Adding A Microcontroller to The Beginner Bot

Part 3

by Gordon McComb

The study of robotics is a lifelong journey. Each new robot is a new adventure, and a fresh chance to expand your learning. For the last several issues, I've written about the Beginner Bot: an affordable and expandable platform that explains basic robotic concepts in easy steps.

In the first installment, you learned how to construct the Beginner Bot platform using wood or plastic, and how to steer it using mechanical switches from a tethered control panel. From there, you learned how to replace the switches with fully electronic influence, adding twin "eyes" and a simple one-chip brain so that your bot could follow the beam of its master's flashlight.

In this article, you'll discover how to add a low cost microcontroller, replacing hard-wired control circuitry with software programming. By moving up to a miniature computer to operate your Beginner Bot, you'll be able to modify its action and behavior just by rewriting a few lines of code.

Using the PICAXE AXE401 Development Board

I'm a sucker for ease of use, but I also like to save a dime here and there. The AXE401 development board for the PICAXE nicely combines both. The '401 is form factor-compatible with the Arduino Uno, and even accepts expansion shields designed for the Arduino.

Like the Arduino, the board provides 20 input/output (I/O) pins, six of which can serve as both analog inputs or standard digital I/O. Instead of using an Atmel ATmega328P microcontroller — as the Arduino does — the AXE401 uses a PICAXE 28X2.

If you're not yet familiar with it, the PICAXE family of microcontrollers is based on various versions of the Microchip PIC, pre-coded to permit easy programming using a Basic-like language. All the PICAXE chips come with numerous built-in features handy for robotics. These features include remote control decoding, I²C interfacing, R/C servo operation, sound and tone making (through an external speaker), and — for the more advanced chips like the 28X2 — multi-program storage. With the latter, you can store up to four separate programs in the chip's memory and run them at will.

As with all microcontrollers, you program the PICAXE from your computer. You then download these programs to the PICAXE via a connecting cable. Free software is provided for building and downloading PICAXE programs. Visit the PICAXE website at **www.rev-ed.co.uk/picaxe** for details. There are versions of the programming editor (and other tools) for Windows, Macintosh OS X, and Linux.

The PICAXE controllers are affordably priced, starting at about $3. The AXE401 comes in kit form and combines the PICAXE 28X2, circuit board, connectors, and all the electronics. The cost is about $18. The '401 only uses through-hole parts. Soldering isn't difficult, and takes about 20 minutes. A completed AXE401 board is shown in **Figure 1**.

