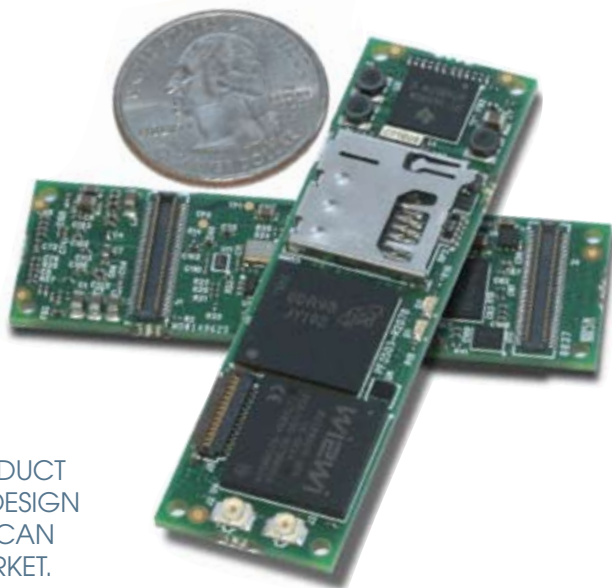


**M**any designers are familiar with open-source software, such as Linux, in which the source code is available to all. However, fewer are familiar with organizations offering open-source hardware. These organizations release free information, including schematics, BOM (bill-of-materials) information, and PCB (printed-circuit-board)-layout data, covering the overall hardware design. Designers with this information can build or add to a freely available design. In many cases, open-source software supports the original design, providing additional advantages. Some aspects of open-source hardware go beyond the sharing of the design itself. These aspects can save time and



BASING YOUR PRODUCT ON A REFERENCE DESIGN OR DEMO BOARD CAN SPEED TIME TO MARKET.

# TAKE ADVANTAGE of OPEN-SOURCE HARDWARE

BY GERALD COLEY • TEXAS INSTRUMENTS

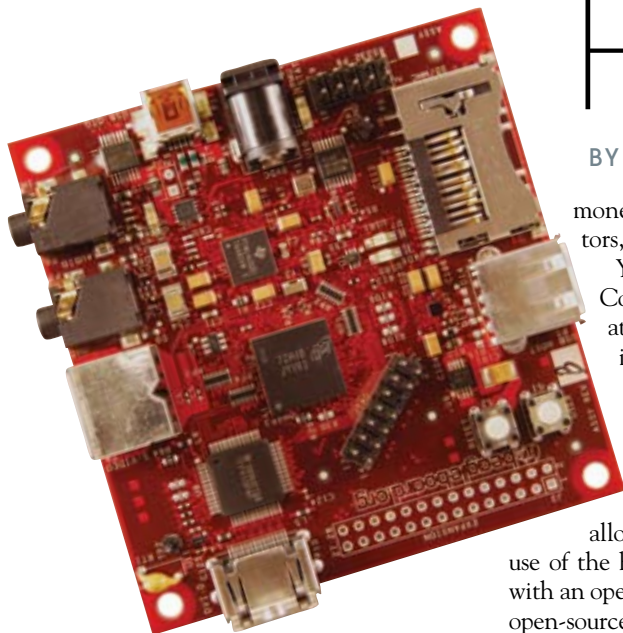


Figure 1 The BeagleBoard (bottom left) and the Overo expansion boards from Gumstix (top right) are examples of available open-source hardware.

money for not only hardware developers but also PCB designers and fabricators, contract manufacturers, and even software developers.

You can license open-source projects from organizations such as Creative Commons, which offers the Attribution-ShareAlike licensing program. Creative Commons stipulates that a user must attribute the open-source work in the manner that the original designer specifies but not in a way that indicates that the original designer endorses the user's work. Likewise, if users provide that work as open-source hardware, releasing it back to the community for access by others, then they must provide that work under the same Attribution-ShareAlike licensing (**Reference 1**). Other licenses, such as the modified BSD (Berkeley Software Distribution),

allow for the assignment of copyrights and provide certain restrictions to the use of the hardware design (**Reference 2**). Be sure to read the license that comes with an open-source design before using it. If users are considering creating their own open-source design, they need to figure out which license works best for them.

