

Capacitive Touch Design Flow for MSP430™ MCUs With CapTIvate™ Technology

Yiding Luo

MSP430 Applications

ABSTRACT

This application report describes the steps to develop your own capacitive touch implementation using MSP430™ MCUs with CapTIvate™ capacitive touch sensing technology. This document presents the entire development flow from understanding capacitive touch sensing to production readiness.

Contents

1	Introduction	3
	1.1 CapTIvate Technology Guide	3
2	Design Flow Overview	4
3	Step 1: Understand the Basics of Capacitive Touch Sensing and CapTIvate Technology	5
4	Step 2: Define the System Requirements and the Product Operating Conditions	6
5	Step 3: Concept Feasibility Evaluation and Device Selection	7
	5.1 Concept Evaluation	7
	5.2 Device Selection	8
6	Step 4: Hardware Development	11
7	Step 5: Firmware Development and System Tuning	19
8	Step 6: System Integration and Prototyping	28
9	Step 7: System Validation and Field Testing	28
10	Step 8: Mass Production	29
11	References	29

List of Figures

1	Design Flow Overview	4
2	Stackup of Basic Capacitive Touch Design	5
3	VREG Pin Decoupling Capacitor	11
4	Series Resistors on CAP I/Os	12
5	EMI Noise Filter Capacitor	12
6	Reset Pin Resistor and Capacitor	12
7	Auto-Assign Feature	13
8	Scan Time Cycles	13
9	Button Shape for Self Capacitance	14
10	Button Shape for Mutual Capacitance	15
11	Mutual-Capacitance Matrix	15
12	Self Capacitive Slider and Wheel	15
13	Mutual Capacitive Slider and Wheel	16
14	Sensor Traces Routing	16
15	Mutual-Capacitance Routing Considerations	17
16	LED Routing	17
17	Ground Pour	18
18	CapTIvate Design Center	20

19	CapTlvate Tuning Process.....	20
20	Sensor Parameters	21
21	Sensor Properties Window.....	22
22	Controller Properties Window	22
23	Enable Advanced Mode	22
24	Tuning Consideration	22
25	Delta	23
26	Proximity Threshold.....	24
27	Touch Threshold	24
28	Change Touch Threshold	24
29	SNR Measurement.....	25
30	Proximity and Touch Debounce	25
31	Scan Time Estimation	26
32	Response Time Equation	26
33	Update Source Code	27

List of Tables

1	Comparison of Self Capacitance and Mutual Capacitance	5
2	Define MCU Requirements	6
3	Define Sensor Requirements	6
4	Define Mechanical Requirements	6
5	Define Operating Condition Requirements	7
6	CapTlvate Family Devices.....	9
7	Device Generations	9
8	Comparison of First-Generation and Second-Generation Device Features	9
9	CapTlvate Key Parameter Performance Under Different Test Conditions	10
10	Schematic Design Checklist	11
11	PCB Layout Design Checklist	14
12	Mechanical Design Checklist	18
13	Sensitivity Tuning Checklist.....	23
14	Reliability Tuning Checklist	24
15	Response Time Tuning Checklist	25
16	System Validation Check List.....	28
17	Mass Production Check List	29

Trademarks

MSP430, CapTlvate, BoosterPack, LaunchPad, Code Composer Studio are trademarks of Texas Instruments.

IAR Embedded Workbench is a registered trademark of IAR Systems.

All other trademarks are the property of their respective owners.

1 Introduction

Capacitive touch sensing is a unique human-machine interface technology that enables the creation of differentiated user interfaces that can bring value to a wide variety of products. With capacitive sensing, mechanical switches and knobs can be replaced with elegant buttons, sliders, and scroll wheels that integrate seamlessly into the enclosure of a product to simultaneously improve the aesthetics and functionality of the product.

These aesthetic and functional improvements come with a different set of challenges for product designers. For example, capacitive touch requires more attention to detail in the PCB design than a simple mechanical switch does. It also requires more firmware to determine the state of the user interface. Factors such as these can make capacitive touch seem like a challenging technology to develop with. However, the reality is that capacitive touch is not extremely challenging—it is just *new* and *different*.

Like many other technologies, the challenges and risks associated with getting a capacitive touch design concept through development and into production can be reduced by having a proven development flow in place. If you are a designer who is new to capacitive touch, or an experienced designer that is new to TI's CapTIvate technology for capacitive sensing, this document provides a step-by-step design flow to go from first concept to mass production. In addition, this document provides checklists at specific points in the development flow to reduce the risk of unforeseen issues late in the development cycle.

1.1 CapTIvate Technology Guide

The [CapTIvate™ Technology Guide](#) is a comprehensive resource on capacitive touch design with CapTIvate technology. This design flow guide includes many links to the *CapTIvate™ Technology Guide* to support your design.

Refer to [CapTIvate MCUs](#) to learn more about MSP430 capacitive touch sensing microcontrollers.

2 Design Flow Overview

Figure 1 shows the typical flow of a capacitive touch sensing product design cycle with CapTIvate MCUs.

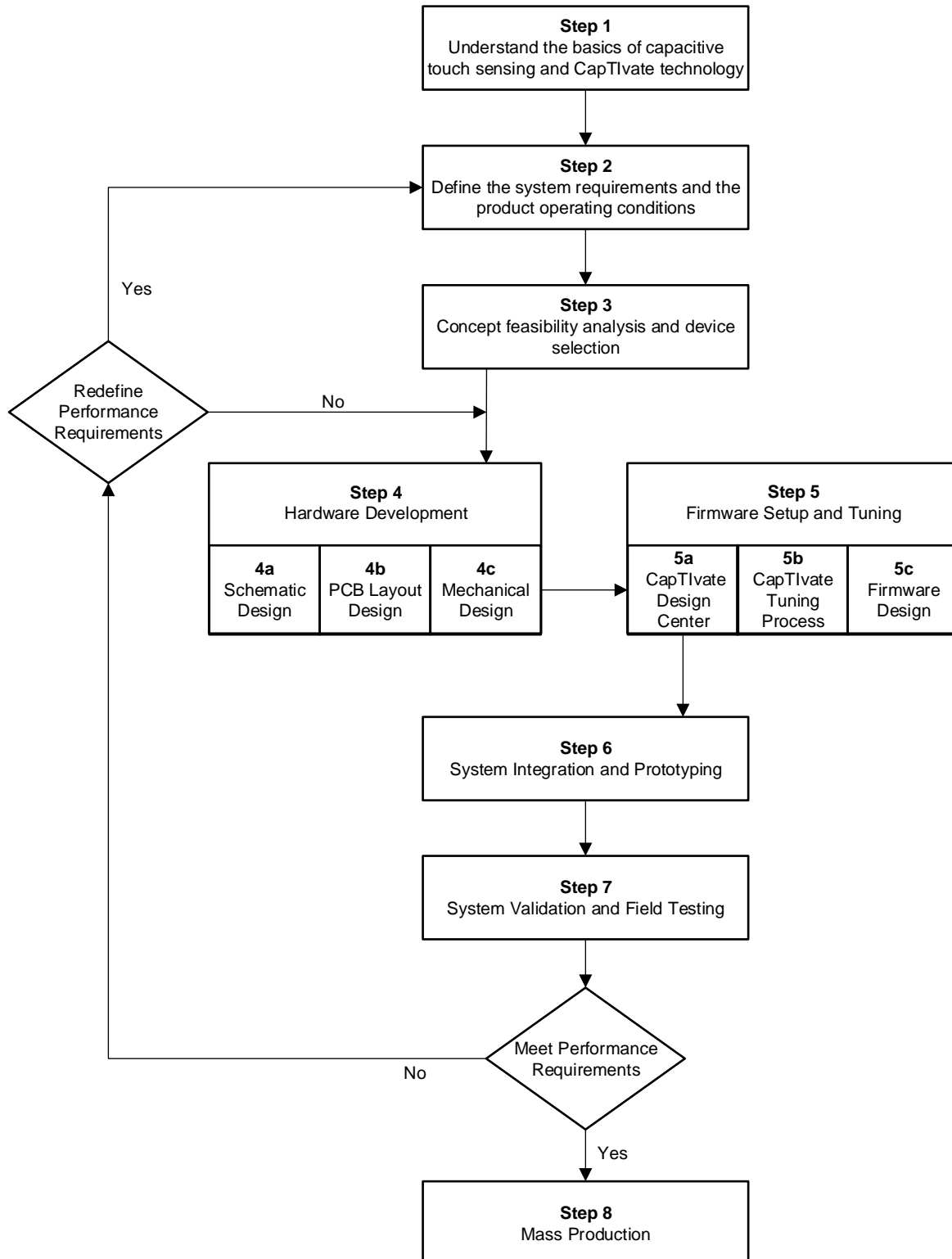


Figure 1. Design Flow Overview

3 Step 1: Understand the Basics of Capacitive Touch Sensing and CapTIvate Technology

The first step of designing a successful capacitive touch sensing system is to understand the basic principles of capacitive touch sensing. Capacitive sensing performs a measurement to detect a change in the capacitance of a sensor element. A sensor element can be any conductive material (for example, a copper PCB plane or a wire). The change can be due to human interaction, such as a finger, ear, or hand. This is often termed capacitive touch, and Figure 2 shows the stackup of a basic capacitive touch design.

For additional information, visit [Capacitive Sensing Basics](#).

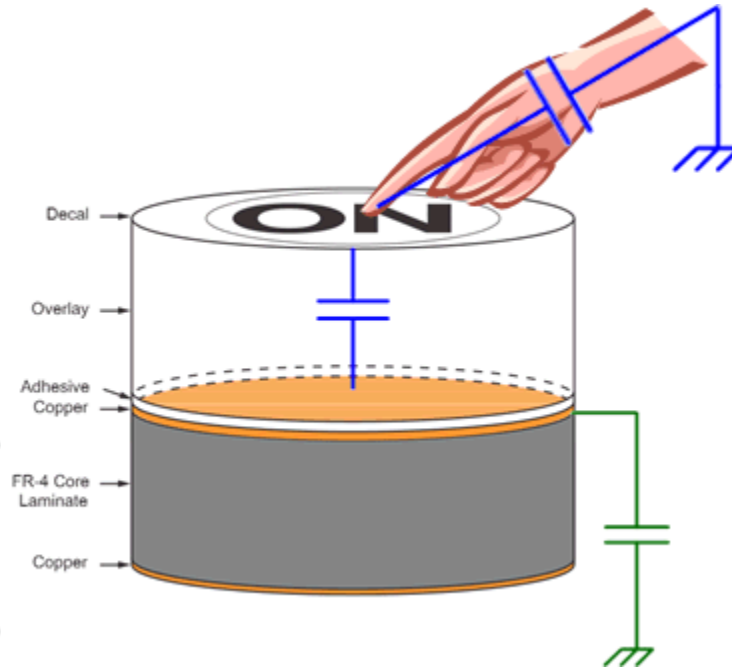


Figure 2. Stackup of Basic Capacitive Touch Design

Two different capacitive sensing measurement methods are available: self capacitance and mutual capacitance. Table 1 compares these two sensing measurement methods. Review the fundamentals of these two sensing methods to choose the one that best suits the application requirements. CapTIvate MCUs support both sensing methods in a single design.

