



White Paper

Maximizing Touchscreen Performance by Pairing Powerful Circuitry with Intelligent Software

Abstract

Capacitive touchscreens have become the dominant user interface technology for smartphones and tablet computers, and are being rapidly adopted in many other device categories.

Product designers who want to include a capacitive touchscreen in their device need to satisfy a large number of requirements simultaneously. When these requirements conflict with each other, designers have always had to endure painful trade-offs in the quality of their design: "Should I compromise on the responsiveness of the screen to improve robustness to noise?"

Some touchscreen controllers rely on a hardware-centric approach to overcome these issues, achieving noise robustness at the expense of performance and design simplicity. More sophisticated touch controllers combine novel hardware with intelligent firmware, enabling them to meet all of these requirements without compromises.

Designing a Touch Interface

The touch input system is one of the most crucial parts of the user interface in today's consumer devices. So, the design of a high-quality touch sensing system must start by considering the user experience. Some of the key criteria are listed below.

Responsiveness

Consumers expect an instantaneous reaction as they move their fingers around on a touchscreen. A good touch controller enables this by sending a very rapid stream of update messages to the operating system so that the application always knows the current finger position. The number of updates per second is called the report rate.



Long Battery Life

Consumers want their battery-operated devices to last for a long time without charging. This means that each component in the system, including the touch controller, has to draw as little power as possible.

Portability and Affordability

The trend in modern consumer devices is to make them thinner, lighter and more affordable. As a result, companies are keen to reduce the number of layers of glass, plastics and glue used to construct a touchscreen.

Stylus Support

Although finger input is great for content consumption, the pen or stylus offers many advantages for content creation, allowing versatility of writing and drawing. A good touch interface should support both forms of input.

Robustness

The touchscreen interface should operate normally in all the environments it is likely to encounter during normal usage. It should not be prone to electrical interference from radio frequency antennae, liquid crystal or OLED displays, battery chargers, fluorescent lights, or other common sources of electric fields.