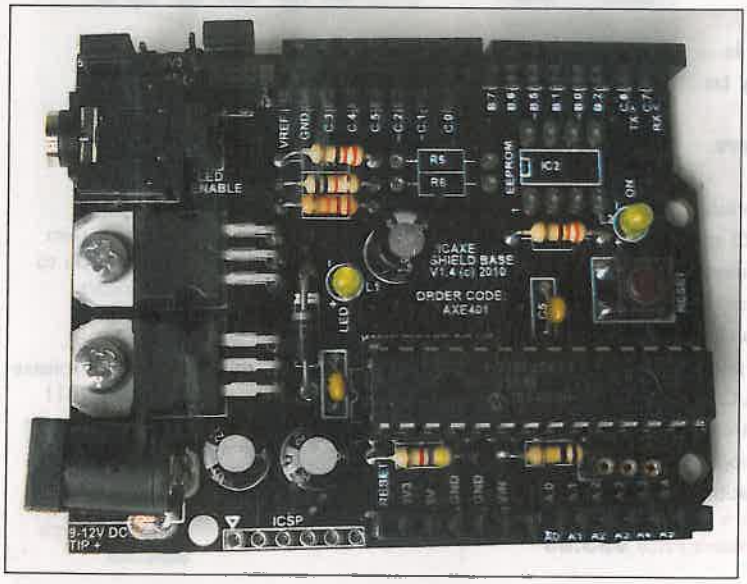
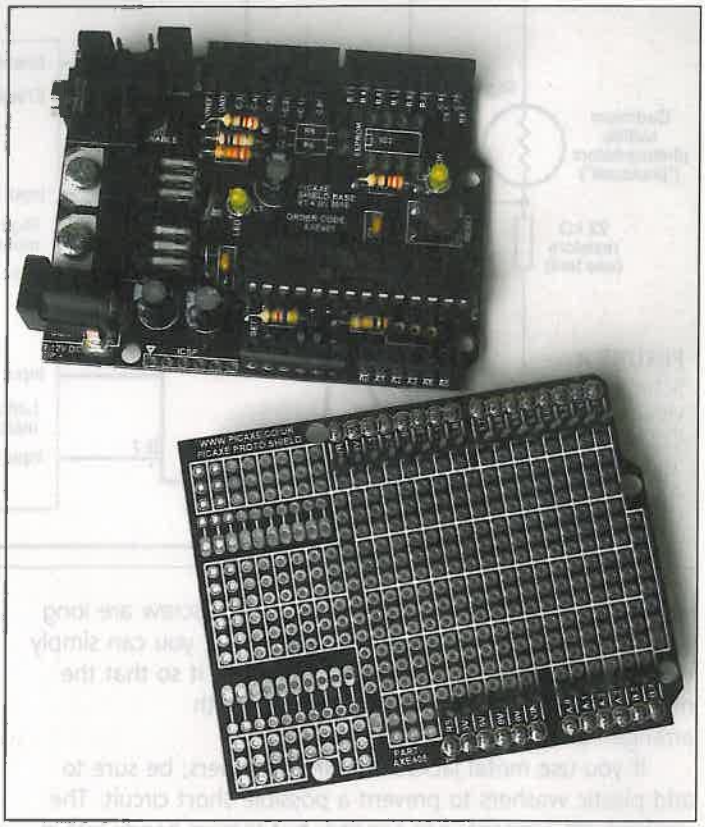


FIGURE 1. The AXE401 development board contains a PICAXE 28X2 microcontroller, voltage regulators, and assorted electronics for quick and easy hookup to your projects.

FIGURE 2. The prototyping board lets you stack or sandwich a shield on top of the AXE401. In the middle of the shield, you can place a mini solderless breadboard for rapid prototyping.



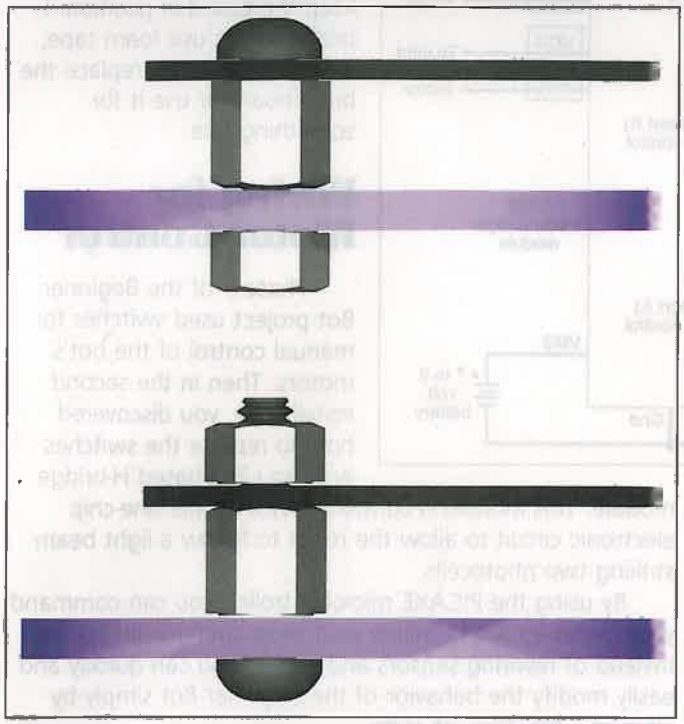


FIGURE 3. Attach the AXE401 board using jackscrews. You can insert the male threaded end of the jackscrew from the top or bottom of the '401 board. You can also use plastic or metal standoffs.

