Do-It-Yourself Fabrication of Electronic Devices

This article analyzes how the digital nature of design files and the flexibility of fabricating them into physical objects affect the design and production of enclosures and circuit boards, as well as component sourcing, circuit assembly, and programming.

David A. Mellis
MIT Media Lab

The increasing accessibility of digital fabrication and embedded computation suggests that anyone should be able to produce their own electronic devices. It's possible to design a circuit board on the computer, get a few copies of the board made, and assemble the components it uses. You can design an enclosure to fit the circuit board, 3D-print one, and end up with both a one-off device and a digital design that can be reproduced, shared, and modified.

There are, however, many challenges in fabricating a device in this way. Soldering individual circuit boards by hand is a time-consuming and error-prone process—and many components are too small for this approach. Other parts might not be available to individuals. Many software design tools (for both electronics and enclosures) were designed with experts in mind, making them difficult to introduce to someone who's just starting out in do-it-yourself (DIY) device production. Digital fabrication itself might be robust and reliable, but it can be limited in the geometries it can produce or the materials it can process.

To investigate this space, I've designed and built a number of devices combining custom electronic circuit boards with digitally fabricated enclosures: radios, speakers, mice, and cellphones (see Figure 1).[1–3] In addition, I've worked with others to make these devices and modify the designs. The case studies apply digital fabrication and DIY practices to create devices for use in daily life, revealing various opportunities and limitations. Before jumping into the details of the case studies, however, let me first present their context.

Digital Fabrication and Embedded Computation

Digital fabrication has enabled individuals and small businesses to create new products in a variety of domains, such as jewelry, furniture,[4] and toys. These products are designed on the computer and prototyped using digital fabrication. Once the design is finalized, online services (such as Shapeways and Ponoko) or other vendors can produce multiple copies using the same fabrication processes, either for distribution by the designer or through direct consumer sales. The use of the same process for both prototyping and production makes it straightforward to try new designs and scale production. It also allows individuals to create or purchase custom variations on a given design, because there's little manufacturing overhead in fabricating a one-off artifact.

At the same time, there's a healthy ecosystem of DIY electronics. Platforms such as Arduino,

 Published by the IEEE CS ■ 1536-1268/14/$31.00

Raspberry Pi, and .NET Gadgeteer[5] let individuals experiment with a wide array of sensors and actuators and build a variety of interactive prototypes. Printed circuit board production, a form of digital fabrication, has created a thriving ecosystem of electronic modules for DIY prototyping and experimentation.[6]

There are fewer examples of people combining these two domains to produce complete consumer electronic devices. Perhaps the biggest proponent of this approach is MIT professor Neil Gershenfeld.[7] In his teaching of the MIT course "How to Make (Almost) Anything" and his establishment of the FabLab network, Gershenfeld promotes a fundamentals approach to electronic device production. Students in his class and visitors to the FabLabs learn to design circuit boards and mechanical structures in software tools and then produce them using digital fabrication processes. My work has been heavily inspired by Gershenfeld's approach.

Some commercial products combine electronic circuit boards with digitally fabricated structures. For example, Adafruit Industries, a hobbyist electronics company, sells digital clock kits with laser-cut enclosures. Some digital fabrication machines are themselves made from fabricated parts. Examples include the RepRap 3D printer (with 3D-printed parts), the Ultimaker 3D printer (with laser-cut parts), and the Othermill (with computer numerical control [CNC] milled parts). Still, such devices seem relatively underexplored—one motivation for the case studies.