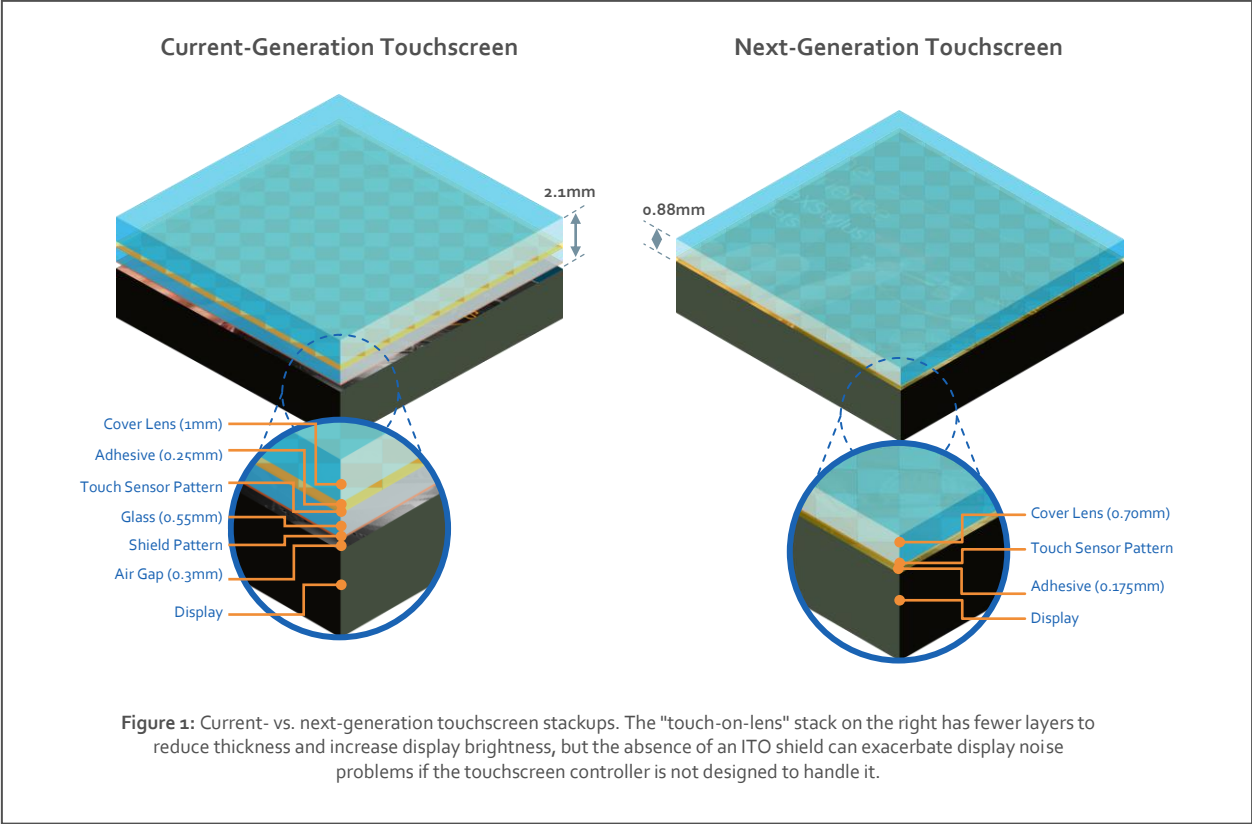
Time to Market

System designers should choose components that are simple to integrate into their devices. Touch controllers that come pre-loaded with standard firmware can drastically reduce development and debug cycles. Instead of writing complex touch sensing code, the designer simply selects a set of parameter values to suit the specifics of his or her project and applies these over a standard 2-wire or 3-wire interface.

Implementation Challenges

With so many requirements to fulfill simultaneously, creating a high-quality touch interface is very challenging. The different requirements can sometimes conflict with each other, resulting in painful trade-offs unless the system is designed in a smart enough way.

For example, the requirement for portable, affordable devices drives manufacturers to make touchscreens with fewer layers of materials. To achieve this, sensors are now being built without the indium tin oxide (ITO) "shield" layer that traditionally protected the touch sensor against interference from the display device underneath.



Without a shield, the interference from the display underneath the sensor can be many times stronger than the signal from a finger above. Dealing with this condition presents serious challenges for the touchscreen controller chip.

Another challenge for portable touch systems comes, perhaps surprisingly, from battery chargers. In-car chargers, wireless chargers, after-market chargers and others often use switching techniques to control their output voltage. Noise at the switching frequency can impact touchscreen operation.

The following illustration shows the image displayed by a smartphone drawing application when a user traces out a set of diagonal lines with his finger. Each image was captured on the same smartphone, with different chargers connected. This smartphone clearly uses a touchscreen controller that does not have sufficient robustness to charger noise.

