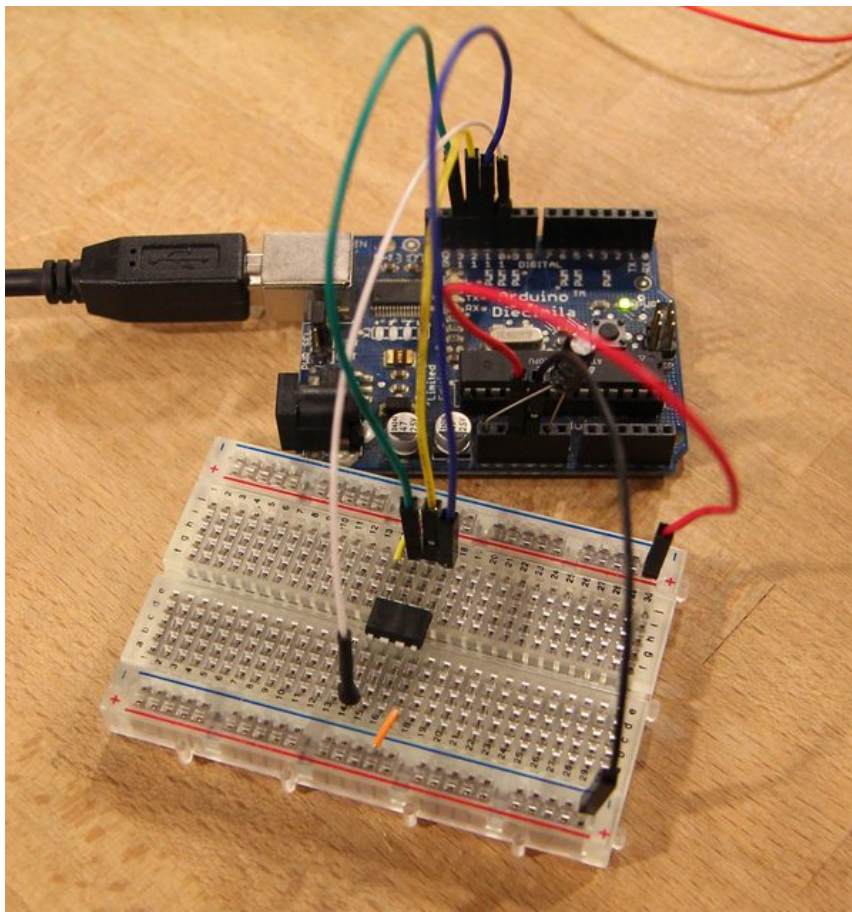




How to program an ATTiny85 (or ATTiny45) with an Arduino + Arduino ISP + Arduino 1.5.x(<http://dev.mikamai.com/post/78652180658/how-to-program-an-attiny85-or-attiny45-with-an>)

Yes, this is yet another tutorial on how to program an ATTiny with Arduino, but time changes, things are evolving fast, and tutorials need to be updated.

This is a very short and concise tutorial, so I won't explain what an ATTiny is and why it is useful to program it with an Arduino, as these arguments are largely covered. Here you will find instructions on how to program an ATTiny85 (or 45) with an Arduino Leonardo, Arduino Yún, Arduino UNO and Diecimila with the most recent version of the Arduino Software (1.5.x)



- Arduino Leonardo, Yun, UNO or Diecimila
- A Breadboard
- Hookup wire
- ATtiny85 (or ATtiny45)
- 10uF 16V electrolytic capacitor (for Arduino UNO or Diecimila)
- 220ohm 1/4 watt resistor
- LED

Step 1

First of all, you will need to download the Arduino Tiny plugin from [here\(https://code.google.com/p/arduino-tiny/downloads/detail?name=arduino-tiny-0150-0020.zip&can=2&q=\)](https://code.google.com/p/arduino-tiny/downloads/detail?name=arduino-tiny-0150-0020.zip&can=2&q=), so your Arduino software will be able to program the ATTiny.

If you want to make it work with Arduino 1.0.x software, just download [this file\(https://code.google.com/p/arduino-tiny/downloads/detail?name=arduino-tiny-0100-0018.zip&can=2&q=\)](https://code.google.com/p/arduino-tiny/downloads/detail?name=arduino-tiny-0100-0018.zip&can=2&q=) instead!

Then unzip it and copy the "Tiny" folder into an "hardware" folder that you have to create into your sketches folder.

The final result should be

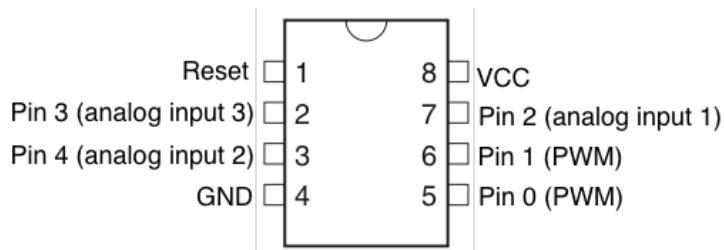
arduino_sketches_folder/hardware/tiny

Inside the "tiny" folder, you will find an "avr" folder. Inside this folder, there is a file named "Prospective Boards.txt". Just rename it into "boards.txt"

When you will start the Arduino Software, you will find new boards under the **Tools > Board** menu, as in this picture.

