

We're in the process of moving Radar to the new oreilly.com. Check it out.



## What you need to know for the hardware-software convergence

Core competencies and essential reading from hardware, software, manufacturing, and the IoT.

by [Jon Bruner](#) | [@JonBruner](#) | [+Jon Bruner](#) | December 18, 2014

As I noted in "[Physical and virtual are blurring together](#)," we now have hardware that acts like software, and software that's capable of dealing with the complex subtleties of the physical world. So, what must the innovator, the creator, the executive, the researcher, and the artist do to embrace this convergence of hardware and software?

At its core, this is about a shift from discipline toward intent. Individuals and institutions — whether they're huge enterprises, small start-ups, or nonprofits — must be competent in several disciplines that increasingly overlap, and should be prepared to solve problems by working fluidly across disciplines.

To use [Joi Ito's](#) example, someone who wants to develop a synthetic eye might begin to approach the problem with biology, or electronics, or software, or (most likely) all three together. Many problems can be solved somewhere in a large multidimensional envelope that trades off design, mechanics, electronics, software, biology, and business models. Experts might still do the best work in each discipline, but everyone needs to know enough about all of them to know where to position a project between them.

Below you'll find the core competencies in the intersection between software and the physical world, and our favorite books and resources for each one.

### Electronics for physical-digital applications

- [Practical Electronics](#), by John M. Hughes: To know what's possible and where to start, it's essential to understand both the analog and digital sides of electronics. This is O'Reilly's authoritative introduction to both analog and digital electronics, with information on circuit design, common parts and techniques, and microcontrollers.
- [Raspberry Pi Cookbook](#), by Simon Monk: The Raspberry Pi is rapidly becoming the standard embedded computing platform for prototyping and experimentation, with enough computing power to run familiar interpreted programming languages and widely supported operating systems.
- [Arduino Cookbook](#), by Michael Margolis: The Arduino microcontroller offers a fluid interface between digital and physical; it's highly extensible and accessible to people with no prior experience in either electronics or code.

### Design for manufacturing and experience design

- [Manufacturing Processes for Design Professionals](#), by Rob Thompson: Understanding both the constraints and possibilities of manufacturing is key to designing great products. This accessible, beautifully illustrated encyclopedic reference describes hundreds of manufacturing processes and their applications.
- [Microinteractions](#), by Dan Saffer: Design in the newly united world of bits and atoms is about creating excellent experiences across both the physical and virtual components of products. Everyone should become comfortable with design thinking, and the concept of microinteractions makes for a good introduction to [experience design](#).
- [Designing Connected Products](#), by Claire Rowland, Elizabeth Goodman, Martin Charlier, Alfred Lui, and Ann Light: A single product might involve interactions on many devices: through the physical product itself, through other devices that connect

with it, through the product's web portal, and through other platforms that are integrated with its API. This book offers an introduction to user experience design in creating these sorts of distributed products.

## Manufacturing and the hardware business model

- [The Hardware Startup](#), by Renee DiResta, Brady Forrest, and Ryan Vinyard: Hardware companies differ from software companies in fundamental ways — crucially, in the need to manufacture and distribute physical goods. This is a practical introduction to hardware business models and the process of developing products and supply chains. (This title is an early release. The manufacturing chapters are forthcoming.)
- [Prototype to Product](#), by Alan Cohen: Scaling is particularly difficult in hardware compared to software. This introduction walks through the full process of product development from prototyping through small-run, medium-run, and large-run manufacturing.

## Software thinking and frameworks

If you're already a comfortable programmer, these aren't essential resources. If you're not a comfortable programmer, though, you'll find that an introduction to programming will be immensely helpful in understanding which kinds of problems can be solved with software as opposed to another discipline. Python is emerging as the standard programming language for general computing and is relatively easy to learn; JavaScript is the standard programming language for the web, and it's finding its way into servers and embedded systems through frameworks like Node.js as well.

- Introductory: [Learning Python](#), by Mark Lutz
- Intermediate: [Programming Python](#), by Mark Lutz
- Introductory: [Learning JavaScript](#), by Shelley Powers
- Intermediate: [JavaScript: The Definitive Guide](#), by David Flanagan

## Connected devices

Each of these books takes a look at one popular technology for making devices interact with each other and with the web.

- [Getting Started with Bluetooth Low Energy](#), by Kevin Townsend, Carles Cufí, Akiba, and Robert Davidson
- [Building Proximity Applications with iBeacon](#), by Matthew Gast
- [Beginning NFC](#), by Tom Igoe, Don Coleman, and Brian Jepson

The fields of hardware and software are evolving quickly, and the areas where they intersect are evolving even faster. We've got several titles coming out next year that promise to bridge the software-hardware gap and help both developers and entrepreneurs move between software and hardware. We'll keep this list updated, and if you've got any favorite references in any of these areas, please let us know in the comments section below.

tags: [APIs for the Physical World](#), [collision of hardware and software](#), [Every Company is a Software Company](#), [experience design](#), [frictionless manufacturing](#), [Industrial Internet](#), [Interaction Design and Connected Devices](#), [Internet of Things](#), [iot](#), [manufacturing](#), [resources](#), [Software Intelligence in the Physical World](#), [Software Replaces Physical Complexity](#)