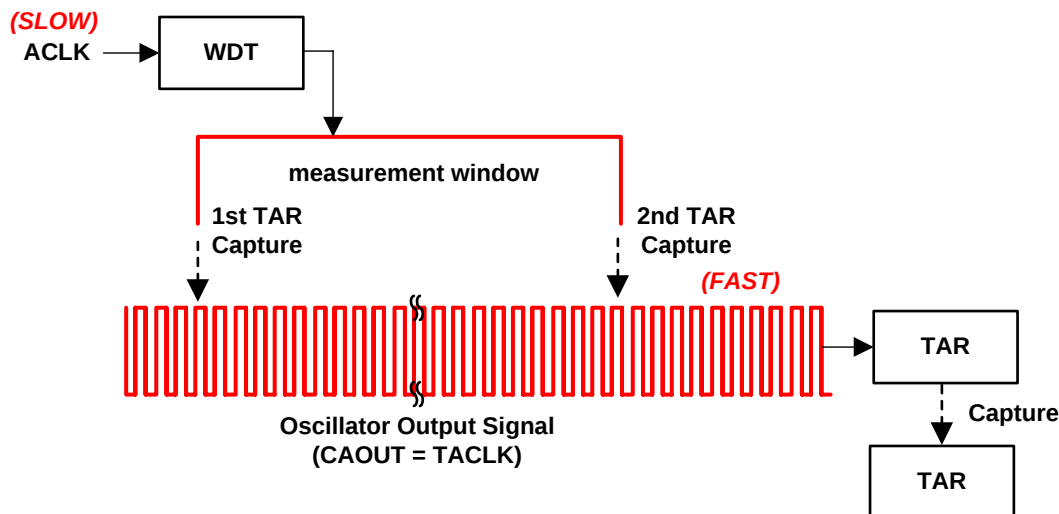


**Figure 7. Basic Capacitive-Dependent Relaxation Oscillator**

The R ladder network, when P<sub>x,y</sub> is high, creates a reference for the comparator that changes with its output. This toggling reference is opposite in polarity to the charge or discharge of the sensor capacitor ( $C_{\text{SENSOR}}$ ), resulting in a continuous oscillation. With equal R's in the ladder network providing  $1/3V_{\text{CC}}$  and  $2/3V_{\text{CC}}$  trip points, the frequency of oscillation is given by:

$$f_{\text{OSC}} = 1/[1.386 \times R_C \times C_{\text{SENSOR}}]$$

By counting the oscillation periods over a fixed time duration, the frequency can be determined and the capacitance measured. For this application report, a sensor resistor,  $R_C$ , of 100k is used. This results in approximately a 600-kHz oscillation frequency for a typical ~10-pF sensor. Figure 8 shows this counting principle as it can be implemented in the MSP430 hardware.

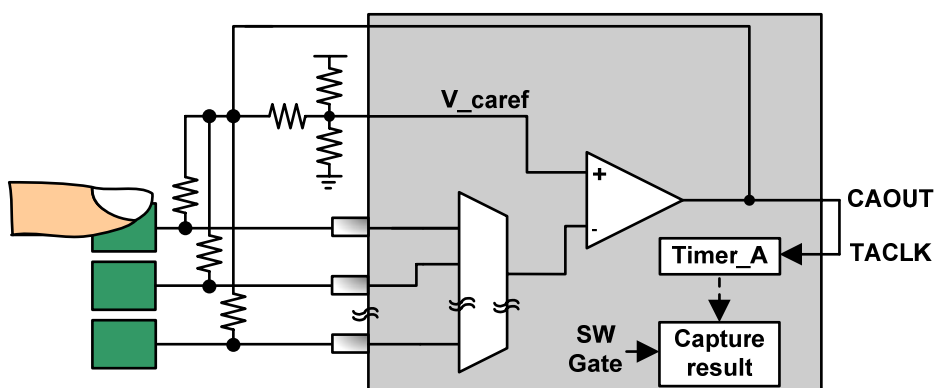


**Figure 8. Frequency Measurement Principle**

Figure 8 show use of the very slow ACLK signal, in this case the 12-kHz integrated VLO, to clock the WDT to create a measurement window. With each WDT interrupt, the CPU in software takes a snapshot of the continuously counting Timer\_A register, TAR. The difference between two of these snapshots, or captures, is the measurement result.

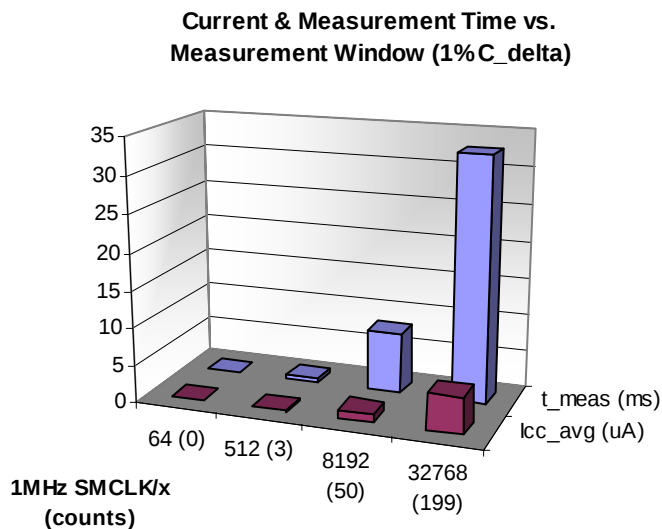
In reality, the actual capacitance is of no interest, only the change in capacitance between the baseline measurement and a touched sensor. To determine this only the actual number of counts captured during the measurement window is of any importance. By storing a base capacitance count used for comparison of future measurements, the relative change in capacitance can be determined.

While one sensor can be easily measured using Comparator\_A as shown in Figure 7, to implement multiple sensors using the oscillator method requires use of Comparator\_A+ enabled devices. The analog multiplexer built into Comparator\_A+ allows for multiple capacitive sensors to independently be measured using the oscillator technique. As shown in Figure 9, a multi-sensor system using Comparator\_A+ can resolve the position of a single touch along a slider.



**Figure 9. Multi-Sensor System Using Comparator\_A+**

Using one external resistor per sensor and three additional resistors for the comparator reference, a simple multi-sensor system can be realized using the MSP430. With 100k reference resistors enabled as needed by a free GPIO, a sensitive yet ultra-low power capacitive touch interface system can be created. Figure 10 shows the average current consumption and measurement time for a single capacitive sensor using this methodology.