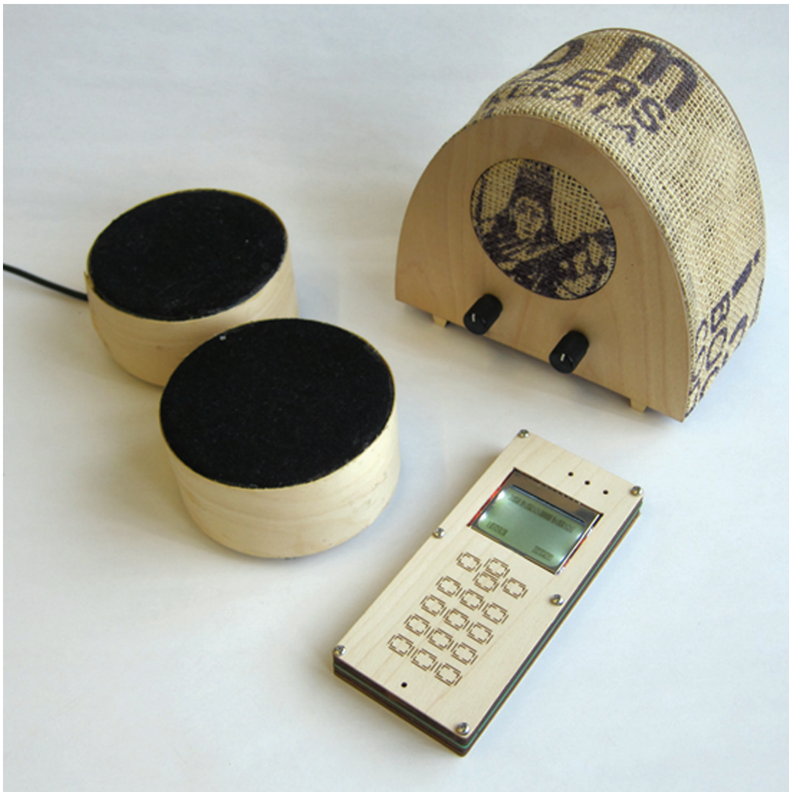


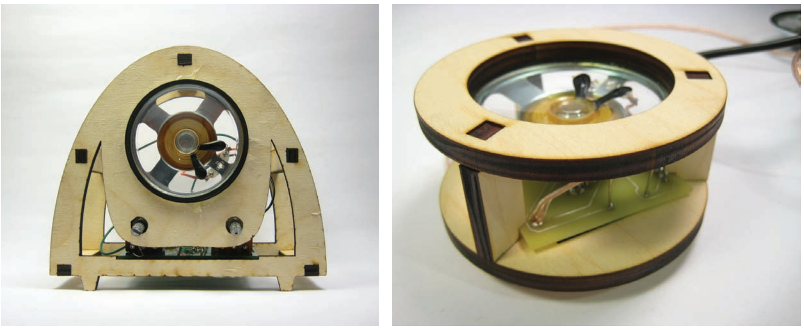
Figure 1. The radio, speakers, and cellphone, fully assembled in their standard forms.

Radios, Speakers, Mice, and Cellphones

During my last four years as a graduate student at the MIT Media Lab, I've tried several approaches to making electronic devices and helping others to assemble or modify them. (In this process, I've been inspired and guided by my former advisor, Leah Buechley.) In making the devices, I've taken advantage of the relatively well-resourced shop at the lab, which includes laser cutters, CNC routers, and small desktop CNC milling machines. Although these machines might not be present in most hobby workshops, it feels meaningful to investigate their implications for DIY practice, because they seem likely to become increasingly available.

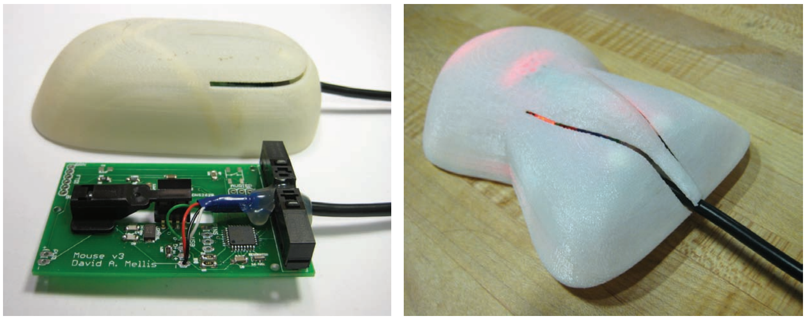
I've deliberately chosen to work with common devices such as radios and cellphones so I can focus on how the devices are made, rather than experimenting with new possibilities for interfaces or functionality. For each device, I've designed the electronics and enclosure together, balancing the need to produce the devices myself with the desire to build robust and attractive products. I describe these devices below, both as examples of the DIY electronic products and to facilitate the subsequent analysis. The details of each device will later serve to illustrate more general points about the impact of digital fabrication on DIY device production.

