As society becomes more and more reliant on information technology, the need for secure computing systems is greater than ever before. Most research in information security has focused on software: operating systems, server software, and desktop and mobile apps, and so on. But there is an equally important piece of the puzzle which is often neglected: the hardware. "Secure" software is not secure if the hardware it runs on is insecure.

This dissertation establishes both rigorous mathematical techniques and a set of programming tools that enable hardware designers to construct verifiably secure hardware systems. Verifiable security means that the security properties we desire are mathematically provable. Rather than designing and fabricating a hardware circuit, waiting for the bad guys to find its security flaws, and fixing these flaws one by one (often at great expense) as they are discovered, verifiable security provides certainty that security flaws do not exist in the first place. Specifically, I have developed a programming language called ReWire that enables engineers to specify hardware systems in a way that takes security into account from the very beginning of the design process. I have also developed a compiler for this language, which translates these security-aware designs into actual, running hardware circuits.