For additional information, visit these topics:

- [Self Capacitance Fundamentals](#)
- [Mutual Capacitance Fundamentals](#)

Table 1. Comparison of Self Capacitance and Mutual Capacitance

Characteristic	Self Capacitance	Mutual Capacitance
Sensor layout	Simpler	More complex
Allow overlay air gap	Yes	No
Number of PCB layers	Typically 1 or 2	Typically 2
Pins required for 12 buttons	12	7 (3 TX and 4 RX)

At this point, you should have a good understanding of the capacitive touch sensing fundamentals. Refer to these topics to find out how TI CapTIvate technology applies these fundamentals:

- [CapTIvate Charge Transfer Technology](#)
- [The CapTIvate Peripheral](#)

4 Step 2: Define the System Requirements and the Product Operating Conditions

The second step of designing a successful capacitive touch sensing system is to define the system requirements and the operating conditions of the end product. This is a critical step to ensure the desired performance in the product. In many cases, there are unique areas of expertise within a company, with separate groups or teams that implement each function. Sometimes one or more of these functions may even be outsourced to a third-party developer or contract manufacturer. Therefore, it is strongly recommended to have all the function teams working together. [Table 2](#) lists several key system requirements and operating conditions to consider when defining a product with CapTIvate technology. You might not have all the answers for these requirements at this step of the design process, but the overall design process is more efficient with more requirements being defined early.

Table 2. Define MCU Requirements

Requirement	Description
Memory configuration	Estimate the memory requirements based the functions implemented on the MCU. Refer to Memory Configuration Considerations .
Pin counts	Estimate the number of GPIOs based on the application. CapTIvate capacitive touch sensing MCUs support up to 19 GPIOs.
Packages	Define the package that is suitable for the application. CapTIvate capacitive touch sensing MCUs are available in QFP, TSSOP, QFN, and DSBGA packages.
Communication interface	Define the type and number of communication interfaces that the device need to support. CapTIvate capacitive touch sensing MCUs support I ² C, UART, and SPI interfaces.
Bootloader	Define if the application requires bootloader for programing during the prototyping phase, final production, and in service.
Supply voltage range	Determine supply voltage range for MSP430 MCU provided by system.
Additional peripherals	Define additional peripherals required for the application such as ADC, timers, and comparators.
Number of CapTIvate I/Os	This number depends on the capacitive sensor requirements. Refer to CapTIvate Device Selection .
Number of CapTIvate blocks	Each CapTIvate block has 2 or 4 CapTIvate I/Os depends on the part. The number of CapTIvate measurement blocks on a device determines the number of sensing electrodes that can be measured simultaneously. CapTIvate MCUs have 1, 2, or 4 CapTIvate blocks depends on the device. Select a device with more blocks if the application includes many capacitive sensors and has strict power consumption and response time requirements. Refer to CapTIvate Device Selection .

Table 3. Define Sensor Requirements

Requirement	Description
Function	The functionality of the capacitive sensor in the application (human detection, object detection)
Type	The type of the capacitive sensor in the application (button, slider, wheel, proximity, metal touch)
Quantity	The number of each type of the sensors.
Size	The dimension of each type of the sensors based on the desired touch area and the space constraints
Resolution	The resolution of the slider, wheel, and proximity sensors
Shape	The shape of each type of the sensors based on the desired touch area and the space constraints
Material	The material of the sensors (PCB, FPC, conductive wire, copper foil, ITO)
Sensor orientation	The orientation of the capacitive touch sensors (vertical, horizontal)

Table 4. Define Mechanical Requirements

Requirement	Description
Overlay properties	Properties of the overlay on top of the capacitive touch sensors including material, thickness, stackup, and bounding
Enclosure properties	Properties of the enclosure for the system including material, thickness, stackup, and bounding
PCB interconnection	Interconnection between different PCBs in the system

Table 5. Define Operating Condition Requirements

Requirement	Description
Power supply	System power supply configuration (wall power, battery power)
Indoor or outdoor	Operating environment for the application
Temperature variance	Temperature variance on the CapTIvate MCU including the rate of change and the range of the expected temperature condition
EMC performance	Type and stress level of EMC tests that the system must pass
Moisture and liquid performance	Expected response of the capacitive touch sensors under the typical liquid scenarios of the application.
Power consumption	Targeted average and transient power consumption from the MSP430 MCU
System noise	Potential noise sources in the system
Feedback	User feedback associated with the detection (LEDs, buzzers, haptic)

5 Step 3: Concept Feasibility Evaluation and Device Selection

The third step of designing a successful capacitive touch sensing system is to perform feasibility studies based on the system requirements and to select the best CapTIvate device for the application.

5.1 Concept Evaluation

The CapTIvate MCUs provide a comprehensive set of evaluation tools to help accelerate the evaluation process.

5.1.1 Hardware

- MSP CapTIvate MCU Development Kit** ([Order](#) | [User's Guide](#) | [Design Files](#))
 This development kit helps you evaluate button, slider, wheel, and proximity functions with the self capacitance and mutual capacitance sensing capabilities of CapTIvate technology.
- EVM430-CAPMINI** ([Order](#) | [User's Guide](#) | [Design Files](#))
 The EVM430-CAPMINI is an easy-to-use evaluation board for the MSP430FR2512 capacitive-touch-sensing microcontroller. It demonstrates the key features of CapTIvate technology such as ultra-low power and ease of use. The onboard CapTIvate HID bridge tool supports real-time tuning with CapTIvate Design Center. The board includes 4 touch buttons and 4 LEDs for creating a simple user interface. It also provides a buzzer for advanced application. The board can be powered by a USB cable or an onboard CR1632 coin battery to enable stand-alone demonstration.
- MSP430 CapTIvate Touch Keypad BoosterPack™ Plug-in Module** ([Order](#) | [User's Guide](#) | [Design Files](#))
 The BOOSTXL-CAPKEYPAD lets you extend a LaunchPad™ development kit by adding a 12-key capacitive touch numeric keypad with LED backlighting and proximity wakeup. Even if you do not have a LaunchPad development kit, the BOOSTXL-CAPKEYPAD can be used as a simple CapTIvate technology evaluation module when coupled with the CAPTIVATE-PGMR board included in the MSP CapTIvate MCU Development Kit.
- Touch on Metal Capacitive Sensing Add-on Board for the CapTIvate Development Kit** ([Order](#) | [User's Guide](#) | [Design Files](#))
 This is an add-on board for the MSP CapTIvate Development Kit ([MSP-CAPT-FR2633](#)) that allows designers and engineers evaluate touch on metal technology.
- Customized Hardware**
 Evaluating an application involves laying out a basic PCB with a footprint for the sensor connector, or laying out electrodes with copper tape and wiring them into the connector. You can use the CAPTIVATE-FR2633 board in the MSP CapTIvate MCU Development Kit to evaluate your sensor PCB (refer to the [CAPTIVATE-FR2633 board pinout overview](#)).

5.1.2 Software

- **CapTIvate Design Center Tool** ([Download](#))
 - GUI
 - Documentation
 - Example projects
 - Real time tuning
 - Generate source code

The CapTIvate Design Center GUI is a one-stop resource for everything related to CapTIvate capacitive sensing technology. It includes tools, documentation and software examples that can simplify and accelerate capacitive touch designs. Start by dragging and dropping capacitive sensing elements to the GUI workspace and use the GUI to configure and tune the design in real time. When tuning is complete, the GUI will generate the source code needed for the selected capacitive touch elements.

- **TI Code Composer Studio™ IDE** ([Download](#))

Code Composer Studio is a TI integrated development environment (IDE) that supports TI's microcontroller and embedded processors portfolio.
- **IAR Embedded Workbench® for MSP430 IDE** ([Download](#))

IAR Embedded Workbench is a third-party integrated development environment (IDE) that supports TI's microcontroller and embedded processors portfolio.

5.2 Device Selection

TI offers a wide range of programmable CapTIvate microcontrollers. Use the requirements defined in [Step 2](#) to select the best CapTIvate device for the application.

For additional information, visit these resources:

- [CapTIvate family devices table](#)
- [CapTIvate family devices data sheets](#)

[Table 6](#) summarizes the number sensors and detection supported by the CapTIvate MCUs. [Table 7](#) and [Table 8](#) compare the two generations of CapTIvate technologies. [Table 9](#) lists the CapTIvate key parameter performance under different test condition.

Table 6. CapTlvate Family Devices

		CapTlvate Pins (RX or TX)			
		4 Pins	8 Pins	16 Pins	
Parallel CapTlvate Measurement Blocks	1	MSP430FR2515IRHL MSP430FR2512IPW16			Devices in this row measure one electrode at a time
	2		MSP430FR2522IRHL MSP430FR2522IPW16		Devices in this row can measure up to two electrodes in parallel for faster scanning in applications with many electrodes
	4		MSP430FR2633IYQW MSP430FR2632IYQW MSP430FR2632IRGE MSP430FR2532IRGE	MSP430FR2633IRHB MSP430FR2633IDA MSP430FR2533IRHB MSP430FR2533IDA MSP430FR2676TPT MSP430FR2676TRHA MSP430FR2676TRHB MSP430FR2675TPT MSP430FR2675TRHA MSP430FR2675TRHB	Devices in this row can measure up to four electrodes in parallel for the fastest scanning in applications with many electrodes
		Devices in this column have 4 CapTlvate pins and support up to 4 electrodes	Devices in this column have 8 CapTlvate pins and support up to 8 electrodes in self-capacitance mode or 16 electrodes in mutual-capacitance mode	Devices in this column have 16 CapTlvate pins and support up to 16 electrodes in self-capacitance mode or 64 electrodes in mutual-capacitance mode	

Table 7. Device Generations

First-generation devices	MSP430FR2512, MSP430FR2522, MSP430FR2632, MSP430FR2633, MSP430FR2532, MSP430FR2533
Second-generation devices	MSP430FR2675, MSP430FR2676
Device-dependent features	FRAM, RAM, capacitive-touch I/Os, sensing blocks, package, power consumption, other peripherals

Table 8. Comparison of First-Generation and Second-Generation Device Features

Generation-Dependent Feature	First-Generation Devices	Second-Generation Devices	Advantages of Second Generation Over First Generation
Sensing modes	Self capacitance and mutual capacitance	Self capacitance and mutual capacitance	—
Electrode charge voltage	VREG mode (1.5 V)	VREG mode (1.5 V) DVCC mode (2.7 V to 3.6 V)	Improved SNR and conductive noise immunity with DVCC mode
Total electrode capacitance	300 pF at 4-MHz conversion frequency	300 pF at 4-MHz conversion frequency	—
Input bias current	No	Yes	Improved conductive noise immunity
Conversion processing (noise filter, drift compensation, detection, event timing)	Hardware state machine	Hardware state machine	—
Noise immunity processing (frequency hopping, oversampling)	Software	Hardware state machine	CPU no longer required for frequency hopping and oversampling

