



Paperduino 2.0 with Circuit Scribe - Paper Arduino

by [Electroninks](#) on December 11, 2013

Table of Contents

Paperduino 2.0 with Circuit Scribe - Paper Arduino	1
Intro: Paperduino 2.0 with Circuit Scribe - Paper Arduino	2
Step 1: Editing the layout in EAGLE	2
Step 2: Exporting the board layout	3
Step 3: Importing layout to Silhouette Studio	5
Step 4: Printing the Arduino	6
Step 5: Placing components	7
Step 6: Burning the Arduino Bootloader & Programming	7
Step 7: What would you make?	8
Related Instructables	8
Advertisements	8
Comments	8



Author:Electroninks Visit our Kickstarter!

Circuit Scribe is a rollerball pen that writes with conductive silver ink. It makes creating circuits as easy as doodling! Visit our kickstarter to get your own pen!

Intro: Paperduino 2.0 with Circuit Scribe - Paper Arduino

What if making an Arduino, or wiring up an Arduino was as easy as printing one out? In this tutorial we printed our own Arduino Pro Mini board using a pen plotter and the Electroninks Circuit Scribe (a rollerball pen with highly conductive ink). Within 15 minutes we printed the board, placed components down with glue or tape, and uploaded a sketch.

Materials needed:

[Circuit Scribe conductive ink pen](#)

Silhouette SD plotter

8.5x11" glossy photo paper

Electroninks Arduino connection kit

FTDI cable

USBtinyISP bootloader or a pre-bootloaded Arduino chip

Superglue (Loctite gel) or [Z-tape](#)

Software:

[EAGLE PCB](#) (if you want to modify the board)

[Arduino IDE](#)

[Silhouette Studio](#)

Downloads:

[Paperduino 2.0 Schematics and Layout \(EAGLE\)](#)

Components list:

[View BOM \(ordering list\) on Google Spreadsheet](#)

2x 10 uF capacitors (1206 package)

3x 0.1 uF capacitor (1206)

2x 10 KOhm resistors (1206)

330 Ohm resistor (1206)

ATmega328P-AU

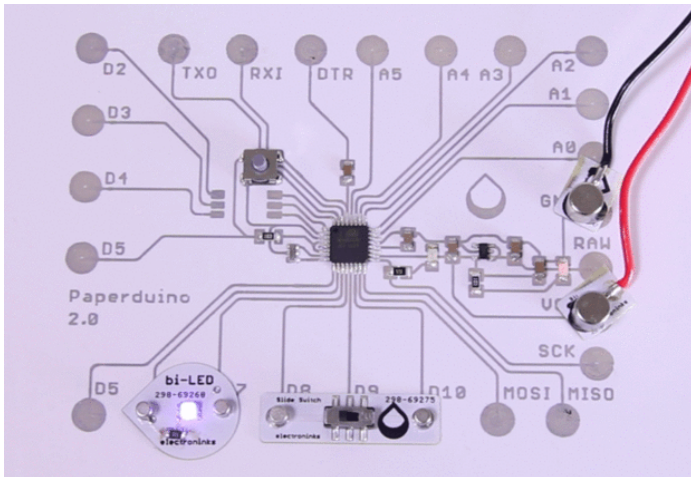
Red LED (1206)

Green LED (1206)

5V voltage regulator (5 pin)

8 MHz resonator

Tactile switch



Step 1: Editing the layout in EAGLE

Our Paperduino design is based on the open source design for the Arduino Pro Mini. We used EAGLE PCB to design the board.

Download a schematic

The Arduino Pro Mini board schematic can be found on the Arduino website.

Our final Paperduino 2.0 EAGLE files can be found [here](#).

Adapting the board for paper

In order to make the schematic paper-friendly, we replaced the blocks of pin headers with our own custom round pad. Through-hole components such as the pin headers are difficult to use on flexible surfaces like paper. We added the new pad design to our Electroninks component library. Since the pen plotter only prints lines, but not fill patterns, the pads were composed of overlapping concentric circles. The tool path for plotting will be shown in Step 3.

