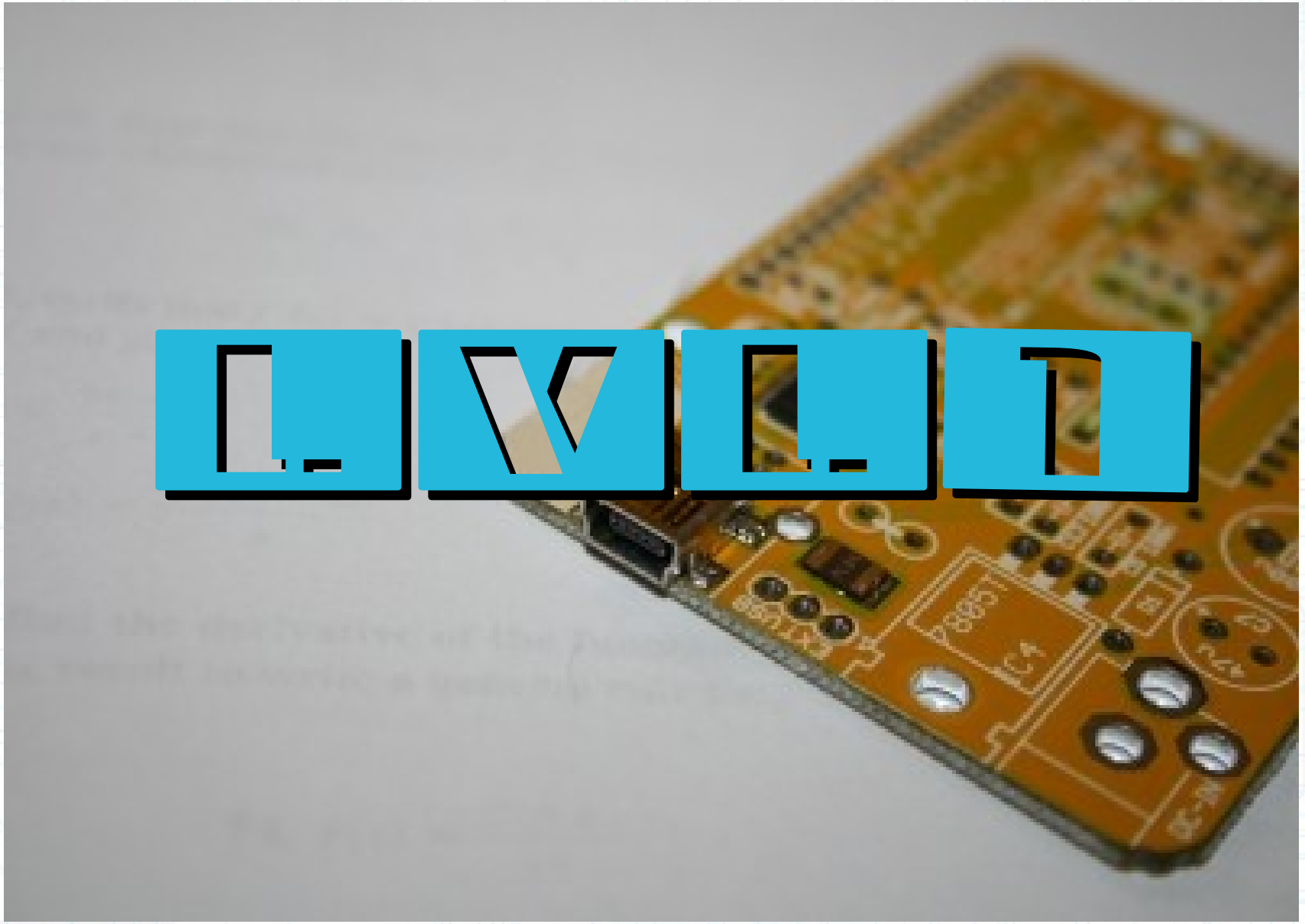
Solder Your Own Freeduino Workshop



Greetz!

Welcome to the LVL1 workshop for Solder Your Own Freeduino. We will be covering the following:

- Basic soldering technique
- Component identification
- Basic Programming of the Freeduino.