Table 9. CapTIvate Key Parameter Performance Under Different Test Conditions

Parameter	Description	Test Condition	1st Generation	2nd Generation
Proximity Detection Range	Proximity sensors are electrodes designed to detect a hand or other conductive object at some distance using greater sensitivity compared to buttons. Proximity sensor design involves carefully balancing sensor size, sensor configuration, ground shielding and system stability.	<ul style="list-style-type: none"> The test PCB has a proximity sensor of 80 mm by 50 mm with 10-mm wide rectangular ring shape with centered hatched GND. Simulated outstretched finger approaching the sensor. 	Typical: 25 mm	Typical: 70 mm
		<ul style="list-style-type: none"> The test PCB has a proximity sensor of 80 mm by 50 mm with 10-mm wide rectangular ring shape with centered hatched GND. Simulated flat hand approaching the sensor. 	Typical: 75 mm	Typical: 110 mm
Sensor Electrode Size	The sensitivity of a sensor can depend on a variety of parameters, but those parameters that have the greatest impact are the overlay material, overlay thickness and type of electrode and electrode size. It is critical to balance the electrode size to achieve desired sensitivity and reliable touch performance.	<ul style="list-style-type: none"> The test PCB is a 2 layer design with approximately 25% hatched GND on top and bottom layers with 4-mm plastic overlay 	Minimum Recommended: 64 mm ²	Minimum Recommended: 36 mm ²
		<ul style="list-style-type: none"> The test PCB is a 2 layer design with approximately 25% hatched GND on Top and Bottom layers with 4-mm glass overlay 	Minimum Recommended: 50 mm ²	Minimum Recommended: 36 mm ²
Response Time	Response time is defined as the time between a finger touchdown event on the sensor electrode and the touch controller generating an response signal. This parameter is particularly important because it directly translates to how fast users can interact with the touch panel. More signal processing provides a more reliable system but the trade-off is longer response time.	<ul style="list-style-type: none"> The test PCB has 16 self mode buttons, conversion frequency 2 MHz and conversion count set to 500. Gen1: MSP430FR2633, Gen2 MSP430FR2676 Frequency hopping and oversampling not enabled. 	Typical: 4 ms	Typical: 4 ms
		<ul style="list-style-type: none"> The test PCB has 16 self mode buttons, conversion frequency 2 MHz and conversion count set to 500. Gen1: MSP430FR2633, Gen2 MSP430FR2676 Frequency hopping enabled. 	Typical: 14 ms	Typical: 10 ms
		<ul style="list-style-type: none"> The test PCB has 16 self mode buttons, conversion frequency 2 MHz and conversion count set to 500. Gen1: MSP430FR2633, Gen2 MSP430FR2676 Frequency hopping and 2x oversampling enabled. 	Typical: 26 ms	Typical: 15 ms
Noise Immunity	Capacitive touch sensing involves the measurement of very small changes in capacitance so the system that are going to be used in noisy environments must be designed with noise immunity in mind from the start. Capacitance electrodes become more sensitive to touch when the system is subjected to conducted noise. This error in touch sensitivity could lead to false touch when the user gets close to the button but does not actually touch the button.	<ul style="list-style-type: none"> CAPTIVATE-EMC EVM was used for this test. A simulated human finger was placed 5 mm above the sensor electrode during the test. 3 Vrms conducted noise coupled directly into DC power supply, noise frequency was swept from 300 kHz to 80 MHz 	PASS with No False Touch (Requires additional software algorithm) ⁽¹⁾⁽²⁾	PASS with No False Touch (No additional software algorithm required) ⁽¹⁾⁽³⁾
		<ul style="list-style-type: none"> CAPTIVATE-EMC EVM was used for this test. A simulated human finger was placed 5 mm above the sensor electrode during the test. 10 Vrms conducted noise coupled directly into DC power supply, noise frequency was swept from 300 kHz to 80 MHz 	PASS with No False Touch (Requires additional software algorithm) ⁽¹⁾⁽⁴⁾	PASS with No False Touch (No additional software algorithm required) ⁽¹⁾⁽⁵⁾

⁽¹⁾ The error in touch sensitivity is defined as conversion result over touch threshold when noise is present.

⁽²⁾ Maximum error in touch sensitivity: self mode: 46.2%, mutual mode: 13%

⁽³⁾ Maximum error in touch sensitivity: self mode: 7.7%, mutual mode: 4.3%

⁽⁴⁾ Maximum error in touch sensitivity: self mode: 150%, mutual mode: 45.5%

⁽⁵⁾ Maximum error in touch sensitivity: self mode: 46.2%, mutual mode: 21.7%

6 Step 4: Hardware Development

The fourth step of designing a successful capacitive touch sensing system is to develop the hardware for the application. This section provides the hardware development guidance for CapTIvate MCUs.

Step 4a: Schematic Design Checklist

Table 10. Schematic Design Checklist

No.	Components	Recommendations
1	VREG pin decoupling capacitor	1 μF , low ESR $\leq 200 \text{ m}\Omega$, placed next to VREG pin
2	DVCC pin decoupling capacitor	4.7- μF to 10- μF tank, 0.1- μF bypass, placed next to MCU
3	Series resistors on CAP I/Os	470 Ω
4	EMI noise filter capacitor	68 pF, required for mutual capacitance mode RX pins if require conducted noise immunity
5	Reset pin resistor and capacitor	47-k Ω pullup with 1-nF pulldown
6	I ² C communication lines pullup resistor	2.2 k Ω
7	Bootloader (BSL)	Refer to the bootloader section of the device-specific data sheet for BSL pin requirement and functions.
8	CAP I/O pins assignment (if possible)	Use CapTIvate Design Center to auto-assign the CAP I/O pins. Leverage all the CapTIvate blocks. Use dedicated CAP I/O pins first. Refer to CapTIvate Pin Selection Guide .

Step 4a.1: VREG Pin Decoupling Capacitor

VREG is the decoupling capacitor of the CapTIvate regulator. The recommended value for the required decoupling capacitor is 1 μF , with a maximum ESR of $\leq 200 \text{ m}\Omega$. Connect VREG as close as possible to the MCU.

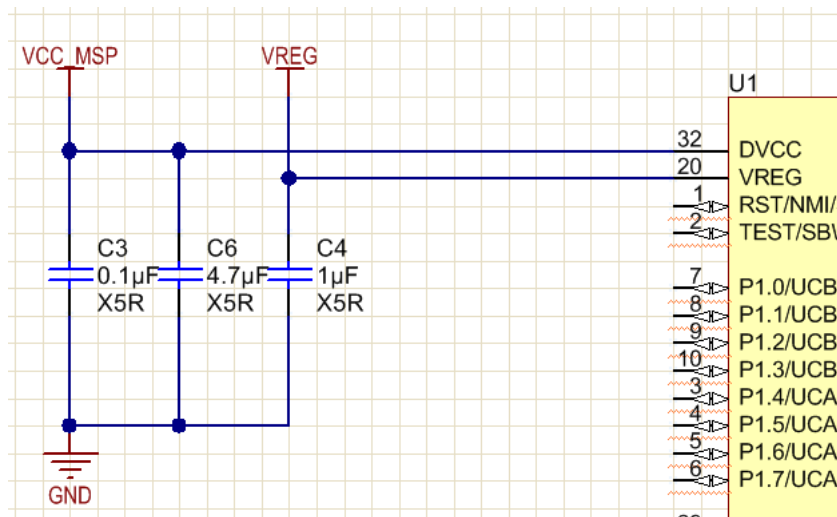


Figure 3. VREG Pin Decoupling Capacitor

Step 4a.2: DVCC Pin Decoupling Capacitor

Refer to the device-specific data sheet for details.

Step 4a.3: Series Resistors on CAP I/Os

These are multipurpose series resistors that work with a TPD1E10B06 transient-voltage suppressor (TVS) diode if the system requires [Electrostatic Discharge \(ESD\)](#) protection. These resistors also help to lower the system emissions and improve RF noise immunity.

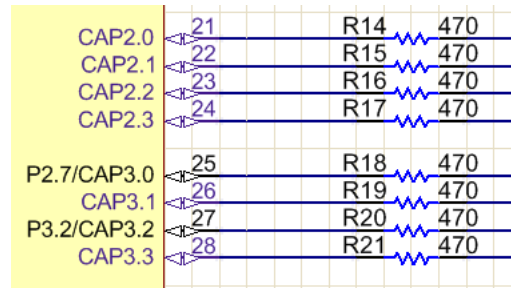


Figure 4. Series Resistors on CAP I/Os

Step 4a.4: EMI Noise Filter Capacitor

For conducted noise immunity considerations, add a 68-pF capacitor to only mutual capacitance mode RX pins. The additional capacitance helps to minimize the effects of EMI. If DVCC mode is enabled, this capacitor can also help to ensure that the ratio of RX parasitic capacitance (C_p) to RX-TX mutual capacitance (C_m) is within the range that is specified in the data sheet.

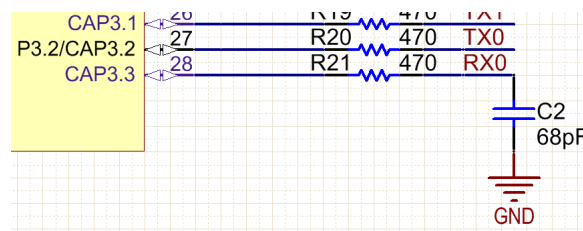


Figure 5. EMI Noise Filter Capacitor

Step 4a.5: Reset Pin Resistor and Capacitor

Refer to the device-specific data sheet for details.

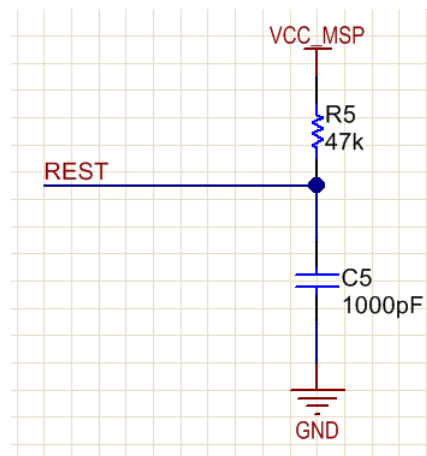


Figure 6. Reset Pin Resistor and Capacitor

Step 4a.6: I²C Communication Lines Pullup Resistor

Refer to the device-specific data sheet for details.

Step 4a.7: Bootloader (BSL)

Refer to the device-specific data sheet and the [MSP430 FRAM Device Bootloader \(BSL\) User's Guide](#) for details.

