REACH
TECHNOLOGY INC.

Need help adding a touch screen to your embedded product?

by Jonathan More, President
Reach Technology, Inc.

"My boss thought I was crazy when I told him I could add an LCD touch screen to our embedded product in a couple of weeks. Then I showed him the finished product."

TEL (510) 770-1417 FAX (510) 657-5055 4575 Cushing Parkway Fremont, California 94538 info@reachtech.com www.reachtech.com

"Touch screen technology is new to me. Where do I start?"

That question is common nowadays. Most manufacturing companies are seeing the value – maybe the necessity – of touch screen technology. Many of them don't have a long-term or close association with the technology, yet they expect their embedded engineers to handle the project successfully and on a tight schedule. These engineers often have questions...

- How much am I going to have to learn to get the job done?
- I've heard that LCD suppliers were not like other suppliers. But, how so?
- What don't I know that could shift the project from "exciting" to "doomed."

You have choices

Probably the three major questions that crop up when you need to add an LCD touch screen to your product are these:

1. Should I use a full-blown, embedded operating system, like Windows CE, CE Linux or QNX?
2. How much work does it take to develop an in-house LCD system from scratch?
3. Do I have other options?

The answer to the first two questions is a resounding "maybe," (depending on what you need to accomplish). The answer to the third question is, probably "yes." In most cases, there is another option.

Who should read this paper?

If you are an embedded engineer who is thinking of adding a touch screen to your product, and if:

- You need to know what is involved in adding color touch controls to your product.
- You need to understand the risks (both known and hidden) involved in LCD technology.
- Your main area of expertise is *not* LCD technology.
- You don't want to re-focus your time to acquire color LCD technology expertise.

If you find that any of the statements above voice your concerns, you may find this paper worth reading.

"The iPhone shows how color touch control can revolutionize a product category. How can you quickly and easily add a touch screen to your product?"

TEL (510) 770-1417 FAX (510) 657-5055 4575 Cushing Parkway Fremont, California 94538 info@reachtech.com www.reachtech.com

When tacking a new project, engineers often consider how schedule, features and overall cost are impacted by the direction they take. *Use this comparison chart to help you organize and analyze your priorities, then continue reading to learn more about the solution that best fits your requirements.*

	Embedded Operating System	In-house Design	Reach Technology Products
Schedule Issues			
Low complexity (keep it simple)			✓
Reuse existing development tools	?	?	✓
No specialized training required			✓
Easy to test			✓
Easy to debug			✓
Features			
Instantaneous boot and shut-down			✓
Consistent interface responsiveness			✓
Supports Flash animation	✓		
Native networking	✓		
Access to centralized databases	✓		
Connects to PCs	✓		✓
Cost			
Lowest production cost per unit		✓	
Low cost for small production volumes			✓
Low development costs			✓
Risk			
Protection from changes in LCD supply (see sidebar on page seven: "The LCD business is very different from others.")			✓
Fast production ramp			✓
One source technical support			✓

Option One: The full-blown, embedded operating system

There is a common belief that still lingers: that a full-blown operating system is required to implement a touch screen. Certainly Microsoft would have you believe this is the case. Today, there are other options.

If you're simply upgrading an existing product with touch screen technology, it's not likely that a full-blown, embedded operating system would be your logical choice.

A complex operating system is appropriate if:

1. You're designing a new product and it needs a high-level interconnection with other, similar OS-based systems.
2. The system will have off-the-shelf peripherals, such as a fingerprint reader, attached to it.
3. You're well-versed in the operating system you're going to use, or are willing to commit the time to become comfortable with it.

Considerations

- **Processing power:** A full-blown operating system needs a full-blown single-board computer (SBC) to run on. If your current product is based on a microcontroller, you have to make a major hardware platform change.
- **Operating system experience:** You and your team must be (or become) familiar enough with the operating system to make accurate risk assessments. A typical statement of concern: "Is there some implementation detail that I don't know about and don't know I have to consider? They say it'll do what I need it to do, but *will it*? If I have a problem, who do I call?"
- **Start-up and shut-down time:** Boot time and shut-down time are issues to consider. High-end operating systems are not inherently designed to do either of these as quickly as your application might require.
- **Memory leaks:** The specter of memory leaks is almost inevitable with high-end operating systems. Consider this issue if memory leaks could pose a problem for your application. For the uninitiated, memory leaks and the consequent low-level processor abort is the major cause of PC "crashes."
- **Reliability:** The more code, the more bugs. It is just that simple. Operating systems inherently have a lot of code.

