According to the World Health Organization, 39 million people worldwide are blind and another 246 million have low vision (www.who.int/mediacentre/factsheets/fs282). To access electronic content, these individuals have two options: text-to-audio conversion—which isn’t always appropriate or desirable—and tactile displays. Tactile displays use pins that raise and lower to form braille characters. In addition to being nearly silent, such braille displays are more useful for conveying highly technical text and mathematical notations.

A braille tactile display provides one row of up to 80 characters. Each character consists of six or eight dots arranged in rows of two. The characters are refreshed by raising or lowering pins as the user navigates the computer screen. However, these one-line displays can be limiting for readers—especially compared with printed braille, which allows for 2.5D graphics that readers can use to naturally navigate to different parts of the page. However, multiline tactile displays have been difficult to design because of space constraints, the force required for mechanical actuators, and the actuation speed needed to make them readable and usable.

Polymer Braille is developing a 2.5D braille display using electroactive polymer (EAP) technology with piezoelectric actuators that raise and lower the braille dots. While the company has focused on the actuator technology, students in North Carolina State University’s electrical and computer engineering senior design class have been developing system prototypes. For these proof-of-concept systems, they used LEDs instead of actuators to represent the braille output. The LEDs let sighted software design engineers use the prototypes for content development and feature testing.

The 2014–2015 senior design team built an initial prototype that controlled the LEDs to translate text to braille characters. This was an output-only device, so users couldn’t send text or commands to the computer. The 2015–2016 team enhanced the initial prototype: they added Bluetooth wireless communication, two-way communication using a braille keyboard with navigation buttons, and an enclosure that supported a 12 × 30 character display.

**HIGH-LEVEL DESIGN**

Figure 1 is a high-level block diagram of the system components. From the computer, text is translated into
braille by the NonVisual Data Access (NVDA) open source software package (www.nvaccess.org). The braille characters are transmitted via USB or Bluetooth to the braille display device, where they’re displayed on the LED array. Information can also be transmitted from various input devices to the computer, in this case, an 8-key braille keyboard and a scroll wheel. The keyboard follows the Perkins Brailler standard: each key corresponds to one dot of the braille character.

The device has two microprocessors: a main controller and a peripheral controller. The main controller manages communication between the device and the computer. When a row of characters has been buffered, the main controller sends the braille data to the LED array, which displays them. The main controller also receives information (text and control signals) from the peripheral controller, which it then sends to the computer via the USB or Bluetooth channel.

**HARDWARE CHALLENGES**

The students made several changes to the previous year’s prototype, including adding a keyboard and changing the display from 4 × 40 to 12 × 30 characters.

They designed a new motherboard to accommodate the keyboard, as shown in the lower part of Figure 2. The motherboard includes both microcontrollers. The keyboard buttons communicate with the peripheral controller, which interprets the signals and sends characters to the main controller. The different components communicate using the I²C (Inter-Integrated Circuit) serial bus protocol.

The main controller also drives the LED cards for the display, communicating directly with each card over I²C. This is an improvement over the previous version, which used dedicated controllers for a pair of rows. The reduction

---

**PROJECT DETAILS**

- **Title:** Tactile Braille Display
- **School:** North Carolina State University, Raleigh, NC
- **Industry mentor:** W. Shepherd Pitts (Polymer Braille, Raleigh, NC)
- **Faculty mentors:** Bobby Compton, Rachana Gupta, and Jean Shortley
- **Support:** National Science Foundation (SBIR Phase II grant IIP-1353625)
in hardware allows the device to be thinner. As mentioned, the LEDs are temporary substitutes for the actuators that will raise and lower dots to form the braille characters. The actuators require relatively high voltage (hundreds of volts), which would require significant changes to the circuit board design. This requirement was determined to be out of this specific prototype’s scope because of time limitations.

To implement the braille keyboard, the students connected key pads to linear switches that were connected to the peripheral controller. They chose a four-way navigational switch with a scroll wheel, rather than a joystick, because it felt more intuitive. Thumb-activated space, backspace, and enter keys were added along the bottom of the device.

The team also designed the outer enclosure by carefully modeling all of the internal printed circuit boards and components. They completed the modeling in SolidWorks (www.solidworks.com) and 3D-printed the enclosure with Shapeways (www.shapeways.com).

SOFTWARE CHALLENGES

This prototype’s main software challenge, relative to the initial version, was the two-way communication between the device and the computer. The previous prototype was only a display, taking character data from the computer. This latest version requires the transmission of keyboard data from the device to the computer as well.

The NVDA software was used to translate text to braille. To handle input from the user, the student team implemented an NVDA plug-in that allows the software to respond to users’ navigational and braille keystrokes. These events are translated into actions on the main computer screen.

In addition to input data from the computer, the main controller receives input from the peripheral controller. The peripheral controller detects the keystrokes and converts the braille to ASCII characters. It also detects when users have pressed the directional buttons on the scroll wheel or scrolled the wheel. This information is encoded and sent to the main controller over the I2C bus.

The students successfully demonstrated their prototype at the university’s May 2016 Design Day (watch an interview with the team at youtu.be/RjAXwlVyzNU?list=PL2Tk3txD5ShlQ2hbsj77IFruFS4-9KvjsGD1). Moreover, they learned valuable lessons about communication and project management—although each member had clear responsibilities, they had to collaborate and adapt to the challenges posed by a complex project. Two of the students continue to work with Polymer Braille. The company anticipates bringing their product to market, offering new ways for people with visual impairments to interact with technology and more actively participate in the global digital community.

**Figure 2.** Final prototype internal components. The LED display is on top, with the microcontrollers and keyboard below.