Step 2

Now we have to make the proper connections between the Arduino Board and the ATTiny



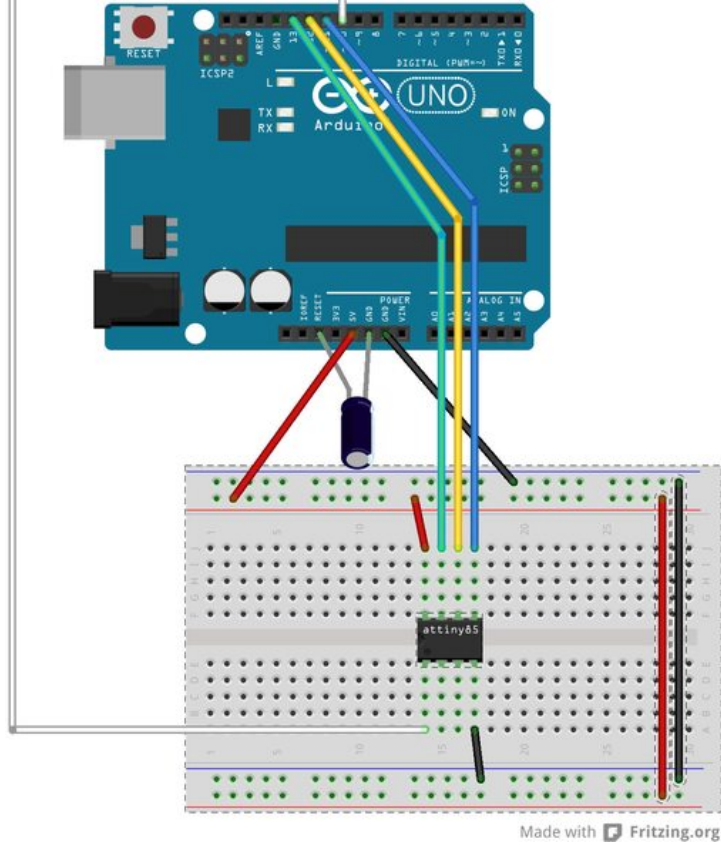
Arduino UNO/Diecimila

```

Arduino PIN 10 --> ATTiny Reset PIN
Arduino PIN 11 --> ATTiny PIN 0
Arduino PIN 12 --> ATTiny PIN 1
Arduino PIN 13 --> ATTiny PIN 2
Arduino GND PIN --> ATTiny PIN GND PIN
Arduino +5V PIN --> ATTiny PIN VCC PIN

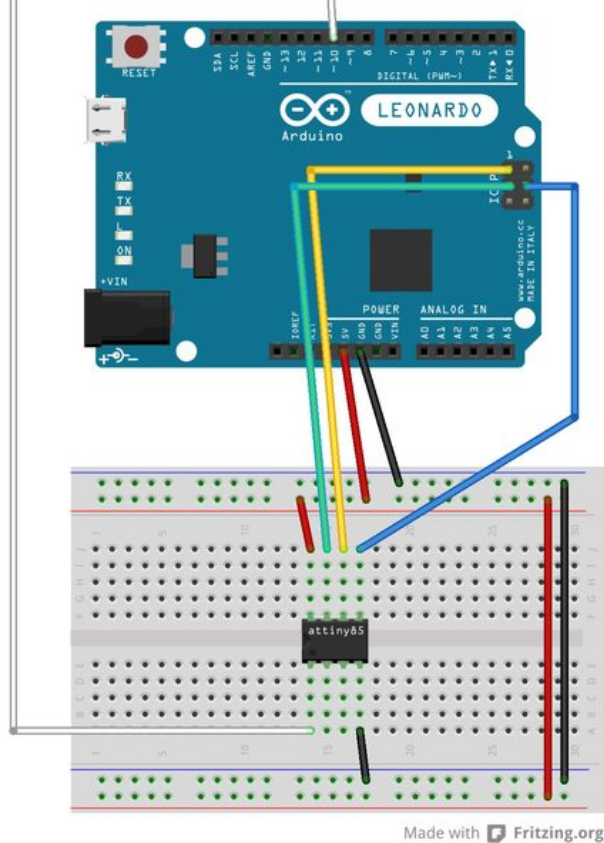
```

as in the picture below



or for Arduino Leonardo/Yún

Arduino PIN 10	--> ATTiny Reset PIN
Arduino ICSP MOSI PIN	--> ATTiny PIN 0
Arduino ICSP MISO PIN	--> ATTiny PIN 1
Arduino ICSP SCK PIN	--> ATTiny PIN 2
Arduino GND PIN	--> ATTiny PIN GND PIN
Arduino +5V PIN	--> ATTiny PIN VCC PIN



Step 3

Open the **File > Examples > ArduinoISP** Sketch and upload it on your Arduino

Arduino Leonardo/Yún

Only for these platforms, you have to make this changes to the Arduino ISP sketch, BEFORE uploading it

```
// January 2008 by Randall Bohn
// - Thanks to Amplifcar for helping me with the STK500 protocol
// - The AVRISP/STK500 (mk I) protocol is used in the arduino bootloader
// - The SPI functions herein were developed for the AVR910_ARD programmer
// - More information at http://code.google.com/p/mega-isp(http://code.google.com/p/mega-isp)

#include "pins_arduino.h"
#define RESET      10

#define LED_HB      13
#define LED_ERR     8
#define LED_PMODE  7
#define PROG_FLICKER true
```

Now upload it to your Arduino.