**Figure 10. Current Consumption and Measurement Time for One Sensor**

Each measurement above is made for a 1% change in sensor capacitance. While small, this is not an uncommon amount of change for many systems due to mechanical constraints. As the measurement window is increased using a larger WDT divider resulting in a longer time between consecutive interrupts, the number of counts for a given change in capacitance increases. More delta counts correlates to more sensitivity and better usability in system. This high count delta can be achieved for a given delta C at the price of a longer measurement time and in turn, increased average current.

#### System Current Contributors

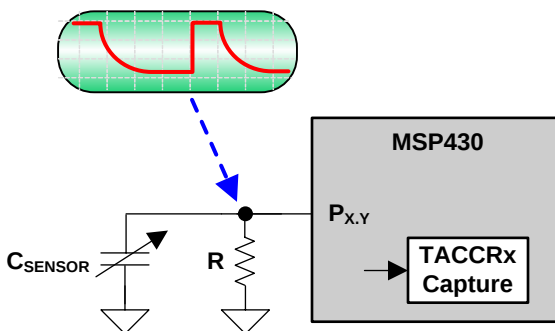
- DCO: ~85  $\mu$ A at 1 MHz
- Comp\_A+: ~45  $\mu$ A
- CA Vref:  $V_{cc}/(1.5R)$  (for 100k Rs, ~20  $\mu$ A)

In order to get the most out of the application, defining t\_measure for adequate counts for the application, yet small enough to keep current consumption to a minimum is the key.

- A larger delta C means a smaller t\_measure window can be used
- Design to fewest counts needed for lowest current

### 3.2 Resistor-Based Capacitive Measurement

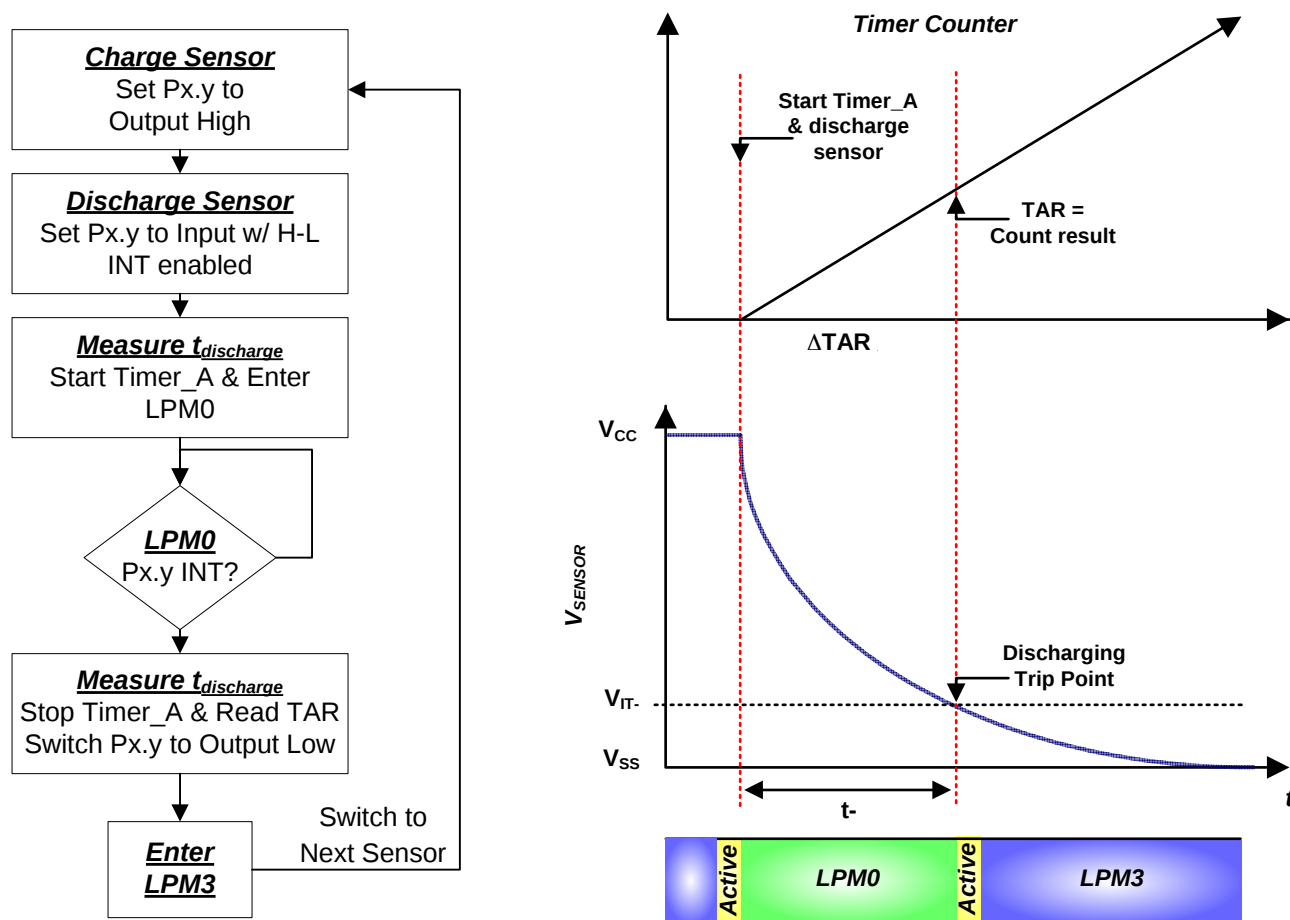
The second methodology to be described uses an external resistor to charge or discharge the given capacitive sensor. Using the port pins of the MSP430 to charge or discharge the sensor cap, the internal Timer\_A can be used to measure the corresponding charge or discharge time. Given a fixed external resistance to provide the charge/discharge path, the capacitance of the sensor can be measured. Figure 11 shows this system as implemented with an MSP430 for a discharge-only measurement.



**Figure 11. Basic Resistive Discharge Capacitance Measurement**

With the same small  $C_{\text{SENSOR}}$  of  $\sim 10$  pF, it is clear that R needs to be quite large in order to provide any realistically measurable discharge time. In this implementation, R is chosen to be 5.1M, giving a discharge window from V<sub>CC</sub> to near ground of  $\sim 250$   $\mu$ s (5 $\tau$ ). In this configuration, P<sub>x.y</sub> can be an output high to charge the sensor capacitor. It can then be switched to an input, allowing  $C_{\text{SENSOR}}$  to discharge through R. Given the  $\pm 50$  nA max port pin leakage of the MSP430, such a measurement implementation is possible with little discharge contribution via the port pin structure.