+ Advantages of embedded operating systems

Industry-standard, embedded operating systems have their advantages. Among them:

1. They're industry standards.
2. High level networking connectivity features are built in.
3. You can connect to centralized databases using standard software interfaces.
4. You can connect to PCs on a peer-to-peer basis.
5. They permit the use of Flash animation.
6. They're "future-proof." Maybe your product doesn't need networking, database connectivity, Flash, or any of that now, but, if you think those features will be required in the future, a full-blown operating system could be the way to go.

- Disadvantages of embedded operating systems

Some projects require the power and versatility of high-end operating systems, but such systems also have substantial disadvantages:

- **Increased complexity:** Full-blown embedded operating systems are extremely complex. Unless you and your team already possess deep expertise in their use, you'll spend months gaining the necessary knowledge and experience.
- **Boot time:** In most embedded applications, boot time is a big issue. Without special programming, full-blown operating systems require 15 – 20 seconds to boot. You can improve that, but to do so requires an expert level of knowledge.
- **Shut-down issues:** What happens when power fails and the system is not shut down "gracefully"? Is shut-down time important to your application? If so, be aware that high-end operating systems require substantial time to shut down correctly. As with boot time, shut- down times can be reduced, but doing so requires substantial know-how.
- **Test complexity:** For an embedded test, all the permutations of the run-time environment must be identified and incorporated into the test suite. This is a huge job and it's rarely fully- accomplished, as is obvious whenever an embedded "system" crashes.

"Is there some implementation detail that I don't know about and don't know I have to consider? They say it'll do what I need it to do, but will it? If I have a problem, who do I call?"

- **Interface responsiveness:** We have become used to our PC's having the occasional "hiccup," as one of its many background processes eats up processor cycles. However, if your microwave oven took varying times to respond to the "start" button, you'd probably think it was broken, or on its last legs. Embedded user interfaces need consistent response times and this is more difficult to achieve with a complex multitasking operating system.

Sample Windows CE Code Used to Create a Frame and Add a Button

```
int CMainFrame::
OnCreate(LPCREATESTRUCT lpCreateStruct)
{
  if (CFrameWnd::OnCreate(lpCreateStruct) == -1)
    return -1;
  // Add the buttons and adornments to the
CommandBar.
  if (!InsertButtons(tbButtons, nNumButtons,
IDR_MAINFRAME, nNumImages) ||
    !AddAdornments(dwAdornmentFlags))
  {
    TRACE0("Failed to add toolbar buttons\n");
    return -1;
  }
  return 0;
}
```

Option Two: A "from scratch" in-house design

Engineers generally enjoy developing a system from scratch. After all, it gives them a chance to look at the most modern processors and development tools, and "new and exciting" beats "old and boring." That is, until they see the schedule. Then old and boring — read known and predictable — starts looking mighty good. The choice to build from scratch or use what's available comes down to the time, money, and people that are available for the project.

There are many cases when designing from scratch is the only way to go:

1. The production run will be large enough to amortize development expenses.
2. The form factor or feature set mandates a fully custom solution.
3. The feature set cannot be implemented with off-the-shelf solutions.

Considerations

- **Analysis paralysis:** There are so many options — processor, software toolset, operating system versus bare metal, graphic library, and so forth — that it's easy to spend more time evaluating options than creating the design.
- **The software toolset and development environment:** If the tools are new to you and your team, how much time and cost will be involved in getting up to speed?
- **Graphics library:** There are many libraries to choose from, and the advantages and disadvantages of each might become apparent only *after* you're deep into implementation.
- **The changing face of LCD technology and markets:** Because of the fast-changing nature of LCD technology and markets, you must consider the real probability that the parts you initially began using will not be available at some point in the future. How will you deal with these changes? (See sidebar on page seven: "The LCD business is *very* different from others".)
- **The nature of LCD technology:** Unless you are already familiar with LCD technology, you will need to learn about it. If you want to make changes, or if changes are forced on you, you must be prepared to adjust to them, understanding that a change in one component will likely mean changes in all of these components:
  - LCD controller board
  - LCD screen
  - LCD touch control
  - Connectors and cabling

"I could buy an LCD panel from one vendor, and buy other components from another vendor, but then I would have to spend the rest of my life getting the two to work together."