Say hello to your neighbors

- They will be your strongest allies.
- Check with each other before soldering components. This is the best way to AVOID MISTAKES!

About Freeduino

Arduino is a protected trademark on an open source hardware design. Freeduino is compatible open source hardware with unrestricted use of the name.

About Freeduino (cont.)

- Based around an 8-bit Atmel AVR microcontroller.
  - On-chip Flash memory
  - A/D converter
  - PWM output
  - General I/O
  - Tons of features really...
- Shield format for easy modularity.
- Easily programmable via USB/Serial.

Soldering Iron

- Used to transfer heat to components intended to be soldered together.
- Higher wattage can transfer more heat.

Solder

- Low melting temp
- Conductive
- Fusable Alloy
- Tin/Lead - Rosin Flux mix
- Rosin improves electrical conductivity and mechanical strength.

Avoid Common Soldering Mistakes

- Use the soldering iron to transfer heat to ALL components needing to be joined together.
  - Don't 'carry' solder to the components. Don't melt solder on the iron, melt solder on the components leads / vias.
- Keep your tip CLEAN!
  - Use a sponge or pad to clean the tip. Use tip cleaner. Tin the tip with some solder.

Avoid Common Soldering Mistakes (cont.)

- Timing is everything.
  - It should only take a couple of seconds to heat up the components enough to melt solder. If you miss your window, stop for a second and let it cool off again. A heavy hand can melt and/or damage components.
- Avoid 'Cold Solder Joints'.
  - A dull blob of solder that doesn't flow around the joint is neither electrically conductive, nor mechanically strong.

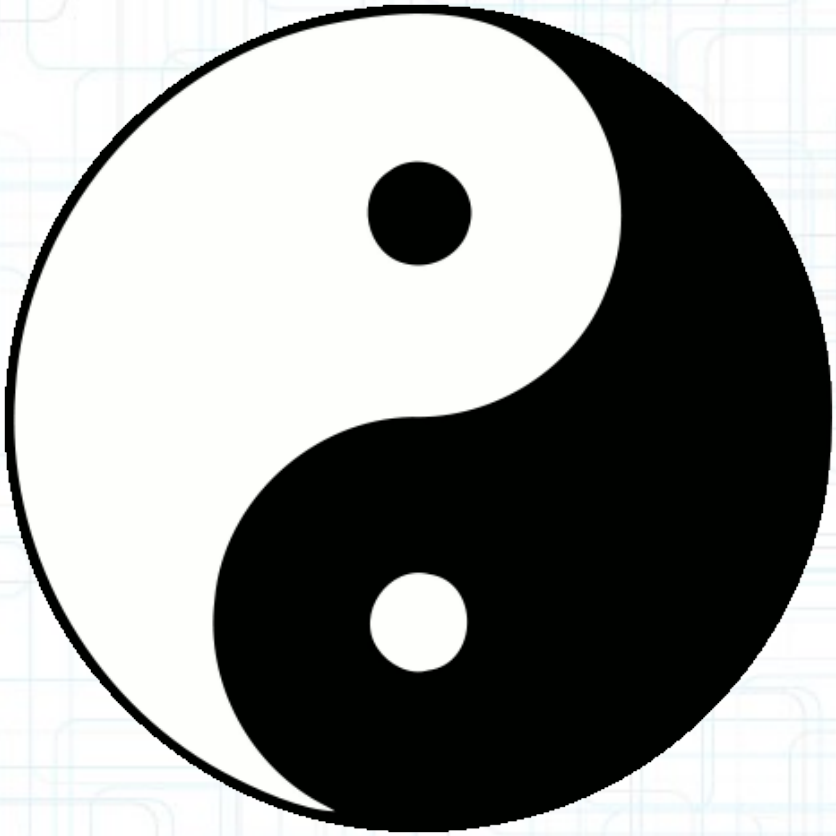
Soldering is a Manual Skill and you are not Neo

Soldering is a Manual Skill (cont.)

- Your hominid brain cannot master soldering by reading or otherwise 'downloading' information about correct techniques. It uses entirely different nerve pathways.
- You will only master soldering by actually soldering.
  - Like playing a guitar, or shooting an arrow, or removing an appendix.