If P<sub>x.y</sub> is an interrupt-enabled GPIO (P1.x or P2.x in all MSP430's), the internal low-level threshold trip voltage can be used as a discharge reference, which when reached, an interrupt can be generated. Using this interrupt, the CPU can take a snapshot of the Timer\_A register using the capture logic of the Timer\_A module, storing the time taken to discharge the given sensor. Using the internal DCO, the timer can be clocked at frequencies up to 8 MHz or 16 MHz depending on the MSP430 used (1xx, 2xx, or 4xx devices). The higher the frequency the higher the count delta can be for a given change in sensor capacitance. Figure 12 details the overall measurement flow using the Timer\_A peripheral.

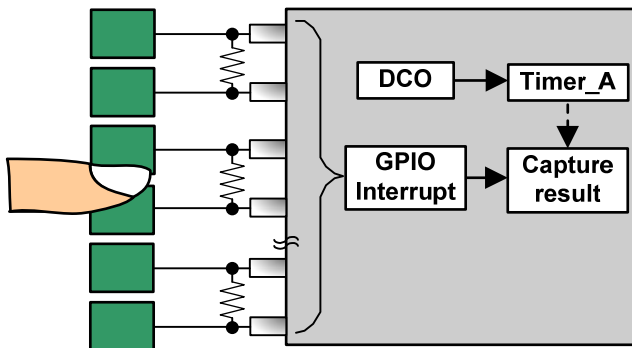


**Figure 12. Measurement Methodology Using the GPIO Threshold and Timer\_A**

The flow chart and graph in Figure 12 show a single measurement cycle. When the timer is started from zero for the measurement, the TAR value after the trip point is reached is the resulting number of counts for the measurement. Alternatively, the timer can be allowed to run continuously, in which case a timer capture needs to happen at both the start of the discharge and at the completion, then the difference in counts from the two points is the count result.

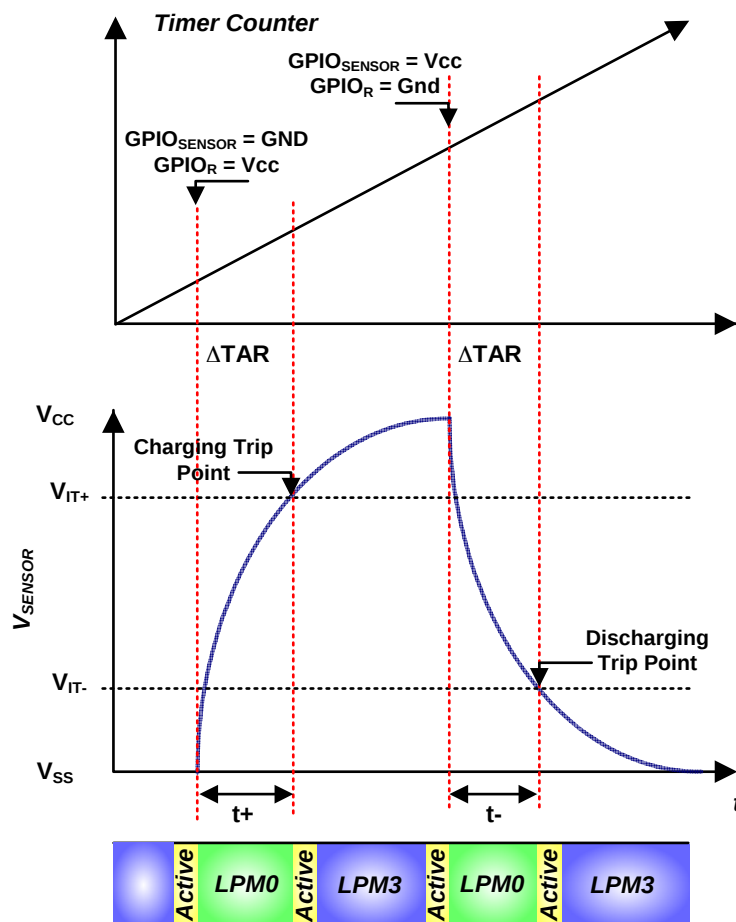
As the sensor capacitance increases, the time to discharge also increases and the number of counts measured goes up. The more counts that can be realized between an untouched sensor and a touched sensor, the better the sensitivity of the system.

The described setup in Figure 11 shows one port pin and one resistor for each sensor in the system. The setup can be further optimized by sharing a single resistor for each pair of sensors. During measurement of one sensor in the pair, the GPIO connected to the other sensor and the other side of the resistor can be set to a low output, creating the ground point for the discharge. These orientations can be swapped for the measurement of the second sensor in the pair. Figure 13 shows this configuration, optimizing the resistor count to  $\frac{1}{2}$  the number of sensors in the system for an even sensor count.



**Figure 13. Multi-Sensor Charge/Discharge Configuration**

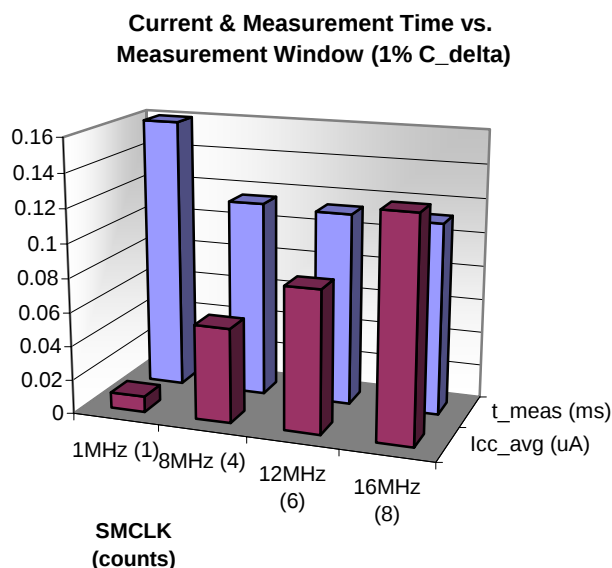
Another benefit of this configuration is that each sensor can be measured in both directions: charged from ground to the high level threshold and then discharged from  $V_{CC}$  to the low level threshold trip point. Figure 14 shows this method.



**Figure 14. Single Measurement Cycle for Improved Noise Rejection**

The measurement count is now the combination of the two results. These can be averaged or simply summed, as the absolute result is not of interest, but instead the difference from the base measurement result. By measuring both the charging and the discharging phases, system noise such as 50/60-Hz mains noise can be better rejected from influencing the final result.

Average current consumption for the charge/discharge system is quite low. For the given sensor and a 1% capacitance change, Figure 15 charts the average current consumption and measurement time for a single sensor based on the DCO frequency used to clock Timer\_A.