+ Advantages of in-house development

- Lowest production cost.
- Greatest freedom to add or change features.
- Greater control of your supply chain.
- Total control of (and responsibility for) LCD system maintenance, which includes changes in available products, supply, and so forth.

- Disadvantages of in-house development

Unless you have very compelling reasons for developing a system in-house, there are distinct disadvantages, too. Among them:

- **Changes in LCD product availability** (See sidebar on page seven: "The LCD business is very different from others.")
- **Cost of design:** It takes time to research the components of any system. Deciding on the right processor (see options below), the most appropriate operating system and determining reliable sources of LCD components (see LCD vendor options at right) can be time-consuming, expensive tasks. Do you have the resources?
- **Production:** Each new part needs an Approved Vendor List (AVL), which requires a huge amount of work to do correctly. At the same time, how many are single-sourced with perhaps a 3,000 piece reel minimum and a possible ten-week lead time?
- **The learning curve:** If you and your team are not already familiar with the components of the system you have in mind, the learning curve can (and will) be steep. It will involve taking time away from your core responsibilities. It will also involve the cost of training, experimentation and other facets of a learning environment. Do the costs make sense in your situation?

LCD Vendor Options:

AUO	Okaya
CMO	Optrex
Data Image	OSD
Densitron	Powertip
EDT	Prime View
Giantplus	Samsung
Hantronix	Sanyo
Hitachi	Sharp
Hosiden	Tianma
Kyocera	Toshiba
LG Philips	TPO Displays
Microtips	Truly
Nanya	Wintek
NEC	

The bottom line: Are the higher costs and greater risks worth it?

There's little doubt that designing your own LCD system from scratch is the riskiest and most costly option you have. The question is: Given all that, does it still make sense to do it? Examine your time constraints, time to market, the cost, and the features you must have. This option has the lowest potential per-unit production cost, but you need to amortize the development cost over the production life to get a true per-unit cost. If the flexibility, control, and other features of an in-house system outweigh the negatives, the answer is yes.

Processor Options (with LCD controller):

AT91SAM926x	Intel Atom	ML67505x	S3C2xxx
EP93xx	LH7952x	NS9360	Tegra
Geode	LPC32xx	OMAP35xx	TMPA910x
H8X	MAC71xx	LPC2478	TMS320DM644x
i.MXxx	MCF532x	PXAxxx	

The LCD business is very different from others

The LCD business is not like the traditional passive and IC components business that most embedded designers are familiar with. First, the lead times can be very long — up to 14 weeks — depending on many unpredictable factors. Second, due to the cost of running an LCD production line, most low-cost LCD displays start out as custom designs for a single high volume customer. Think of the Palm Pilot, a portable DVD player, the Garmin GPS, and so forth. This customer dictates every detail — the screen size, interface connector type, pinout, and location, mounting holes, and so forth.

The LCD company might then decide to try to market this panel to other customers to get more business. At this stage, a datasheet is created and sent out.

At some point, the original LCD customer makes a design change and stops ordering the original display.The LCD manufacturer, having lost its bread-and-butter customer, will generally stop making that specific LCD, even if there are existing customers, because the run rate is too low. Unfortunately, the panel end-of-life notification is not always forthcoming or timely.

If you've already designed this LCD into your product, you have a problem: You can't get them anymore. You likely face the prospect of having to make changes in your system. You have to change the LCD display, the touch control, the cabling, the mechanical mounting, and perhaps other aspects of your system. This is a very expensive proposition, but it happens all the time, because of the nature of the LCD industry. Even when a panel doesn't go obsolete, if its original buyer has a surge in demand, your lead time can go from a few weeks to several months. How can you protect your business from this unpredictable supply issue?

One possibility: Use an LCD specifically designed for the embedded market. These are long-lifetime designs from NEC, Kyocera and others, and are priced accordingly. The other possibility is to use a display module supplier such as Reach that supplies the total solution and understands the LCD module business.

Protection against product changes

When you buy your LCD products from Reach Technology, supply problems are greatly reduced. Here's why: When Reach develops an LCD controller/display module, it uses LCD form factors that are "de-facto" standards. Reach has direct, long term relationships with LCD suppliers, so it is the first to know about end-of-life or specification change issues. If an LCD becomes obsolete, a substitute is offered that has the same mechanical mounting to avoid packaging redesign. There are no changes or adjustments you need to make. Reach also aggregates LCD demand for low-volume customers so they won't be stuck with sales-killing lead time increases.

