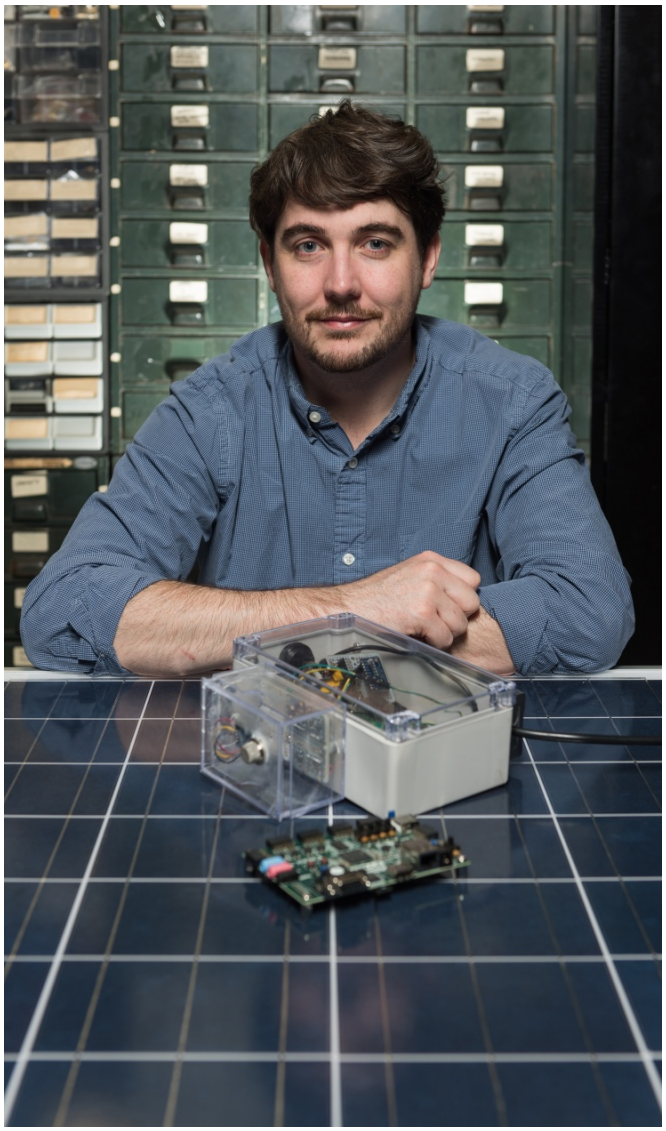
Research advances security and trust in reconfigurable devices

29 April 2015, by Rick Robinson



Researcher Lee W. Lerner displays an FPGA board along with a custom sensor box built by the GTRI team for research purposes. Credit: Rob Felt

A research team at the Georgia Tech Research Institute (GTRI) is studying a range of security challenges involving programmable logic devices – in particular, field programmable gate arrays (FPGAs).

FPGAs are integrated circuits whose hardware can be reconfigured – even partially during run-time – enabling users to create their own customized, evolving microelectronic designs. They combine hardware performance and software flexibility so well that they're increasingly used in aerospace, defense, consumer devices, high-performance computing, vehicles, medical devices, and other applications.

But these feature-rich devices come with potential vulnerabilities – the very configurability of an FPGA can be used to compromise its security. The slightest tweak, accidental or malicious, to the internal configuration of a programmable device can drastically affect its functionality. Conversely, when security and trust assurances can be established for these devices, they can provide increased, higher-performance resilience against cyber attacks than difficult-to-assure software-based protections.

The GTRI researchers have identified multiple issues that could become serious threats as these devices become increasingly common.

"Because FPGAs are programmable and they tightly couple software and hardware interfaces, there's concern they may introduce a whole new class of vulnerabilities compared to other microelectronic devices," said Lee W. Lerner, a researcher who leads the GTRI team studying FPGA security. "There are entirely new attack vectors to consider, ones that lie outside the traditional computer security mindset."

Conventional protections such as software or network-based security measures could be undermined by altering the logic of a system utilizing programmable devices.

"The potential to access and modify the underlying

hardware of a system is like hacker Nirvana," Lerner said.

Traditional hardware security evaluation practices – such as X-raying chips to look for threats built-in during manufacturing – are of little use since an FPGA could be infected with Trojan logic or malware after system deployment. Most programmable devices are still at risk, including those embedded in autonomous vehicles, critical infrastructure, wearable computing devices, and in the Internet of Things, a term that refers to online control devices ranging from smart thermostats to industrial systems.