You must consider several factors, including power, cost, and documentation, when selecting an open-source-hardware platform. Make sure that the license provides comprehensive, high-quality documentation, including schematics, BOM, and PCB data. The documentation must have the support of a large community of users, and it must align with your product's needs.

Some popular open-source-hardware platforms include Gumstix, Arduino, and

the BeagleBoard (Figure 1). Gumstix uses an open-source-hardware model in posting the schematics and layouts of all the company's Overo-series expansion boards. Arduino employs a microcontroller as its hardware and has its own community of designers and hobbyists. Arduino's schematics come in Eagle and PDF (portable-document format), and the PCB information is in Eagle. The Creative Commons license covers licensing for the platform, and extensive libraries and software support are available. The BeagleBoard uses a Texas Instruments OMAP3530 (Reference 3). A large, Linux-based open-source-software community supports the BeagleBoard, and schematics are available in PDF and Cadence's OrCAD. The BOM is in Microsoft Excel, and PCB information is in Cadence's Allegro and Gerber files.

Due to its high technology level, the BeagleBoard presents some interesting challenges for users of the OMAP3530 device, but you can overcome the challenges by taking advantage of open-source hardware. The 515-pin OMAP3530 device uses 0.4-mm-pitch balls and supports POP (package-on-package) technology, which mounts the memory devices on top of the processor. This technology can create challenges, including schematic design and PCB layout, fabrication, and assembly.

### STREAMLINE THE DESIGN

You can reduce risk by basing designs

#### AT A GLANCE

- ▾ Open-source hardware offers an advanced start on your design.
- ▾ Open-source software complements open-source hardware.
- ▾ Open-source hardware prepares your PCB (printed-circuit-board)-fabrication and -assembly houses for high-volume production.
- ▾ You may want to share your improvements by making them open-source additions, as well.

on open-source hardware and taking advantage of a proven design that has operated successfully in the past. Thus, you can work from a known starting point and easily see what's there, what's missing, and what is unnecessary because you have access to both the finished hardware and the complete design. It also saves development time. Having access to the complete BOM with part numbers enables you to quickly adjust it. You can look for places to substitute your favorite capacitors and resistors and get access to the information on any unfamiliar parts. As long as the cost and specification of the part in use are on target, you should have no problem with using it. If you are unfamiliar with a part, however, you can improve the design by selecting a part better suited for your design's needs.

For instance, the OrCAD schematic

tool lets you quickly and easily add devices to the schematic. It takes advantage of the unused pins on the processor and even replaces devices on the open-source-hardware design. Creating components can take a lot of effort, so using those on the schematic saves time and reduces the risk of errors. Alternatively, you can use a PDF version of the hardware schematic to create schematics with your favorite tool. This approach can be an advantage because re-creating the design gives you more in-depth knowledge of how the design works and where problems might arise. The Beagle-Board's documentation includes a fairly detailed reference manual that can answer questions about the design. Users also can access the support community for additional help. Don't be afraid to ask for help; open-source hardware is all about learning from other people's mistakes.

Using the Allegro file format is the shortest path to completing a layout. You can import footprints from this database to your library, saving a significant amount of time. Because you already have a working board, you can use these footprints with confidence. If your board or prototype is similar to the BeagleBoard design and you need to add several components, you can simply do an ECO (engineering-change order), meaning that you use the current layout and add only key components so that you don't disturb the basic layout.