Tip! You can read more about the PICAXE family of processors — including the AXE401 — in Ron Hackett's columns in *Nuts & Volts*, the sister publication to *SERVO*. Each month, Ron explores the world of the PICAXE, and demonstrates various programming concepts and projects you'll want to try. Be sure to also read the three-part PICAXE manual that's provided in PDF format. It's included in the programming editor software download.

You need a download cable to program the 28X2 chip on the '401 board. Two versions of cable are available from the PICAXE folks, or you can make your own if your computer has an older RS-232 serial port. Ready-made download cables are available for RS-232 serial (about $6) and USB.

The cost of the USB cable is about $25, but keep in mind you only need one cable, no matter how many PICAXE chips or development boards you have. This makes the PICAXE particularly attractive in classroom settings, where the one download cable can be shared among all the students. The cable is only needed when downloading programs to the PICAXE. Once the program is downloaded, the cable is removed from the AXE401 board, so that the Beginner Bot can trawl around on its own.

Remember that to use the USB cable, you need to first install the PICAXE USB drivers. They're included with the programming editor software. The USB drivers have their own installation program which you should run before plugging in the cable. Once the drivers are installed, the USB cable will appear as a serial communications port to your computer.

Included with the AXE401 is a bare prototyping shield. It's intended for expanding the '401 with additional external components. It sandwiches with the main board via a set of header pins as shown in **Figure 2**. I'm showing the board with two sets of male headers: one set points down and mates with the AXE401 board; and the other set points up, allowing connection to a mini solderless breadboard on top. I'll discuss connecting the AXE401 to a solderless breadboard later in this article.

Mounting the AXE401 Board on the Beginner Bot

Use the second "deck" we added in Part 2 of this series for mounting the AXE401 development board. If you followed along and built that version, you'll want to carefully remove the wiring and components from the mini solderless breadboard and put them aside — you'll be reusing some of the parts.

Remove the second deck from Beginner Bot and drill two holes to mount the AXE401. Pick the holes in the upper right and bottom left of the board. Center the board on the deck (but leave a little more space toward the front) and mark the position for the two holes with a scribe or small nail. Carefully drill new holes using a 1/8" drill bit.

Use a pair of 4-40 jackscrews, nuts, and 4-40 x 1/4" machine screws to mount the AXE401. Jackscrews are like miniature standoffs with male threads on one end, and female on the other. If you used 1/4" thick material for the

