



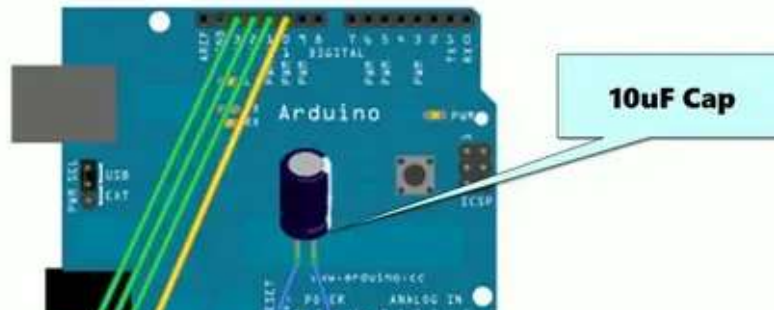
If you are looking for a small and low powered Arduino board the ATtiny is a really good option, its surprisingly featureful for it's size. It has 5 GPIO pins, 3 of which are analog pins and 2 which have PWM output. It is also really flexible to the voltage that it runs off (2.7V to 5.5V) so it's perfect for running off batteries. Did I also mention it only costs about \$1!? The trouble is with the ATtiny is that you can't just plug a USB cable in to program it, but it actually isn't hard to build a programmer for it and that is what we are going to go through in this tutorial.

There are already lots of guides for building a shield, but there is a step missing when using newer versions of the Arduino IDE in the software setup in all the ones I checked that I will go through here as well. Check out the above video where I go through all the information that is in this tutorial.

Let's get to it!

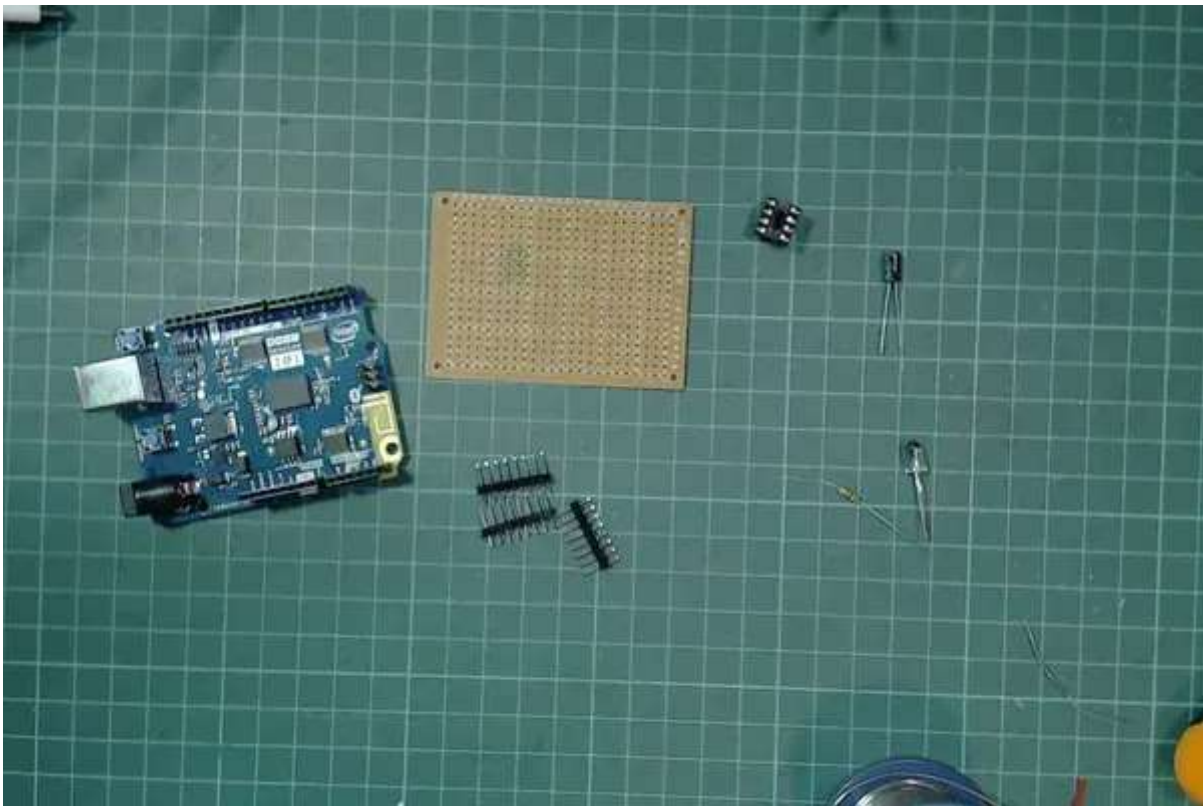
Step 1: Breadboard Programmer Circuit

This circuit will work with the programming steps that are later in the video



I think it's worth noting that you can use a breadboard circuit to program the ATtiny too if you prefer to not have to build a shield. I wanted the shield so I would have something more permanent to use in the future. If you opt for the breadboard programmer, the software steps later on are the exact same as for the shield. Skip to step 5 for this.

Step 2: What You'll Need



You will need the following parts to build the programmer:

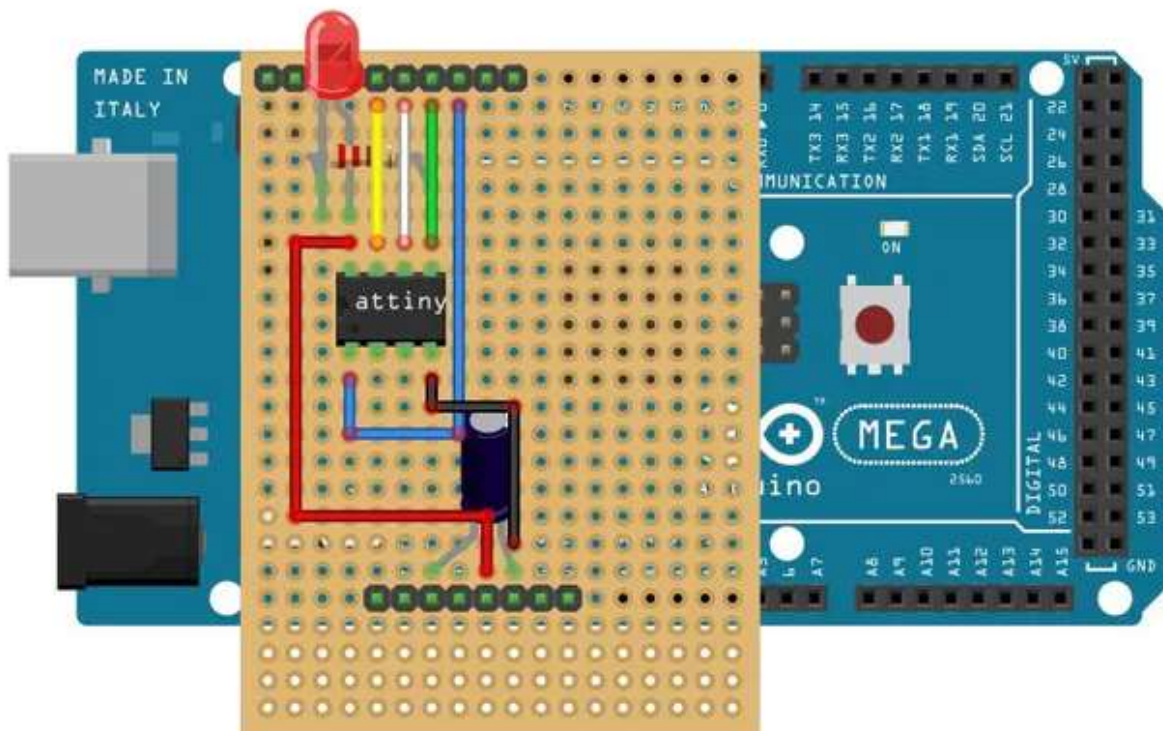
- ATtiny85* (<http://s.click.aliexpress.com/e/6uRbQ7u>) - Probably going to need one of these :)
- Protoboard (10 pieces) (<http://s.click.aliexpress.com/e/YZvrzFE>)* Male Header Pins (<http://s.click.aliexpress.com/e/ZNzzj6l>)*
- 120 Piece Capacitor Set (has a 10uF that we need) (<http://s.click.aliexpress.com/e/jUfAUBY>)*
- IC socket (20 pack) (<http://s.click.aliexpress.com/e/Rj6aqvV>)*
- Basic starter kit (has LED and 1K resistor that we need) (<http://s.click.aliexpress.com/e/Imm6euB>)*