We also replaced the package for many of the small components with the 1206 package (meaning 1.2mm long x 0.6mm wide) so they would be easier to mount later on.

Designing the board layout

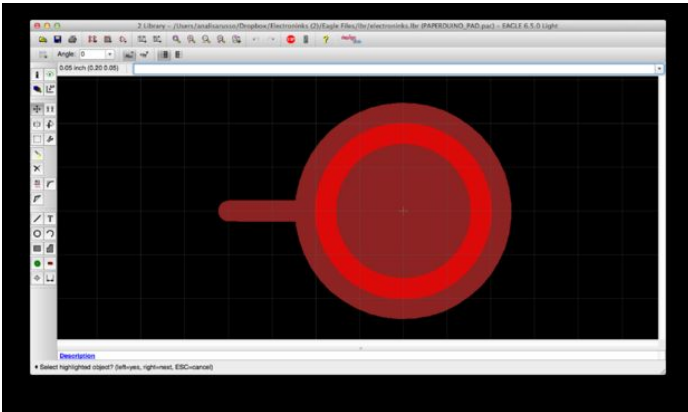
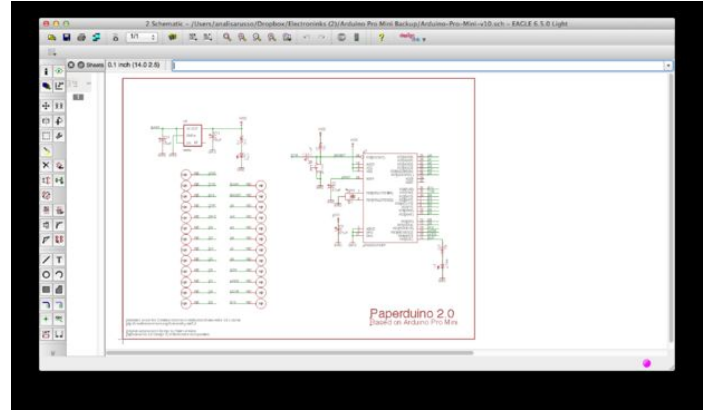
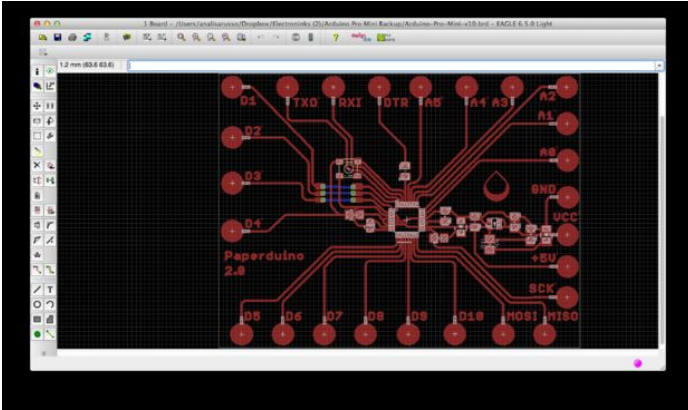
In the board layout, we set the top layer line width to 0.6 mm to match the width of the pen trace.

<http://www.instructables.com/id/Paperduino-20-with-Circuit-Scribe/>

We designed the board layout so the Paperduino could be printed on one layer. In many cases, the surface mount components act as jumpers. It is important to keep components close together to minimize line resistance between them.

In EAGLE PCB, the component pads will not be filled in by the pen plotter when exported as a DXF -- only the outlines will be drawn. To solve this issue, overlay the pads with meandering line features so they will be filled by a raster pattern.