You Will Make Mistakes

- An equally important skill is desoldering
- It is the yang to soldering, allowing you to undo what has been done.
- When you screw up, you can always desolder.



Desoldering

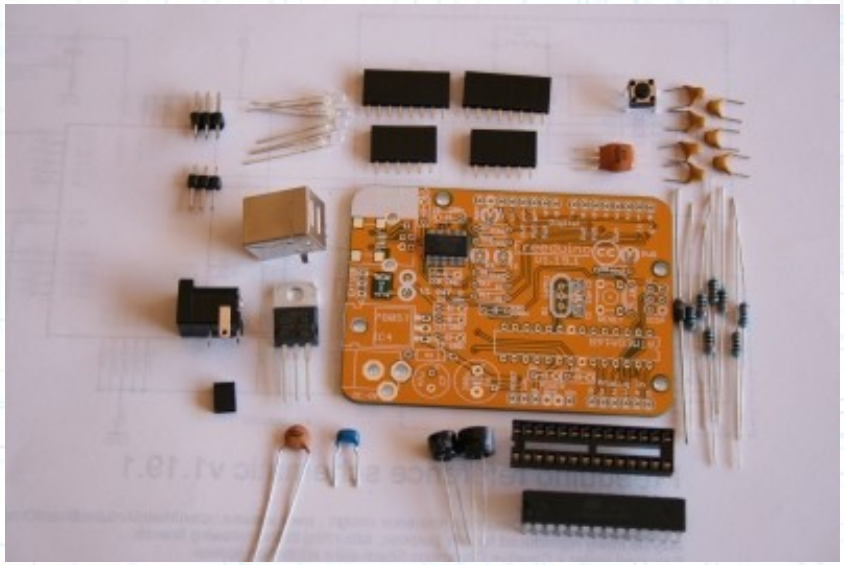
- Soldering wick does what it says.
- There are also soldering suction devices that will remove solder.

Download Arduino Software

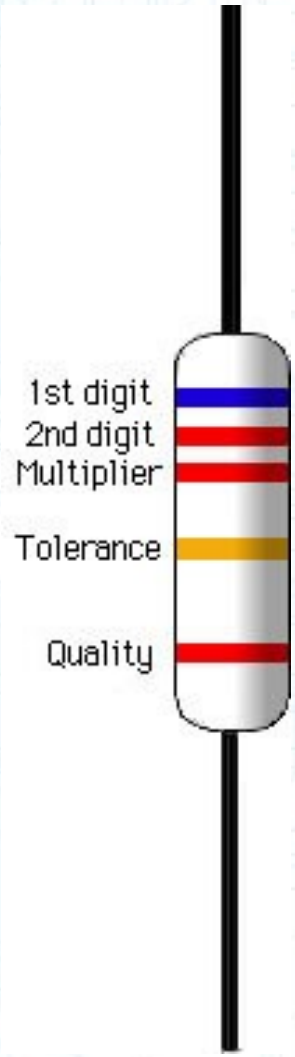
- The open-source Arduino environment makes it easy to write code and upload it to the i/o board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing, avr-gcc, and other open source software.
- http://www.arduino.cc/en/Main/Software
- Extract the archive and make sure it runs correctly on your system.