Myriad possibilities

FPGA chips are constructed from heterogeneous logic blocks such as digital signal processors, block memory, processor cores, and arrays of programmable electronic logic gates. They also include a vast interconnected array that implements signal routing between logic blocks. Their functionality is dictated by the latest configuration bitstream downloaded to the device, commonly referred to as a design.

An FPGA's adaptability gives it clear advantages over the familiar application-specific integrated circuit (ASIC), which comes from the foundry with its functionality permanently etched in silicon. Unlike an ASIC, for instance, an FPGA containing some sort of error can often be quickly fixed in the field. One example application which utilizes this flexibility well is software-defined radio, where an FPGA can function as one type of signal-processing circuit and then quickly morph into another to support a different type of waveform.

The earliest FPGAs appeared 30 years ago, and today their logic circuits can replicate a wide range of reconfigurable devices including entire central processing units and other microprocessors. New internal configurations are using high-level programming languages and synthesis tools, or low-level hardware description languages and implementation tools, which can reassemble an FPGA's internal structures.

Depending on how they are set up, FPGAs can be configured from external sources or even internally by sub-processes. Lerner refers to their internal configuration capability as a type of "self-surgery" – an analogy for how risky it can be.

Additionally, because FPGA architectures are so dense and heterogeneous, it's very difficult to fully utilize all their resources with any single design, he explained.

"For instance, there are many possibilities for how to make connections between logic elements," he said. "Unselected or unused resources can be used for nefarious things like implementing a Trojan function or creating an internal antenna."

Anticipating attacks

To exploit an FPGA's vast resources, bad actors might find ways to break into the device or steal design information. Lerner and his team are investigating ways in which hackers might gain the critical knowledge necessary to compromise a chip.

One potential avenue of attack involves "side-channels" – physical properties of circuit operation that can be monitored externally. A knowledgeable enemy could probe side-channels, such as electromagnetic fields or sounds emitted by a working device, and potentially gain enough information about its internal operations to crack even mathematically sound encryption methods used to protect the design.

In another scenario, third-party intellectual property modules or even design tools from FPGA manufacturers could harbor malicious functionality; such modules and tools typically operate using proprietary formats that are difficult to verify. Alternatively, a rogue employee or intruder could simply walk up to a board and reprogram an FPGA by accessing working external test points. In some systems, wireless attacks are a possibility as well.

FPGAs even contend with physical phenomena to maintain steady operation. Most reprogrammable chips are susceptible to radiation-induced upsets. Incoming gamma rays or high-energy particles could flip configuration values, altering the design function.

Lerner points to a real-world example: Google Glass, the well-known head-mounted optical technology, which uses an FPGA to control its display.

Multiple security techniques

To provide assurance in programmable logic designs, Lerner and his team are developing multiple techniques, such as:

- Innovative visualization methods that enable displaying/identifying/navigating patterns in massive logic designs that could include hundreds of thousands of nodes and connections;
- Applications of high-level formal analysis tools, which aid the validation and verification process;
- System-level computer simulations focused on emulating how heterogeneous microelectronics like FPGAs function alongside other system components.

The GTRI team is also engaged in other areas of research that support design security analysis, including exact- and fuzzy-pattern matching, graph analytics, machine learning / emergent behavior, logic reduction, waveform simulation, and large graph visualization.

The team also researches architectures to support trustworthy embedded computing in a variety of applications, such as cyber-physical control. They have developed the Trustworthy Autonomic Interface Guardian Architecture (TAIGA), a digital measure that is mapped onto a configurable chip such as an FPGA and is wrapped around the interfaces of process controllers. Its goal is to establish a "root-of-trust" in the system, a term that refers to a set of functions that can always be trusted, in this case to preserve system safety and security.

TAIGA monitors how an embedded controller process is functioning within the system, to assure that it's controlling the process within specification. Because TAIGA can detect if something is trying to tamper with the physical process under control, it removes the need to fully trust other more vulnerable parts of the system such as supervisory software processes or even the control code itself.

"TAIGA ensures process stability – even if that requires overriding commands from the processor or supervisory nodes," Lerner said. "It's analogous to the autonomic nervous system of the body, which keeps your heart beating and your lungs respiring – the basic things that your body should be doing to be in a stable state, regardless of anything else that's going on."

The team has installed a version of the TAIGA system on a small robot running the Linux operating system. Georgia Tech students and other interested persons are invited to manipulate the installation and the robot online to try to compromise its control system at the team's main website, configlab.gatech.edu , when the experiment is ready.

"We provide formal assurances that TAIGA will prevent anyone from hacking critical control processes and causing the robot to perform actions deemed unsafe," Lerner said. "However, if someone figures out how to run the robot into a wall or damage its cargo, for instance, then obviously we'll know we have more work to do."

Provided by Georgia Institute of Technology