The first device is the Fab FM, an FM radio (receiver), housed in a plywood, veneer, and fabric structure (see Figure 2a).[3] It has a simple, classic interface, with one knob to control the volume (and a "click" at the end to turn the radio off) and another for tuning the station. These interface with an Arduino-compatible microcontroller, a digital radio receiver module, and a simple audio amplification circuit. The radio's plywood frame and veneer faces were designed using the open source graphics tool Inkscape and were laser cut, with the fabric glued by hand. The Fab FM was developed in collaboration with Dana Gordon and was initiated in Gershenfeld's course at MIT. We designed the circuit board using Eagle,



(a) (b)

Figure 2. The plywood frame and circuit board of the (a) Fab FM radio and (b) speakers.



(a) (b)

Figure 3. The 3D-printed mouse: (a) circuit board and standard enclosure, and (b) enclosure variation.

a software tool popular with electronics hobbyists, and milled initial prototypes of the board in the Media Lab shop. Later, we ordered circuit boards from an online vendor for assembly in a workshop. The Fab FM is now available as a kit (with circuit board, electronic components, and laser-cut parts) from SparkFun Electronics.

The speakers are derived from the radio, retaining its combination of laser-cut plywood, veneer, and fabric housing (Figure 2b).[1] They include a battery-powered (3 AAA) amplification circuit, using two op-amps. There's no microcontroller or software in the speakers, and the only interface is the audio plug (standard 3.5-mm format) for input and an on-off switch. The main design for the speakers consists of two cylindrical housings (left and right speakers). An alternative variation combines left and right speakers into a single housing designed to be hung on a wall; instead of batteries, it plugs into a standard power outlet. As with the radio, the initial circuit boards for the speaker were milled in-house, with later batches ordered from online vendors. The plywood parts were primarily laser cut in-house, although I've successfully ordered them from Ponoko, an online digital fabrication service.

For the mouse, I explored 3D printing, enclosing the circuit board with a 3D-printed base plate and top (Figure 3).[1] I designed the housing using the software design tool Rhino and printed it primarily in ABS plastic on a Stratasys Dimension printer at the Media Lab. Screws hold the 3D-printed parts together and attach them to the circuit board inside. The circuit contains a chip (the ADNS-2620) designed specifically as a mouse sensor; it images the surface below the mouse to calculate its movement. The chip's maker (Avago) also provides a lens and LED clip specifically designed to accompany it. These are held in the appropriate locations by the 3D-printed housing. The circuit also includes an Arduino-compatible microcontroller that communicates with the mouse chip and talks over a USB cable to the computer. For initial prototyping, I used a breakout board from SparkFun that included both the ADNS chip and the Arduino-compatible microcontroller. (In fact, only the lack of buttons prevented this board from serving as a fully functional mouse.) Later, I designed and ordered custom circuit boards based on this breakout board.

The cellphone is an attempt to push the limits of DIY production by making a more complex—and more relevant—device. It's a fully functional mobile phone, capable of making and receiving phone calls, sending and receiving text messages, storing a list of contacts, and serving as an alarm clock. The phone's circuit is based on the Arduino GSM Shield, a prototyping module for connecting to cellular networks. The phone uses the same GSM module (the Quectel M10) as the Arduino shield, which does the main work of connecting to the cellular network (for example, AT&T or T-Mobile in the US) and processing audio. An Arduino-compatible microcontroller communicates with the GSM module and controls the phone's interface, which consists of a set of buttons and a low-resolution display (an LCD on some variants, an LED matrix on others). Several enclosures have been produced for the phone—some by me and some by others—including a laser-cut

plywood and veneer combination, CNC routed solid wood, 3D-printed plastic, and even hand-cut cardboard (Figure 4). The phone's circuit boards were ordered from online services because they're too complicated to easily produce in-house, although I've milled smaller boards to test various portions of the circuit.

Figure 4. Variations on the cellphone enclosures.