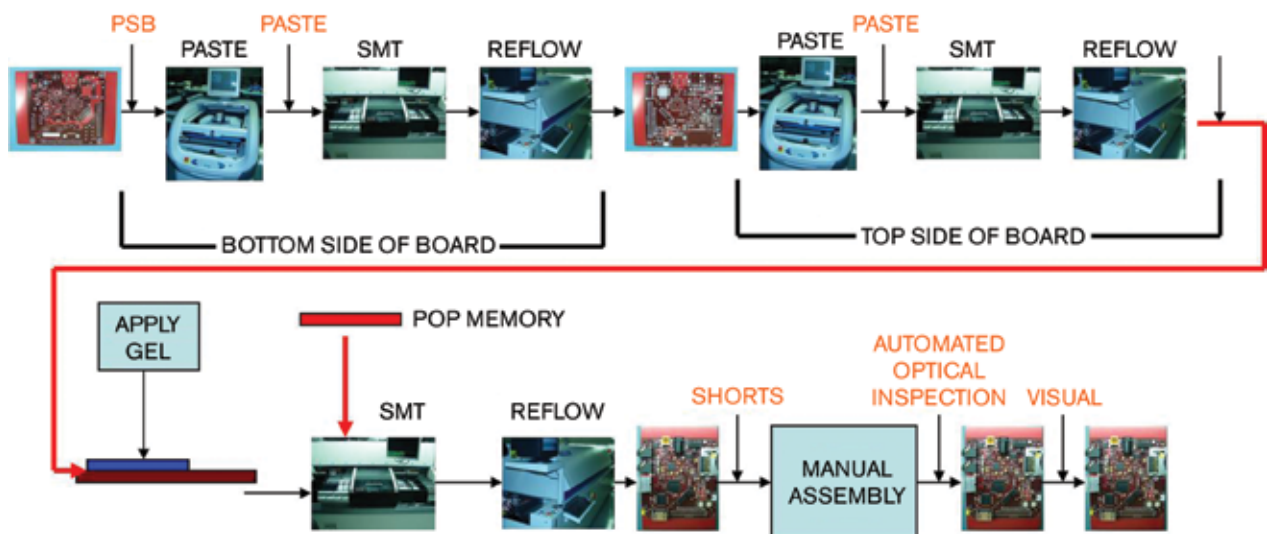


Figure 2 A board-assembly flow begins with a PCB and ends with a visual inspection of the work.

Allegro has a useful free viewer for reading design files. This viewer exposes the layer stacks and all of the PCB information, which provides a guide when designing the board on your own tool. In this way, you can see board routing, trace widths, and trace spacing. Another option is to convert the database to other tools using available and third-party tools. You must be careful not to lose design information when using this method, however. You can use Gerber files to double-check the information and as learning tools. These files are not user-friendly but can be helpful in creating your layout. The OMAP3530 device also includes a design-guideline application note to assist in layout.

### GETTING UP TO SPEED

Many new devices use more advanced technology, including blind vias, stacked vias, narrow traces, and via-in-pad technology, than the technology from your PCB supplier. Although the suppliers may be able to handle these advances, they may not have previously needed to provide it. If a user is comfortable with a PCB vendor because it's a known entity, though, it may be worth getting the vendor to the point at which it can handle the new needs. However, depending on your requirements and confidence level, you may be better off going to a vendor that has experience with the type of board you are working on. Using the available CAD data, the PCB vendor can get up to speed and prototype an initial board so that you are confident that the company can build it. If the vendor needs to adjust certain aspects of the design, it can modify the Gerber files for its normal processes.

Expected yield determines the cost of building boards. Building the board as it is gives vendors insight into yield. They can then prevent any problems that arise when building the new board. For example, if they have problems with via-

in-pad technology, you should avoid using it in your design. The assembly house also may encounter problems in assembling your board. You can get the assembly house up to speed by using open-source hardware. Alternatively, you can use your own assembly house, if applicable, to build the entire board or a subset of the board.

In many cases, it can take several passes to work out the wrinkles in assembly. By using the open-source-hardware design and assembling as many boards as you need, you can solve any problems. It is better to work on these problems with a proven and tested board instead of your first prototype, on which it may be more difficult to find issues. The BeagleBoard also has a POP assembly guideline (**Reference 4**). As in the case of the PCB-layout guidelines, it also uses the BeagleBoard design (**Figure 2**).

