# *Palm rejection on resistive touchscreens*

*by Geoff Walker*

Have you ever used a device such as a webpad and had the cursor jump around or disappear under your hand as you tried to write something on the screen? If so, you've experienced a lack of palm rejection.

Palm rejection allows a user to rest their hand or other object on a touchscreen without activating it. Palm rejection is usually significant only in devices with screens larger than about four inches, since with smaller screens the user's hand rests mostly on the screen bezel. It's also usually significant only in devices that are intended to be used with a pen or stylus, since if a device is meant to be used only with pure finger-touch, there's no reason for the user to rest their hand on the screen.

Palm rejection isn't really about your palm; it's about the part of your hand that pokes the writing surface when you hold a pen. For some people, it's the skin protrusion resulting from the creases on the edge of your hand; for others, it's the knuckles on your little finger. Both of these are relatively sharp points that act like a finger-touch. Resistive touchscreens can't handle multiple touches, so the cursor goes somewhere other than where it should be.

How common is poor palm rejection? Unfortunately, it's very common. It seems as though almost every pen tablet created in the last five years has had poor palm rejection (Tablet PCs don't count, since they use active electromagnetic digitizers rather than touchscreens). Hardware OEMs just don't seem to think about this issue during design, and the touchscreen vendors contribute by minimizing the issue or appearing uninterested in doing the required customization.

Poor palm rejection can kill a product. Here's an example from my consulting experience. A few years ago a large hardware OEM bought into the Microsoft hype and created a very nice "Smart Display" (originally known as a "Mira" device). Unfortunately, since Microsoft's basic concept was seriously flawed, the product didn't sell very well. The OEM tried re-purposing the product as a vertical-market data collection tablet (i.e., a wireless thin client), at which point they discovered that none of their enterprise prospects were willing to buy the device. Prospects found that having to hold their hand above the screen while using the product (due to lack of palm rejection) was just too uncomfortable to tolerate for any length of time. That OEM's Smart Display is still a product in search of a market.

**How Palm Rejection Works:** Figure 1 shows the basic components of a typical resistive touchscreen. A touchscreen contains two layers of transparent conductor (ITO, indium tin oxide), one on top of the glass substrate and one on the underside of the PET (polyethylene terephthalate) top layer. The two layers are kept apart by transparent spacer dots. When a finger or pen presses on the top surface, the transparent conductors make contact and the touchscreen controller determines the touch position.

The key to palm rejection on a resistive touchscreen is the spacing of the spacer dots. Most touchscreen vendors use spacing (pitch) of around 3.5 mm and a dot size of around 0.1 mm. This is a compromise that allows the touchscreen to work both with large, blunt objects such as a finger and with small, sharp objects such as a pen.

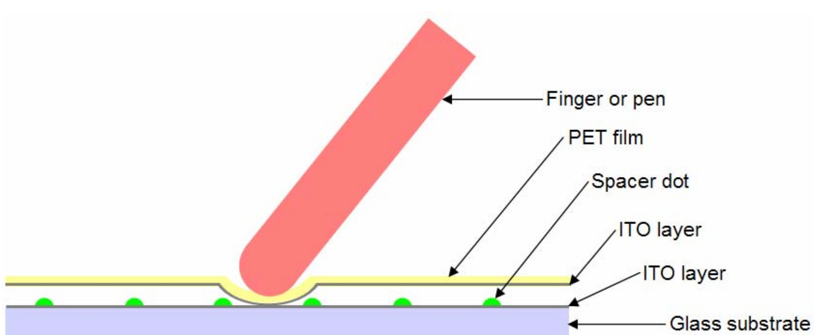


*Figure 1: Basic components of a typical resistive touchscreen*

In fact, almost all standard resistive touchscreens are marketed as being able to work with "finger, fingernail, gloved hand, pen or stylus". This positioning in itself is a warning that most standard resistive touchscreens don't include palm rejection.

If the spacer dots are far apart, pressure from either a large-radius object or a small-radius object can cause the conductors to touch. If the spacer dots are close together, only a small-radius object can cause the conductors to touch, since a large-radius object covers too many spacer dots. Of course, if the user exerts enough pressure, any size object can cause the conductors to touch.