Components

- Resistors
- Capacitors
- LEDs and Diodes
- Odds and Ends



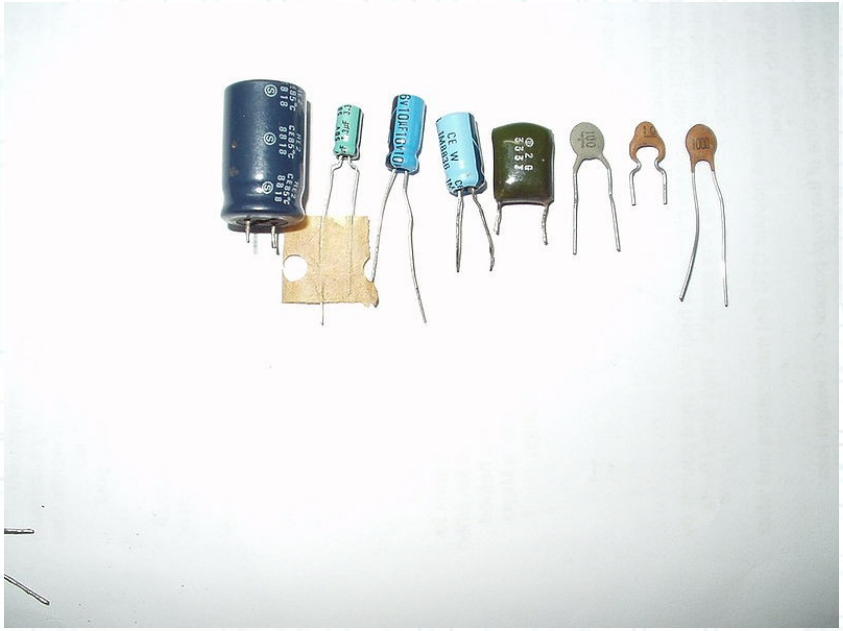
Resistors



- Color-coded bands tell you the ohm resistance value.
- Bend the leads after inserting into the PCB to keep them secure.

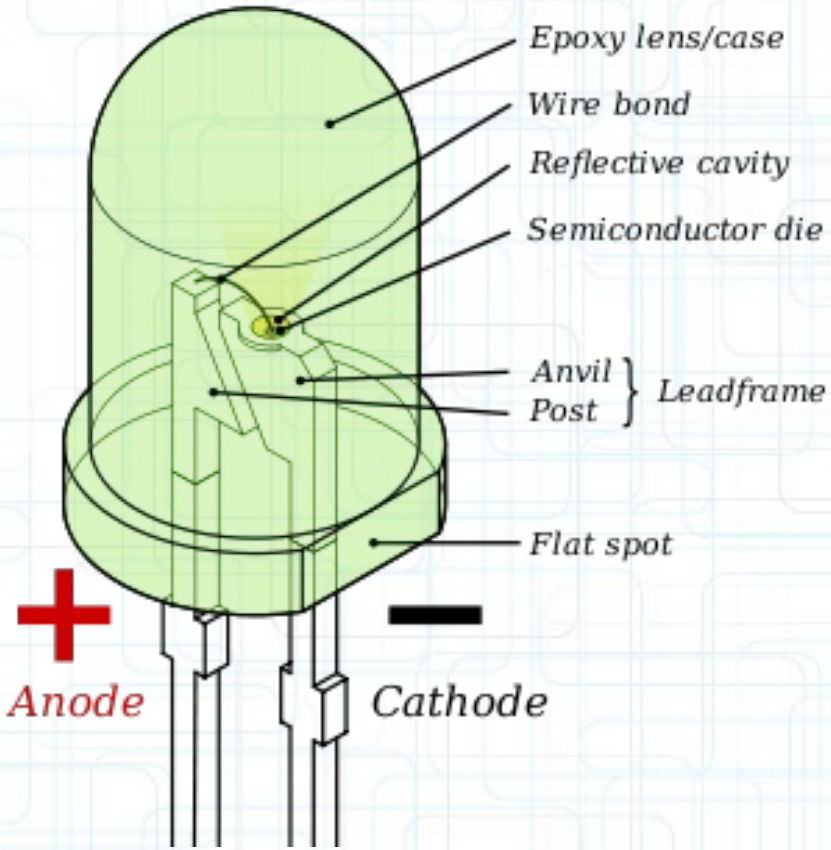
Capacitors

- Values are either labeled literally (on electrolytics) or 3-digit code (i.e. 101, 475)
- Electrolytic capacitors have polarity. Longer lead is ”+”.