Elements of a Fabricated Electronic Device

Analyzing the elements of the devices in the case studies reveals the ways in which digital fabrication affects (or doesn't) the process of producing a device. Consider the following aspects:

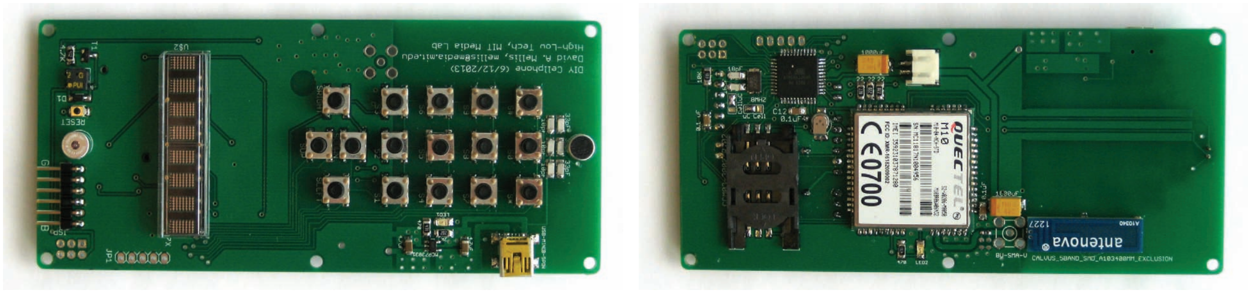
- selection of electronic components and materials,
- design and fabrication of the electronic circuit,
- assembly (soldering) of the electronic circuit,
- design and fabrication of the enclosure,
- assembly of the enclosure, and
- programming of the microcontroller.

Digital fabrication dramatically transforms some of these aspects, but others very little. Let's consider them in turn.

The selection of components is at the core of the process of creating an electronic device. With the right components, the circuit can be trivial; without them, it might be impossible to build. This is true regardless of digital fabrication. At the heart of the radio, mouse, and cellphone is a component that handles much of the device's core functionality (radio receiver chip, optical mouse sensor chip, and GSM module). Without these components, I probably would have never made the devices. Complicating matters is the fact that not all components are available in all countries, or to all individuals. Furthermore, there's no guarantee that components will remain available; the original radio chip and the mouse sensor have since disappeared, and the GSM module might become obsolete as carriers retire the cellular networks it uses. Even worse, in my experience, the most important and difficult-to-replace components are the ones that are most likely to become obsolete. Intermediate businesses targeting hobbyists, such as SparkFun or Adafruit, play an important role in making components available to individuals, but they, too, are limited by their suppliers' decisions and product lines. In a sense, then, digital fabrication lets you create an electronic device around existing components, a powerful possibility but one that is shaped by the industrial forces driving the development of the required components.

The design and production of the printed circuit boards (PCBs) that connect these components provide DIY practitioners with a powerful tool for creating and sharing electronic circuits. PCB production is one of the most mature forms of digital fabrication, with a variety of vendors offering standardized, accessible services in quantities ranging from one to thousands of boards. These PCBs can be designed to accommodate a variety of components, many of which would be difficult or impossible to work with using manual prototyping processes. The PCB files for a circuit capture much of its design, such as its components and how they're connected and arranged. Optimizations to the circuit's function or geometry are also captured, and all of this can be shared in digital form for others to reproduce using a convenient fabrication process or service. I've been able to smoothly transition, for example, from milling individual prototypes in-house, to ordering a few boards from a quick online service, to getting bigger batches of boards from slower, cheaper vendors. The ability to capture and share in digital form information about a circuit's design becomes increasingly valuable as its complexity increases. So, whereas the simple amplification circuit in the speakers could be easily reproduced in a new PCB design or as a hand-assembled circuit, recreating the cellphone (Figure 5) without my files would be difficult and time consuming.

The ease of ordering PCBs increases the emphasis on the accessibility of the software tools for designing them. Many traditional circuit board design tools are expensive and complex, designed for professionals. Because these are software tools, we can exploit the power of the abstractions offered by the computer and the learnings about interface design from human-computer interaction (HCI) research and practice. Recently, some effort has been made on



(a) (b)

Figure 5. Cellphone circuit board: (a) top and (b) bottom.

improving open source options (KiCad [www.kicad-pcb.org], for example) and creating new, easy-to-use tools for beginners (Fritzing,[8] for example). Still, it seems like there is a long way to go—and a lot of exciting opportunities—toward making circuit board design software as accessible and popular as, say, desktop publishing software.