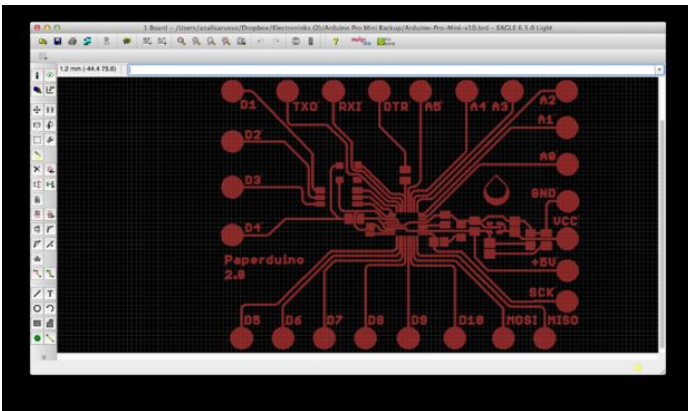
Finally, if any lines require a paper jumper (like our digital pins 1, 2, and 3), they can be created in the bottom layer (blue). After the first layer is printed, paper stickers can be placed over the lines and the bottom DXF layer can be used to print over the stickers. We will publish a jumper sticker tutorial in a future Instructable.

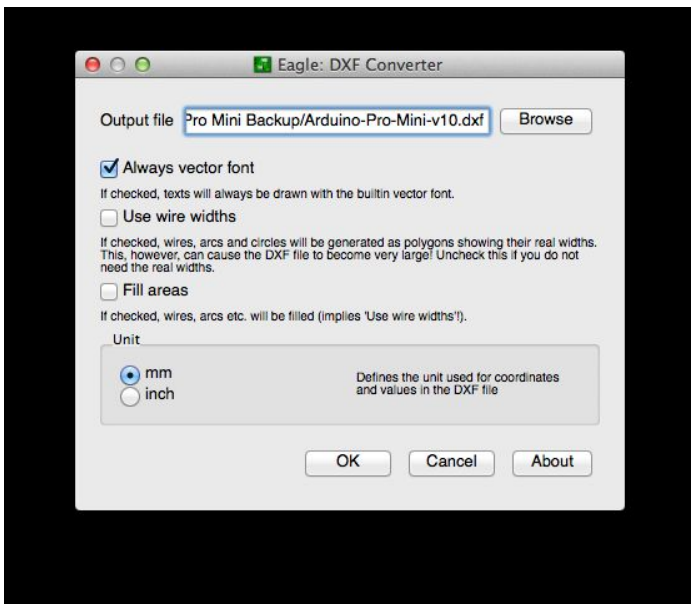
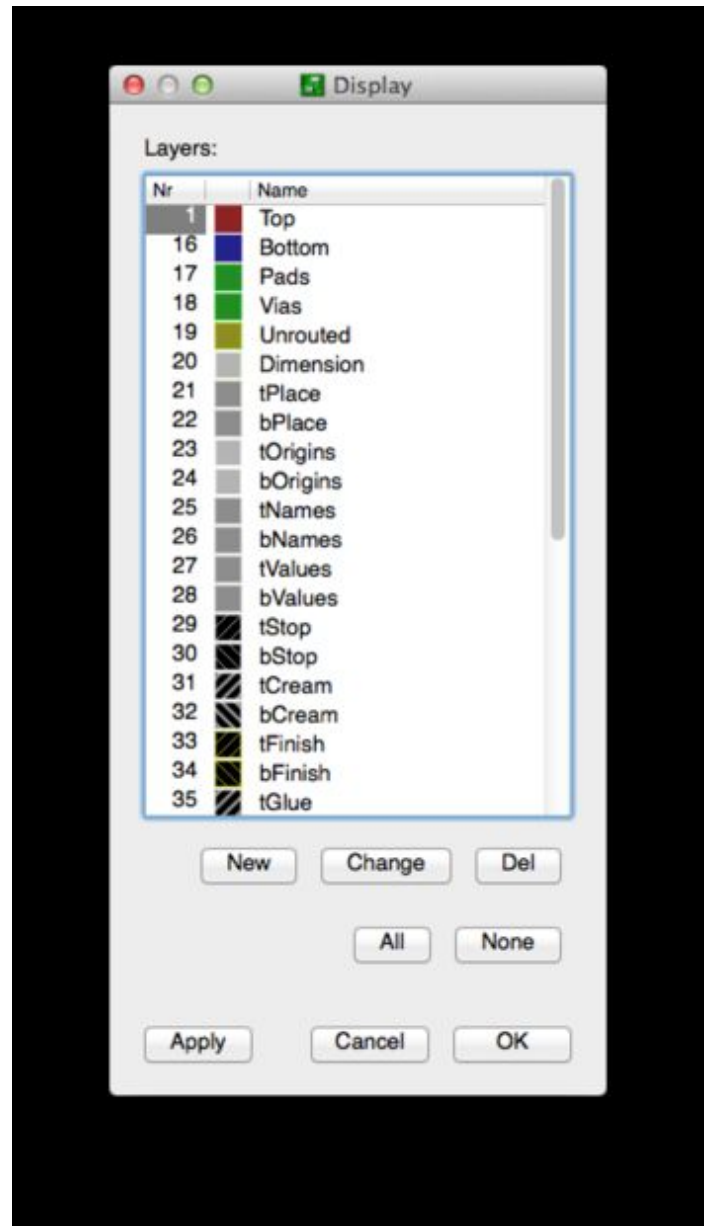
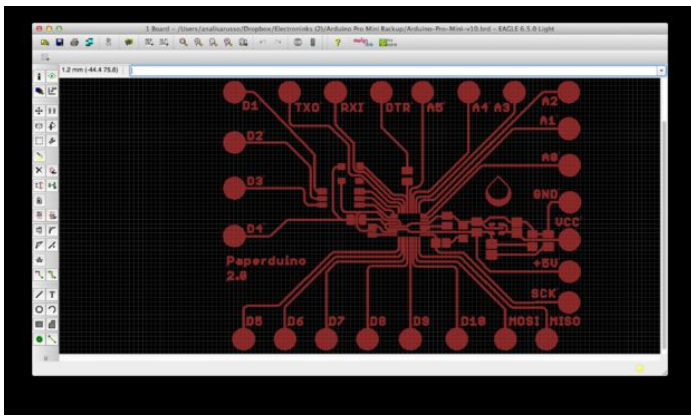


