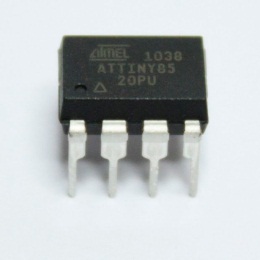
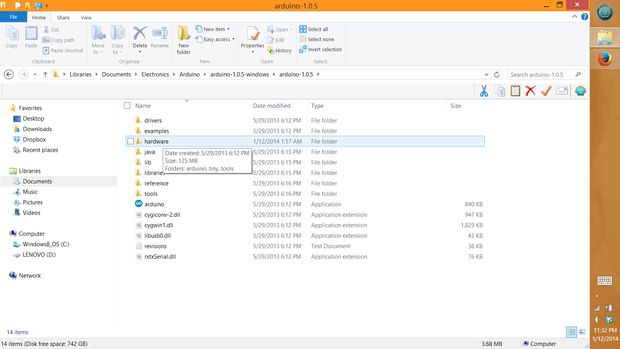
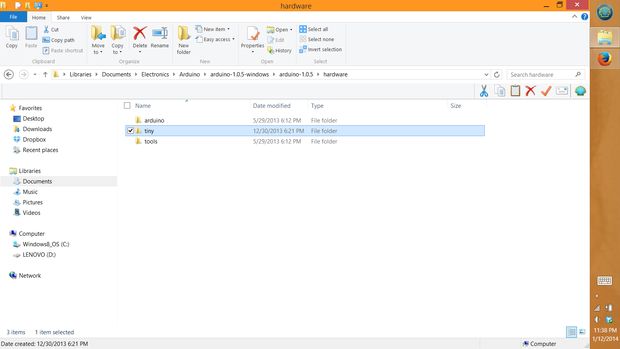
**Programming the ATtiny**

[](http://cdn.instructables.com/FEN/BB5H/HQCPMKSZ/FENBB5HHQCPMKSZ.LARGE.jpg)

This is a tutorial for programming the ATtiny chip. This tutorial will be more in depth than most. I will give you every step and tell you the things nobody told me when I first started programming ATtiny.  
  
Things You'll Need  
- Arduino Uno( Can be done with other boards but I will be using the UNO)  
- ATtiny 85 or equivalent  
- 10 uF Capacitor  
- Jumper wire  
- Breadboard  
- LED

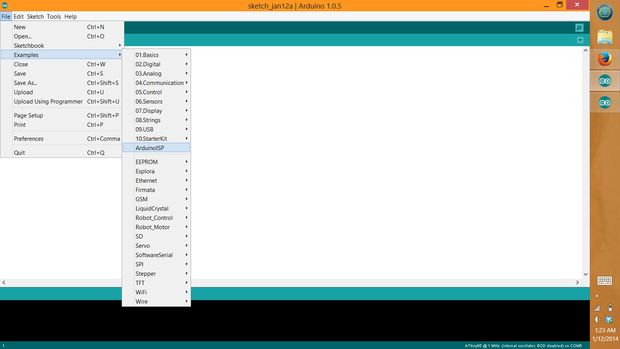
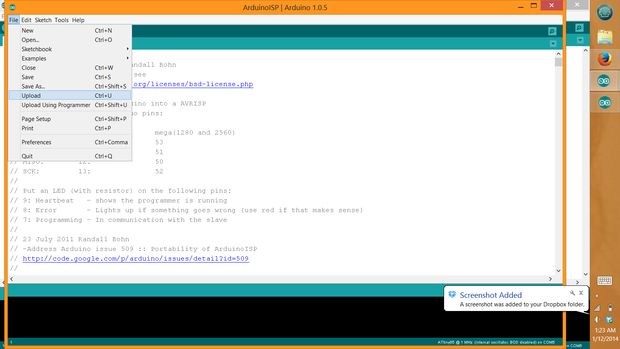
**Step 1: Software**

[](http://cdn.instructables.com/FOO/WO5Z/HQCWG6UI/FOOWO5ZHQCWG6UI.LARGE.jpg) [](http://cdn.instructables.com/F1R/F2V6/HQCWG6VG/F1RF2V6HQCWG6VG.LARGE.jpg)