Assembling the circuit inside a device (by soldering the electronic components to the PCB) can be a time-consuming manual process. There are a lot of small parts, and mistakes can be difficult to locate and correct. Automated soldering processes require more setup and negotiation than makes sense for the small quantities in which I've been making products. As a result, the devices I've designed have been hand-soldered, typically by the person who will be using it. This process allows that individual to become familiar with the parts that make up the device and to develop a personal investment through the effort they put into assembling it. However, because it establishes a minimum amount of effort required to make a device, it can serve as a barrier to certain people or contexts. (More later on people's involvement in the DIY device production process.)

Digital fabrication has its most transformative effects on the production of a device's enclosures (or other mechanical parts). Rather than mass-producing thousands or millions of, say, injection molded plastic parts from an expensive and constrained mold, I've been able to laser cut or 3D print parts directly from digital design files in the quantities required. This is also different from hand-crafting (or manually machining) each enclosure because the design—and, therefore, much of the work—is captured in the digital file. Each prototype is thus also an original master that can be reproduced, shared, or further modified. To the extent to which a given digital fabrication process offers the required resolution and tolerances, the process of making a part is largely reduced to a software problem—one that can draw on the numerous advances in computing power and interface design.

Digital fabrication also lends itself to more materials, constructions, and aesthetics than mass-produced parts. I've had good luck with laser-cut wood, for example, a material that's not frequently found in commercial devices. Because it primarily cuts flat parts, however, the laser cutter puts constraints on the possible geometries and typically requires some postfabrication assembly. The 3D printer allows for a much broader range of shapes (such as those found in the mouse enclosures) but it's more constrained in the materials it can work with. Without postprocessing, 3D-printed parts also tend to display characteristic ruled or stair-step surfaces, a result of the layer-by-layer production process. These constraints aren't necessarily more or less limiting than those of mass production (or manual crafting), but they're definitely different, requiring a dedicated design process and aesthetic.

Assembling the digitally fabricated parts, like soldering together the circuit boards, physically involves the individual. There is, again, an opportunity to invest personal meaning in the process and in the crafting of certain elements unspecified by the digital design files (such as the fabric in the radio and speakers). It's also necessary to spend a certain amount of time and effort in the process, although in some cases this can be much less than that required to solder the circuit (particularly with 3D-printed parts, which might only need to be snapped or screwed together). Minimizing the number of different machines and processes involved in fabricating a device's parts makes them much easier to produce, enabling faster iteration and easier production of variants because there are fewer steps to coordinate. Still, it's important to keep in mind that digital fabrication doesn't remove all manual parts of the assembly process, and there are opportunities and challenges in involving the individual in these activities.

Of course, a device's electronics and enclosure need to fit together. Coordinating the two presents its own challenges, particular when using hobbyist

design tools that don't integrate with each other (such as Eagle, Inkscape, and the other programs I've used). The digital nature of the designs does, however, usually make it possible—if not easy—to ensure that the two align precisely. Digital fabrication also helps by making it quick and easy (or, at least, quicker and easier) to make physical prototypes that can be assembled, tested, and refined. Working with a laser cutter, for example, allows for multiple iterations in an afternoon, which is particularly valuable when working with 2D parts that need to be assembled to get a clear understanding of the resulting 3D form. Working with 3D modeling allows for a more complete understanding of the overall geometry in digital form, which is good because 3D printing tends to be slower than laser cutting. Fabricating and assembling a circuit board can be a relatively slow process, which suggests that multiple iterations of an enclosure will be prototyped for each revision. This makes it useful to build in some flexibility to the PCB's design, for example, so that additional parts can be soldered on if needed for a quick test.

The process of programming the processors contained in an electronic device might seem unrelated to digital fabrication; however, the ability to easily reproduce and modify a program is an important part of supporting the replication of digitally fabricated products. For example, by putting the software for my cellphone online, I've enabled other people in other parts of the world to reproduce the device without needing to write any code. In developing the phone, I was aided by the availability of existing, open source libraries for interfacing with many of its electronic components, such as the GSM module and display. This is partly due to my use of accessible and easily solderable components, which, for the same reason, are likely to be used by the electronics hobbyists who are likely to develop such libraries. This process was greatly supported by my use of Arduino-compatible microcontrollers, because a wide range of open source libraries have been developed for them. In this way, the constraints that limit the potential selection of electronic components amenable to DIY practice can promote collaboration by encouraging people to use the same parts in their projects.