Step 4

Load the **File > Examples > 01.Basic > Blink** Sketch, and change this line

```
int led = 0;
```

this is because the ATtiny has only 6 output pins (0,1,2,3,4,5) and so we will use the pin 0 instead of the pin 13

Step 5

Choose the proper board settings (i.e. Attiny85 @ 1Mhz), then choose **Tools > Programmer > Arduino as ISP**. Now you can upload the "blink" example to the ATtiny85

If everything went fine, you will see something like

```
avrdude: please define PAGEL and BS2 signals in the configuration file for part ATtiny85
avrdude: please define PAGEL and BS2 signals in the configuration file for part ATtiny85
```

in the output window of the Arduino Software. Don't care about the warnings, they're normal.

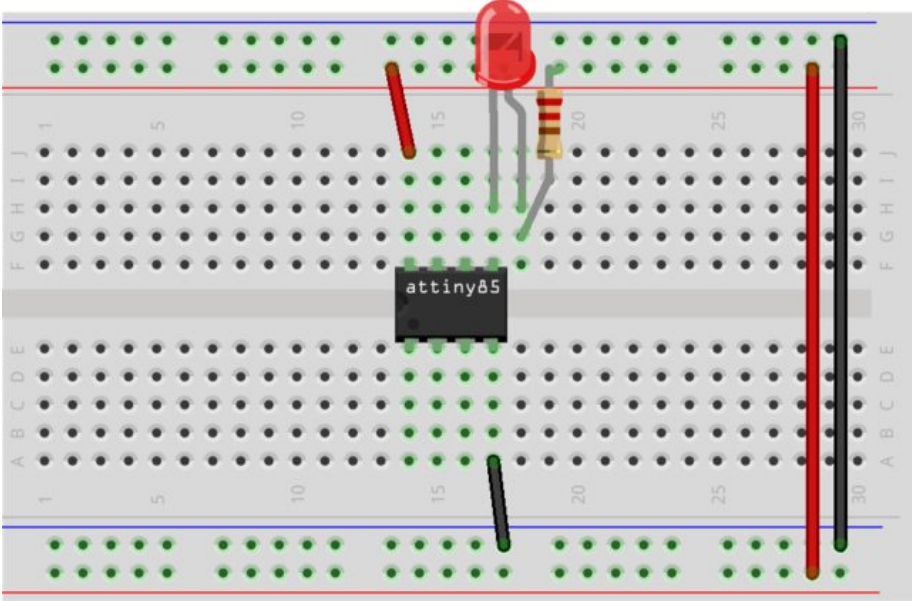
If something went wrong, you will find a message like

```
avrdude: please define PAGEL and BS2 signals in the configuration file for part ATtiny85
avrdude: Yikes!  Invalid device signature.
        Double check connections and try again, or use -F to override
this check.
```

This means that you have to check your connections again, it happend a lot to me the first time I tried to use an Arduino Leonardo instead of an Arduino UNO, because the ICSP Pins are mapped in a different way on the Leonardo/Yún boards.

Step 6

Now we can test it! To build a simple test circuit, for the “blink” example, just build this simple circuit



I hope this tutorial is useful to you as it would be useful to me if I was able to find one: I got more then an headache trying to put all the pieces together: the proper plugins, the connections and the little software modifications...

Cheers!

Jeko

 By Stefano Guglielmetti(<https://twitter.com/jeko>)

Posted on **Wednesday, 5 March 2014**(<http://dev.mikamai.com/post/78652180658/how-to-program-an-attiny85-or-attiny45-with-an>) ·

0 notes(<http://dev.mikamai.com/post/78652180658/how-to-program-an-attiny85-or-attiny45-with-an#notes>)

Read Later

Share

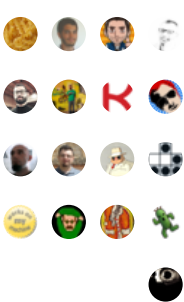
[⌵\(http://dev.mikamai.com/post/78652180658/how-to-program-an-attiny85-or-attiny45-with-an\)](http://dev.mikamai.com/post/78652180658/how-to-program-an-attiny85-or-attiny45-with-an)

Previous post(<http://dev.mikamai.com/post/78453410376>)

Next post(<http://dev.mikamai.com/post/78664776898>)

Site authors

About us



  the official Mikamai developers blog

Find me on

[Penman](#)
[Theme](#)(<http://penman.stylehatch.co>)
created by [Style](#)
[Hatch](#)(<http://stylehatch.co>)
Powered by
[Tumblr](http://www.tumblr.com/)(<http://www.tumblr.com/>)