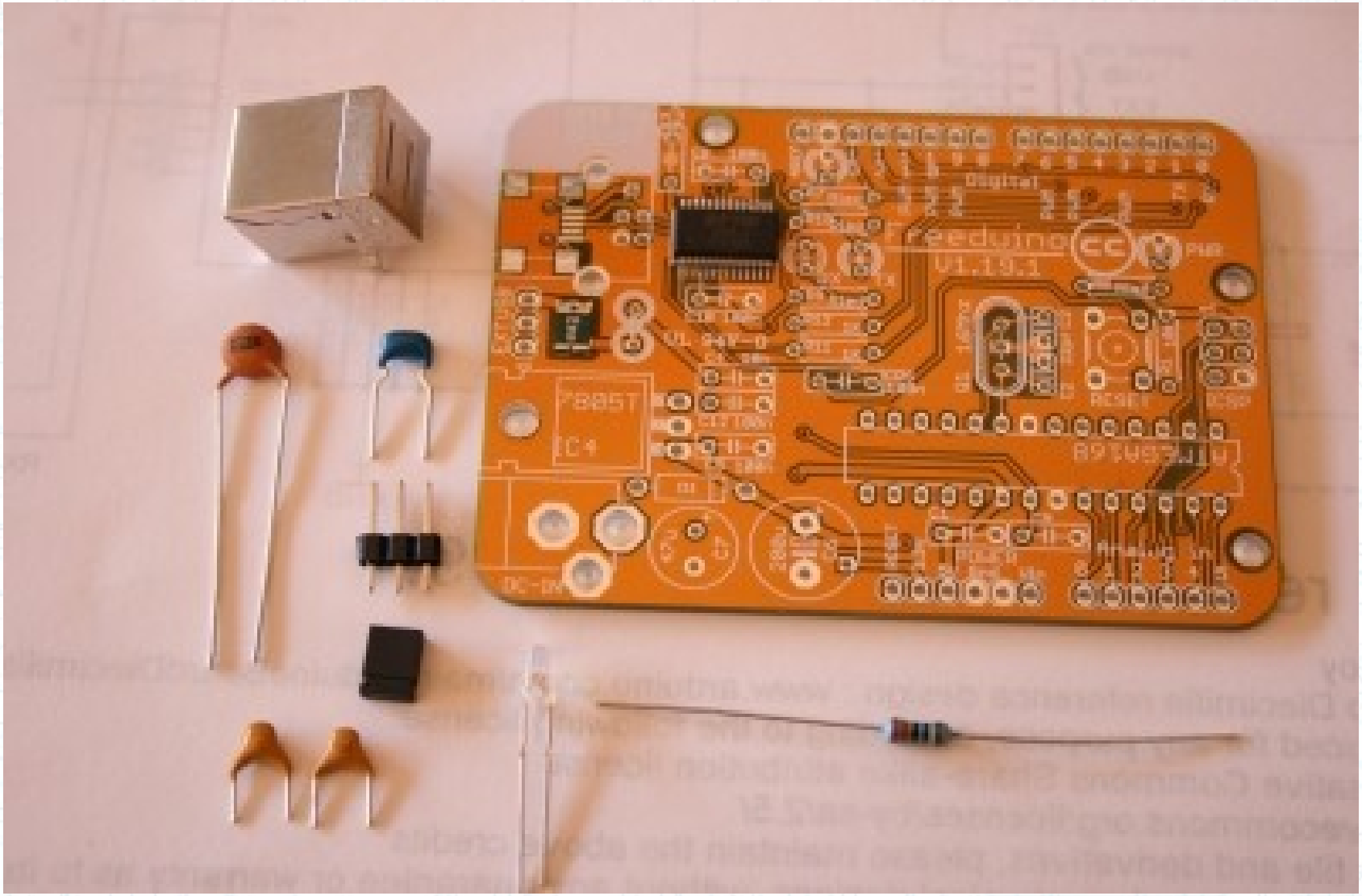


LEDs and other Diodes

- LED polarity is indicated by lead length. Longer lead is ”+”.
- Diode polarity is marked by a strip indicating ”+”. Match the PCB graphic.