**Figure 15. Resistive Charge/Discharge  $I_{cc}$  and Measurement Time: Single Sensor**

Given a 5.1M resistor and using the charge/discharge combined measurement, a single sensor can be measured in a fraction of a millisecond. Since the measurement time is set by the time taken to charge or discharge the sensor, the actual time needed is quite small due to the very small sensor capacitance. This time is essentially fixed in this method of measurement, so the variable under design control is the frequency used to drive the timer. The higher the frequency used for the Timer\_A TAR, the larger the count result for a given change in capacitance. In the measurement shown in Figure 15, using 1 MHz provides a count delta of only 1 count, not usable for button determination. At 16 MHz, the count delta realized goes up to ~8, providing some margin for a touched/untouched switch detection event.

As the change in capacitance goes up, the larger the resulting measurement number will be for use in the system. Increasing the amount of influence of the user by reducing the thickness of the overlay is a good method to increase the usable count delta.

## 4 Software Implementation

Once the raw measurement result is obtained, the user software must now interpret the data for the given application. Given the sensitivity of the measurement, noise due to power supply, measurement clock frequency shift, and external factors such as 50/60-Hz mains noise can all lead to noisy results.

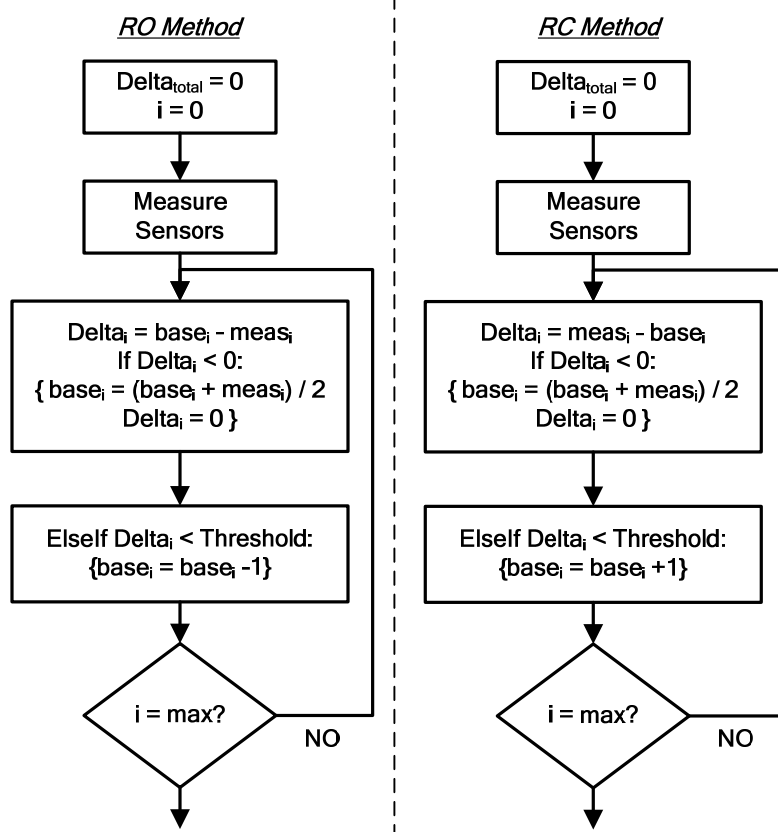
Often it may be good enough to simply ignore some number of LSBs of the result. When a large change in measured counts are achieved for a given touch event, this approach is likely acceptable. For instance, when doing simple key press detection for on/off function, noisy LSBs can often be ignored. When the application requires better resolution, e.g., implementation of a multi-point slider, it may be necessary to handle the data more carefully. Low pass filtering of the data and simple averaging of multiple samples can help to smooth out position detection in such a system. As system constraints such as power budget and specifically the insulator thickness of the sensor overlay are tightened, the resulting LSB content is more critical to extract.

### 4.1 Tracking Sensor Base Capacitance

Whether implementing a simple button sensor or more complex slider, tracking of each sensor's base capacitance is a key element of any touch sensing software algorithm. The base capacitance is the capacitance of a given sensor when untouched by the user. Voltage stability, PCB mechanics, insulator properties as well as ambient conditions such as temperature and proximity to other objects all play a role in the base measurement of a PCB-based capacitive sensor.

Without a dynamic ability to track this changing baseline value, instability can result in false press detection or "stuck key" behavior. Consider a simple button that has a preset on/off threshold without dynamic baseline tracking. In the case that the baseline result drifts, it may move towards the trip threshold for a valid key press. If there is enough drift in the base (untouched) result, it can reach the pre-set trip point and create such a false trigger.

One method for dynamically measuring and adjusting (tracking) the base capacitance is shown in Figure 16. Keep in mind that this must be done for each sensor independently. Also note that an increase or decrease in the variable "base" in the Figure is not necessarily reflecting a respective increase or decrease in base capacitance. Only the variable used in the algorithm is being adjusted. The sign of the "Delta" calculation and "base" adjustment is different for the RO and RC methodologies. (For RO the measured count decreases when the capacitance of the sensor increases; for RC the measured count increases when the capacitance of the sensor increases.)



**Figure 16. Example Algorithm for Tracking Baseline Sensor Capacitance**

When a measurement is completed, it must first be determined if a valid touch is occurring. This can be done by looking for a threshold that would represent the smallest real touch to be detected. For a simple on/off button this can be a number much smaller than the on/off threshold.

Once determined that no touch is occurring, the base value can be adjusted. How the base value is adjusted is dependent on the direction in which it is perceived to have changed. For instance, if the result of the measured sensor indicates that the base capacitance is decreasing, the base value is automatically adjusted down. Since a touched sensor will increase in capacitance, the decreasing result is taken as a genuine decrease in the base value. This is implemented by a simple average of the old base value with the new measured value, resulting in the new base value to be used in the next measurement.

When the measured result indicates an increase in base capacitance, it is recommended that the base value be tracked more slowly. While an increase may mean that it is truly shifting up, it may also mean that a finger is nearby that will soon be in contact with the sensor in question. If the base value is adjusted upwards too quickly, the change calculated when the finger is really in contact with the sensor may not be enough to indicate a press. For a detected increase in an untouched sensor, the base value is simply adjusted by a single count in the algorithm represented in Figure 16.

## 4.2 Implementing Button Function

Implementing simple button functionality is relatively straight forward using capacitive touch sensing. Because a single level of detection is required for a simple on/off button, the sensitivity and magnitude of the change in capacitance for a touch event is minimal. Figure 17 shows a graphical representation of the measured counts in a system for four keys. Untouched keys provide a count representing the base capacitance. The second key that is touched provides a larger delta count result, due to the increased capacitance. Sensitivity of each key is tuned by adjusting the threshold to a higher or lower value between the base and maximum count results.