Mega board I used (<http://s.click.aliexpress.com/e/fmQVbEy>)* - any Mega or Uno will work though.

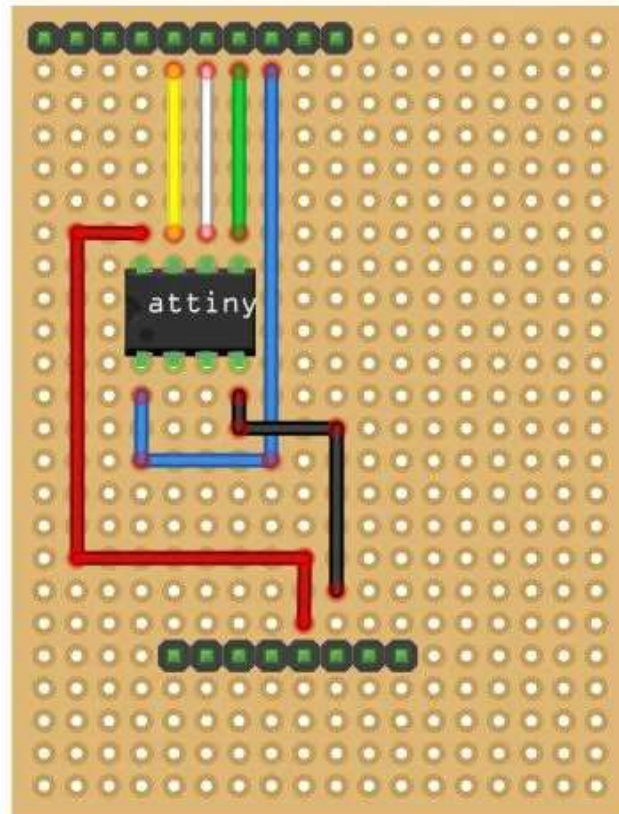
You will also need a soldering iron and some wires,

*= Affiliate Links

Step 3: Layout of the Shield



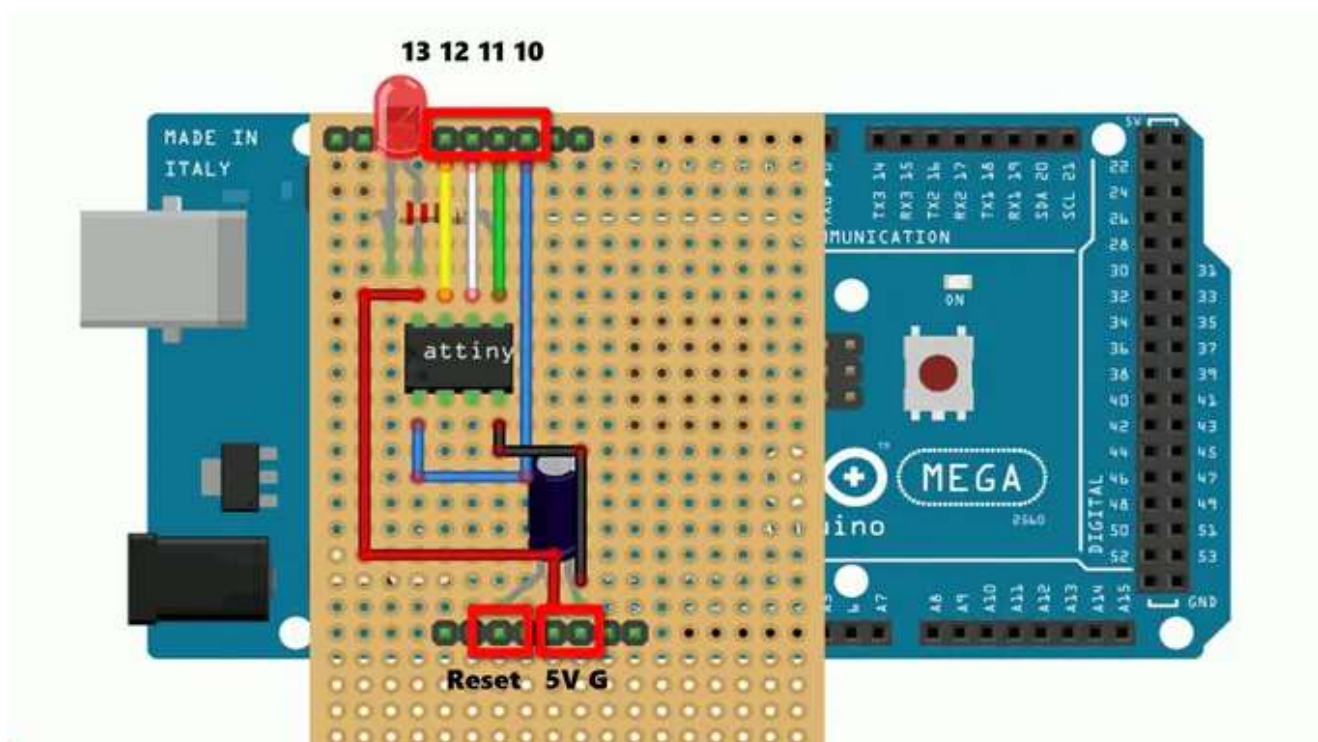
In the above images you can see the layout of the shield we are going to make. I find the image with components and wire is a little too crowded so I made up the circuit using just the wires and just the components to make it easier to read

[< PREVIOUS](#)