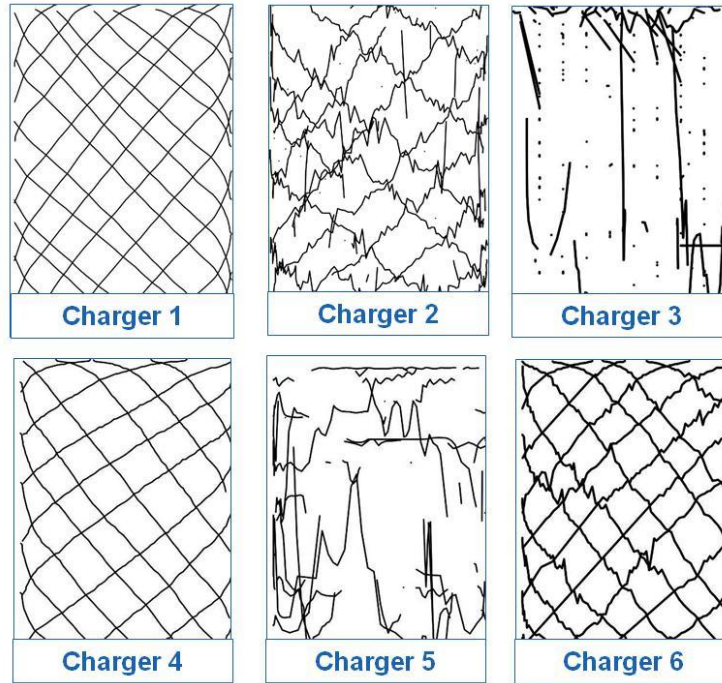


Figure 2: Smartphone touchscreen line drawing test results with various chargers. This touchscreen exhibits false touches, poor linearity and high jitter with some of the chargers – hallmarks of a poor user experience.

Current Solutions

There are some relatively straightforward methods that can be used to combat these challenges individually. But to tackle all of the challenges at the same time, more sophisticated systems are needed.

Let us review these simpler techniques before considering more advanced systems.

High Voltage Drive

The touch controller IC works by driving voltage signals into the sensor and measuring how much each capacitor in the array charges up as a result. Noise from displays and chargers changes the effective signal voltage, thus upsetting the measurements. But if the signal voltage is increased, the interference appears relatively smaller, resulting in a cleaner measurement.

This technique is simple and effective, so it is being widely adopted in touchscreen controllers. For example, devices in the maXTouch® S Series from Atmel utilize up to 24V equivalent high-voltage drive to increase signal-to-noise ratio in noisy conditions.

Display Synchronization

The noise generated by LCD, OLED and other display devices is not constant in time. It follows a repeating pattern of noisy and quieter periods as it continually refreshes the image. Starting with a horizontal line of pixels at the top of the image, it updates the color values on each horizontal line until it reaches the bottom of the screen, before returning to the top again to start the next frame.

There are usually slight pauses in electrical activity between successive horizontal lines and longer pauses between frames. Some touch controllers try to take advantage of these pauses (known as “blanking intervals”), since the level of noise is generally much lower during these times.

However, there are several practical limitations to this approach. Firstly, very few display devices are designed to provide synchronization signals to other components such as touch controllers. This makes it harder for the touch controller to know when to expect quiet periods. Secondly, the blanking intervals are not always quiet and, in some cases, are too short to be useful. Finally, and most importantly, synchronizing to the display device means that the touch controller loses the freedom to choose its operating frequency. So although it successfully avoids the display noise, it becomes more difficult to avoid noise from other sources such as chargers.

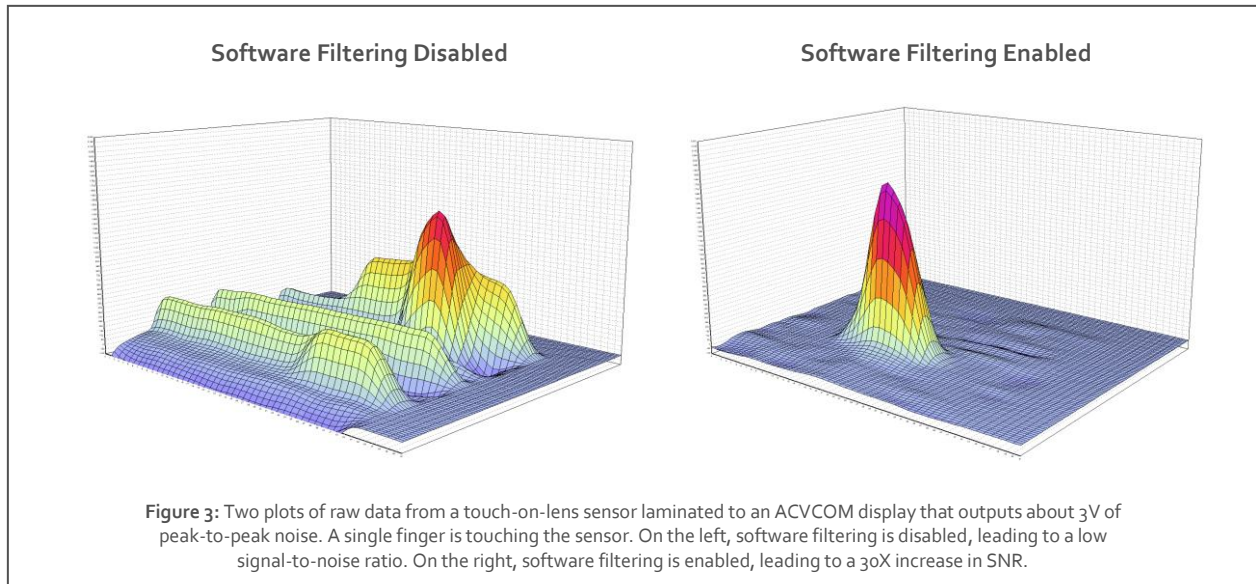
For these reasons, display synchronization is not a practical approach in a real system.