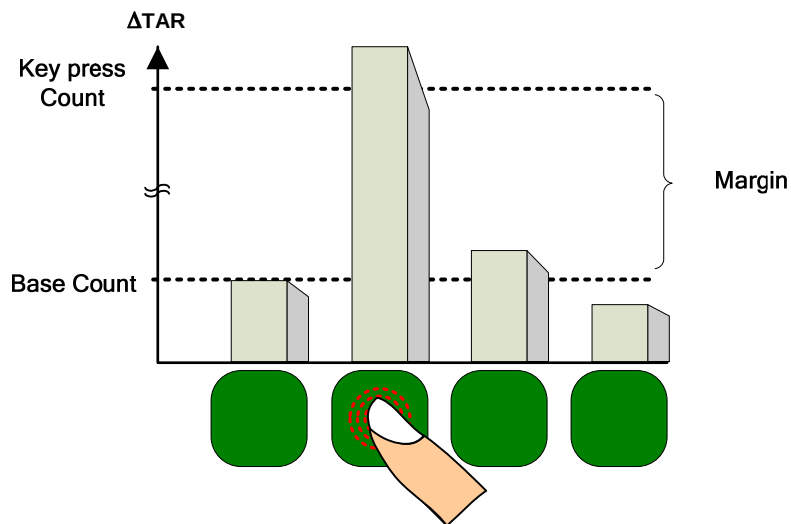
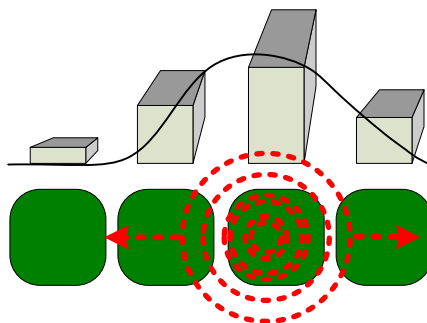


Figure 17. Button Press in a Four-Key System

The threshold set point should be above any noise in the measurement to allow for a robust and accurate key press result.

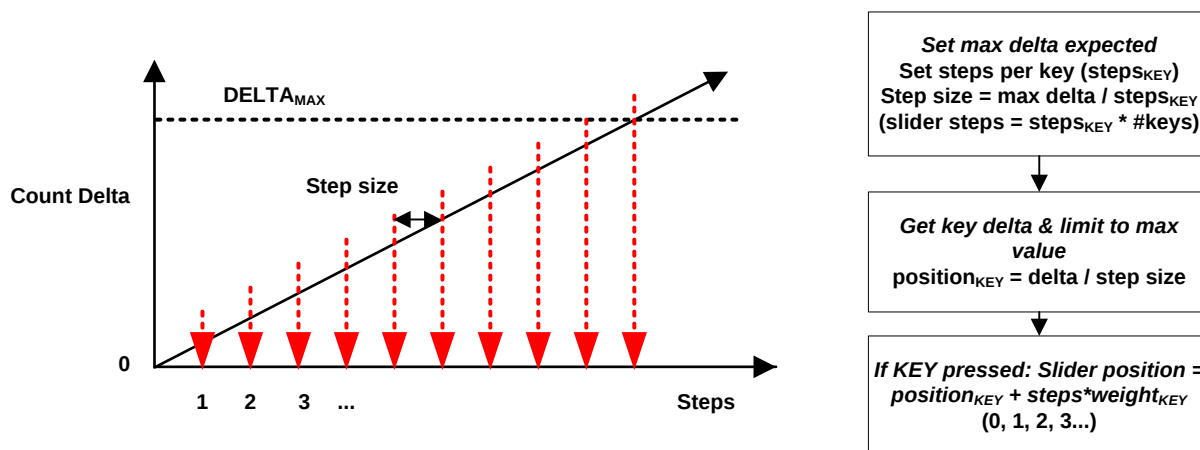
## 4.3 Implementing Slider Function

Extending beyond simple buttons with on/off functionality, it is possible to determine more than one threshold using the capacitive sensor to implement multiple “positions” for a single “button”. As more of the capacitive sensor is interacted with by the user, the larger the capacitance will be. Figure 18 represents the response as a single touch moves across the four sensor elements.



**Figure 18. Fundamental and Neighboring Sensor Response**

A simple slider implementation can be realized by assigning multiple positions to each sensor. The example here establishes 16 positions to each sensor of the four sensor configuration providing detection of 64 individual steps. The number of steps a sensor can accommodate is a function of the sensitivity of the sensor to a given touch, which is the amount of capacitive change induced. The greater the change in capacitance, the greater the number of delta counts measured, and the greater the number of individual positions per sensor. Figure 19 defines a simple algorithm that implements a 64 position slider with four sensors.

**Figure 19. Example Slider Position Determination Methodology**

This implementation simply limits the maximum response to an upper value which can always be achieved for a given system. This max delta is then divided by the desired steps per key. Each key is weighted to linearly result in steps 1 through 64 (position “0” is defined by no key press). The most significant sensor touched is used for determining the position. Table 2 indicates the associated steps from each key.