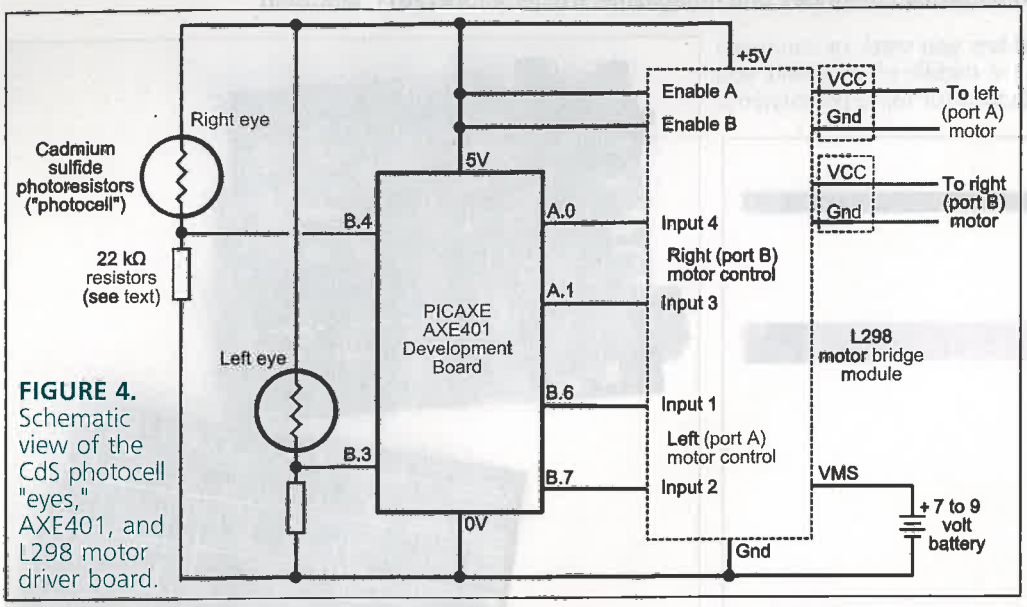


FIGURE 4. Schematic view of the CdS photocell "eyes," AXE401, and L298 motor driver board.

second deck, be sure the threads of the jackscrew are long enough to catch a nut on the other side. Or, you can simply turn the jackscrew upside down and mount it so that the male threads point up. See **Figure 3** for both arrangements.

If you use metal jackscrews and fasteners, be sure to add plastic washers to prevent a possible short circuit. The washers are probably not needed, but they're handy just in case a fastener gets too close to an exposed circuit part.

No jackscrews handy? Most any 1/4" to 1/2" metal or plastic threaded standoff will work. These are sold by numerous online retailers, including All Electronics, Pololu, and others. Pick those made for #4 hardware; the holes in the AXE401 board won't accommodate the larger stuff.

Once the AXE401 board has been mounted, you can attach the prototyping shield on top with the mini solderless breadboard in the middle. Most mini breadboards come with self-adhesive foam tape. For my prototype, I just soldered in some additional header pins front and back to keep the board in position. I prefer to not use foam tape, so that I can easily replace the breadboard or use it for something else.

Wiring for Motor Control

Phase 1 of the Beginner Bot project used switches for manual control of the bot's motors. Then in the second installment, you discovered how to replace the switches with an L298-based H-bridge module. This module is controlled by a simple one-chip electronic circuit to allow the robot to follow a light beam striking two photocells.

By using the PICAXE microcontroller, you can command your Beginner Bot to follow your programmatic instruction. Instead of rewiring sensors and inputs, you can quickly and easily modify the behavior of the Beginner Bot simply by altering a few lines of code.

See **Figure 4** for how to connect the AXE401 development board to the L298 H-bridge module. Use the mini breadboard as an interface between the motor bridge, the two photocells, and the AXE401 board. **Figure 5** shows the circuit in breadboard view.

The jumper wires you use between the AXE401 and the breadboard will depend on the kind of header pins you've installed on the protoboard shield. For my prototype, I used standard male header pins which require male-female jumpers. I'm using SchmartBoard #920-0022-01 5" male-female jumpers. You can also create your own using crimp-on breadboard pins (Electronix Express #03BBPINS), with your choice of 22 or 24 gauge stranded conductor any length you wish.

Another method is to use two sets each of 1x6 and 1x8 female headers, then connect to the breadboard with the more commonly available (or made) male-male jumpers. One such product is Pololu #1016. You'll still need the male headers underneath to connect the shield to the AXE401 board.