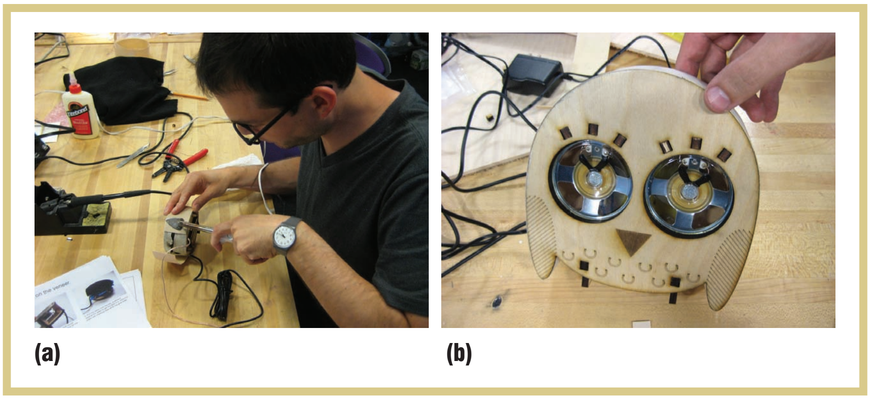


Figure 6. Workshop participants' involvement in building devices: (a) assembling the speakers, and (b) an owl-shaped speaker created by an attendee.

People, Motivations, and Skills

Given the number of different elements required to make an electronic device (even with the help of digital fabrication), it's perhaps not surprising that there are many ways for people to get involved in the process. In working with various groups to build their own versions of my devices, I've started to tease apart some of the motivations and skills involved in the process.

The most straightforward form of engagement has been involving people in assembling the devices for themselves: soldering components to the PCBs, putting together the laser-cut parts, and so on (see Figure 6a). By putting these skills and activities into a meaningful context—that is, making a useful device—I've attracted participants to my workshops who probably wouldn't have participated in a more general soldering or digital fabrication session. In the course of the workshop, participants get the satisfaction of making something for themselves as well as a better understanding of what goes into the devices they use every day. Participants in the cellphone workshops, for example, were thrilled when they got to make the first call on their newly assembled phones, and talked about how they'd never thought about what made up a cellphone until they put one together. In addition, in assembling a device, an individual can add personal touches such as an engraving on a laser-cut wooden part or a custom fabric.

Going beyond assembly to the design itself requires careful consideration of the parts of the device that people will be modifying, the skills required to do so, and the supporting tools and contexts required. In the speaker workshop, two participants were interested in making a custom enclosure to house the circuit board. One, with experience in vector drawing programs like Adobe Illustrator, quickly modified my design into a new shape that looked something like an owl (Figure 6b). The other, who didn't have experience with such software tools, decided that the task was more than he could accomplish in the workshop and proceeded with one of the existing designs. Although most participants personalized the enclosure with an engraved message, going farther than this would have required either previous background with vector drawing tools or a portion of the workshop specifically dedicated to the process. Building an electronic device could offer a compelling context for

learning or applying such vector drawing skills but only if this is explicitly considered as a goal of the process.

My mouse workshop focused on applying computer-aided design (CAD) skills to create device enclosures. For this workshop, I recruited participants with previous experience in the Rhino 3D modeling software. Most of the workshop was spent designing the enclosures. Participants created varied forms to fit the existing circuit board. For many, this was an exciting opportunity to use their 3D-modeling skills to create a physical, interactive object, which there was little opportunity for in the architecture studies that many of the participants were pursuing. Here again, the construction of an electronic device helped motivate participants to exercise existing skills. It also illustrates the possibilities for collaboration between engineers and designers (or individuals with PCB design skills and those with 3D-modeling skills). For example, two individuals can exploit their complementary skills and digital fabrication to construct a complete device.

In short, DIY electronic devices let individuals express many different skills and interests. These can complement each other, allowing for various forms of collaboration between people with different kinds or levels of expertise and interest in the process. Furthermore, these involvements offer different possible outcomes, whether production of useful devices, learning about technology, or social activities.