Step 4a.8: CAP I/O Pin Assignments

Every CAP I/O pin can be configured as a self capacitance or mutual capacitance sensor. To leverage the parallel scanning feature that reduces the overall power consumption and increases the response rate, you must optimize the CAP I/O pins assignment. CapTIvate Design Center has an auto-assign feature to optimize the CAP I/O pins assignment before schematic design. You can also use CapTIvate Design Center to evaluate an existing pin assignment and attempt to reduce the number of scan cycles. [Figure 7](#) shows the auto-assign feature optimize 12 buttons with only 3 time cycles.

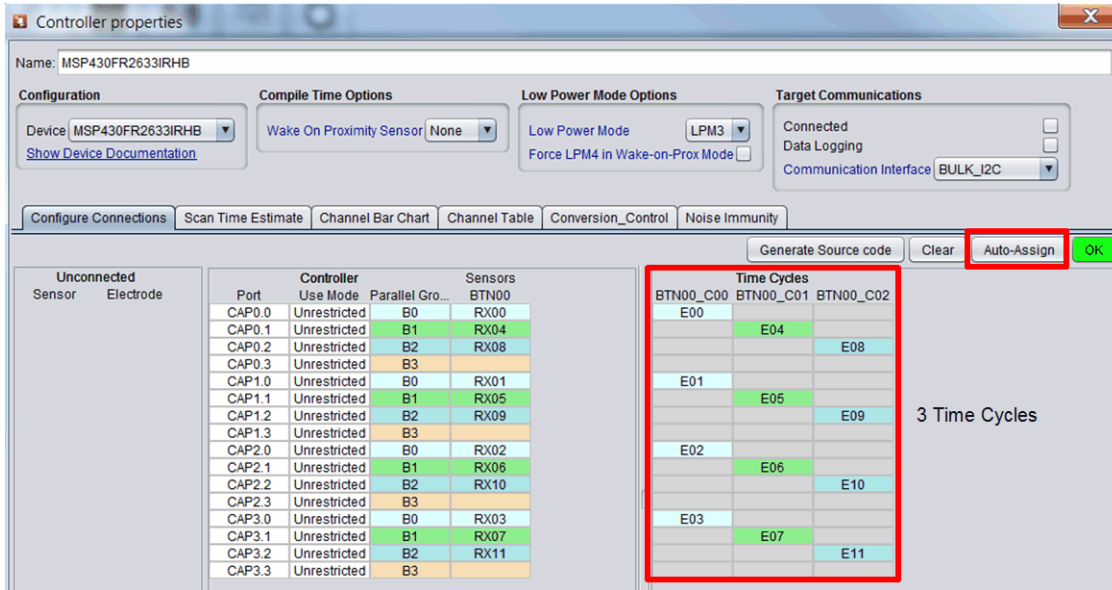


Figure 7. Auto-Assign Feature

Try to use all of the available CapTIvate blocks to enable parallel scanning. The number of CapTIvate measurement blocks on a device determines the number of sensing electrodes that can be measured simultaneously. For example, if a device has 4 blocks and 6 buttons in a design, [Figure 8](#) shows two different pin assignment and the required scan cycles.

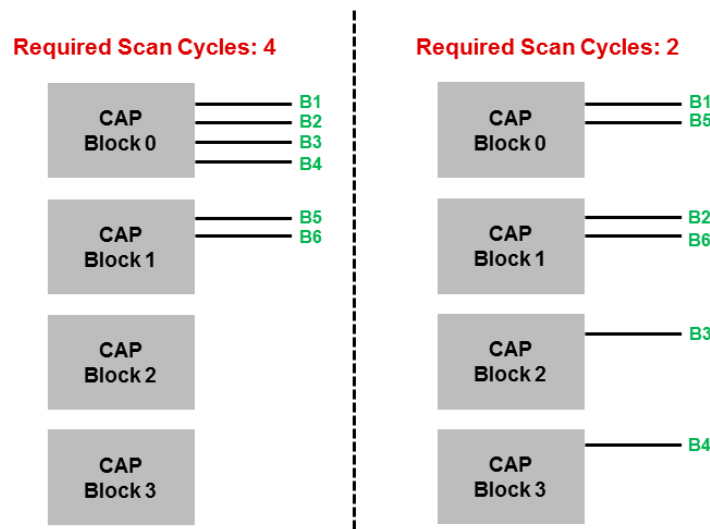


Figure 8. Scan Time Cycles

If the device supports dedicated CAP I/O pins, assign the sensors to those CAP I/Os first. This can save the GPIOs for other uses in the application.

Refer to the [CapTlvate Pin Selection](#) section of the [CapTlvate™ Technology Guide](#) for details.

Step 4b: PCB Layout Design Checklist

Table 11. PCB Layout Design Checklist

No.	Components		Recommendations
1	Buttons (self capacitance)	Size	10 mm and 12 mm (side length of square button, diameter of round button), equivalent to the interaction area
		Shape	Various, typically round or square
		Spacing to surrounding	0.5 x overlay minimum thickness
2	Buttons (mutual capacitance)	Size	10 mm and 12 mm (side length of square button, diameter of round button), equivalent to the interaction area
		Shape	Various, recommend square or shape with corners
		RX thickness	0.5 mm (typical)
		TX thickness	1 mm (typical)
3	Sliders and wheels	Shape	Sliders: Linear shape. Wheels: Circular shape. Refer to Automating Capacitive Touch Slider and Wheel PCB Design Using OpenSCAD Scripts .
		Size	Depends on the required touch area
		Number of electrodes	3 or 4 electrodes
4	Proximity		Varies from design to design
5	Sensor trace	Width	8 mil or as thin as the PCB manufacture allows
		Length	Minimize the length from sensor to controller
6	LEDs		LEDs signals 4 mm away from sensor signals
7	Ground pour		Use hatched pour instead of solid pour. Hatching: 25% (typical) 45°, 8-mil track width, 64-mil grid size
8	Moisture and liquid		Refer to Moisture Design Consideration and the Liquid Tolerant Capacitive Touch Keypad reference design
9	Metal touch		Refer to Capacitive Touch Through Metal Using MSP430™ MCUs With CapTlvate™ Technology
10	Noise immunity		Refer to Noise Immunity Hardware for details.

Step 4b.1: Buttons, Self Capacitance

- A self capacitive button sensor is a single electrode. Self capacitive buttons are simple to layout and each button is assigned to only one pin on the MCU.
- Button size ranges from 6 mm to 15 mm (side length of square button, diameter of round button) and depends on the required touch area.
- The typical shape for a self-capacitance button is rectangular or round.

Refer to [Self Capacitance Buttons Design](#) and [Electrode Size and Overlay Thickness](#) for details.



Figure 9. Button Shape for Self Capacitance

Step 4b.2: Buttons, Mutual Capacitance

- A mutual capacitance button sensor requires two electrodes, one as a TX and the other for RX. TX on outside and RX on inside.
- Button size ranges from 6 mm to 15 mm (side length of square button, diameter of round button), depending on the required touch area.

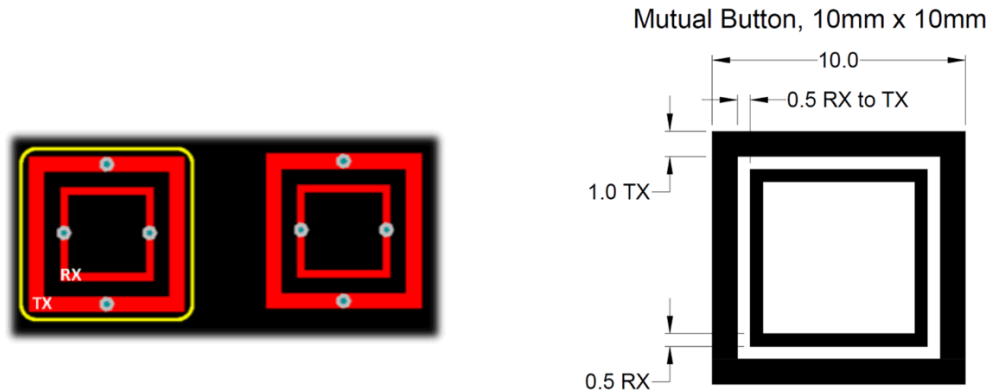


Figure 10. Button Shape for Mutual Capacitance

- [Figure 11](#) shows an optimized layout for selecting TX and RX pins in an application with 12 mutual-capacitance buttons. This layout is optimized to reduce the number of pins and scan cycles.

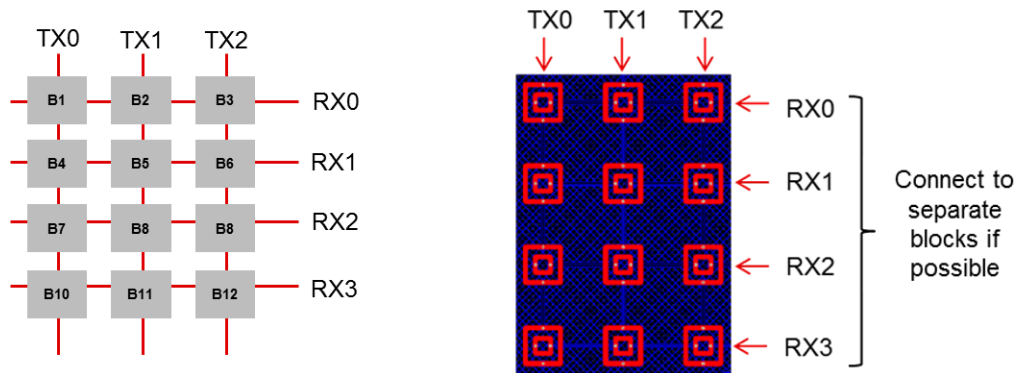


Figure 11. Mutual-Capacitance Matrix

Refer to [Mutual Capacitance Buttons Design](#) and [Electrode Size and Overlay Thickness](#) for details.

Step 4b.3: Sliders and Wheels

- Sliders and wheels have the best performance when they have well-balanced sensitivities across all electrodes.
- The area of the electrode is not as critical as the percentage of coverage across multiple electrodes. Interdigitated slider and wheel designs provide the most efficient and optimal coupling but can be complicated to create. Simpler designs are possible but require experimentation.
- Self capacitance slider and wheel

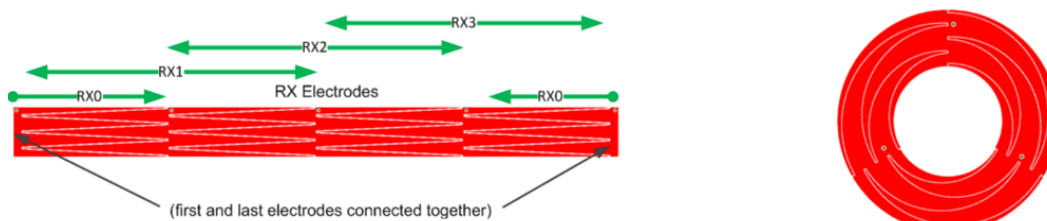


Figure 12. Self Capacitive Slider and Wheel

- Mutual capacitance slider and wheel



Figure 13. Mutual Capacitive Slider and Wheel

Refer to [Slider and Wheels Design](#) and [Implement custom slider/wheel position algorithm](#) for details.