For an all-in-one solution, there's the "stackable" header which combines long male header pins on the bottom with female connectors on the top. These are available among numerous sources, including

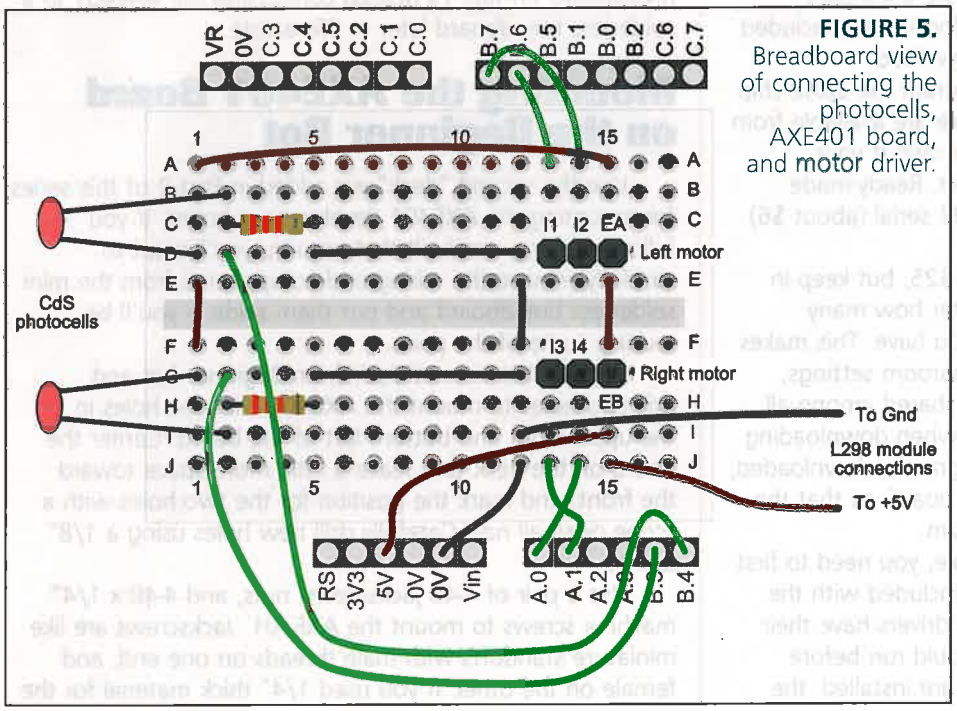


FIGURE 5. Breadboard view of connecting the photocells, AXE401 board, and motor driver.

SparkFun (#PRT-10007) and Adafruit (item #85). See the **Sources** box website addresses for these and other online retailers.

Here's how the circuit works: Two cadmium sulfide (CdS) photocells detect the amount of light falling on them. This type of photocell exhibits a change of resistance depending on the amount of light: The less light, the higher the resistance; the more light, the lower the resistance. For each CdS "eye," a 22 kΩ resistor turns the resistive output to a varying voltage – the CdS cell resistor and the fixed resistor form a voltage divider circuit.

The voltage produced at the junction between these components stretches from between zero and five volts. The outputs of the sensors are connected to two of the AXE401's analog inputs – the pins marked B.3 and B.4.

The value of 22 kΩ for the resistors connected to each CdS cell is determined experimentally. There are no standards in CdS photocells, and their dark and light resistance can differ greatly – even among components of the same type. You'll want to try different values to determine the best sensitivity for the photocells you use. You want the highest sensitivity while maintaining the widest possible swing between zero and five volts.

(Note: The pin IDs reflect the nomenclature used on the 28X2 chip itself, and are printed on the AXE401 board. For your reference, the *B* is a *port* on the chip containing many pins, and the *3* or *4* is a specific pin on that port. The 28X2 has three ports, labeled A, B, and C. There are different numbers of pins available on each of the ports.)

Recall from last month that the L298 H-bridge module requires two inputs per motor. The direction of the motor is determined by the instantaneous value of these two inputs, according to **Table 1**.

By setting the pins LOW (zero volts) or HIGH (five volts) in programming, you can control the operation and direction of either motor. You'll see exactly how this is done in the next section.

You may have noticed that the AXE401 contains its own power plug and five volt voltage regulator – it even has a second regulator for 3.3 volts. For this project, we won't be using these features, as the Beginner Bot instead draws its five volt power from the regulator on the L298 H-bridge that we've selected for the project. This power drives the *logic* portion of the L298 module, and also operates the AXE401.

