News
Little LEDs could soon be finding their way on to your clothing

Industrial
Seeking out security risks in industrial settings

Test & Measurement
The truth behind real-time jitter measurement

Displays & GUIs
How optimised ASSPs could provide faster time to market
Frameworks Fuelling the Future

Efforts to ease software development now extend to user interfaces, as Philip Ling discovers.

Despite their similarities, no two Cortex-M based MCUs are really truly alike; the fact that manufacturers including ST, Freescale, Texas Instruments and Renesas have a licence to take the core and adorn it with their own choice of peripherals and enhancements guarantees their ‘uniqueness’, albeit with the same basic instruction set. The same can be said of GUIs; they may all have common features but the fact remains that one size will not fit all. With growing expectation of simpler interfaces in everyday objects, the commercial opportunity for a configurable GUI targeting Cortex-M MCUs is potentially as big as the market for the devices themselves. Not surprisingly, their development is now drawing the attention of embedded software specialists such as Express Logic.

The company’s latest product offering in this space, GUIX, has been designed for embedded systems in general and the ARM architecture in particular (both -M and -A cores), as such it can be implemented in a small code footprint, plays nicely with the company’s real-time OS, ThreadX, and comes complete with a PC-based development suite, GUIX Studio. A handful of major silicon manufacturers have already worked with Express Logic to port it to their own ARM-based devices and according to Ken Maxwell, Director of GUI Engineer at Express Logic, the roadmap intends to extend its support: “We will keep up with the latest silicon vendor offerings,” said Maxwell, “which may require new designs within a UI framework to effectively take advantage of hardware functionality.”

Maxwell has a long history of GUI development, being the creator of tools including PEG+, PEGPro and WindowBuilder. But things have changed: “When I wrote those tools, the focus was on bringing the desktop UI experience to embedded devices; we tried to emulate the look and feel of a Windows desktop in a way that could be replicated on much smaller and less powerful targets.” Today, customers require more customised solutions and, as a result, the design of the toolkit is quite different.
Fundamentally, Maxwell intends for GUIX to deliver modern UI designs using very limited resources: “GUIX will fit within the on-board Flash storage of many devices in which many similar toolkits just can’t squeeze,” he said. “We are primarily targeting a range of devices that the big name solutions, such as Android and Qt, just aren’t going to fit, including those based on ARM’s Cortex-M architecture.”

**Out of the Box**  
Like many graphics frameworks, GUIX will provide out-of-the-box support for widgets such as buttons, keyboards and scroll bars, as well as supporting touch-screen and other input methods. All will be customisable and the design suite will enable engineers to develop their own. “We have also invested a lot of resources in creating real-world demonstration applications for several vertical markets,” said Maxwell. While he doesn’t necessarily expect them to be used as-is, full source code is provided to allow customisation. The real value will likely be in its support for on-chip features from different vendors, be they graphic-based such as sprite engines, or more generic touch controllers.

The GUIX run-time engine provides a framework for the UI, such as routing events to graphic elements, handling events and menu systems — these can be complex and Maxwell claims the framework will allow developers to build very complex applications that are also robust and deterministic.

Importantly, GUIX Studio offloads a lot of the processing to keep the runtime component as small as possible. “Other toolkits might impose some processing on the target, such as TrueType font rendering, JPG image decoding and colour space mapping. We implement these functions within GUIX Studio, so that our runtime component is smaller and faster.”

As the Internet of Things grows, the ARM architecture is expected to empower a significant number of nodes, which Maxwell believes will lead to increased GUIs: “In these devices, networking and GUIs will experience the most spectacular growth as the expansion of the Internet moves forward. Given the importance of ARM-based systems in this industry, ARM will continue to dominate such devices.” This is expected to fuel demand for embedded solutions: “Our ThreadX RTOS, NetX networking software and GUIX are right in the middle of this exciting growth.”