When you get started, you happen to choose an LCD panel that might not have interoperability options, we'll let you know in advance, so you don't inadvertently choose components that will adversely affect product availability and flexibility.

We've been guiding customers through changes such as these for 20 years. With your cooperation, we will help you ensure a steady and uninterrupted supply for your customers.

LCD maintenance issues

LCD maintenance issues arise because of changes in the supply. When you use Reach products, any adjustments that are called for will be made before you receive the equipment. Therefore, you have no maintenance issues.

Support for LCD systems/knowledge-sharing

Reach Technology has combined experience in LCD technology and associated markets of some 20 years. Our embedded engineers are always willing to discuss your concerns and to share their LCD experience with you.

The Other Option: Use LCD color touch from Reach Technology

Unlike the full-blown embedded operating system, and the in-house LCD system designed from scratch, you should consider the Reach option if what you want to do is upgrade an existing product. Although Reach can be a valid choice in other situations, this is where our products truly shine.

Today's options for embedded systems designers have expanded dramatically. It used to be that full blown operating systems were the only choice when color touch, networking, or other advanced features were needed in an embedded system. Now there are easy-to-use board-level products for embedded Web servers, wireless subsystems, USB hosts, and LCD control surfaces.

Using the technology that Reach offers, you can add LCD color touch control to your new or existing products with very little time spent on researching, identifying and coping with the risks that might be involved.

Considerations

- **Time to market:** What's your schedule? If it's tight, Reach products should be a consideration.
- **Operating systems:** If you would prefer not to deal with a full-blown embedded operating system and, especially, if you don't absolutely require one for other reasons, Reach LCD color touch products could be your answer.
- **LCD supply:** Do you have the resources and the LCD industry knowledge it will take to control your supply sources and to respond to changes in availability? (See sidebar: "The LCD business is very different from others".) Reach Technology forms a buffer between you and the vagaries of the industry.
- **Available experience and expertise:** Do you or your team have deep knowledge and experience with LCD technology? You need little of either when you use Reach Technology LCD products.

+ Advantages of Reach technology

Under the right circumstances, the Reach LCD color touch technology makes a lot of sense. The reasons are:

- **It works with any processor,** from small 8-bit microcontrollers to 64-bit multi-core processors.
- **It works with any operating system:** Reach LCD products work with, or without, a host OS. From full-blown, embedded systems like Windows CE, CE Linux or QNX, to small RTOS, or a "bare metal" main loop, the interface is the same.
- **No graphics library required:** The library is built into the Reach product. Your embedded system does not require additional memory overhead for library code and has more resources to perform tasks such as controlling your product.
- **Low risk:** Reach products are used by hundreds of companies worldwide, and there are an estimated 20,000 units in operation. All products are fully-tested and they're designed by highly-experienced LCD engineers.

- **Short time to market:** There is no extensive learning curve involved and little programming required. Your products can be running with color touch technology within weeks.
- **Reliability:** The Reach command protocol is simple and robust. There is no code added to your system that could cause problems. You don't have to be concerned with a library in which a wrong order of calls could cause memory leaks or worse, or where a bad parameter could cause a memory abort (both of which mean more debug time). With the Reach system, a command error is simple to recognize and no badly-formed command can crash the system.
- **Easier to debug:** All commands are in easy-to-read ASCII, and the GUI interaction can be easily logged to spot problems.
- **Protection from changes in LCD supply:** Reach Technology works with multiple vendors to insure supply in the face of component and LCD supply changes. (See sidebar on page seven: "The LCD business is *very* different from others".)
- **Instantaneous boot and shutdown.** When you turn on a product equipped with Reach LCD color touch, it comes to life immediately, not in 10 or 20 seconds. Similarly, you can cut power to the product without losing data, and the device is ready for an errorless re-boot.
- **No special knowledge of LCD markets required.** Reach Technology is the buffer between you and the markets.
- **Firmware version control:** Unlike some vendors (Microsoft?) that stop selling popular software versions, Reach allows customers to specify the exact firmware version when ordering. The version of firmware that you approved for production is the same that you get for production, and there are no unexpected changes.