**Table 2. 64-Position Slider Key**

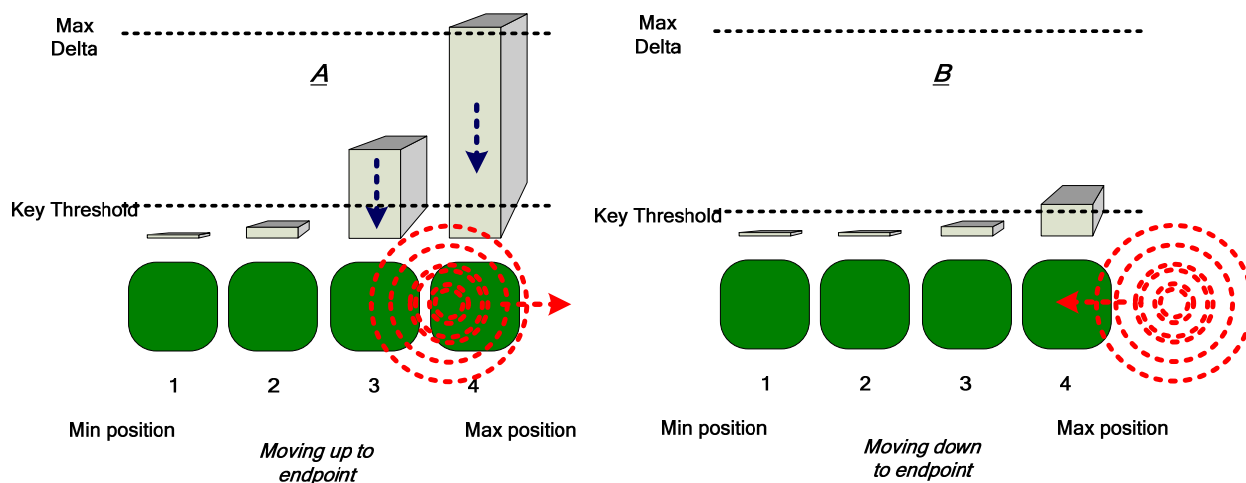
| <b><i>Sensor</i></b> | <b><i>Positions per sensor</i></b> | <b><i>Weight factor</i></b> | <b><i>Resulting Position</i></b> |
|----------------------|------------------------------------|-----------------------------|----------------------------------|
| 1                    | 1 to 16<br>(min to max delta)      | 0                           | 1-16 + 16×0: 1-16                |
| 2                    | 1 to 16<br>(min to max delta)      | 1                           | 1-16 + 16×1: 17-32               |
| 3                    | 1 to 16<br>(min to max delta)      | 2                           | 1-16 + 16×2: 33-48               |
| 4                    | 1 to 16<br>(min to max delta)      | 3                           | 1-16 + 16×3: 49-64               |

One drawback to this simple method is that once the maximum delta is achieved for a given sensor, the position calculated will not change until the next most significant sensor is influenced. While not described in the context of this document, it is possible to realize robust implementations providing a more linear transfer characteristic by interpolating position based on not only the maximally influenced sensor, but also its neighbors.

## 4.4 Handling Slider Endpoints

The simple slider described previously determines position across a 4-sensor array. Position “0” is for no key press and position “1” is the lowest key press. It is associated to the left-most position of the slider. Position “64” is the maximum position and is associated to the right-most position of the slider.

The simple algorithm implemented inherently handles proper increase or decrease in position at the minimum end point and up through the sensors moving left to right. However the maximum endpoint must be managed more directly. This is because the measured result for the most significant end will decrease from the maximum delta once the touch extends beyond the center of the last sensor and will increase from the minimum with a touch beginning at the rightmost edge of the same sensor. Figure 20 represents both scenarios.



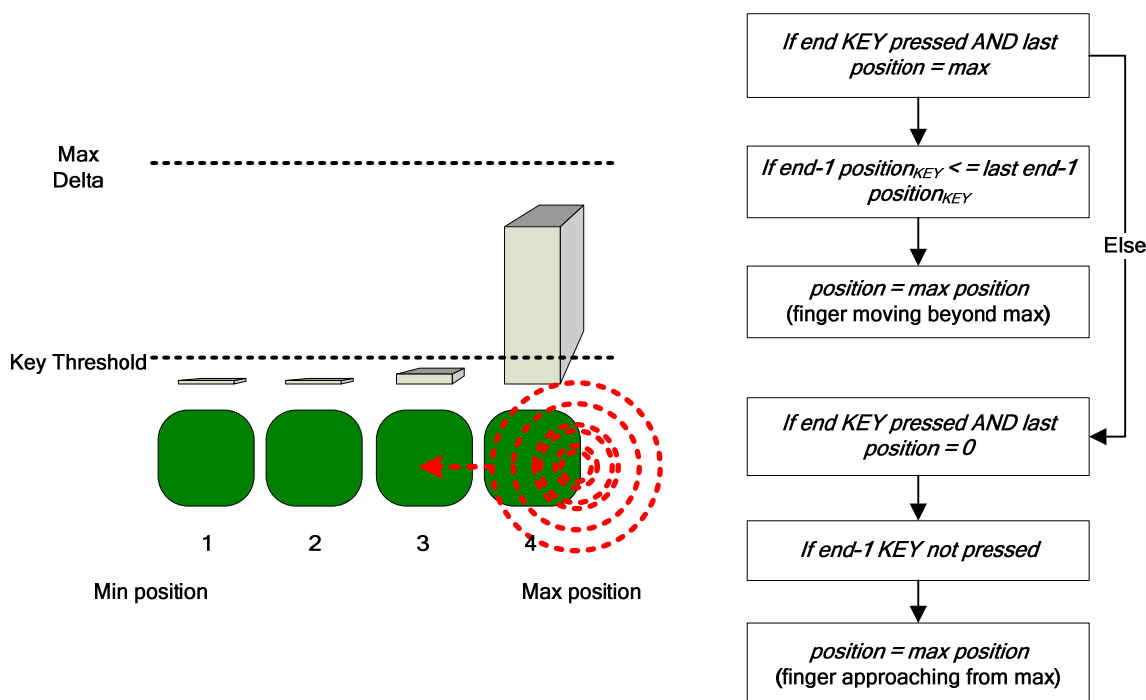
**Figure 20. Representation of Count Measurement Results for Max Endpoint**

In the figure, the A portion represents a touch extending up to and then beyond the desired maximum position and the B portion shows a touch approaching from the maximum position. In both cases, this influence must be correctly assessed and the expected position calculated accordingly.

In the case of this 64-position slider, a touch extending beyond the maximum position, left-to-right, will result in a decrease in measured response for the end key until the key is no longer touched measured as a “0” delta result. Without any special handling, this will appear as a decrease from position 64 gradually down to 49 and then 0, even though the touch did not move in the decreasing direction.

Similarly, a touch approaching from the right-most position, right-to-left, it will appear as an increase in measured response for the end key until the maximum influence is applied resulting in a maximum delta measurement. Without any special handling, this will appear as an immediate change from 0 to 49 and then gradually increase up to 64, even though the touch started at the physical end of the slider.

The algorithm defined in Figure 21 manages both conditions. In the case that condition A in Figure 20 is occurring, it is detected by monitoring the last position calculated as well as the next to last sensor (the third sensor in this example). When the third sensor is determined to be decreasing in influence and the prior position was the max, 64, the new position calculated will be held at 64, even if the actual measured result for the last sensor is decreasing.



