Taking Touch to New Frontiers: Why It Makes Sense and How to Make It Happen

Touch interfaces are appearing in everything from consumer devices to industrial equipment, not because touch is "in fashion," but because it provides a truly better form of human-device interaction. This article examines the advantages of gesture-based touch interfaces and the key steps to building a device with a great touch experience.

by Mark Hamblin

TOUCH INTERFACES are far more than "fashionable" features used as a selling point for consumers. They are truly a more intuitive form of human-device interface, compared to many alternatives. While the keyboard and mouse still have their place with the PC, touch interfaces can spread more pervasively into entirely new applications by replacing simple, "low-tech" interfaces such as buttons, dials, and even paper.

Examples include the electromechanical interface found on a washing-machine dial, the button/menu based interface on some medical equipment, and even the non-interactive paper-based interface of a restaurant menu, all of which may someday be replaced by a well-designed, well-implemented touch interface. There are five reasons why this could happen, as follows:

- Touch can simplify interaction with a device.
- Gesture-based touch interfaces are more intuitive.
- Touch provides for more accessible interfaces.
- Touch helps "futureproof" a device.
- Touch enables convergence of other functions into a device.

Mark Hamblin *is the Founder of Touch Revolution, Inc., a touch systems manufacturer based in San Francisco that he founded in 2008. He can be contacted at 415/335-9123 or mark@touchrev.com.*

Simplified Interface

A well-implemented touch interface can be much simpler to use than a conventional mechanical or button-based interface because it can show the user only those controls that are relevant to a particular operation, while all the controls in a conventional interface are always present. This characteristic also makes it easy to expand the interface's functionality because additional functions can remain hidden until they are needed. A touch interface can also be implemented as a sequential guide to help a user easily get

Fig. 1: *One portion of the touch interface on a 2009 wall-oven from Jenn-Air includes a streamlined representation of temperature settings. Source: Jenn-Air.*

0362-0972/03/2010-036$1.00 + .00 © SID 2010

through a series of control steps, similar to a "setup wizard" on a PC. Such features improve the user experience by making the device easier to understand. As device makers continue to build more functionality into their devices, the need for simple, interactive touch interfaces will continue to grow.

One example of how touch can provide a simpler interface can be seen on a wall-oven produced by Jenn-Air in 2009, which features a 7-in. projected-capacitive touch display (Fig. 1). Although the Jenn-Air oven interface appears much simpler than the typical non-touch interfaces found on other ovens, it includes additional functionality such as a step-by-step guide for adjusting cook settings based on the food category, type, and desired degree of doneness. Buttons or controls that are not relevant to the immediate process are eliminated, which streamlines the interface, reduces visual clutter, and prevents user confusion.

Gesture-Based Touch Interfaces

Touch-interface gestures, defined as two-dimensional finger motions, can further simplify an interface and provide an intuitive user experience that goes beyond the typical "button replacement" found in most simple touch interfaces. Gestures allow a sense of control over interface elements that mirror physical elements, allowing for a concept known as "direct manipulation." For example, swiping emulates the finger motion involved in turning the page of a book, while dragging an interface object around a screen mirrors moving physical objects. Gestures and direct manipulation allow users to employ intuitive actions they already use in the physical world rather than having to learn new actions.

More Accessible Interfaces

Interactive touch interfaces provide a significant benefit over conventional static interfaces because they can be configured individually for each user. Text and image sizes can be enlarged for elderly users, languages can be changed as required, options can be simplified for beginning users, and pop-up help menus can appear automatically. The device can even automatically make these reconfigurations upon sensing information about the user. Accessibility will become increasingly important as touch interfaces move into more devices in our lives and face an increasingly diverse user base.

"Futureproofing" Devices

A reconfigurable touch interface without hardware dependencies can provide the ability to modify and improve the interface over time, and even upgrade and change the functionality of the entire device. New features can be rolled out to devices after the initial sale, bugs can be fixed remotely by updating the software over a network connection, interface reconfigurations can be made after actual field usage data is collected, and new applications can be loaded on a device through an online store or other provisioning system. As device manufacturers continue to add more complex features and interfaces to their products, this ability to futureproof the device will become increasingly important. This advantage has already been realized in automotive and GPS applications, for example.