Step 2: Exporting the board layout

Before exporting the board as a DXF file, only have the top layer (red) selected in EAGLE. If you have more than the top layer selected you may end up drawing more on the board than you desire. If you'd like to create text, vector fonts can be printed out on the pen plotter. If you use the tnames layer to separate traces and text, then select both the top layer and tnames before exporting the DXF file.

When prompted after selecting export as DXF (file>export | DXF), deselect the wire width and fill area options and then save the file to the appropriate directory.





Step 3: Importing layout to Silhouette Studio

Import the file

Import the DXF file into Silhouette Studio by dragging the file onto the blank page. You can see that all of the pads are composed of either raster line patterns or concentric circles.

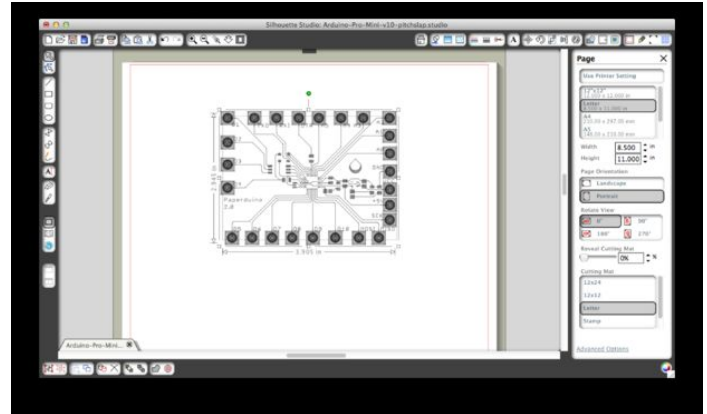
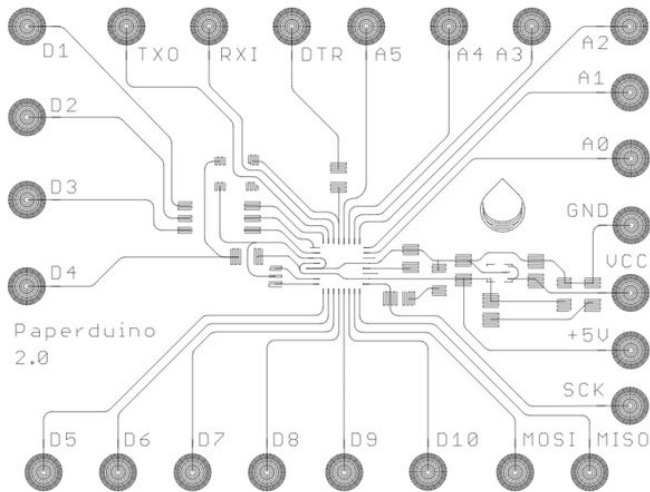
Adjust the layout