• •

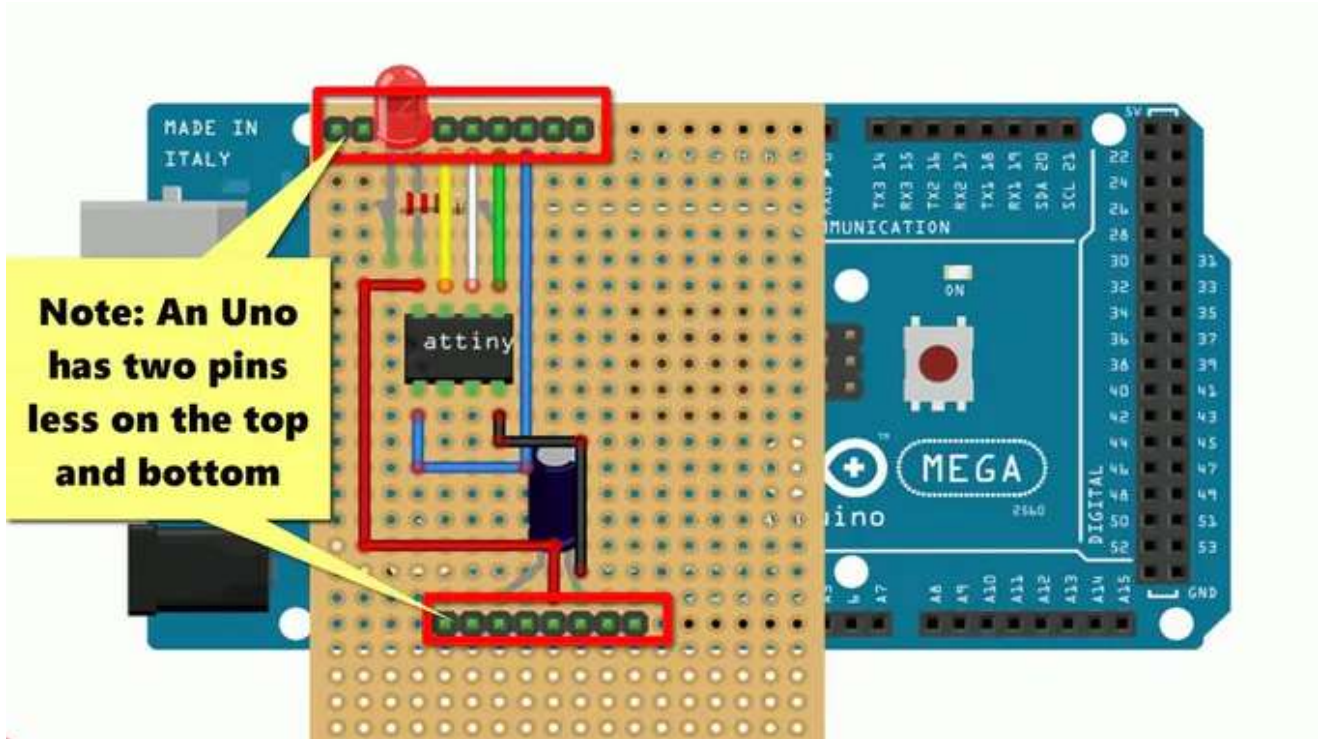
[NEXT >](#)

You don't need to use as many pins as I did, I marked in the image below the pins that are actually needed, I just thought it would be easier to plug in the shield in the correct place if it used all the pins at the top and bottom.



Step 4: Building the Shield

It's a pretty straight forward circuit to build, the most complicated part is probably just getting the pins right.

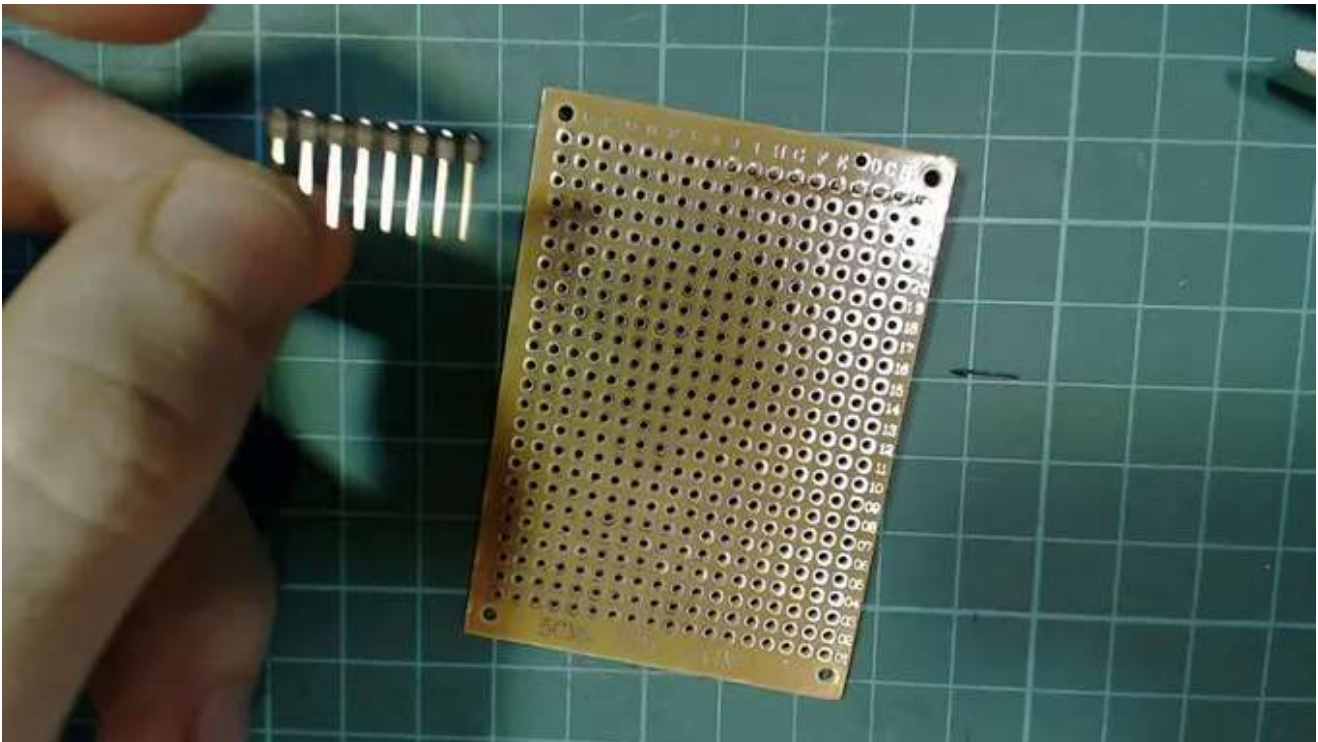


The way I did the pins was:

- Cut the male header pins so they would fit in the top and bottom rows of your Mega/Uno.
- Insert them into the Arduino.
- Place the protoboard on top and mark them using a sharpie.