The Meaning of DIY

Different actors have been involved to different extents in the design and fabrication of the electronic devices described in the case studies. I've put both my creative decisions and my labor into the devices. Others have assembled them from my designs or modified some aspects. There is a temptation to classify these activities, to attempt to decide how much one has to do for oneself for the process to be considered DIY. Instead, I think it's more useful to consider these activities together as an alternative to buying commercial mass-produced devices. By showing how individuals can collaborate in structures other than those of a market or firm, the case studies can be seen as a form of peer production[9] analogous to those found in software or digital media. Because it makes it easy for one person to create and share a design and another to assemble it (or modify it to their own needs or taste), digital fabrication lends itself to these kinds of collaborations and hybrid approaches. This lends increased scale and impact to the expertise of the person creating the initial design. It also offers more sophisticated and appealing designs for individuals to invest personal effort and meaning in making devices from the digital files. The various tiers of involvement possible within peer production systems can serve both to scaffold increased participation and skill in digital fabrication and electronics and to accommodate different levels and types of interest in these activities. As such, they seem likely to remain an important feature of digital fabrication and DIY electronics.

This discussion of DIY, of course, needs to be tempered by the reminder that the electronic devices described in the case studies rely on sophisticated industrial technology, including digital fabrication machines and electronic components. Just as many companies rely on sophisticated supply chains and third-party parts to mass produce devices, DIY devices depend on complex ecosystems. That is, DIY can be seen as the ability of an individual or a group to pull together available parts and processes into a desired product, rather than the extent to which they have made all the pieces from scratch. This makes DIY practice dependent on the technology available—not simply what exists in the world, but what is made accessible to individuals. This, in turn, depends on the decisions of many actors, not all of whom are interested in or supportive of DIY practices. As a result, making specific pieces of technology (such as electronic components or digital fabrication processes) available to individuals can expand the potential range of DIY practice as much as new tools or tutorials. In this respect, platforms and businesses targeting individual hobbyists serve as important intermediaries between industrial production and individual consumption. Still, given an available set of technology, the things people can do with it are largely shaped by the tools and techniques available for them to do so, an area marked by rapid changes and unexplored opportunities.

Making specific pieces of technology available to individuals can expand the potential range of DIY practice as much as new tools or tutorials.

The continuing evolution and increasing accessibility of digital fabrication and embedded computation creates many new technological possibilities. HCI researchers and others are exploring various tools and techniques for exploiting these technologies. Some of these involve novel mechanisms for integrating interfaces and electronics with digitally fabricated parts. Examples include vinyl cutting or printing copper to serve as touch-sensitive electrodes[10]; integrating optical functionality into 3D-printed objects[11]; using a camera to detect the motion of 3D-printed interface elements[12]; and creating sophisticated servo motors and other mechanisms from electronic circuits and

3D-printed structures.[13] Others have created new software or interactive interfaces for designing the objects to be fabricated. These tools include Fritzing, a simplified interface for PCB design[8]; SketchChair, a domain-specific tool for designing chairs[4]; Enclosed, a tool for designing a laser-cut enclosure for .NET Gadgeteer prototypes[14]; and Interactive Construction, a system for interactively specifying object geometries on the surface of a material in the bed of a laser cutter.[15]

These many attempts to investigate the implications and possibilities of digital fabrication raise multiple questions about the future of this technology and its uses:

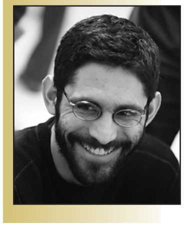
- *Access*. To what extent will advances in underlying technologies (digital fabrication and electronics) be translated into forms that are accessible to individuals? Without this translation, the possibilities for DIY might become increasingly outdated and irrelevant. Furthermore, the relative progress of different technologies (for example, 3D printing versus laser cutting) will shape the constructions and aesthetics of DIY devices as well as the balance between scalable digital processes and personal physical ones.
- *Appeal*. What skills and interests will the tools target for working with new, digital technologies? This will play a large role in influencing who is interested in working with those technologies and what they will make with them.
- *Collaboration*. To what extent will the tools for designing devices foster collaboration between people with different skills and levels of involvement? Will DIY devices remain the province of a few deeply invested individuals, or will it become a peer-production process involving many people in multiple ways?
- *Viability*. Will DIY be a viable method for creating electronic devices? (It's not necessarily the cheapest, easiest, or best method, but one that's close enough to remain appealing for other, more personal motivations.)