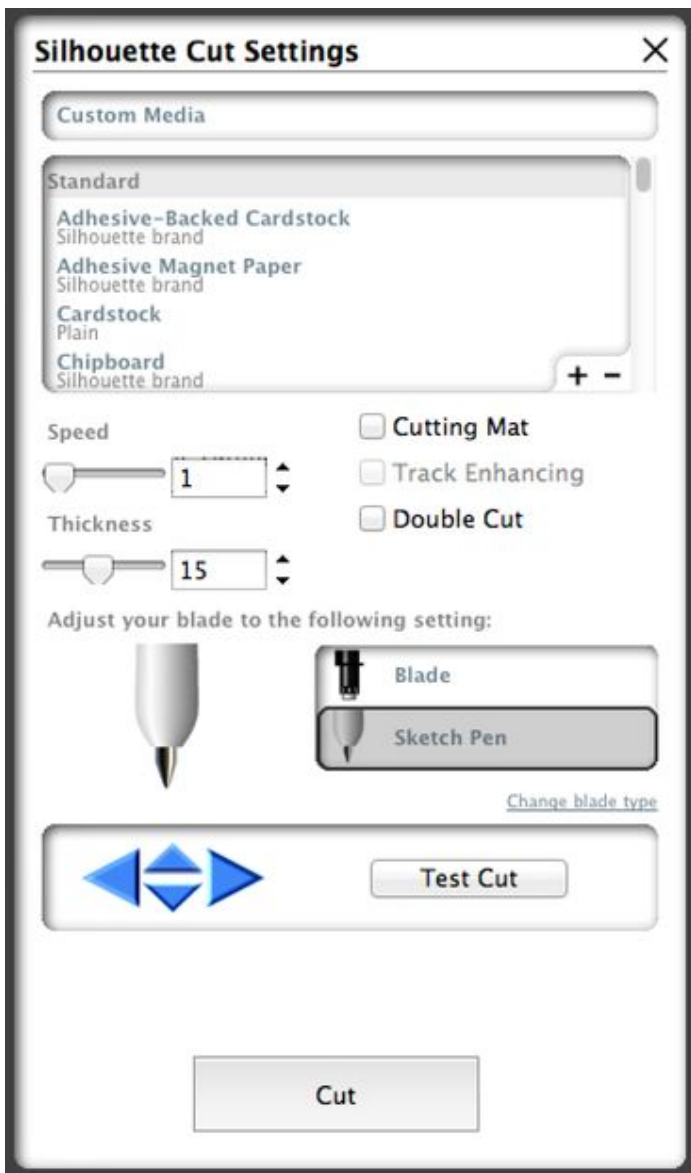
After importing the DXF file to Silhouette Studio, it may need to be rescaled. Go back to the EAGLE board layout and measure the pattern from top to bottom. Then "select all" in Silhouette Studio and rescale the pattern to match.

Next, position the Arduino pattern on the page where desired.

Select print settings

Open up the plotting window. We aren't using a cutting mat, so that option can be deselected. A printing speed of 1 (the slowest speed) works well. Glossy photo paper can be used with a thickness level 15. Finally, make sure the "sketch pen" option is selected.





Step 4: Printing the Arduino

Getting the plotter ready

To prepare the Circuit Scribe for plotting, wrap the pen in masking tape, then wedge it into the plotter head and lock it.