- Remove the headers from the arduino.
- Push the plastic of the headers to one end of the pins (I used the protoboard for this, just pushed it towards the table). They should end up looking like the pins in the picture above

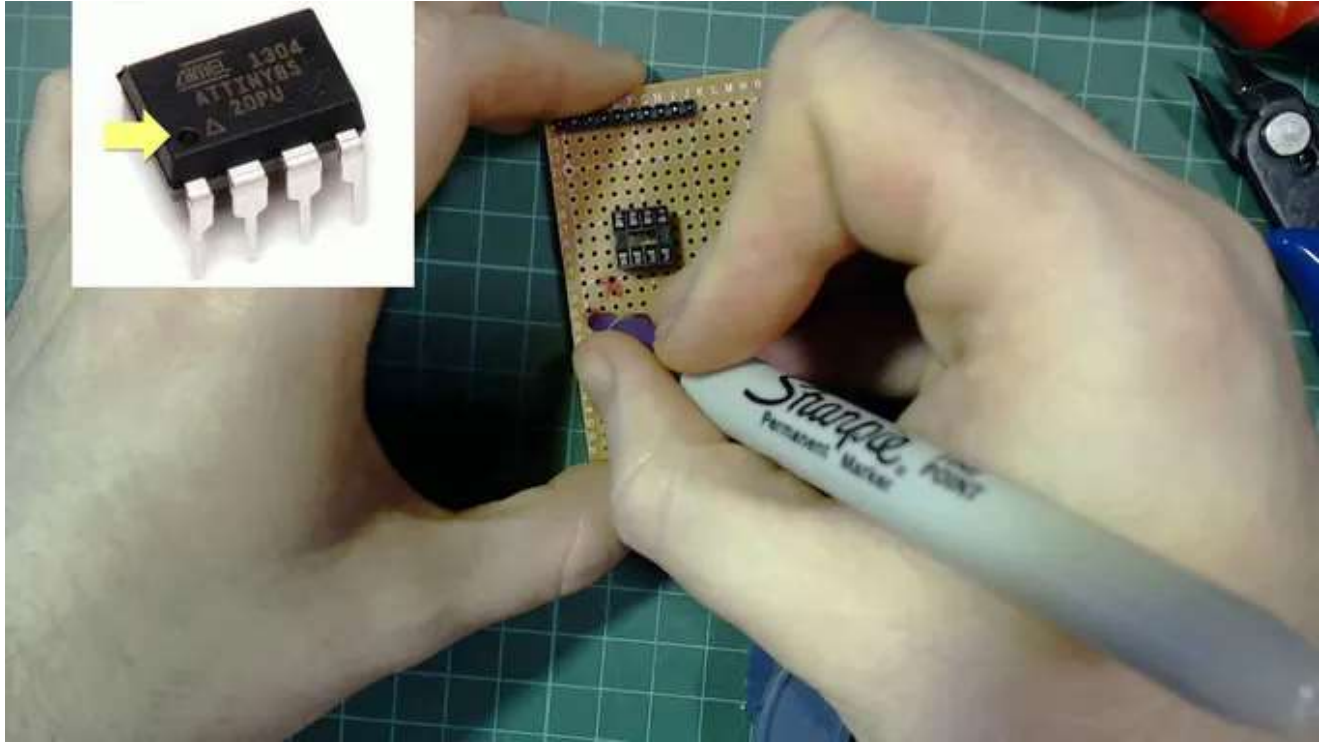


- Put the pins in through the top of the protoboard (plastic on top)
- Solder them in place, solder only enough to hold in place for the moment.

After that it's just a case of building the circuit, insert your components through and bend the pins towards where you need to connect them to and solder the connections together. I like to use blue tack to hold my components in place when I'm soldering.

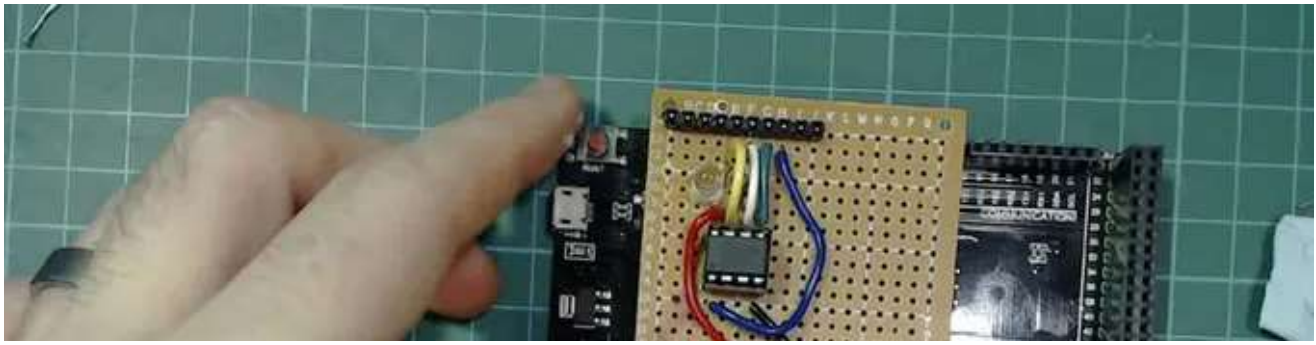
Make sure to double check the direction of the LED and Capacitor before soldering it up. For the LED the resistor should be connected to the short led of the LED. For the capacitor the leg with the silver marking above it should be connected to

ground. Finally it is probably a good idea to give yourself some marking or hint to remind you of the orientation of the ATtiny when plugging it in. If you check the last image above I show a picture of me marking the bottom left corner, this to match up with the dot on the ATtiny.

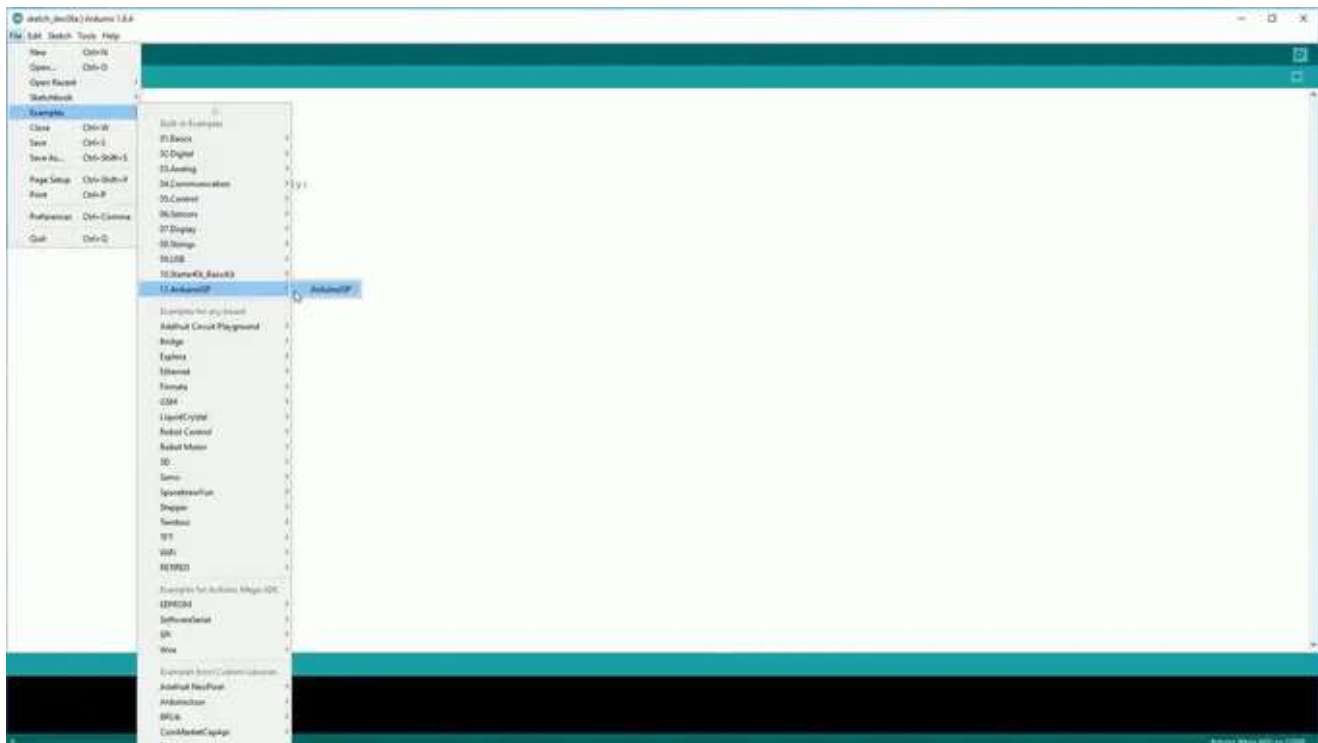


If you have a multi meter, I would suggest testing the pins for any bridges between, especially the bottom pins as they are the power pins.