Step 1 – USB Components

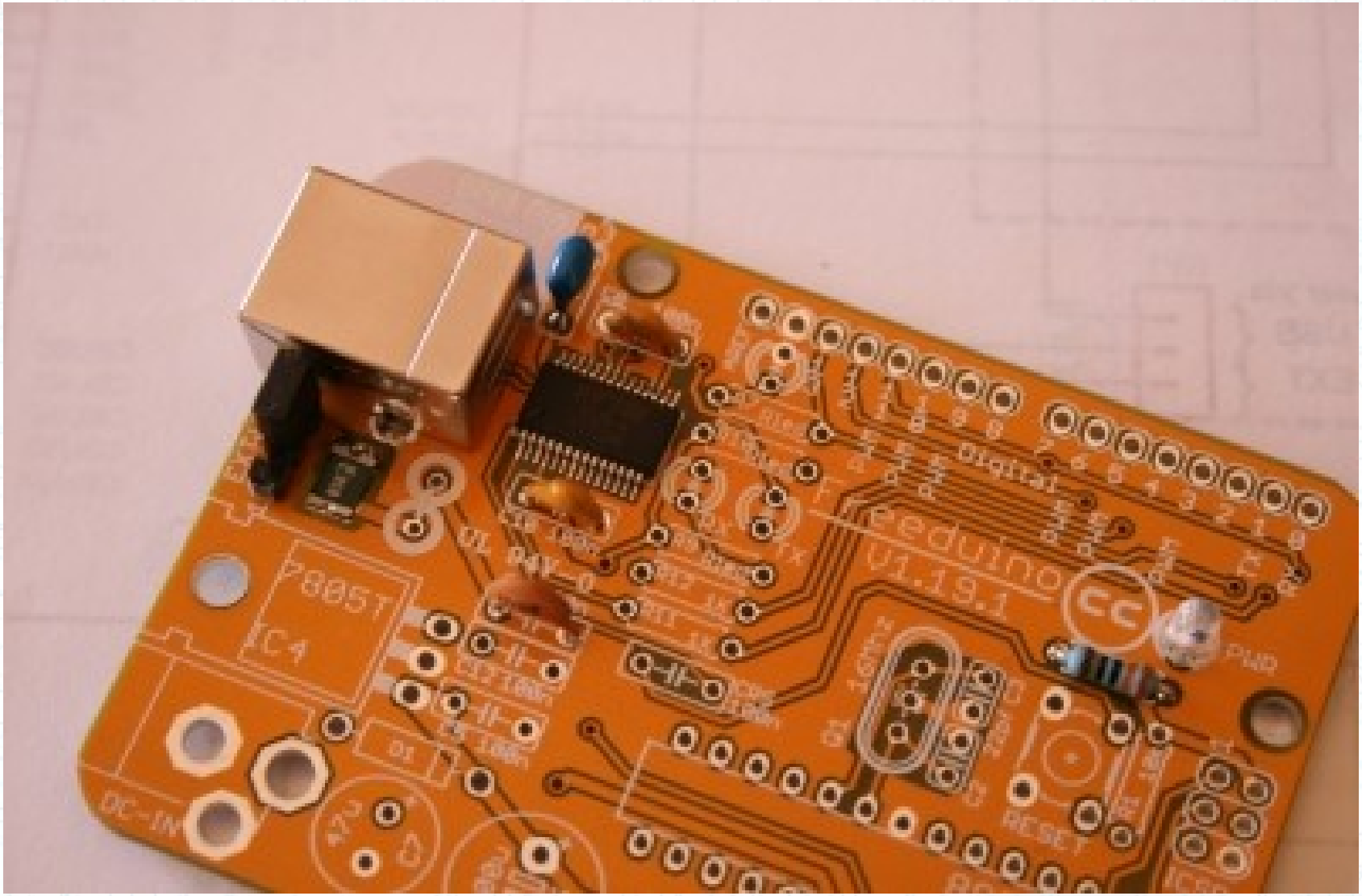


Step 1 – USB Components

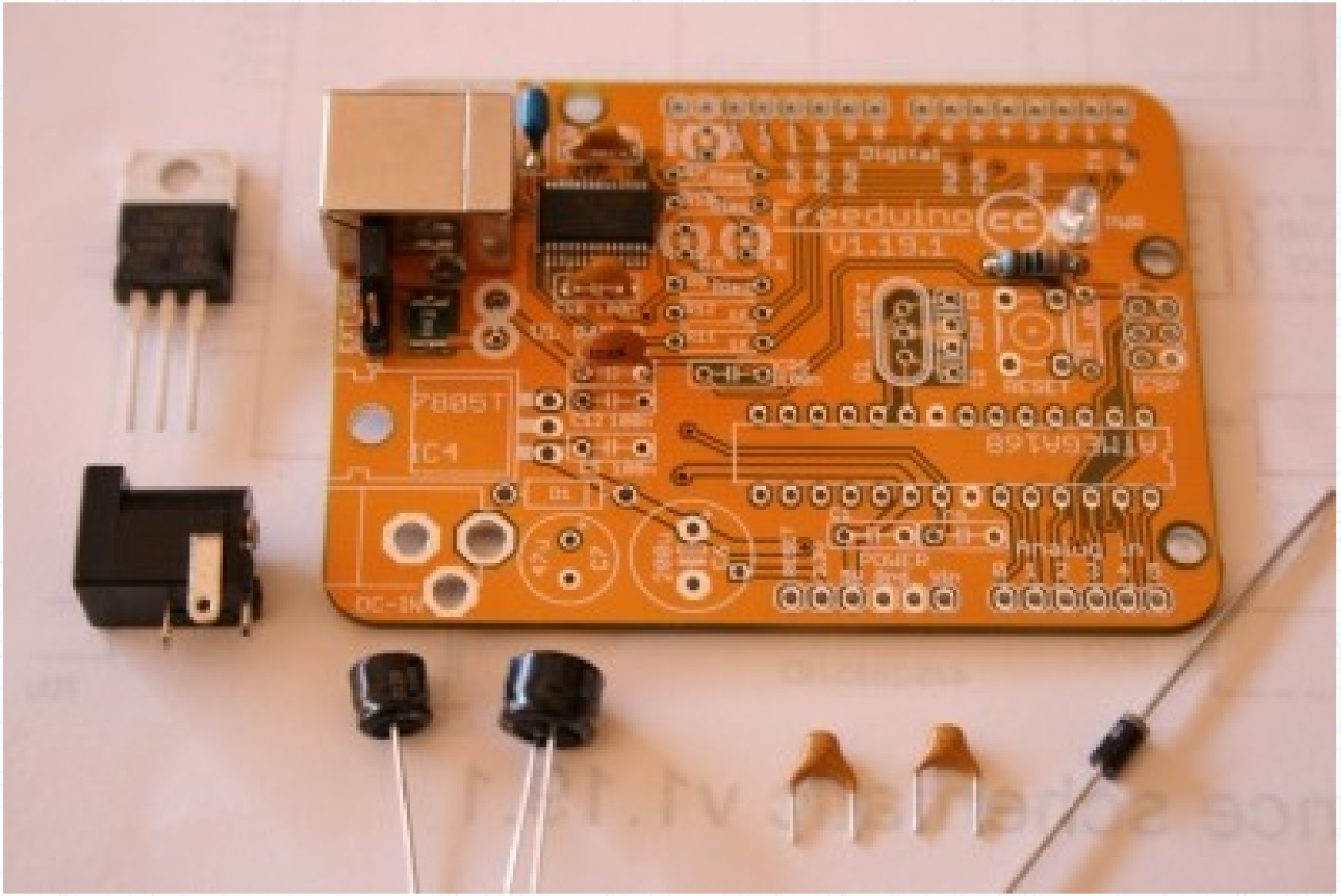
- R8 – 1Kohm resistor ”RLED” (**brown black red gold**)
- C13 – 4.7uF capacitor (yellow, marked **475**)
- LED – 3mm green LED
- C8, C10 – 100nF ceramic capacitor (marked **104**)
- C4 – 10nF ceramic capacitor (marked **103**)
- SV1 – 3 pin male header
- SHUNT – Black shunt

Solder all of these components, place the shunt over 2 pins of the 3-pin jumper labeled ”USB”. Once successfully completed, a connected USB cable will light up the power LED.