Load the plotter with glossy photo paper, which will enhance the conductivity of the ink, compared to standard paper.

Printing

Now simply hit "cut" in Silhouette Studio!

The tool path that Silhouette Studio creates tends to double over most of the lines. This is to our advantage -- drawing over the traces twice reduces their electrical resistance.





Step 5: Placing components

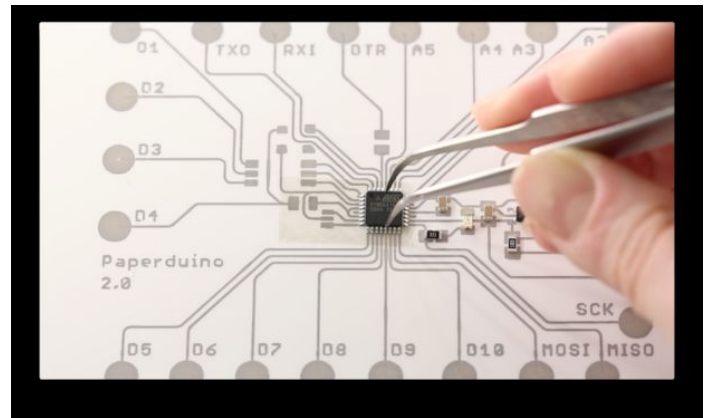
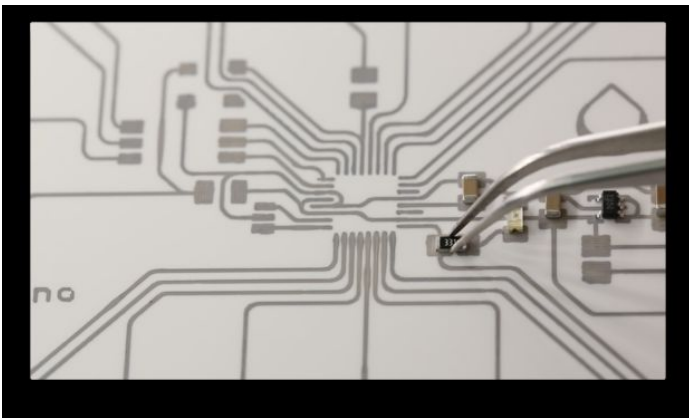
Method 1:

Place a drop of superglue between the component pads. It's ok if glue gets on one of the traces that runs between the pads. Then carefully place a component with tweezers and press in the center.

This method is permanent and the connection between components and pads should be good. However, we sometimes reinforce the connections using conductive epoxy applied with a syringe or an electric fluid dispenser.

Method 2:

Place a strip of z-axis tape over several of the component pads. Then press the surface mount components in place. The components could be peeled up later and reused. This is by far the quickest and easiest method to attach components.



Step 6: Burning the Arduino Bootloader & Programming

Burning the Arduino Bootloader

An Arduino chip isn't ready to program straight from Atmel's factory (the company that makes the microcontrollers)! In order to get the ATMEGA328 to run any Arduino sketches, the Arduino firmware has to be loaded in a process known as bootloading (akin to loading an operating system such as Windows 7).

There are a few programmers you can buy - we use a USBtinyISP from Adafruit. You can also buy Atmel's official programmer (an AVR ISP) from digikey . If you never want to burn the bootloader yourself you can buy the surface mount ATMEGA328P already bootloaded .