Step 5: Setting Up Your Programmer



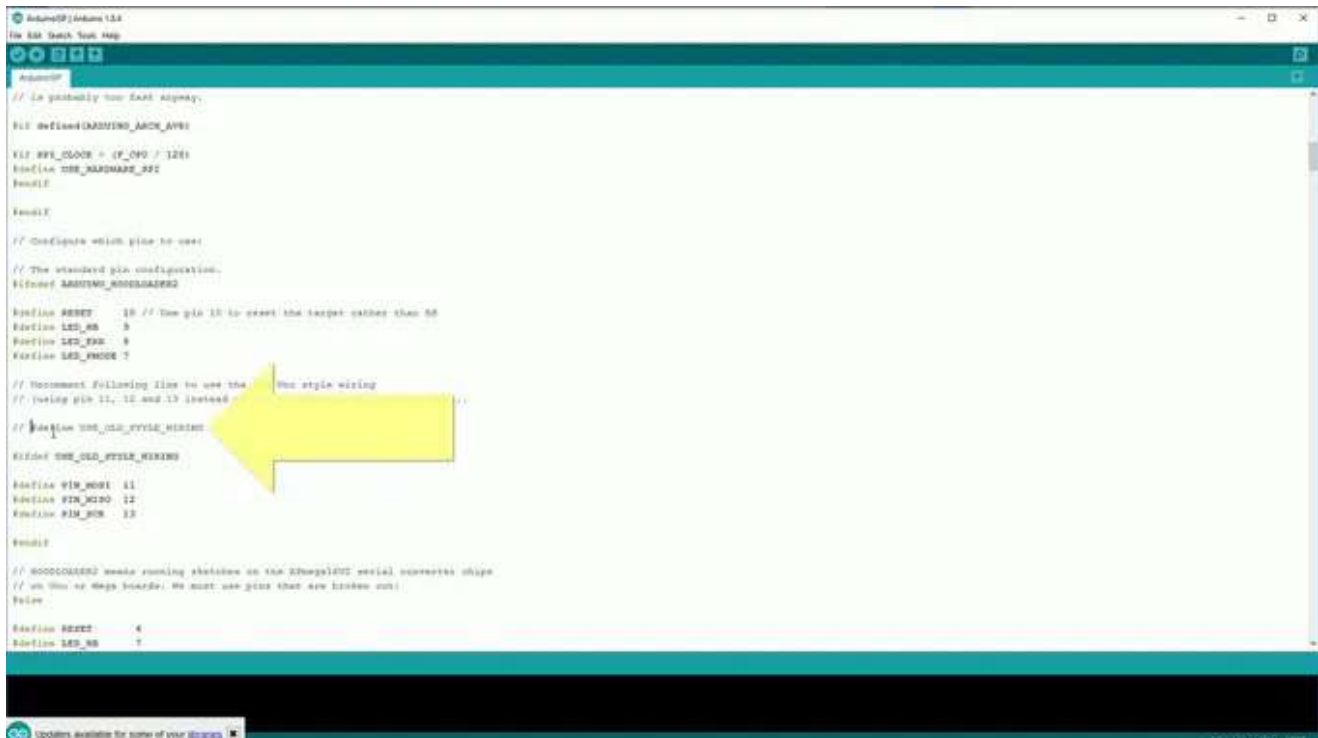
In order to use our Arduino as a programmer we first need to flash a sketch to it. First plug in your shield to your arduino, then plug in the USB cable into your arduino. Open the Arduino IDE, then click **File -> Examples -> 11.ArduinoISP -> ArduinoISP**



We need to make a change to this file, this is the part I found missing from all the other guides.

Scroll down on this file till you see a commented out line `// #define USE_OLD_STYLE_WIRING`

Remove the comment from this line (so it should now look like `#define USE_OLD_STYLE_WIRING`)



```

// In probably too fast anyway.
#define ARDUINO_ARCH_AVR

// If SPI_CLOCK is 0, it's 128
#define USE_HARDWARE_SPI
#undef SPI

// Configures which pins to use

// The standard pin configuration.
#define ARDUINO_WOODGASPER2

#define RESET 10 // Use pin 10 to reset the target rather than 50
#define LED_PB 5
#define LED_PPB 8
#define LED_PMODE 7

// Uncomment following line to use the "Old style wiring"
// (using pins 11, 12 and 13 instead)
// #define USE_OLD_STYLE_WIRING
// #undef USE_OLD_STYLE_WIRING

#define PIN_MOSI 11
#define PIN_MISO 12
#define PIN_GND 13

// WOODGASPER2 needs running sketches on the Ethernet serial connected chips
// (in the 4 Mega boards). We must use pins that are broken out!
#define
#define RESET 4
#define LED_PB 7

```

You can now upload this sketch to your arduino like you would any other sketch.

Step 6: Setting Up the Arduino IDE for the ATtiny

We need to install the ATtiny software through the board manager before we can program to the ATtiny.

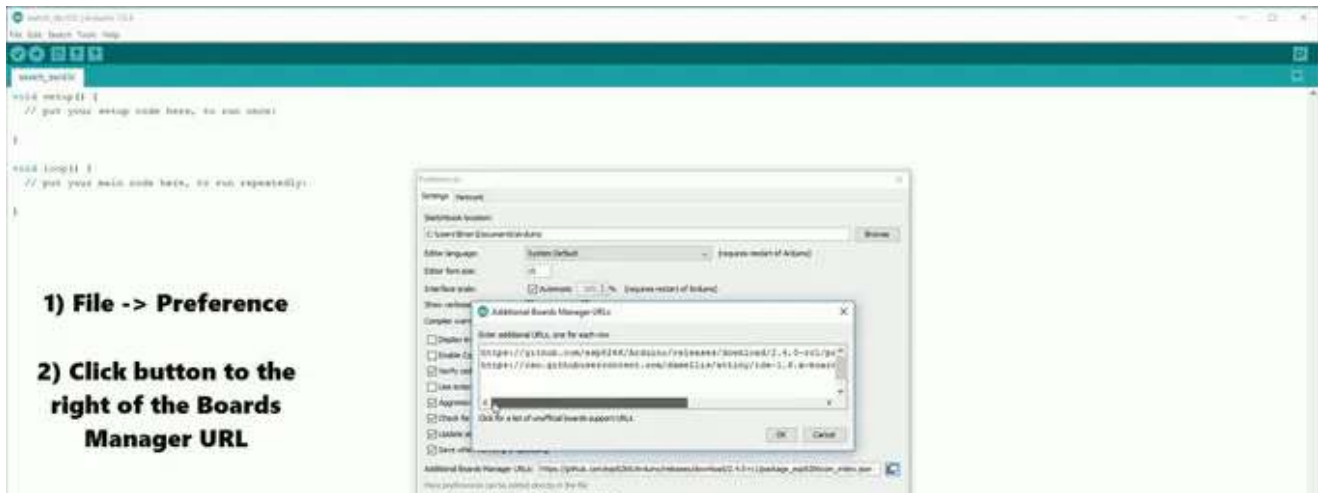
First thing we need to do is add a new line to our Additional Boards Manager URLs which can be found under **File -> Preferences**