Convergence of Other Functions

A touch interface is really just a blank slate on which the control of any application or function can exist. This allows a touch interface to be the common element through which various functions can converge into one device. In the past, the need for different physical interfaces such as buttons determined the need for products and applications to be separate. For example, in a business environment, a physical business-card file or phone list (paper interfaces) is often located beside a desktop phone (button interface). Neither of these two products provides an interface that is convenient for the other. But if a gesture-based touch interface were implemented on the desktop phone, integrating a graphical-user-interface (GUI) based electronic contact directory into the phone's calling functions would probably improve the utility of the phone and the contact list, as well as being an obvious workflow improvement. In ways such as this, touch interfaces can facilitate product convergence between high-tech and low-tech products.

How to Create an Interface with a Great Touch Experience

Device OEMs seeking the benefits of adding touch interfaces to their products are often faced with the question of how to do it. They typically look at some of the leading touch products on the market, such as the Apple iPhone, as the benchmark for the "touch experience" – which can be defined as the collection of factors that affect the ease-of-use, intuitiveness, and overall user experience of the touch interface. OEMs looking to create an iPhone-like interface on their product often quickly realize that creating an intuitive, easy-to-use touch interface is not as simple as buying a capacitive touch sensor and "slapping it on top" of an existing product. It is a complex endeavor involving hardware, software, integration, optimization, and testing.

The following paragraphs provide 10 best practices for delivering a great touch experience with a wide range of products from medical devices to mobile phones to home

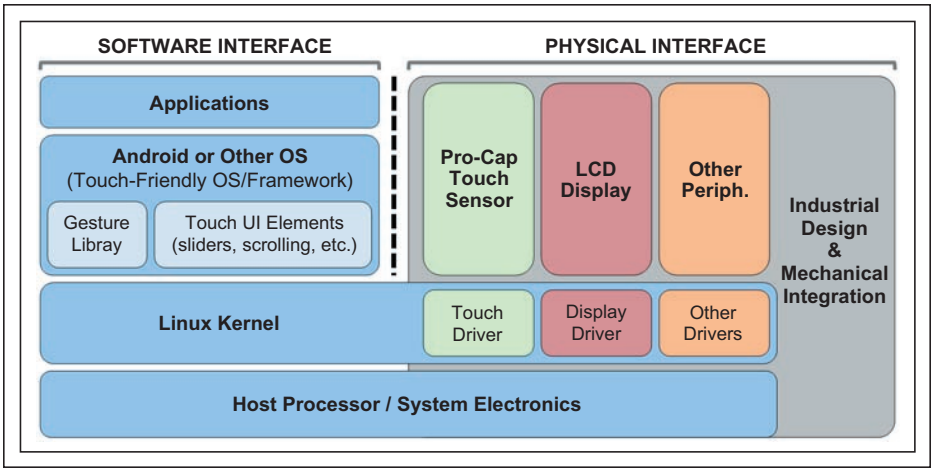

Fig. 2: A block diagram of a touch interface shows the components that must be considered when creating a great touch experience. Source: Touch Revolution.

appliances. The touch-interface block diagram shown in Fig. 2 shows most of the components that are discussed in the following paragraphs.

1. Holistic Design Mindset: Start with the goal of creating a great touch device, not just adding touch to an existing design. The design must be approached holistically. Factors that must be considered from the beginning of a project include user demographics, the product's industrial design, system hardware selection, supported features, and even the product's price-point.

2. Touch-Sensor Technology: Different touch technologies have different advantages and disadvantages, and there are many to choose from – projective capacitive, surface capacitive, analog and digital resistive, surface acoustic wave, optical, *etc.* It is important to note that no one-touch technology solves all problems.

3. Touch-Friendly Operating System: Developing intuitive, attractive, gesture-based touch GUIs and applications can be difficult, especially for OEMs who are new to working with touch interfaces. Giving the users the touch experience they now expect can be made easier through the use of an operating-system (OS) software platform specifically built for touch. These platforms, such as the Google Android, Apple's iPhone OS, and Windows 7 (to some extent), make the software developers' job easier by pre-integrating many common touch user interface (UI) elements such as sliders, selection switches, and gestures such as "flick to scroll." "swipe," and "pinch to zoom."

4. Integration Testing: It is exceptionally important to plan for sufficient integration testing when developing a touch device, especially when using a capacitive touch sensor. Issues such as RF-EMI affecting the touch sensor, software driver optimizations on the LCD and touch controller, cable-routing, application performance affecting touch responsiveness, unwanted optical interaction between the LCD and the touch sensor, ESD concerns, *etc.*, are quite common. The only way to find and fix these issues is to allow significant time for quality-assurance testing and to have engineers with the right background do the troubleshooting. The amount of effort required to integrate all the hardware and software pieces into a cohesive, responsive, and field-ready product is often underestimated, resulting in delayed, over-budget, or even cancelled products.