Step 4b.4: Proximity

The sensing range of a proximity sensor is dependent upon several factors:

- The size and shape of the proximity sensor
- The sensor configuration tuning values
- The surrounding conductors
- The system surrounding environment

Refer to [Proximity Sensing Design](#) for details.

Step 4b.5: Sensor Traces

- Keep the width of the sensor traces less than 8 mil or as thin as the PCB manufacture allows
- Minimize the length of the sensor trace from the sensor electrodes to controller
- Do not have 90° turns, because the sharp turns can pick up noise.
- Sensor traces routing considerations:
 - Try to route the sensor trace on the bottom layer of the PCB so the user will not interact with the sensor traces.
 - Do not run digital signals like PWM signals or communications line like I²C or SPI near the sensor trace. It is recommended to keep these types of signals at least 4 mm away from the sensor trace. If the digital signal and the capacitive touch trace must cross, then it is recommended to keep the crossing at a 90 degree angle.

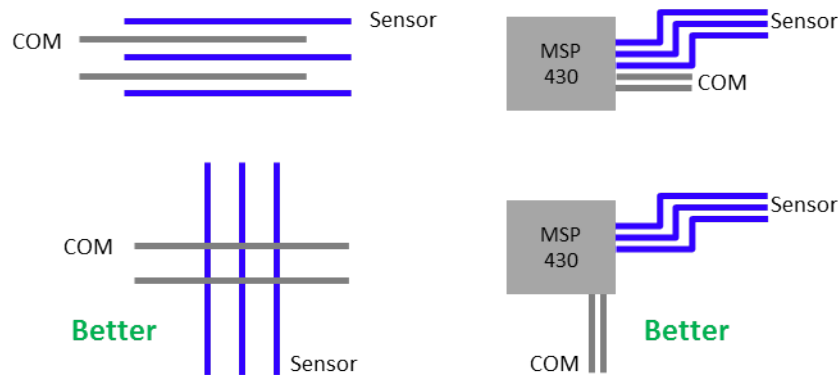


Figure 14. Sensor Traces Routing

- Mutual capacitance routing considerations:
 - Avoid routing TX lines near RX lines when possible. If unable to avoid routing TX and RX on the same layer, put a ground trace between TX and RX. If TX needs to cross RX, route the traces to that they cross perpendicular to each other.
 - Route TX lines next to other TX lines. Route RX lines next to other RX lines.
 - Keep grounds away from the TX and RX traces by half of the panel thickness.

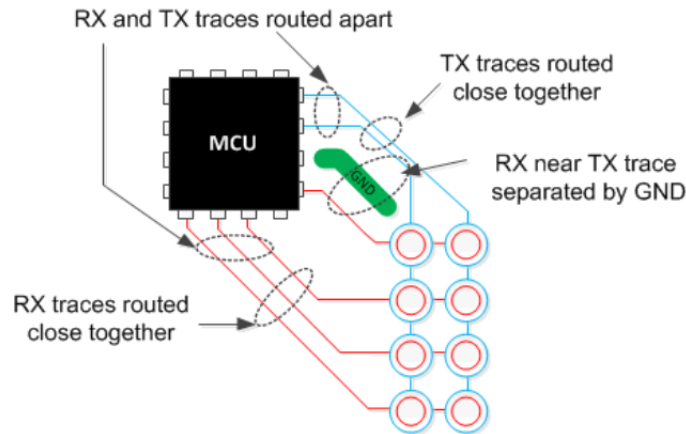


Figure 15. Mutual-Capacitance Routing Considerations

Step 4b.6: LEDs

Signals that drive LEDs (unless the LEDs require high-strength drivers) are similar to other digital signals. As with digital signals, avoid placing LEDs signals near the sensor trace. TI recommends a distance of at least 4 mm for S_{LED} (see Figure 16).

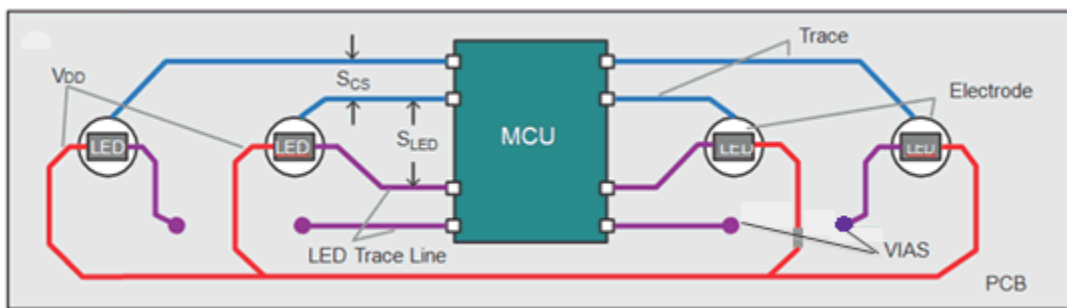


Figure 16. LED Routing

Refer to [LEDs/LED Backlighting](#) for details.

Step 4b.7: Ground Pour

- When adding a ground pour near a sensor electrode, there is a tradeoff between maintaining high sensor sensitivity and increasing the noise immunity of the system.
- The use of a hatched pour instead of a solid ground pour is a good design practice. A hatched pour reduces the parasitic capacitance from the ground pour. Typically, a 25% fill hatch is sufficient, but this percentage can be increased or decreased to improve noise immunity or sensitivity, respectively.
- When designing a button, slider, or wheel of any size or shape, a flexible method to control the distance between any sensor and the ground pour is to provide a poly cut-out region around the sensor (see Figure 17). The gap from the sensors to the surrounding ground can be 0.5 mm to 2 mm and is typically equal to half of the overlay thickness.

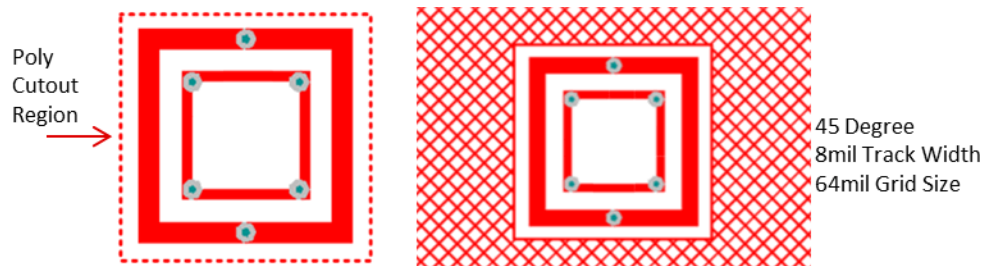


Figure 17. Ground Pour

Refer to [Ground Planes](#) for details.

Step 4b.8: Moisture and Liquid

- Moisture and liquid rejection: The system is designed to block sensor detection if moisture or liquid is detected. Refer to [Moisture Design Consideration](#).
- Moisture and liquid tolerant: The system is designed to work properly when exposed to moisture or liquid with limitations on the sensor spacing and the orientation of the sensor placing. Refer to the [Liquid Tolerant Capacitive Touch Keypad Reference Design](#).

Step 4c: Mechanical Design

[Table 12](#) is a mechanical design checklist.

Table 12. Mechanical Design Checklist

No.	Components		Recommendations
1	Overlay	Material	Typical: glass, polycarbonate, acrylic, ABS. Avoid conductive materials and conductive paints.
		Thickness	10 mm or thinner, depends on material and sensor size
		Stackup	Self capacitance: Avoid an air gap Mutual capacitance: Cannot have an air gap Can use materials to bridge the gap like spring clips.
		Bounding	Typical: optically clear adhesive, screws
2	Enclosure		Varies from design to design
3	Connector		Minimize the trace length from the sensor to the MCU. Make sure that the connector is mechanically stable during operation.
4	Sensor material		Typical: PCB, FPC, copper tape, conductive wire, ITO
5	Noise immunity		Refer to Noise Immunity Hardware .
6	Metal touch		Refer to the metal touch application design guide, Capacitive Touch Through Metal .

Step 4c.1: Overlay

Material and Thickness

- The overlay material's dielectric constant (ϵ_r) and thickness play an important role in determining the strength of the electric field at the surface of the sensor.
- The dielectric constant determines how efficiently electric field propagates through material and the distance between the overlay and the sensor determines how much electric field reaches the target contact area. For example, if all other factors are the same, a 5-mm overlay with ϵ_r of 4 has similar sensitivity to a 2.5-mm overlay with ϵ_r of 2.
- Conductive materials interfere with the electric field. We recommend to not using conductive materials for overlay including the conductive paints.

Stackup and Bounding

- Make sure that there is good contact between the overlay and the sensor electrode to ensure good sensitivity and reliable performance. Always avoid air gaps between the sensor electrode and overlay, especially in a mutual capacitive design. If the mechanical stackup requires a gap, use a filler material.

Pressure sensitive adhesive tape can fill a narrow gap, and mechanical springs can fill a wide gap. No air gap is allowed in a mutual capacitive design.

- The most common bounding method is using nonconductive optically clear adhesive film and this can increase the sensitivity by eliminating the air gap. Typical adhesive used are 3M™ 200MP, 467MP and 468MP.

Refer to [Overlay Design](#) and [Gaps Consideration](#) for details.

Step 4c.2: Enclosure

The choice of enclosure depends on the product industrial design and also the requirement for the capacitive touch subsystem. For example, in moisture tolerant capacitive touch design you should use a non-conductive enclosure if possible. However, metallic enclosure that can be tied to a ground is better for ESD discharge.

Step 4c.3: Connector

- Do not use a connector between the sensor and the MCU if possible, because a parasitic capacitance is associated with the connector PCB footprint and structure.
- If possible, place the MCU on the touch sensor PCB and route power and communication lines through the connector to the main PCB.
- Flex PCB or FPC can be used as an interconnection between PCBs, but make sure that it is mechanically stable during product operation.

Refer to [Connector Consideration](#) for details.

Step 4c.4: Sensor Materials

- Common sensor electrode materials include PCB, FPC, copper tape, conductive wire, and ITO.
- When using copper tape or conductive wire for the sensor electrode, do not run them in close proximity to a metal enclosure or other noise signals in the system.

Refer to [Electrode and Trace Materials](#) for details.

7 Step 5: Firmware Development and System Tuning

The 5th step of designing a successful capacitive touch sensing system is to develop the firmware with the optimized tuning configuration based on the system requirements defined in [Step 2](#) and the hardware developed in [Step 4](#).

TI provides two software development tools that can reduce firmware development effort and time. Download the latest version of the tools.