The URL you need to add is:

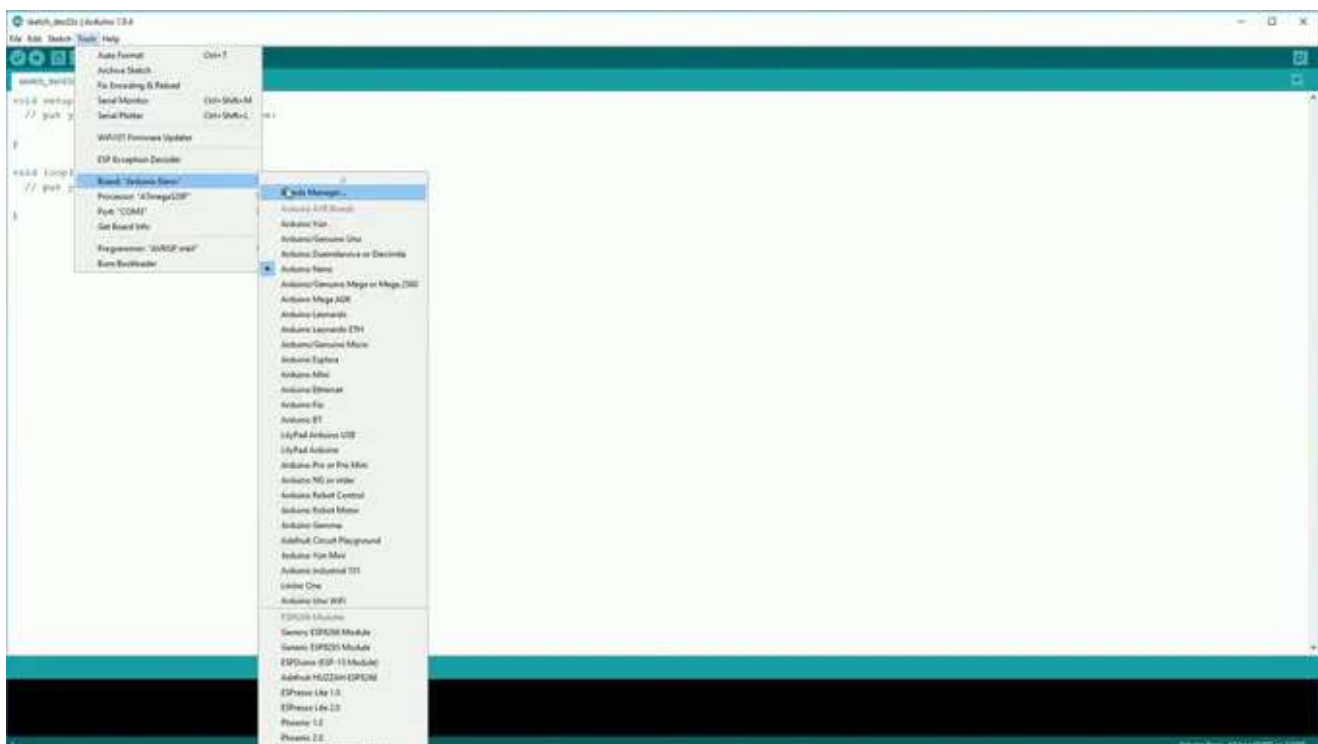
https://raw.githubusercontent.com/damellis/attiny/ide-1.6.x-boards-manager/package_damellis_attiny_index.json

(https://raw.githubusercontent.com/damellis/attiny/ide-1.6.x-boards-manager/package_damellis_attiny_index.json)

Click the button to the right of the Boards Manager URL box and enter the above on a new line.



You now want to open the Boards Manager, go to **Tools -> Board: "whatever is selected" -> Boards Manager**



Search for "attiny" and click install.

[illegible]

- Select **ATTiny25/45/85** from the **Boards** drop down.
- Select **ATTiny85** from the **Processor** drop down.
- Select **Internal 8 MHz** from the **Clock** drop down.

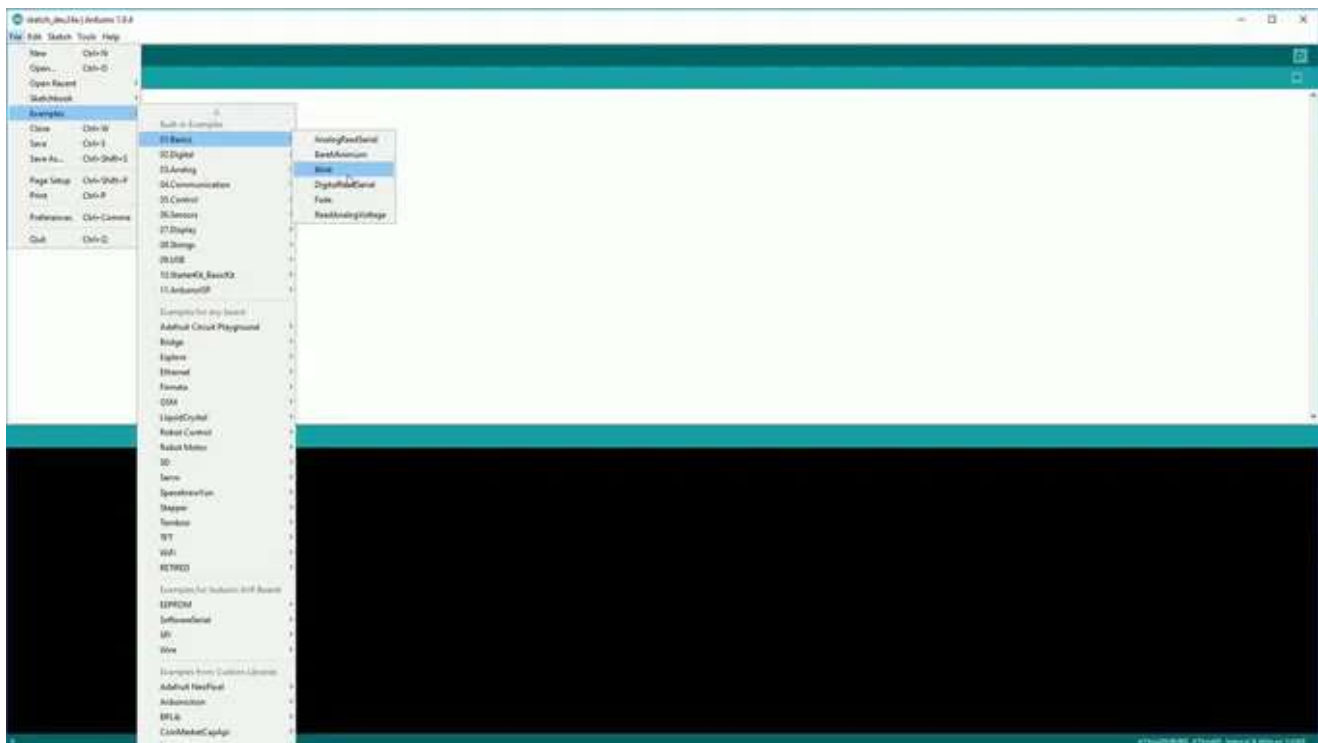
- **Port** should be the Com port to the Arduino you are using as the programmer.
- Select **Arduino as ISP** from the **Programmer** drop down.