There is always pressure to complete the hardware so that you can start the software verification. Basing your design on open-source hardware allows an early start on software development by using the available open-source-hardware boards to begin the development effort. Because open-source hardware typically also has a software component, the software team can jump-start its efforts by using the available code. This approach allows the hardware team to make sure it gets the design right on the first pass.

In addition, the software team may discover that it needs to change the hardware to improve performance or to add a feature. Making a lot of changes to the basic design quickly diminishes this advantage, however. Make sure that the basic design components, including the memory, power management, and key debugging peripherals, remain the same. Additionally, pay careful attention to the available software to see whether it offers the necessary applications or functions. Although software developers can write the software themselves, they typically would rather use the already-available software.

In summary, using the OrCAD schematic design tool or implementing your own version using open-source hardware reduces risk and saves development time. You then need to focus only on what to add or remove from the basic design to complete the final design. You can use the Allegro CAD files or Gerber files to provide a map with which

---

### FOR MORE INFORMATION

**Arduino**  
[www.arduino.cc](http://www.arduino.cc)

**BeagleBoard**  
[www.beagleboard.org](http://www.beagleboard.org)

**Berkeley Software  
Distribution**  
[www.bsd.org](http://www.bsd.org)

**Cadence**  
[www.cadence.com](http://www.cadence.com)

**Creative Commons**  
<http://creativecommons.org>

**Gumstix**  
[www.gumstix.com](http://www.gumstix.com)

**Realtime Technology**  
[www.realtimedsp.com.cn](http://www.realtimedsp.com.cn)

**Texas Instruments**  
[www.ti.com](http://www.ti.com)

you can reuse whatever works correctly. Next, select a PCB vendor to accurately build the technology and save time and cost for the prototype run. This approach increases the chance of first-pass success and reduces the number of costly problems that would require re-spins and debugging during the prototype phase. Then, focus on getting the assembly house to build the open-source-hardware board. Now that the hardware prototypes are ready to run software and you have tested the software from the open-source-hardware board, there is a much higher chance of success.

### GIVING BACK

Open-source hardware is about sharing work with others for everyone's benefit. It is acceptable if you never meant for the product to be open. You need not make your changes available to the community. In the spirit of open-source hardware, however, it's beneficial for all parties to provide upgrades and additions to the community whenever possible so that the next user can add

other enhancements. When you add a function to hardware, it affects the software, which adds a reason to enhance and improve the overall performance of the software to take advantage of the new feature. As developers design products based on this design, another community member has perhaps added the function with the already-completed software work to help make it a better product.

In the future, more companies will offer varying levels of open-source hardware to their customers and the community at large, creating an environment in which developers spend most of their efforts on improving rather than re-creating the design. The community can benefit from this common goal, so keep your

eyes open for the next entrant in the world of open-source hardware. **EDN**

### REFERENCES

- 1 "License Your Work," Creative Commons, <http://creativecommons.org/choose>.
- 2 Nelson, Russell, "The BSD License," Berkeley Software Distribution, Oct 31, 2006, [www.opensource.org/licenses/bsd-license.php](http://www.opensource.org/licenses/bsd-license.php).
- 3 "OMAP3530 Applications Processor," Texas Instruments, [www.ti.com/omap3530-oshw](http://www.ti.com/omap3530-oshw).
- 4 Gutierrez, Keith, and Gerald Coley "PCB Assembly Guidelines for 0.4mm Package-On-Package (PoP) Packages, Part II," Texas Instruments, April 2008, [www.ti.com/litv/pdf/spraav2](http://www.ti.com/litv/pdf/spraav2).

### AUTHOR'S BIOGRAPHY

Gerald Coley is a systems engineer at Texas Instruments, where he has worked for 13 years. Coley focuses on the OMAP processors and has designed more than 20 development boards, including the Beagle-Board.

+ Go to [www.edn.com/090820df](http://www.edn.com/090820df) and click on Feedback Loop to post a comment on this article.

+ For more technical articles, go to [www.edn.com/features](http://www.edn.com/features).