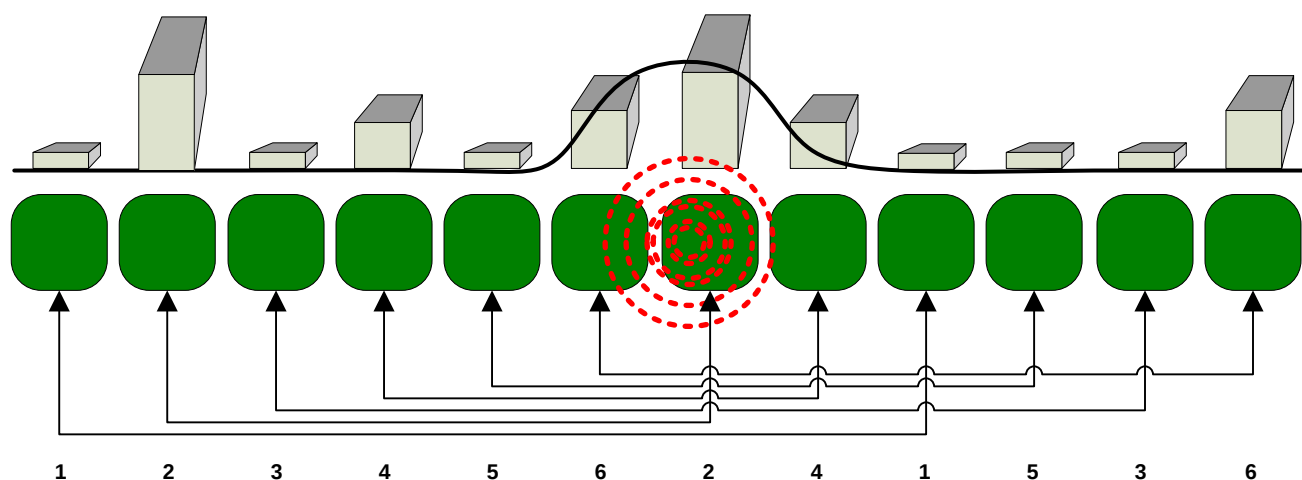
**Figure 21. Example Slider Endpoint Handling**

In the case condition B is occurring, it is detected by determining that the rightmost key is beginning to be touched without any response from the next key to the left. This is interpreted as a touch moving right to left beginning at the rightmost sensor and results in an immediate position determination at the maximum, 64.

## 4.5 Sensor Multiplexing for Sliders

When implementing sliders, it is often possible to extend the number of sensors used beyond the actual number of MSP430 pins available for measurement. This is achieved by connecting more than one sensor to a given measurement pin. To the measurement, the additional sensor simply appears as a larger base capacitance and does not impact the functionality of the system. However, since the base capacitance is increased and only one of the two sensors is influenced at a time, the response of the parallel sensors decreases. A practical limit for the number of sensors connected in parallel is considered to be two.

In terms of the sensor pairs being measured, the MSP430 cannot distinguish between the two and “sees” the same result for both, independent of which sensor in the pair is actually touched by the user. In order to use such a setup for a slider implementation, it is assumed that the mechanical design allows for multiple sensors to be influenced along any point of the slider. If this is achieved, and the sensors are organized in a way that the combined response of multiple sensors being influenced is unique for any position along the slider, the usability of the slider and positions can be extended. Figure 22 show a 6-to-12 pairing and the positions of each of the 6 pairs. Analyzing the figure it can be determined that for a touch anywhere along the slider, a unique response will be established across the group of sensors. Using an interpolating algorithm that accounts for this unique response detection, the sensor count can be extended beyond the actual pin count for a given device.



**Figure 22. Example Sensor Multiplexing of 12 Capacitive Sensor Elements**

## 5 Summary

Two methods of implementing single-touch capacitive sensing on the MSP430 have been discussed. While each method has its own advantages and disadvantages, each can be used to realize a solution when the proper measures are taken both from a mechanical assembly and hardware/software standpoint.

In summary, each method's key takeaways are:

### MSP430 RO Method

- Works in Comp\_A+ devices only
- Number of independent sensors limited by available CA+ mux inputs
- Needs one external resistor per sensor plus reference ladder (three additional resistors)
- Sensitivity limited by current consumption (programmable measurement time)

### MSP430 RC Method

- Can be implemented on any MSP430
- Up to 16 independent sensors (16 interruptible GPIOs)
- Single external resistor per two sensors
- Sensitivity limited by on-chip maximum clock frequency (fixed measurement time)
- Lowest power implementation

Fundamentally, the actual measurement of the capacitance is quite simple; however, the mechanics of the assembly regarding the sensors and touch interface along with the software algorithms used to determine the nature of the touch provide the implementation in an end-equipment with significant challenges. The contents of this application report are not to provide a one-size-fits-all solution, but rather to establish the fundamentals of the application and methodologies used which can be extended and customized to fit a given product.

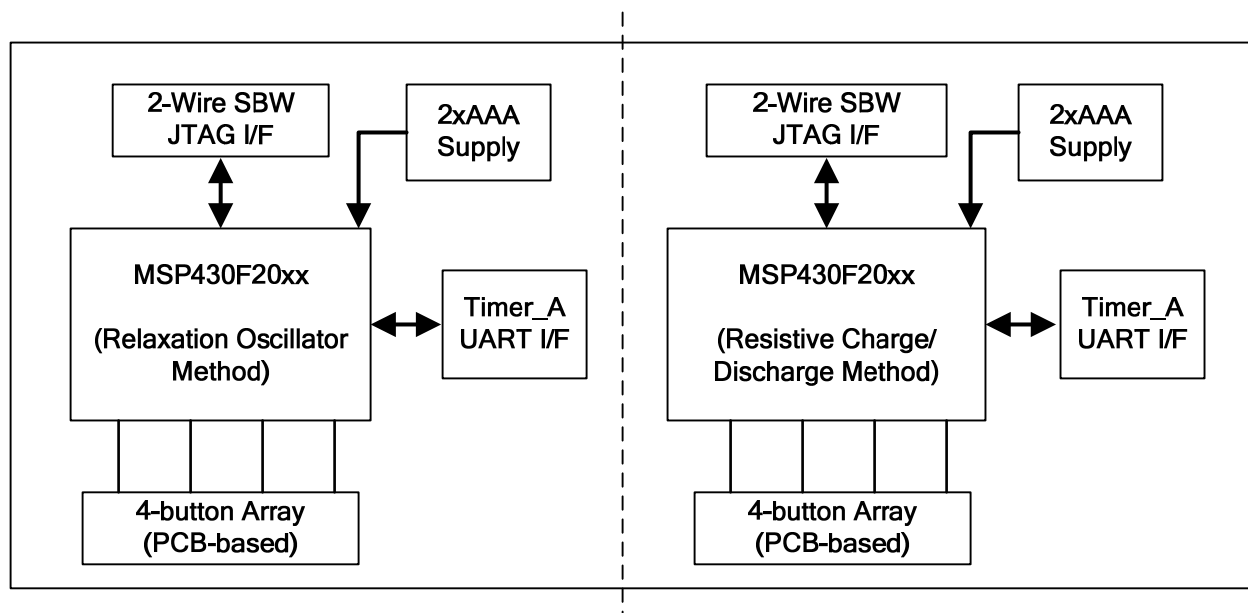
## 6 References

***Special thanks to fellow Tlrs Vincent Chan and Steve Underwood for their significant contributions to this application development.***