This power connection arrangement simplifies the wiring, but know that when it comes time to reuse your AXE401 for some other project that it's capable of being separately powered using its own onboard voltage regulator.

TABLE 1

Input A	Input B	What Happens
Low	Low	Motor stops
Low	High	Motor turns one direction
High	Low	Motor turns the other direction
High	High	Motor stops

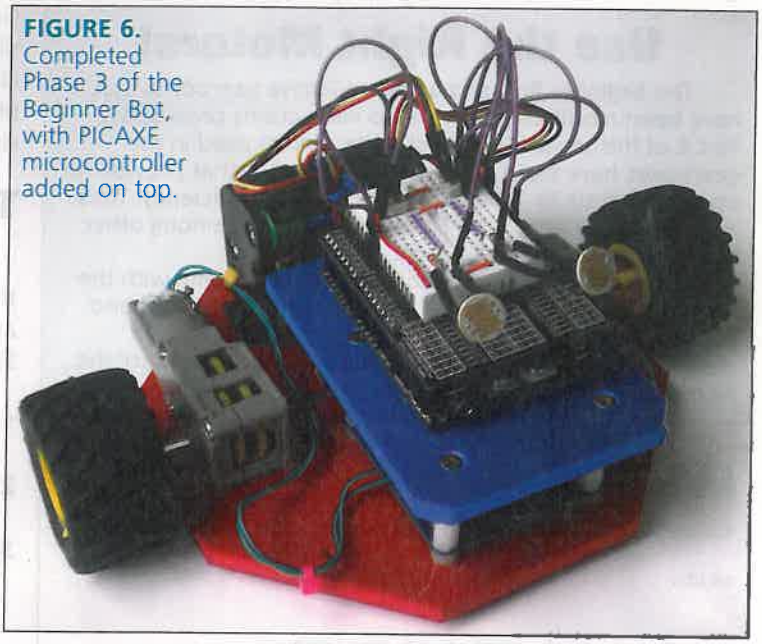


FIGURE 6. Completed Phase 3 of the Beginner Bot, with PICAXE microcontroller added on top.

Figure 6 shows the completed *Phase 3* of the Beginner Bot, with AXE401 board, prototype expansion shield, and populated mini solderless breadboard.

As with the *Phase 2* version of the Beginner Bot that demonstrated control using a hex inverter IC, the two CdS photocells are mechanically attached to the front of the mini solderless breadboard and poke out like snail's eyes.

Sources

Precut and predrilled Beginner Bot base, with all construction hardware:

Budget Robotics
www.budgetrobotics.com

PICAXE documentation, software, sales (UK and EU):

PICAXE Home
www.rev-ed.co.uk/picaxe

Tech Supplies
www.techsupplies.co.uk

AXE401 development board, serial and USB download cables for PICAXE:

HVW Tech*
www.hvwtech.com

PH Anderson
www.phanderson.com

Robotshop*
www.robotshop.com

SparkFun*
www.sparkfun.com

Mini solderless breadboards, jumper wires, header pins, etc.:

Adafruit Industries
www.adafruit.com

All Electronics
www.allelectronics.com

Electronix Express
www.elexp.com

HVW Tech*
www.hvwtech.com

Jameco Electronics
www.jameco.com

Mouser Electronics
www.mouser.com

Pololu
www.pololu.com

RobotShop
www.robotshop.com

Schmartboard
www.schmartboard.com

SparkFun
www.sparkfun.com

Solarbotics
www.solarbotics.com

*At the time of this writing, these resources were not yet carrying the AXE401, but they do stock other PICAXE parts. Check for current availability.

Use the Right Motors!

The Beginner Bot uses a pair of Tamiya gearboxes that have been modified according to instructions provided in Part 2 of this series. Specifically, the motors used in the gearboxes have been replaced with versions that provide for operation at six to 12 volts, and with higher efficiency. These motors are available from Pololu (item #1117), among other sources. Cost is under $2 each.