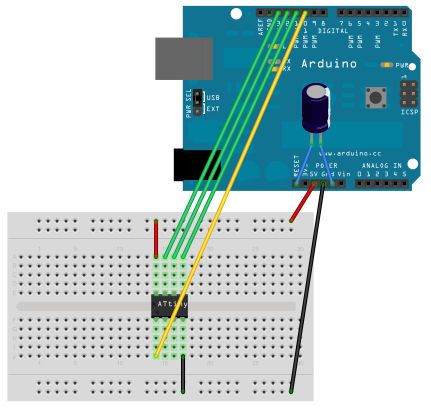
First off, we're gonna need the software. If you don't already have it, go to arduino.cc and download the latest version. Next were gonna need to install the ATtiny core files, which are included down below. Open up the arduino software and go to "file" and click on "preferences" take note of you sketchbook location. You'll need it later. CLOSE the IDE, if you don't it won't work until you do. Download the ATtiny core files I've included below. Go into the sketchbook and go to the hardware folder. Drag and drop the "tiny" file into that hardware folder. Now reopen the IDE and click on "tools" and then click on "boards". There should be a whole new set of boards availible including the ATtiny 85. For now, just leave the board as UNO.

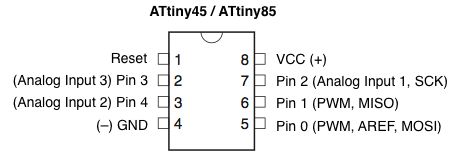
**Step 2: Arduino as ISP**

Now we need to program the Arduino as an ISP( In-System Programmer). To do this go into the examples and look for ArduinoISP. Open this file and upload the sketch to the UNO as you would with any normal sketch. You have now turned the Arduino into an external programmer.

[](http://cdn.instructables.com/F6P/A31F/HQBSQRZA/F6PA31FHQBSQRZA.LARGE.jpg)[](http://cdn.instructables.com/F1V/T6CN/HQBSQRZC/F1VT6CNHQBSQRZC.LARGE.jpg)

**Step 3: Hooking up the ATtiny to the UNO**

[](http://cdn.instructables.com/FK2/TW9M/HQCWFYU3/FK2TW9MHQCWFYU3.LARGE.jpg)[[ATtiny45-85.png](http://cdn.instructables.com/FOX/O3FX/HQCPMJE4/FOXO3FXHQCPMJE4.LARGE.jpg)](http://cdn.instructables.com/FOX/O3FX/HQCPMJE4/FOXO3FXHQCPMJE4.LARGE.jpg)



Next step, attach the ATtiny to the Arduino.  
  
Arduino Pin          Attiny Pin  
    10           to             1  
    11           to             5   
    12           to             6  
    13           to             7  
    5v           to             8  
    GND       to             4  
  
\*note: See pictures for pin configuration of the tiny  
  
  
Now you need to put the 10uF capacitor between the reset and GND. Be sure to observe polarity( the small negative signs go to ground).  
If you don't put the cap in there it won't work.

**Step 4: Program the ATtiny**

You are now ready to program the ATtiny, but first we need to select our ATtiny chip. To do this, go to "tools" click on "boards" and select the the ATtiny 85 with 1MHz internal oscillator. Don't use any of the ones that say external oscillator because you need to hookup a external crystal of the frequency. We should check to make sure everything has worked. Go into the examples and open the "blink" sketch change the line "int led = 13";  to "int led = 0". Now, in order for the sketch to upload you need to go to "file" and click "upload using programmer" it will start uploading and you will get 2 errors that say "avrdude: please define PAGEL and BS2 signals in the configuration file for part ATtiny85" this tells you that the ATtiny has successfully been programmed. Plug in a LED to pin 5(labeled as pin 0) and connect the negative side to pin 4(GND). It should start blinking at 1Hz. Have fun building and creating with your new tinier arduino.  
  
Some things to note about the ATtiny. It doesn't support all the functions that the UNO does but it does support the following:  
- pinMode()  
- digitalWrite()  
- digitalRead()  
- analogRead()  
- analogWrite()  
- shiftOut()  
- pulseIn()  
- millis()  
- micros()  
- delay()  
- delayMicroseconds()  
- SoftwareSerial  
  
\*Another Note: To run the board at 8MHz you must burn the bootloader. To do this, Click on ATtiny 85 with the 8MHz internal oscillator. Click on "tools" and click on burn bootloader. Once that is done the Attiny will run at 8MHz.Top of Form