Next-generation Solutions

Software techniques can also be of significant benefit in noisy conditions.

In fact, it is possible to almost completely remove the effects of display noise using only software filters, without the need for display synchronization or high voltages.

The plots below illustrate the effectiveness of software filtering. In these plots, the vertical Z axis represents the size of the capacitive signal measured by the touch controller at each point on the screen. The X and Y axes represent the surface of the touch sensor itself.



Since this kind of algorithm does not rely on any synchronization information, it can be used with displays of any type and any resolution. Additionally, it performs the filtering without requiring additional listening channels or non-standard modifications to the touch sensor electrode configuration. This allows the system designer to simply choose whatever display he or she desires, secure in the knowledge that it will always “just work”.

Furthermore, this technique means that the touch controller does not have to consider the display characteristics when selecting an operating frequency. With this extra degree of freedom, it is better able to avoid noise and interference from other sources such as chargers. Such flexibility is crucial in real-world consumer devices, which are subject to more than one noise source at a time.

While the above data illustrates that software alone can overcome display noise, it does not address the negative impact that running such algorithms in software can have on screen responsiveness and power consumption. This illustrates the peril of focusing too much on eliminating the effects of noise without addressing the impact of such measures on other aspects of touchscreen performance. It is for this reason that a sophisticated hybrid of software and hardware is necessary to achieve all requirements without compromise.

To run these algorithms without slowing down the touch interface or consuming extra power, the touch controller must efficiently handle the extra processing load. An ASIC

customized for touch applications can use specialized hardware blocks to do the computations with minimum delay and power impact. For example, devices in Atmel's maXTouch S Series feature hardware acceleration of display noise elimination algorithms. This is in contrast with a solution based on standard processors, which would not have the specific hardware functions needed, forcing the designer to trade responsiveness and battery life for noise performance.

This maxim of pairing both hardware and algorithms together applies equally well to charger noise. While high-voltage panel scanning provides improvements in signal-to-noise ratio that scale linearly with the applied voltage, pairing such circuitry with intelligent firmware can lead to additional gains in SNR. For example, devices in the maXTouch S Series employ active noise avoidance in addition to high-voltage scanning to overcome charger noise. Intelligent firmware in the devices continuously monitors the background noise level in the environment, and reacts autonomously to change the scanning parameters of the analog circuitry to avoid frequency bands that exhibit high levels of noise. This level of autonomous decision-making is necessary, since the noise profile of the system will vary over time. This is especially true with battery chargers, as the charger switching frequency (and by extension, its noise profile) will change depending on the battery load current.

Conclusion

To meet the demands of today's user interfaces, touch controllers have to deal with many challenges simultaneously, rather than one at a time. In this paper we have outlined examples of how multiple challenges can be addressed using a combination of techniques that operate in tandem.

Next-generation touchscreen controllers, such as the Atmel maXTouch S Series, utilize both powerful circuitry and intelligent algorithms to meet these challenges and maximize touchscreen performance.