We can now burn the bootloader, there is a good description of what the bootloader does and the advantages/disadvantages here

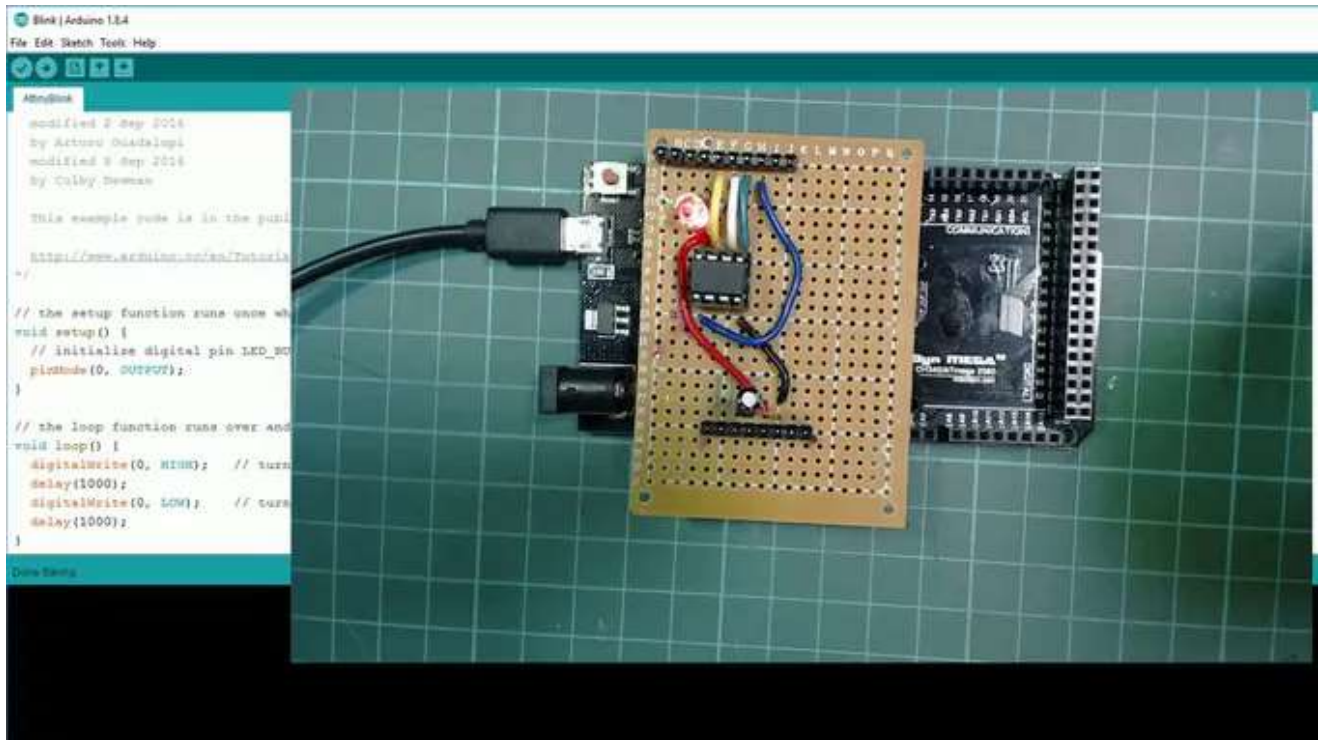
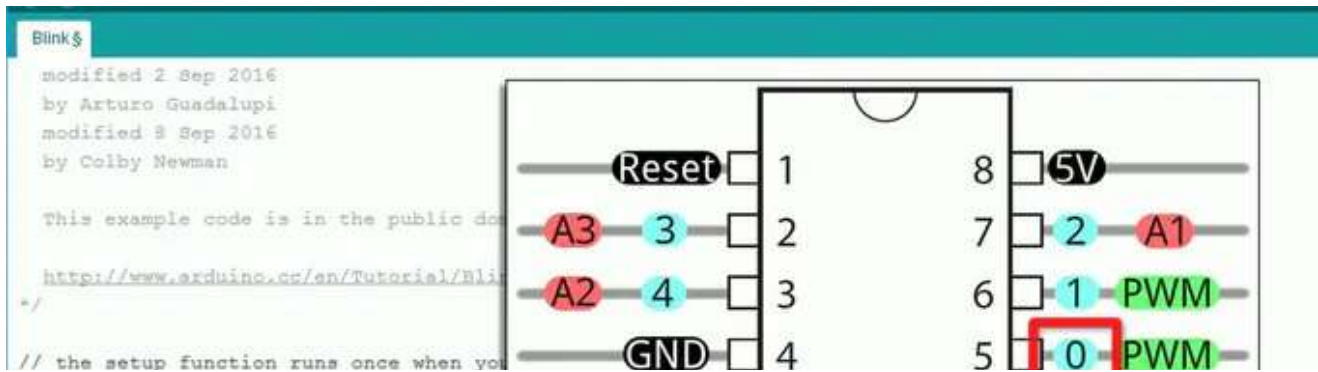
(<https://www.hackster.io/porrey/bootload-your-attiny85-625387#toc-what-is-a-bootloader-0>). Go to **Tools** again and select Burn **Bootloader**.

Next we need to program a sketch to the ATtiny

Open up a basic blink example: **File -> Examples -> Basics -> Blink**



As the ATtiny has no pin for **LED_BUILTIN**, we need to replace that in our sketch with **0** as we have our LED on pin 0. You should then be able to upload this sketch to your board by clicking upload. The LED should be hopefully blinking!