1. Robert E Marin, Roger K Simonson, "Capacitive Keyswitch Sensor and Method" US Patent #3931610, 1976
2. MSP430x2xx Family User's Guide (SLAU144)
3. MSP430F20xx data sheet (SLAS491)
4. MSP430FG4618/F2013 Experimenter Board and Software  
<http://focus.ti.com/docs/toolsw/folders/print/msp-exp430fg4618.html>
5. MSP430 USB Stick Development Tool <http://focus.ti.com/docs/toolsw/folders/print/ez430-f2013.html>

## Appendix A. Flexible Evaluation and Demonstration Hardware

This section is the hardware description of a 4-button MSP430-based single-touch capacitive sensing system described and used for test/measurement in the previous sections of this document. This hardware implements both the oscillator and RC methods on a single platform for evaluation purposes



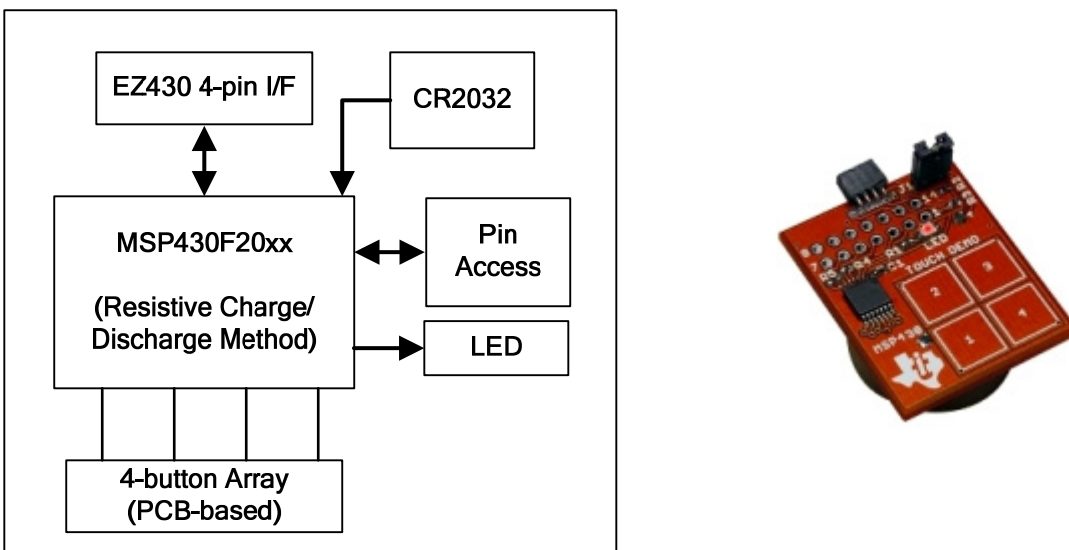
**Figure 23. Flexible 4-Button RO/RC System Block Diagram**

Both methods for capacitive sensor measurement discussed can be realized with this hardware. However, each method is implemented independent of the other, allowing the two instances to be cut apart from a single PCB. In addition to the touch sensing interface, each implementation provides a dedicated 2xAAA battery power supply, 14-pin header for JTAG access to each MSP430 MCU and hardware interface to Timer\_A pins. The Timer\_A connections allow for communication of-board with a PC-based application, for example, providing a means of data transfer and control via standard UART. (Detailed implementation of a Timer\_A UART can be found in additional MSP430 Application Note collateral.)

Corresponding hardware and software materials can be found in the .zip archive associated with this application note, downloadable from [www.msp430.com](http://www.msp430.com).

## Appendix B. Simple Demonstration Hardware

This section is the hardware and software description of a simple 4-button single-touch sensing system using the MSP430. This system implements only the RC-based measurement technique and is designed for a use with the EZ430 development tool.



**Figure 24. Simple 4-Button RC System Block Diagram and Image**

Information regarding the EZ430 development tool can be found at <http://focus.ti.com/docs/toolsw/folders/print/ez430-f2013.html>

Corresponding hardware and software materials can be found in the .zip archive associated with this application note, downloadable from [www.msp430.com](http://www.msp430.com).

## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

| Products         |                                                                    | Applications       |                                                                          |
|------------------|--------------------------------------------------------------------|--------------------|--------------------------------------------------------------------------|
| Amplifiers       | <a href="http://amplifier.ti.com">amplifier.ti.com</a>             | Audio              | <a href="http://www.ti.com/audio">www.ti.com/audio</a>                   |
| Data Converters  | <a href="http://dataconverter.ti.com">dataconverter.ti.com</a>     | Automotive         | <a href="http://www.ti.com/automotive">www.ti.com/automotive</a>         |
| DSP              | <a href="http://dsp.ti.com">dsp.ti.com</a>                         | Broadband          | <a href="http://www.ti.com/broadband">www.ti.com/broadband</a>           |
| Interface        | <a href="http://interface.ti.com">interface.ti.com</a>             | Digital Control    | <a href="http://www.ti.com/digitalcontrol">www.ti.com/digitalcontrol</a> |
| Logic            | <a href="http://logic.ti.com">logic.ti.com</a>                     | Military           | <a href="http://www.ti.com/military">www.ti.com/military</a>             |
| Power Mgmt       | <a href="http://power.ti.com">power.ti.com</a>                     | Optical Networking | <a href="http://www.ti.com/opticalnetwork">www.ti.com/opticalnetwork</a> |
| Microcontrollers | <a href="http://microcontroller.ti.com">microcontroller.ti.com</a> | Security           | <a href="http://www.ti.com/security">www.ti.com/security</a>             |
| RFID             | <a href="http://www.ti-rfid.com">www.ti-rfid.com</a>               | Telephony          | <a href="http://www.ti.com/telephony">www.ti.com/telephony</a>           |
| Low Power        | <a href="http://www.ti.com/lpw">www.ti.com/lpw</a>                 | Video & Imaging    | <a href="http://www.ti.com/video">www.ti.com/video</a>                   |
| Wireless         |                                                                    | Wireless           | <a href="http://www.ti.com/wireless">www.ti.com/wireless</a>             |

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
Copyright © 2007, Texas Instruments Incorporated