- CapTivate Design Center Tool ([Download](#))
 - Code Composer Studio IDE ([Download](#))
- or
- IAR Embedded Workbench for MSP430 IDE ([Download](#))

Step 5a: CapTivate Design Center

The CapTivate Design Center is a rapid development tool that accelerates capacitive touch designs for CapTivate Technology enabled MSP430 devices. By helping guide you through the capacitive touch development process, the CapTivate Design Center can simplify accelerate any touch design through the use of innovative user graphical interfaces, wizards and controls. This GUI tool can generate the source code, perform real time tuning and also provide example projects.

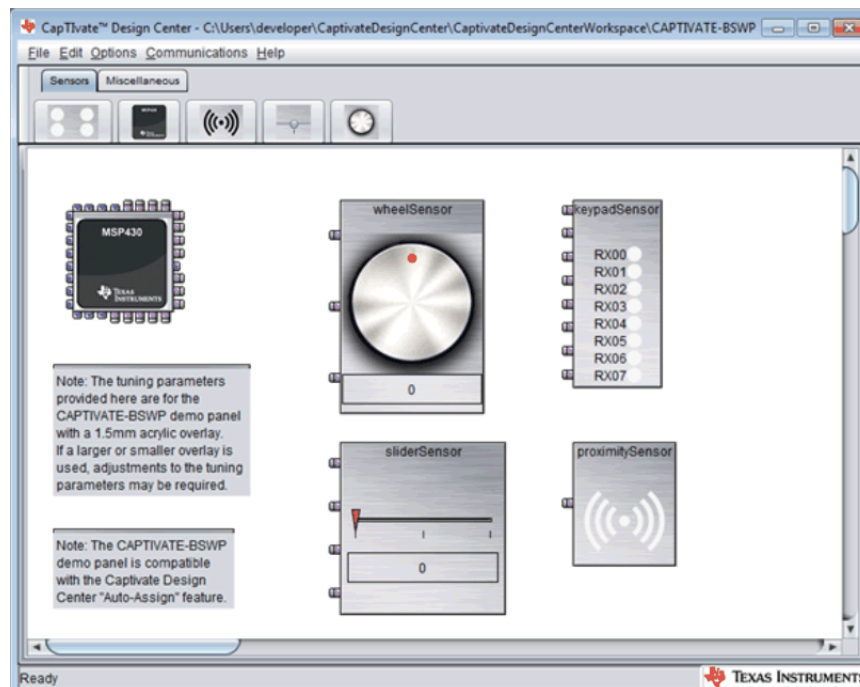


Figure 18. CapTivate Design Center

Step 5b: CapTivate Tuning Process

It is important to first understand how to leverage the software development tools to tune the sensor and then how to actually tune the sensor performance step by step.

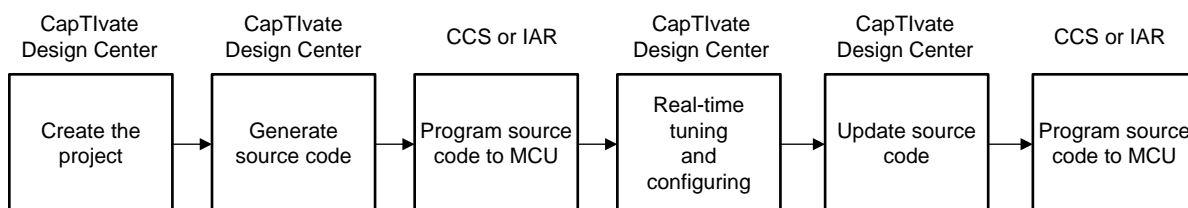


Figure 19. CapTivate Tuning Process

Step 5b.1: Create the Project

- Use CapTivate Design Center to drag and drop capacitive sensing elements to the GUI workspace based on the hardware design.
- Use auto-assign or manually assign the sensor connections to MSP430 capacitive touch I/O pins based on your schematic and layout.
- Select the I²C or UART communication interface to the host PC.

Refer to [Create a new project](#) for details.

Step 5b.2: Generate Source Code

Refer to [Generate source code](#) for details.

Step 5b.3: Program Source Code to MCU

Refer to [Loading and running generated projects](#) for details.

Step 5b.4: Real-Time Tuning and Configuring

After connecting the target MCU to CapTIvate Design Center GUI tool you should be able to display the real-time sensor data, and also to enable reading and modifying the sensor's configuration and tuning parameters.

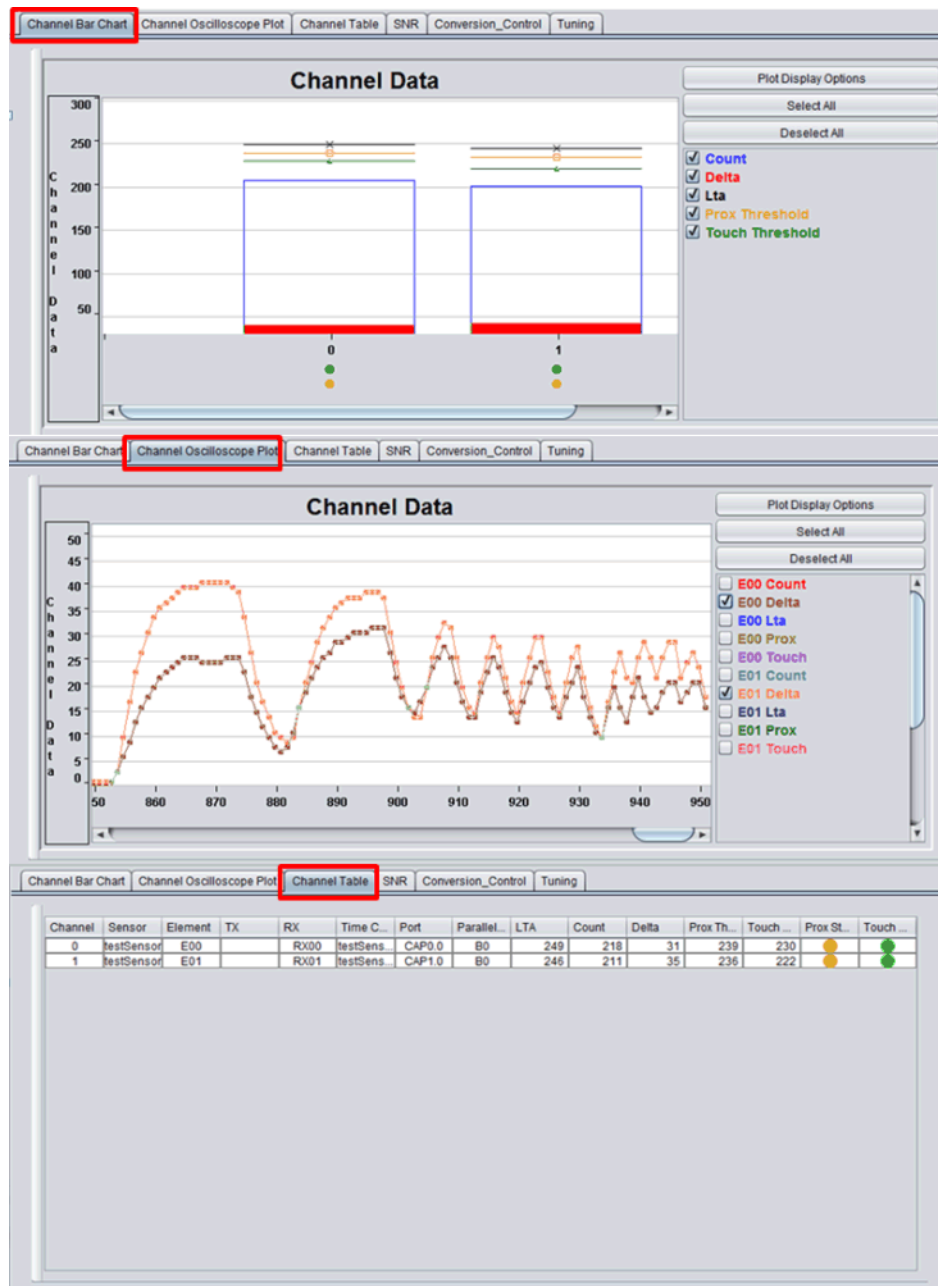


Figure 20. Sensor Parameters

Open the Sensor Properties window by double clicking the sensor element.

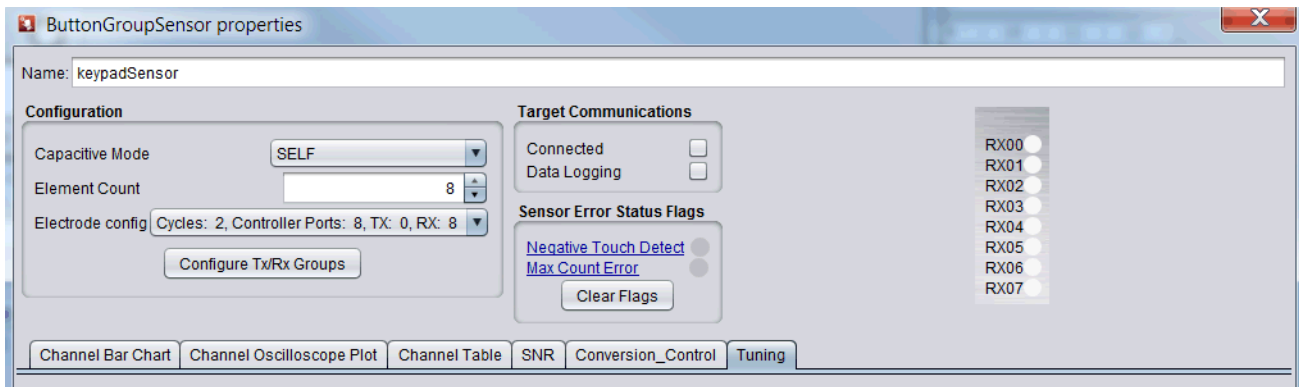


Figure 21. Sensor Properties Window

Open the Controller Properties window by double clicking the controller element.

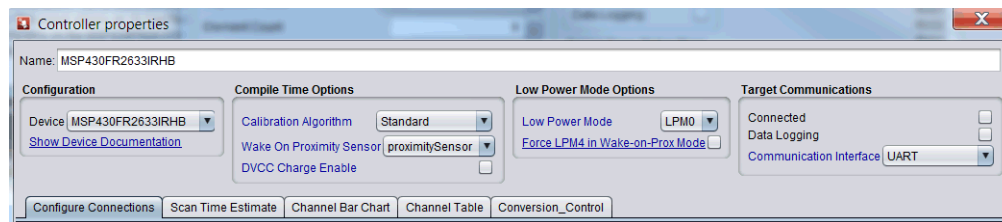


Figure 22. Controller Properties Window

NOTE: Make sure to enable the advance mode to see all the tuning parameters that are listed in this document. In CapTIvate Design Center, click **Options** → **Features** → **Advanced** → **OK** (See Figure 23).