The definition of a "small-radius object" is fairly consistent among the eight well-known touchscreen vendors shown in Table 1 – it's usually a rigid pen tip of diameter 1.6 or 2.0 mm. Touch International is the only oddball at 0.8 mm diameter, and their spec is probably a typo. The definition of a "large-radius object" varies a bit more by vendor. It's usually a silicone rubber hemisphere, ranging in diameter from a low of 6 mm (Gunze) to a high of 25 mm (SMK). If you press your index finger to one side of a piece of glass and measure the flattened area on the other side, you'll probably find that it's around 15 mm in diameter, so the 16.3 mm average in Table 1 is about right. Why is this spec important? A touchscreen that's properly designed for a 6 mm finger should have better inherent palm rejection than a touchscreen designed for a 25 mm finger because the spacer dots in the former should be closer together.

| **Touchscreen Vendor** | **Standard Pen-Only** | **Spacer Option** | **Pen Diameter** | **Finger Diameter** | **URL** |
|---|---|---|---|---|---|
| 3M | No | Yes | 2.0 mm | 15.9 mm | http://www.3m.com |
| Bergquist | No | Implied | 2.0 mm | 19.1 mm | http://www.bergquistcompany.com |
| DigiTech Systems | No | No | 1.6 mm | 16 mm | http://www.digitechsys.co.kr |
| ELO | No | No | No spec. | No spec. | http://www.elotouch.com |
| Fujitsu | Yes | Yes | 1.6 mm | 16 mm | http://www.fujitsu.com |
| Gunze | No | No | 1.6 mm | 6 mm | http://www.gunzeusa.com |
| SMK | No | No | 1.6 mm | 25 mm | http://www.smk.co.jp |
| Touch International | No | No | 0.8 mm | No spec. | http://www.touchinternational.com |
| | | ***Average*** | ***1.6 mm*** | ***16.3 mm*** | |

Table 1: Palm-rejection related specifications from eight touchscreen vendors
(*Data taken from vendor websites on 11/25/05*)

**Designing for Palm Rejection**: If you're designing a product that uses a resistive touchscreen and you want palm rejection, it's almost always necessary to specify a custom touchscreen. Only one of the vendors shown in Table 1 (Fujitsu) offers a standard "pen-only" (palm-rejecting) touchscreen. Fujitsu and 3M both offer the ability to customize the spacer dots; Bergquist implies that they offer the ability without making it an explicit option. None of the other vendors acknowledge the issue of palm rejection in their published specifications.

One of the unfortunate side effects of using a pen on a resistive touchscreen is that it shortens the life of the touchscreen. Lifetime (durability) specs for four-wire touchscreens are typically 1M finger touches or 100K pen touches; for five-wire touchscreens they're typically 35M finger touches or 1M pen touches. This becomes a serious consideration when you're designing a pen-centric product, since in a real-world POS application, 100K touches takes only a few months. There are two aspects to the shortened life – cosmetic and electrical. In the former, the pen causes the top surface of the PET to become cloudy due to mechanical abrasion. This can be eliminated by using a "screen protector" (a replaceable plastic film) on top of the touchscreen. Electrical end-of-life is usually caused by failure of the ITO coating; the pen contributes to this by causing the ITO-coated PET to flex at a sharper radius than the gentler radius produced by a finger.

**Fixing Poor Palm Rejection:** Can palm rejection be added to a finished product? Not really. The right solution is to replace the touchscreen with one that has closer spacer dots. Unfortunately, real-world supply chain considerations often make this impractical once a product is in production. It's theoretically possible to write a smart touchscreen driver that provides palm rejection while you're writing by filtering points both spatially and temporally. However, what if the application is primarily composed of checkboxes, pull-down menus and other such non-handwriting UI elements? From the driver's point of view, the user seems to be tapping randomly all over the screen, and the filtering algorithm falls apart. If the driver attempts to filter just the specific active areas on each application screen (a bad idea from an abstraction point of view), the user's hand can still be resting on one active area while he's tapping another. Most attempts to fix poor palm rejection after the fact have an adverse impact on the user's interaction with the system (e.g., the system doesn't respond when you tap outside of an active area, or it responds in ways that make sense to the driver but not to the user).

The best approach is to fully understand the requirements of an application before designing a product for it, and to work closely with a touchscreen vendor to customize the touchscreen appropriately when palm rejection is required.

**Based in Silicon Valley, Geoff Walker is a consultant with Walker Mobile, LLC. Geoff has worked on the engineering and marketing of mobile computers since 1982 at GRiD Systems, Fujitsu Personal Systems (now Fujitsu Computer Systems) and Handspring. In addition to mobile computers, Geoff's areas of particular expertise include displays and digitizers.**