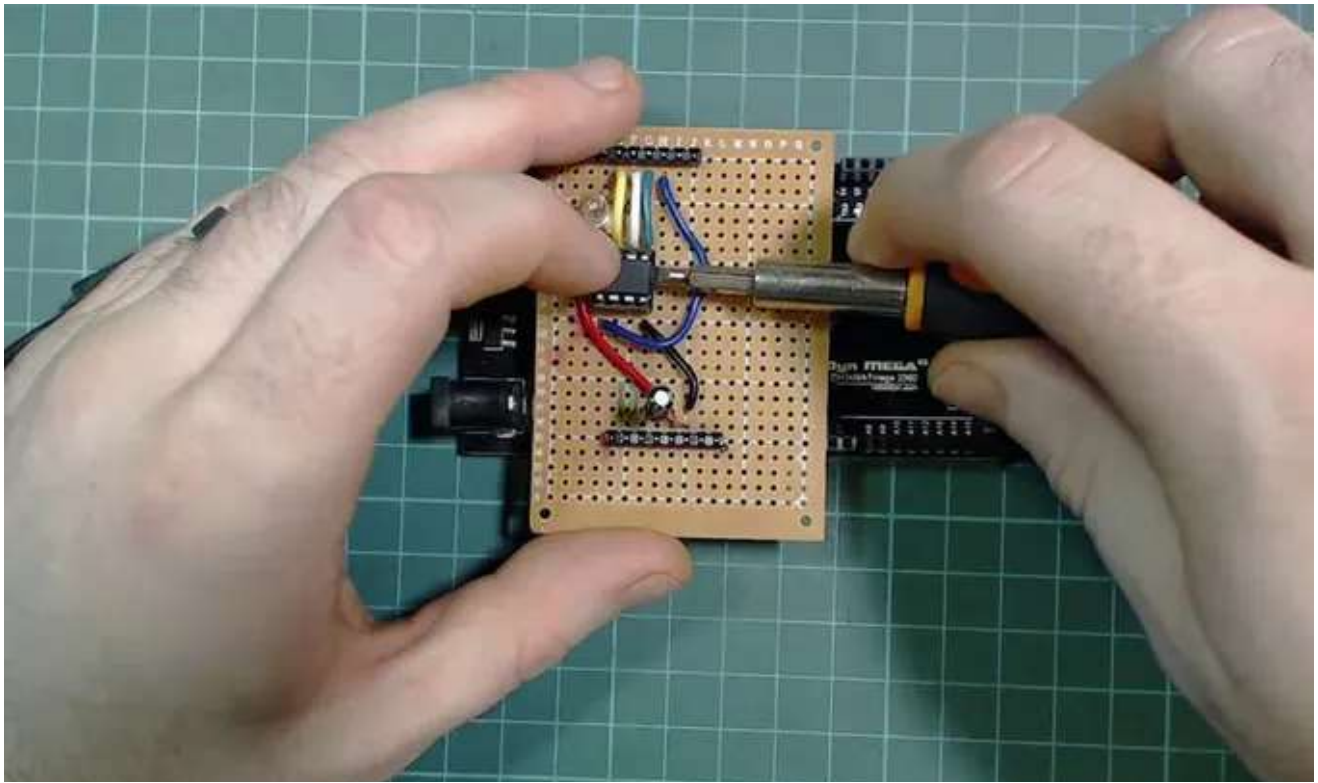


Hopefully your LED should be blinking!

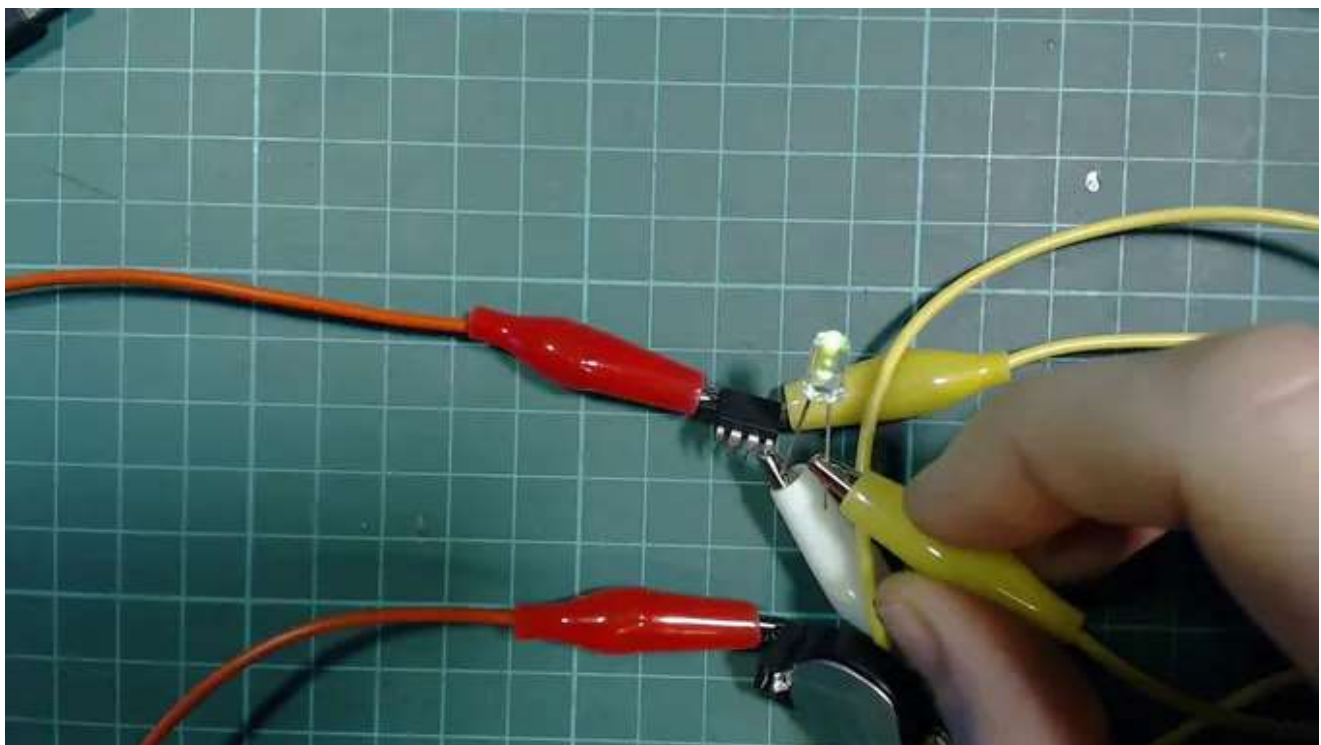
Step 8: Be Free Little Attiny!

Now that it's all grown up its time to remove the attiny from it's Programming shield home.

I find the best way of doing this is using a flat headed screw driver to pop it out. If you just pull it out directly you are very liable to bend the pins. Put the screwdriver underneath the side of chip on one side and gently pry it out, when that side is free move to the other side and repeat.



You can now use the attiny in whatever project you want once you connect V and ground. In the last example I show how you can even use a coin cell battery to power it!



Hopefully you found this tutorial useful, if you have any questions please do feel free to ask! Videos: Check out my YouTube channel for more Videos

(<https://www.youtube.com/channel/UCezJOfu7OtqGzd5xrP3q6WA>)

Live Streams: I live stream working on electronics projects every Monday on Twitch (<https://www.twitch.tv/brianlough>)

And I talk electronics and other random stuff on twitter - @witnessmenow (<https://twitter.com/witnessmenow>)

Brian

COMMENTS

Please log in (/projecthub/users/sign_in?

id=120533&m=project&reason=comment&redirect_to=%2Fprojecthub%2Fbrian-lough%2Fdiy-attiny-programming-shield-6dba38%23comments) or sign up (/projecthub/users/sign_up?

id=120533&m=project&reason=comment&redirect_to=%2Fprojecthub%2Fbrian-lough%2Fdiy-attiny-programming-shield-6dba38%23comments&source=popup) to comment.

Be the first to comment!

AUTHOR



(/projecthub/brian-lough)

Brian Lough (/projecthub/brian-lough)

5 PROJECTS 33 FOLLOWERS

FOLLOW (/PROJECTHUB/USERS/SIGN_UP?ID=301897&M=USER&REASON=FOLLOW&R

PUBLISHED ON

December 6, 2018

RESPECT PROJECT (/PROJECTHUB/USERS/SIGN_UP?ID=120533&M=ARTICLE&REASON=RESPEC...

WRITE A COMMENT