There are other suppliers that seem similar, but…

There are other companies which, at first glance, appear to offer products similar to Reach products. But, as you do your research, ask these questions:

1. What's the reliability of supply? Are they really only selling excess LCD stock that will change due to the vagaries of the LCD markets? (See sidebar: "The LCD business is very different from others".)
2. How long will the product be available, and what happens when changes in supply occur?
3. Do they provide support for their products? What's the nature of it?
4. Do they specialize in LCD color touch technology, or are they simply taking advantage of currently – and temporarily – available product?
5. Do they cater to hobbyists, rather than to businesses? While some hobbyists are very sophisticated, the products they use are often not designed for commercial use.
6. Are they really just IC manufacturers looking for a market for their chips? If so, their products are designed to sell their chips, not to deal with the best solution for you.

- Disadvantages of Reach technology

- **Cost considerations:** Might not be cost-effective with very large runs, particularly for companies with staff experienced in the use LCD technology.
- **Might not be the best choice for use with full-blown, industry-standard operating systems**, where the hardware for LCD support is essentially "free."
- **If you are not comfortable buying board level components** to put into your product, then Reach is likely not for you.

Reach Technology LCD products: Add color touch in weeks

Reach products require very little programming and they interface, via serial port, with whatever microcontroller software you now have. Your products could be updated with LCD color touch within weeks.

Reach Technology LCD controller products carry enormous benefits for the right users.
They include:

- Lower development costs.
- Faster times to market.
- Protection from changes in LCD availability.
- Reduced training requirements.
- Total relief from LCD maintenance & support issues.

Enclosed units

Ready to go for NEMA 4 applications. These fully-enclosed units replace panel switches and LCD character displays, providing a modern, attractive user interface for any product. Each unit contains a Serial LCD (SLCD) controller board, LCD, and touch screen mounted in a waterproof and dustproof unit ready for immediate installation in any application. Standard interfaces include RS232 and RS422/485. Available in 4.3", 5.7" and 8.4" panel sizes.

Module units

Simple graphic-display units come completely assembled. Modules make integration easy. Each unit comes complete with a SLCD controller board, LCD, touch screen, cables and mounting interface. They come in various panel sizes and resolutions.

Serial LCD controller boards

Rapidly integrate an intelligent control system into your product. Low-cost SLCD controller boards need no special OS or library on the host processor. They control multiple panel sizes in QVGA, WQVGA, VGA and WVGA. These boards are at the heart of Reach Technology's enclosed units, modules and development kits.

Low-risk development kits

We offer low-cost, low risk development kits which contain everything you need to get a color touch interface up and running in a matter of days. Kits include: SLCD controller board, LCD, touch screen, cables, sample images, sample code, power supply, and technical support.

How it works

Your microcontroller sends serial commands to a Reach Technology module, which is made up of a serial LCD controller board, an LCD, and a touch screen. When a button or other interactive control is touched, a serial notification is sent back to your controller. There is no special operating system or library required. Programming for Reach LCD color touch technology is simple.

Reach Technology: Your partner in LCD color touch

After reading this paper, you should have a better idea of the ins and outs of adding LCD color touch technology to your products. You've read about the risks involved in the LCD market, the kinds of risks you face in your particular project, and some of the major questions to answer before you get started. Possibly, you've learned that introducing color touch technology doesn't have to be as complicated as it might have sounded. It's possible to achieve lower development costs, reduced risk, and faster times to market, while being insulated from changes in LCD availability, technology and format.

Hopefully too, you've learned about Reach Technology; about the products we offer, and about the benefits of the knowledge and experience we've gained in more than 20 years in the LCD business.

Jonathan More

Jonathan More, President
Reach Technology, Inc.

Is Reach a fit for you and your needs?

Here are some options for next steps:

1. **Download a Technical Requirements Survey** at reachtech.com/support/downloads/ to help outline your project's LCD control technical requirements and project constraints.
2. **Talk with a customer service representative** by calling 510-770-1417 or emailing support@reachtech.com.
3. **Schedule a time with a Reach engineer** after filling out the survey above by sending an email to techsupport@reachtech.com or by calling 503-675-6464.
4. **Download data sheets** for specific products at reachtech.com/support/downloads/.
5. **Order a Reach development kit** which contains everything you need to get a touch interface up and running in a matter of days: SLCD controller board, LCD display, touch screen, cables, sample images, sample code, power supply and include technical support. Nearly 1,300 units sold so far. Watch a 15-minute tutorial video that shows you how to set up hardware, software, and build your first screens and find order forms at reachtech.com/support/downloads/.