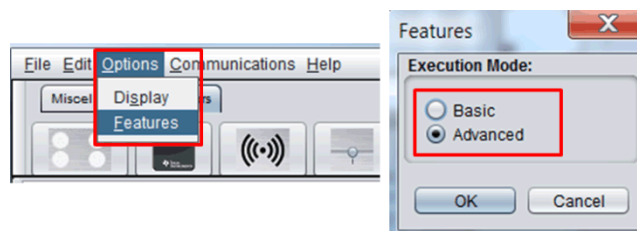


Figure 23. Enable Advanced Mode

Three key areas determine the capacitive touch performance. The purpose of the tuning process is to balance the performance tradeoffs in these three areas.

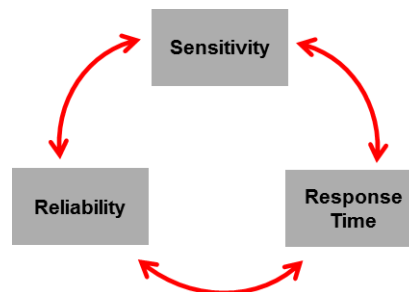


Figure 24. Tuning Consideration

Sensitivity Tuning Checklist

Table 13. Sensitivity Tuning Checklist

No.	Parameter	Location	Recommendations
1	Conversion Gain	Sensor Properties → Conversion_Control	100
2	Conversion Count	Sensor Properties → Conversion_Control	Depends on the sensor and mechanical design. Start from the small number and increase it until reach the desired touch delta.
3	Frequency Divider	Sensor Properties → Conversion_Control	Self capacitance: f/4 Mutual capacitance: f/2
4	Delta	Sensor Properties → Channel Table	Desired touched delta depends on applications

1 and 2. Conversion Gain and Conversion Count

- Conversion Gain and Conversion Count are the fundamental parameters used to establish the sensitivity of the sensor. They are used to adjust the resolution and sensitivity of the sensor.
- Conversion Gain should be set to 100 (Min) for most of the applications.
- Conversion Count should be adjusted starting from the small number and increasing until it reaches the desired touch delta.
- The typical touch delta for buttons is 50 and for sliders and wheels is 50 to 100.
- These two parameters apply to all of the sensor elements in this sensor group.

3. Frequency Divider

- The conversion frequency divider allows for the conversion clock to be divided down from the base rate of 16 MHz. The conversion clock period must be long enough to ensure complete charge transfer phases.
- In typical designs, self-capacitance frequency divider should be set to f/4 and mutual capacitance frequency divider should be set to f/2.

4. Delta

- Delta typically refers to the change in a measurement due to change in capacitance of an electrode caused by a proximity or touch condition.
Delta = Signal Count – Long Term Average (LTA)
- Long Term Average (LTA) is the baseline count of the sensing signal.

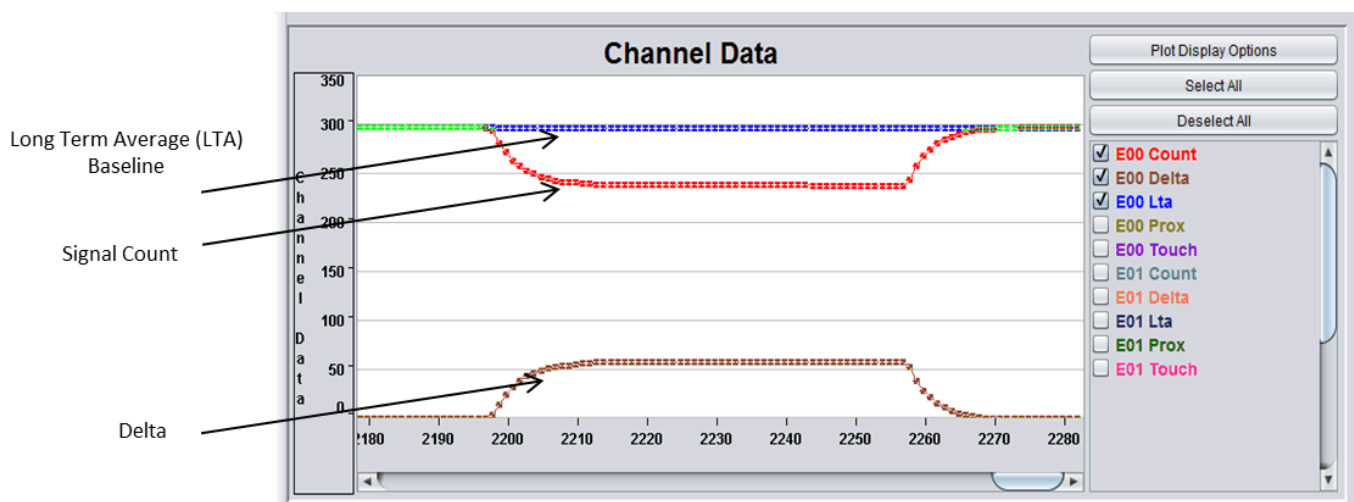


Figure 25. Delta

Table 14 is a reliability tuning checklist.

Table 14. Reliability Tuning Checklist

No.	Parameter	Location	Recommendations
1	Prox Threshold	Sensor Properties → Tuning	This proximity threshold is an absolute value of the counts not percentage. The value depends on applications.
2	Touch Threshold	Sensor Properties → Tuning	This touch threshold is a percentage value of LTA. The value depends on the PCB and overlay design.
3	Signal-to-Noise Ratio Measurement	Sensor Properties → SNR	Run SNR measurement before moving forward Make sure Advice = GOOD
4	Prox Debounce	Sensor Properties → Tuning	Debounce In = 1 Debounce Out = 1
5	Touch Debounce	Sensor Properties → Tuning	Debounce In = 2 Debounce Out = 2
6	Noise Immunity	Controller Properties → Compile Time Options	If noisy environment and system → enable noise immunity
7	Error Threshold	Sensor Properties → Tuning	Long term average (LTA) x 2
8	Sensor Timeout Threshold	Sensor Properties → Tuning	65535 (off) or depends on application
9	Count Filter Beta	Sensor Properties → Tuning	1 or 2
10	LTA Filter Beta	Sensor Properties → Tuning	7

1. Prox Threshold

- This is used to set the level of interaction required by the user for proximity detection.

$$\text{Sensor State} = \begin{cases} \text{Proximity On} & \text{if } \Delta > \text{Prox Threshold} \\ \text{Proximity Off} & \text{if } \Delta < \text{Prox Threshold} \end{cases}$$

Figure 26. Proximity Threshold

- If the application is proximity detection, put your hand at the distance where proximity detection should trigger. Record the delta for that sensor and set the Prox Threshold to be the delta. Also makes sure the Prox Threshold is larger than the delta caused by the noise.
- If the application is only touch detection, just need to make sure the Prox Threshold is larger than the noise caused delta and also larger than the Touch Threshold delta.

2. Touch Threshold

- This is used to set the level of interaction required by the user for touch detection.
- It is a percentage of the Longer Term Average (LTA) baseline value. And it depends on the PCB and overlay design.

$$\text{Sensor State} = \begin{cases} \text{Touch On} & \text{if } \Delta > \text{LTA} * (\text{Touch Threshold} / 128) \\ \text{Touch Off} & \text{if } \Delta < \text{LTA} * (\text{Touch Threshold} / 128) \end{cases}$$

Figure 27. Touch Threshold

- You can select the Touch Threshold for individual elements in a sensor group.

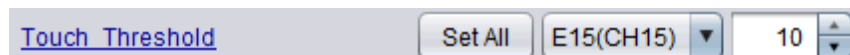


Figure 28. Change Touch Threshold

3. SNR Measurement

- After you completed the Sensitivity Tuning Checklist and set the Prox Threshold and Touch Threshold. Using the SNR Measurement tool to check the system reliability based on the tuning parameters. This is a critical step of the tuning process.
- Make sure the advice from the tool is **GOOD** and then move on to the tuning process. Otherwise, go back to adjust the tuning parameters or change the hardware design. Refer to [Sensitivity, SNR, and Design Margin in Capacitive Touch Applications](#) for details.

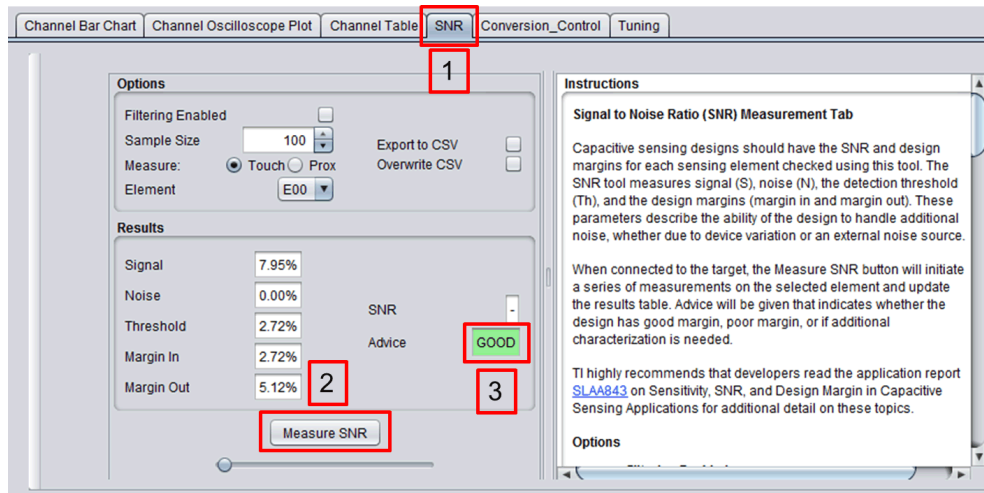


Figure 29. SNR Measurement

4 and 5. Prox Debounce and Touch Debounce

- They are used to select the number of consecutive samples that the sensor must be in detection to generate a detection state to the system. This is to prevent transient noise signal to cause false touch detections.

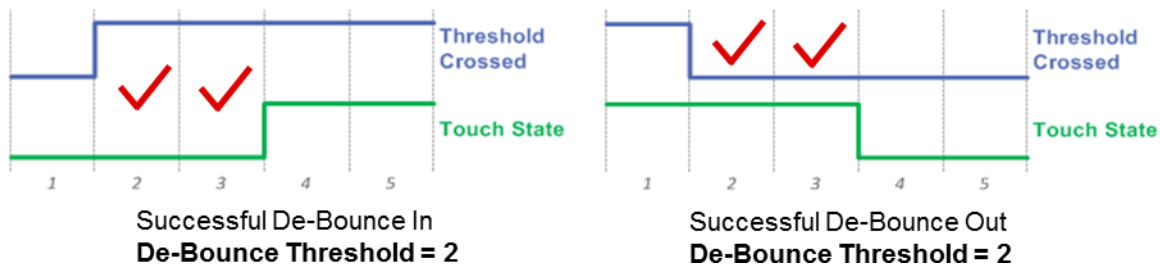


Figure 30. Proximity and Touch Debounce

- The Touch Debounce parameters are generally set to 2 for reliable sensor status detection. This value can be raised or lowered based on the noise aspects of the system.