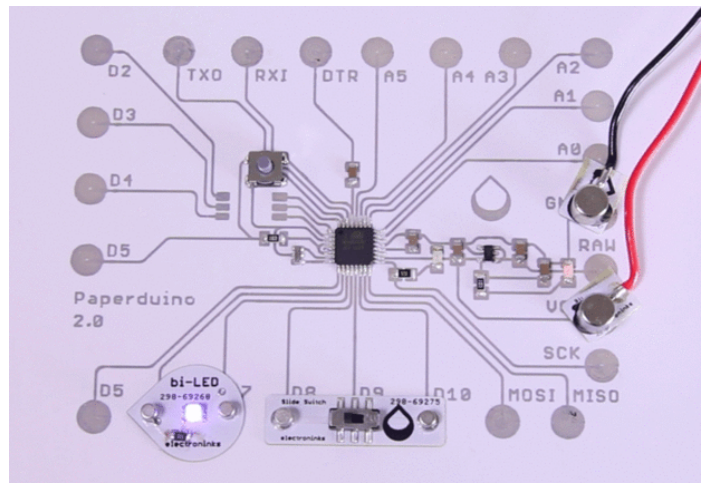
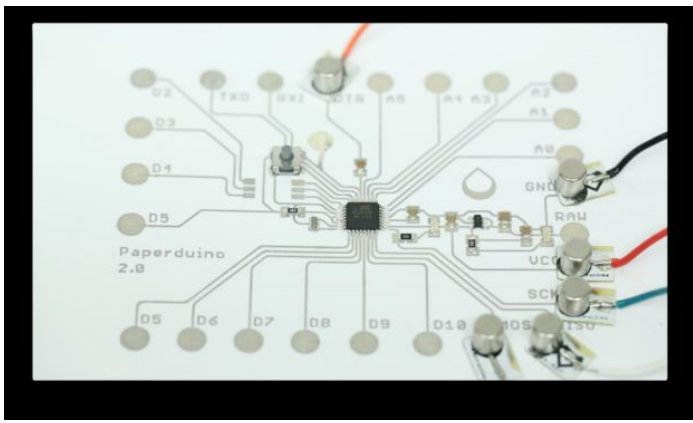
Burning the bootloader is simple. First connect 5 pads on the Paperduino (GND, VCC, MOSI, MISO, SCK and DTR) to the programmer. Once connected, select the correct board type in Arduino IDE (an Arduino Pro Mini 328 at 5V), and then click "burn bootloader." If it works, the bootloader will be uploaded. If needed, additional help can be found here .

Uploading a sketch to the Paperduino

Our Arduino does not have an FTDI chip built on the board. The FTDI essentially bridges the gap between USB and the serial communication of the ATMEGA328 chip. What we need is the simple FTDI chip, which can be found packaged in a USB cable, and purchased from Sparkfun here . In future versions of the Paperduino we'll actually integrate an FTDI chip and make a paper USB connector to plug directly into a USB port on your computer.

Once you have the FTDI cable in hand, the connection is simple. Wire up the GND, VCC, TXD, RXD and DTR to the appropriate spots on the cable (we used our magnet-tipped cables from the Arduino Connection Kit). Once wired, all you have to do is upload a sketch. Uploading the blink example will blink the green surface mount LED.

We uploaded a simple sketch that changes the direction of our magnetic bi-direction LED component from blue to green. Using two digital outputs, the direction is changed easily by making output 6 low, and 7 high. And to reverse, make output 6 high and 7 low. Voila!



Step 7: What would you make?

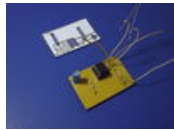
We can't wait to see what you come up with!

Share your own project ideas with us in the comments section!

Related Instructables



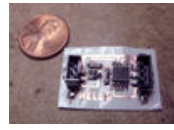
XY-Plotter by bdeakyne



QCB: QUICK CIRCUIT BOARD by Captain Vyom



Cheap Arduino Controlled 3-Axis Pen Plotter (Photos) by davidhend



How to make circuits with a Roland CAMM sign cutter by nadya



Switch mode LED torch by neelandan



Polargraph Drawing Machine by Euphy

Comments

3 comments

[Add Comment](#)



ZeroTruths says:

I'm wondering if there's any way to do double sided circuits.

Dec 18, 2013, 12:38 PM [REPLY](#)



SayntCigol says:

I would love to see this pen in the silhouette form factor (so you don't have to tape it up)

Dec 18, 2013, 11:11 AM [REPLY](#)



Electroninks says:

How about a 3D printed holder? It's just a cylinder that is hollow. We can post a part for that.. no problem. :)

Dec 18, 2013, 12:32 PM [REPLY](#)