Be sure to not use the stock motors that come with the Tamiya gearboxes. These are rated for only three volts and can consume copious amounts of current. This current exceeds the rating of the L298 H-bridge used to control the motors.

LISTING 1

```
#picaxe 28x2            ; Specify 28X2 PICAXE chip

main:                   ; Main program loop

  ; Go through series of motions
  gosub robot_forward
  pause 2000

  gosub robot_backward
  pause 2000

  gosub robot_right
  pause 2000

  gosub robot_left
  pause 2000

  gosub robot_stop
  pause 2000

  goto main             ; Repeat main program

robot_forward:          ; Motion control routines
  high B.6              ; Right motor controlled by
  low B.7               ;   pins B.6 and B.7
  high A.0              ; Left motor controlled by
  low A.1               ;   pins A.0 and A.1
  return

robot_backward:
  low B.6
  high B.7
  low A.0
  high A.1
  return

robot_right:
      high B.6
      low B.7
      low A.0
      high A.1
      return

robot_left:
      low B.6
      high B.7
      high A.0
      low A.1
      return

robot_stop:
      low B.6
      low B.7
      low A.0
      low A.1
      return
```

Gently bend the leads of the cells so that they point slightly upward and outward, like that in **Figure 7**. I've added some unshrunk heat shrink tubing over the photocell leads to provide both insulation from short circuits and an extra bit of mechanical support.

Testing Motor Operation

Listing 1 shows a demonstration program for testing the basic operation of the AXE401 board, the H-bridge, and the motors. Type or download this program from the *SERVO* website, then:

1. Place a small block under the Beginner Bot base to lift the wheels off your worktable.
2. Connect the battery to apply power to the H-bridge and AXE401 board.
3. Connect the programming cable between your PC and the AXE401, and be sure its communication port is selected in the PICAXE programming editor (choose *View>Options>Serial Port*).
4. Download the program to the AXE401. You'll be prompted if there are connection errors.

The downloaded program starts automatically. Assuming the motors have been connected properly, both motors should turn the same direction forward and back. The motors will stop after one cycle. You can press the Reset button or momentarily break power from the batteries to rerun the motor demonstration.

If one or both motors turn in the wrong direction, remove power and flip the terminal wiring from the affected motor on the H-bridge.

Let There Be (Flash) Light

In the previous installment, you learned how to control the Beginner Bot using a flashlight, shining the light into the photocell eyes. The simple circuit depicted in this article reacted to the bright light, steering the robot toward it.

Listing 2 extends the concept; this time, in a purely programmatic way. The program tells the PICAXE microcontroller to read the value from both photocells. A series of *If* tests determine if there's enough light to follow, and if so, in what direction the robot should travel. This is a good example of conditional logic.

The program first sets a threshold value to determine the boundary between dark and light. I've set this value to 180 — out of a range of 0-255 — as a starting point. Try higher or lower values to see what works best with your particular CdS cells.

When both cells receive light over the threshold, the robot drives forward. When only one cell receives light over the threshold, the robot turns in the direction of the light. If neither cell receives light over the threshold, the robot stops.

Let's test all this. Download the program in **Listing 2**, and when the download is complete remove the