6. Noise Immunity

Based on the operating conditions defined in Step 2, if the application requires noise immunity feature, you can enable the software noise immunity feature in CapTIvate Design Center. Refer to [Noise Immunity Tuning Check List](#) for details.

Table 15 is a response time tuning checklist.

Table 15. Response Time Tuning Checklist

No.	Parameter	Location	Recommendations
1	Active Mode Scan Rate (ms)	Controller Properties → Conversion Control	Typical HMI application, 33 ms (30 Hz)
2	Scan Time Estimation	Controller Properties → Scan Time Estimation	
3	Touch Debounce	Sensor Properties → Tuning	Debounce in = 2 Debounce out = 1

1. Active Mode Scan Rate

- The active mode scan rate specifies the period (in milliseconds) to refresh the user interface at when in active mode.
- To convert to samples per second (sps), divide 1000 by the specified scan rate period. For example, a scan rate of 50 ms equates to 20 sps.
- A shorter scan rate period increases application response rate, but the tradeoff is higher power consumption. Typical HMI applications use a scan period of 33 ms (30 Hz).

2. Scan Time Estimation

Make sure the scan rate period is longer than the time takes to scan all the sensors. This window shows the estimated:

- Time takes to scan each sensor
- Time takes to scan all the sensors
- Scan rate period

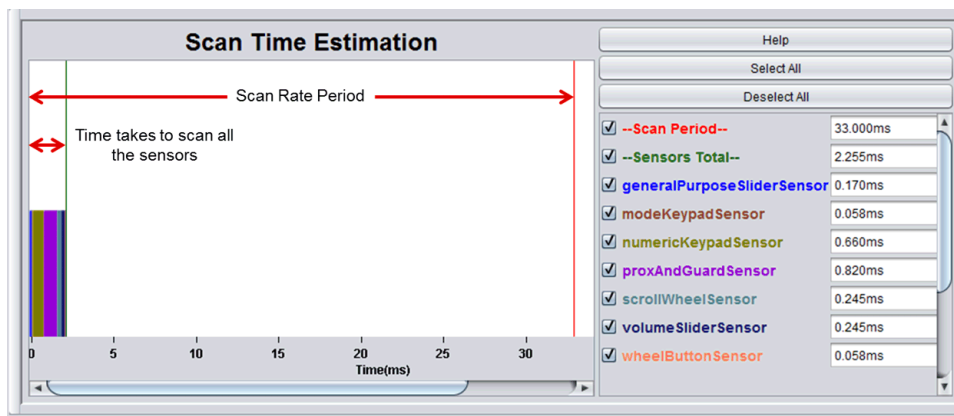


Figure 31. Scan Time Estimation

3. Touch Debounce

- This parameter was set in the reliability checklist and also affects the response time.
- The time it takes for the system to report a touch signal after the sensor touch is detected can be calculated as shown in Figure 32.

$$\text{Sensor Response Time} = \frac{\text{Debounce In}}{\text{Sample per Second}}$$

Figure 32. Response Time Equation

Step 5b.5: Update Source Code

After real-time tuning and configuration, you can use CapTlvate Design Center to update the source code with the latest tuning parameters. The output directory must be the IDE source code project directory.

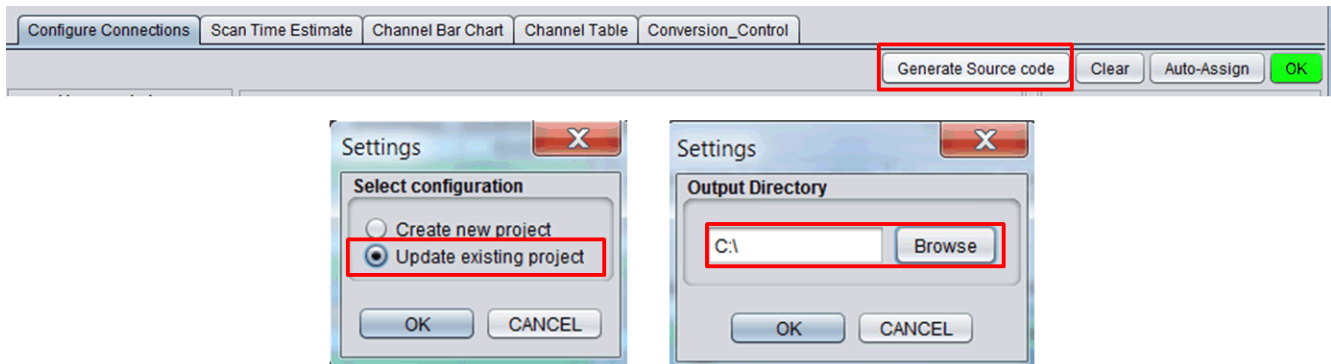


Figure 33. Update Source Code

Step 5c: Firmware Development

After the basic CapTlvate tuning process, the next step is to complete the firmware development based on specific application requirements.

Step 5c.1: Accessing the Sensor Data

- In general, the CapTlvate software library operates on the principle of refreshing data in objects, rather than returning results directly through a function call.
- It is the responsibility of the application to access the results of a measurement in the appropriate object data structure. A callback function can be registered with any sensor that must be called whenever the sensor data is refreshed.
- The following links are to examples of how to access the sensor data:
 - [Register a callback function](#)
 - [Access element state data](#)
 - [Access element measurement data](#)
 - [Access dominant button](#)
 - [Access slider or wheel position](#)

Step 5c.2: Customize Communication Code

- Many applications require an interface to communicate the capacitive touch status to a host or driver in the system (for example, another MCU, an LED driver, or a motor driver).
- This interface could be I²C, SPI, UART, or GPIO.
- The CapTlvate library includes a communications module to connect CapTlvate MCUs to external components through the UART or I²C serial interface. Refer to [Communications Module](#) for details.

Step 5c.3: Optimize Low Power Consumption

- If the application is battery powered and requires extremely low power consumption to extend the battery life and product shelf life, it is critical to optimize the system power consumption.
- The CapTlvate peripheral on MSP430 MCUs enables user interface designs with extremely low power consumption. Low power consumption is possible because the CapTlvate peripheral includes a processing state machine that performs basic functions while the MCU is in LPM3 without any CPU interaction.

Refer to [Ultra Low Power Optimization](#) for details.

8 Step 6: System Integration and Prototyping

The sixth step of designing a successful capacitive touch sensing system is to integrate all the system components together and build the prototype units for validation and field testing. The components include PCBs, overlay, enclosure, firmware, and power supply.

TI recommends building the prototype units as much like the final production units as possible to accurately validate the performance and reliability of the system design.

9 Step 7: System Validation and Field Testing

The seventh step of designing a successful capacitive touch sensing system is to perform the system validation with the prototype units under all the expected operating condition. TI recommends building 20 to 50 units for field testing.

If the system meets performance requirements, the design can move to mass production. If the system does not meet performance requirements, you can either adjust the performance expectation or revisit the system design.

Table 16 is a system validation check list.

Table 16. System Validation Check List

No.	Design For Test	Recommendations
1	CapTIvate Design Center Communication Interface	Place a UART or I ² C communication interface on the board for connecting to CapTIvate Design Center. This can be removed on production units.
2	Function Testing	Test the unit with all the functions enabled in the design.
3	EMC Testing	Define the EMC testing standard and pass criteria.
4	Environment Drift Testing	Define the environment drift testing standard and pass criteria.
5	Signal-to-Noise Ratio Margin Analysis	Perform SNR measurement using CapTIvate Design Center with all of the noise sources enabled in the design. Make sure Advice = GOOD

1. CapTIvate Design Center Communication Interface

- It is much easier and more reliable to monitor the capacitive touch performance by using the CapTIvate Design Center real-time tuning feature.
- It is highly recommended to leave a UART or I²C communication interface connection for CapTIvate Design Center on the prototype and field testing units. They can be disabled or removed on production units.

2. Function Testing

To ensure the reliability of the product, make sure to test the capacitive touch performance with other functions enabled in the design and also with switching other functions on and off. The other functions may include: LED driver, motor driver, wireless communication, near-field communication(NFC), battery power supply.

3. EMC Testing

- Define the EMC testing standard and stress level as well as the pass criteria.
- Example: Class A pass/fail criteria based on IEC 61000-4 standard for equipment under test (EUT)
 - The EUT must not exhibit any false touch detections during or after the test
 - The EUT must always detect valid touches during and after the test.
 - If the EUT contains slider or wheel sensors, the position must be reported accurately to within an acceptable limit during and after the test.
 - The EUT must not experience a processor reset during the test.

Refer to [Noise-Tolerant Capacitive-Touch Human-Machine Interfaces Design Guide](#) for complete details on EMC testing.

4. Temperature Drift Testing

- Define the product operating temperature range, fastest expected change rate and the pass criteria.
- Define the product operating humidity percentage range and pass criteria.

5. Signal-to-Noise Ratio Margin Analysis

Perform SNR Measurement using CapTivate Design Center with all the noise sources enabled in the system and make sure you are getting the GOOD advice. Refer to [Sensitivity, SNR, and Design Margin in Capacitive Touch Applications](#) for details.

10 Step 8: Mass Production

The final step of designing a successful capacitive touch sensing system is to prepare for mass production.

[Table 17](#) is a mass production check list.

Table 17. Mass Production Check List

No.	Items	Recommendations
1	Production Test Plan	Work with your manufacturer on the production test plan and specifically on capacitive touch performance pass criteria.
2	Production Firmware	Remove the development software.

1. Production Test Plan

- TI strongly recommends that you work closely with your manufacturer on the production test plan. Make sure that the production line process is consistent on all units to ensure reliable performance.
- The capacitive touch production test plan includes capacitive touch test fixtures and system pass/fail criteria.

2. Production Firmware

Remove all the development software like CapTivate Design Center communication interface code. Add production-line system diagnostic code, if necessary.

11 References

1. [CapTivate™ Technology Guide](#)
2. [MSP430™ Capacitive Touch Sensing Microcontrollers](#)

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from May 2, 2019 to August 14, 2019	Page
• Updated the "Pins required for 12 buttons" for mutual capacitance in Table 1 , <i>Comparison of Self Capacitance and Mutual Capacitance</i>	5
• Added Table 9 , <i>CapTivate Key Parameter Performance Under Different Test Conditions</i>	10
• Changed the minimum button size to 6 mm in "Step 4b.1: Buttons, Self Capacitance" in Section 6 , <i>Step 4: Hardware Development</i>	14
• Changed the minimum button size to 6 mm in "Step 4b.2: Buttons, Mutual Capacitance" in Section 6 , <i>Step 4: Hardware Development</i>	15

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL 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 AND 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 skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI 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. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2019, Texas Instruments Incorporated