5. Graphics and Processing Horsepower: A powerful touch interface can consume a lot of processor cycles. It is important to consider where this processing takes place – in the touch-screen controller's CPU, the host's CPU, or the host's graphics processing unit (GPU). An advanced GUI is pointless unless the hardware has enough horsepower in the right places to run it well, without lags, delays, or choppiness. Cutting back on hardware performance to save cost can severely limit the potential of your GUI.

6. Display Selection: Choosing the right display to use in a touch device is especially difficult because of the numerous dependencies between the display and the touch sensor. Important factors to consider include understanding RF-EMI interference issues between the display and the touch sensor, matching the active area and viewing angles, minimizing optical losses, and bonding/sealing the display and touch sensors properly, among many others.

7. Mechanical Integration: Most touch sensors (projective capacitive included) are made of glass, which has many benefits but also adds significant constraints when being integrated into a product. The touch sensor must be integrated correctly to prevent breakage in the event of mechanical stresses, to prevent slight deflections of the sensor that could interfere with the sensing baseline, to prevent dust or other contamination from interfering with viewing quality, and to prevent ESD from damaging the touch sensor or system, among others. Environmentally sealing the touch screen (if required by the device application) can be more difficult than sealing just a display, depending on the touch technology.

8. Industrial Design: By their very nature, touch devices are intended to be highly interactive with the user. This means that ergonomics, usability, and intuitiveness are critical. This is important not only for the GUI design, but for the physical design as well. If the device is portable, how does the user hold it? Is there room for a firm grip without touching the screen? If not, does the touch screen use multi-touch to provide "grip suppression"? Is it designed for users of all ages, sizes, and disabilities?

9. Optimized Touch Software: With an advanced touch interface, there are many software layers involved in translating the motion of your finger on the touch screen into a responsive action on the LCD and in the application software. The firmware running on the touch controller, the touch and display drivers running in the OS, and the application software itself must all be tested and optimized for responsiveness. Any lags in this software stack will result in a sub-optimal user experience.

10. Great GUI: That a touch interface should include a great GUI seems fairly

Fig. 3: *This touch-screen interface for a washer-dryer was shown as a demonstration product at the 2010 Consumer Electronics Show in Las Vegas. Source: Touch Revolution.*

obvious, but many OEMs still do not seem to get it. A touch interface should be much more than just a series of "virtual buttons" to provide an intuitive, accessible, inviting, fun, and satisfying user experience.

Resources

The above "Top 10 Keys to Great Touch Design" is an excellent start, but certainly not complete. Delivering a great touch product to market can be a daunting task, especially for OEMs new to the world of touch. There are many helpful resources OEMs can use to make their touch application a success. A number of touch-controller IC companies offer solutions pre-integrated with a touch screen and OS drivers, minimizing sourcing and technical integration challenges. Some LCD and touch-screen makers are beginning to offer integrated display-and-touch-screen modules. In deciding on a development strategy, OEMs must carefully consider the trade-offs between cost, schedule, reliability, and overall project risk. While bringing touch interfaces to new applications can be difficult, a well-executed touch product can provide a big payback by attracting new customers, enabling new product features, and even opening new sources of revenue.

Future Touch Applications: The Focal Point of Innovation

The future of touch is bright – there will be new technologies, new companies, and new markets. But perhaps even more exciting than the evolution of the touch industry is the impact that touch will have on other industries and applications. Touch interfaces have already helped revolutionize the mobile-phone industry, greatly affecting consumer usage models, network bandwidth requirements, and even enabling major shifts in brand market-share and revenue streams. Touch is likely to do the same for other markets as well.

Take, for example, the home appliance market, and the touch-screen washer-dryer pictured below in Fig. 3, shown as a demonstration product at the 2010 Consumer Electronics Show. Not only does this product have a better, more user-friendly interface compared to the dial-and-button-covered interface of most washing machines today, but the touch interface adds new functionality to the appliance. With the washer connected via WiFi or a 3G network, the user could download laundry-specific applications such as a stain-removal guide, a laundry-symbol decoder, and even an e-commerce portal for purchasing laundry supplies. The user could also install applications for other home-related functions such as lighting control or energy monitoring. All of these possible features create new opportunities and potential revenue streams for the appliance maker, product user, application developers, and numerous other third parties. ■