Although the continuing advance of technologies for digital fabrication and embedded computation shows no signs of abating, the extent to and ways in which these technologies will translate into DIY practice depends on many unanswered questions.

REFERENCES

1. D.A. Mellis and L. Buechley, "Case Studies in the Personal Fabrication of Electronic Products," *Proc. ACM Conf. Designing Interactive Systems* (DIS 12), 2012, pp. 268–277.
2. D.A. Mellis and L. Buechley, "Do-It-Yourself Cellphones: An Investigation into the Possibilities and Limits of High-Tech DIY," *Proc. ACM CHI Conf. Human Factors in Computing Systems* (CHI 14), 2014, pp. 1723–1732.
3. D.A. Mellis, D. Gordon, and L. Buechley, "Fab FM: The Design, Making, and Modification of an Open-Source Electronic Product," *Proc. Int'l Conf. Tangible and Embedded Interaction* (TEI 11), 2011, pp. 81–84.
4. G. Saul et al., "Sketchchair: An All-in-One Chair Design System for End Users," *Proc. Int'l Conf. Tangible and Embedded Interaction* (TEI 11), 2011, pp. 73–80.
5. N. Villar et al., ".NET Gadgeteer: A Platform for Custom Devices," *Proc. Int'l Conf. Pervasive Computing* (Pervasive 12), 2012, pp. 216–233.
6. D. Mellis and L. Buechley, "Collaboration in Open-Source Hardware: Third-Party Variations on the Arduino Duemilanove," *Proc. ACM Conf. Computer Supported Collaborative Work* (CSCW 12), 2012, pp. 1175–1178.
7. N. Gershenfeld, *Fab: The Coming Revolution on Your Desktop—From Personal Computers to Personal Fabrication*, Basic Books, 2007.
8. A. Knörig, R. Wettach, and J. Cohen, "Fritzing: A Tool for Advancing Electronic Prototyping for Designers," *Proc. Int'l Conf. Tangible and Embedded Interaction* (TEI 09), 2009, pp. 351–358.
9. Y. Benkler, *The Wealth of Networks: How Social Production Transforms Markets and Freedom*, Yale Univ. Press, 2006.
10. V. Savage, X. Zhang, and B. Hartmann, "Midas: Fabricating Custom Capacitive Touch Sensors to Prototype Interactive Objects," *Proc. ACM Symp. User Interface Software and Technology* (UIST 12), 2012, pp. 579–588.
11. K. Willis et al., "Printed Optics: 3D Printing of Embedded Optical Elements for Interactive Devices," *Proc. ACM Symp. User Interface Software and Technology* (UIST 12), 2012, pp. 589–598.
12. V. Savage, C. Chang, and B. Hartmann, "Sauron: Embedded Single-Camera Sensing of Printed Physical User Interfaces," *Proc. ACM Symp. User Interface Software and Technology* (UIST 13), 2013, pp. 447–456.
13. P. Schmitt, "Original Machines: Developing Tools and Methods for Object-Oriented Mechatronics," PhD thesis, Media Arts and Sciences, Massachusetts Institute of Technology, 2011.
14. C. Weichel, M. Lau, and H. Gellersen, "Enclosed: A Component-Centric Interface for Designing Prototype Enclosures," *Proc. Int'l Conf. Tangible, Embedded and Embodied Interaction* (TEI 13), 2013, pp. 215–218.
15. S. Mueller, P. Lopes, and P. Baudisch, "Interactive Construction: Interactive Fabrication of Functional Mechanical Devices," *Proc. ACM Symp. User Interface Software and Technology* (UIST 12), 2012, pp. 599–606.

the AUTHOR



David A. Mellis is a PhD student at the MIT Media Lab. His research interests include digital fabrication and DIY electronics. Mellis has a master's degree in interaction design from the Interaction Design Institute Ivrea, Italy He is one of the creators of Arduino, an open source hardware and software platform for electronic prototyping. Contact him at mellis@media.mit.edu.