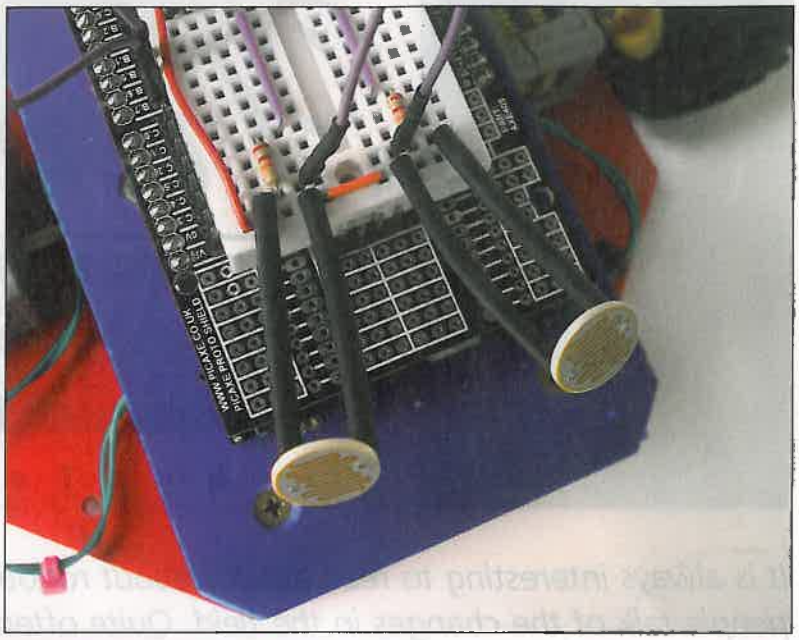


FIGURE 7. Spread out the sensor surface of the photocells so that you can direct the beam of a flashlight into either both or just one at a time.

programming cable. Move to a darkened room, apply power to the robot, and place it on the ground – tile or wood floor is better than thick carpeting due to the limited clearance under the Beginner Bot.

To start, aim a bright flashlight (preferably one with a strong narrow beam) *away* from the Beginner Bot. Both motors should remain off. Now, shine the flashlight directly into the photocells. The robot should move toward you.

Get close to the robot and aim the flashlight into just one photocell; you may need to gently spread the cells apart if they're too close together. The robot should turn toward the photocell with the light shining into it.

If your robot moves when there's no light falling on the CdS cells, try changing the threshold value to something higher. Conversely, if the light from the flashlight seems to make no difference, enter a lower threshold and try a darker room. Keep in mind that any significantly bright light source will "blind" the robot to the flashlight.

If you're still not getting results, try the simple test program in **Listing 3**. When run, it displays the numeric values obtained from both CdS photocells in the PICAXE editor's Debug window. Keep the download cable attached to the AXE401 during this test, and note the values in the *b0* and *b1* boxes (upper left corner). Aim the CdS cells at a bright light source and the value should climb high – up to the maximum of 255. Block all light and the value should fall close to zero.

If nothing like this happens, double-check your wiring. Try different CdS cells, and re-run the test.

LISTING 2

```
#picaxe 28x2
symbol threshold = 180          ; Define threshold between
                                ;  dark and light

main:                           ; Main program loop

  readadc B.3, b0               ; Read analog pins B.3 & B.4
  readadc B.4, b1

  ; Determine direction of robot based on sensor input
  if b0 > threshold and b1 > threshold then gosub robot_forward
  if b0 > threshold and b1 < threshold then gosub robot_left
  if b0 < threshold and b1 > threshold then gosub robot_right
  if b0 < threshold and b1 < threshold then gosub robot_stop

goto main                       ; Repeat program loop

(repeat motor control routines from Listing 1 here)
```

LISTING 3

```
#picaxe 28x2
main:
  readadc B.3, b0
  readadc B.4, b1
  debug b0
  goto main
```

Next Up: Arduino-based Beginner Bot

The Beginner Bot isn't tied to just one kind of microcontroller. It's a universal design that lets you use your choice of microcontroller. I wanted to start with the PICAXE – and especially the AXE401 development board – because it's inexpensive and easy to use. You're free to try other microcontrollers, like the Arduino, BASIC Stamp, or Parallax Propeller.

In the next installment, we'll do just that. You'll see how to connect the Beginner Bot to the popular Arduino Uno development board, and how to adapt the light seeking abilities of your little robot to Arduino programming code. You'll also discover how to add touch sensors and other features. SV