Step 1 – USB Assembled



Step 2 – Power Components

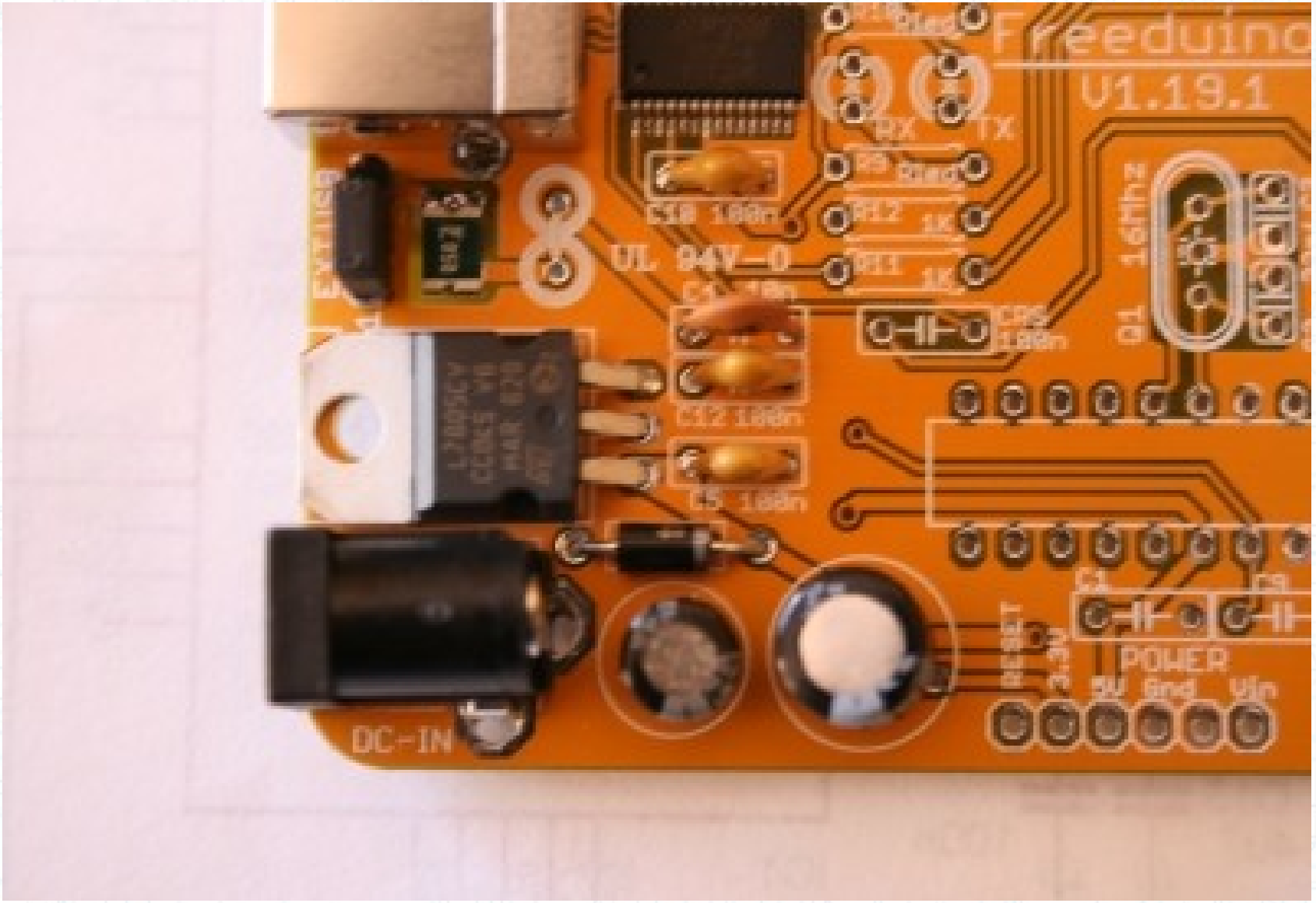


Step 2 – Power Components

- DC1 – 2.1 mm barrel DC power jack
- D1 – 1N4004 diode
- C5, C12 – 100nF ceramic capacitor (marked **104**)
- C6 – 100uF electrolytic capacitor
- C7 – 47uF electrolytic capacitor
- IC2 – 7805 5V positive voltage regulator

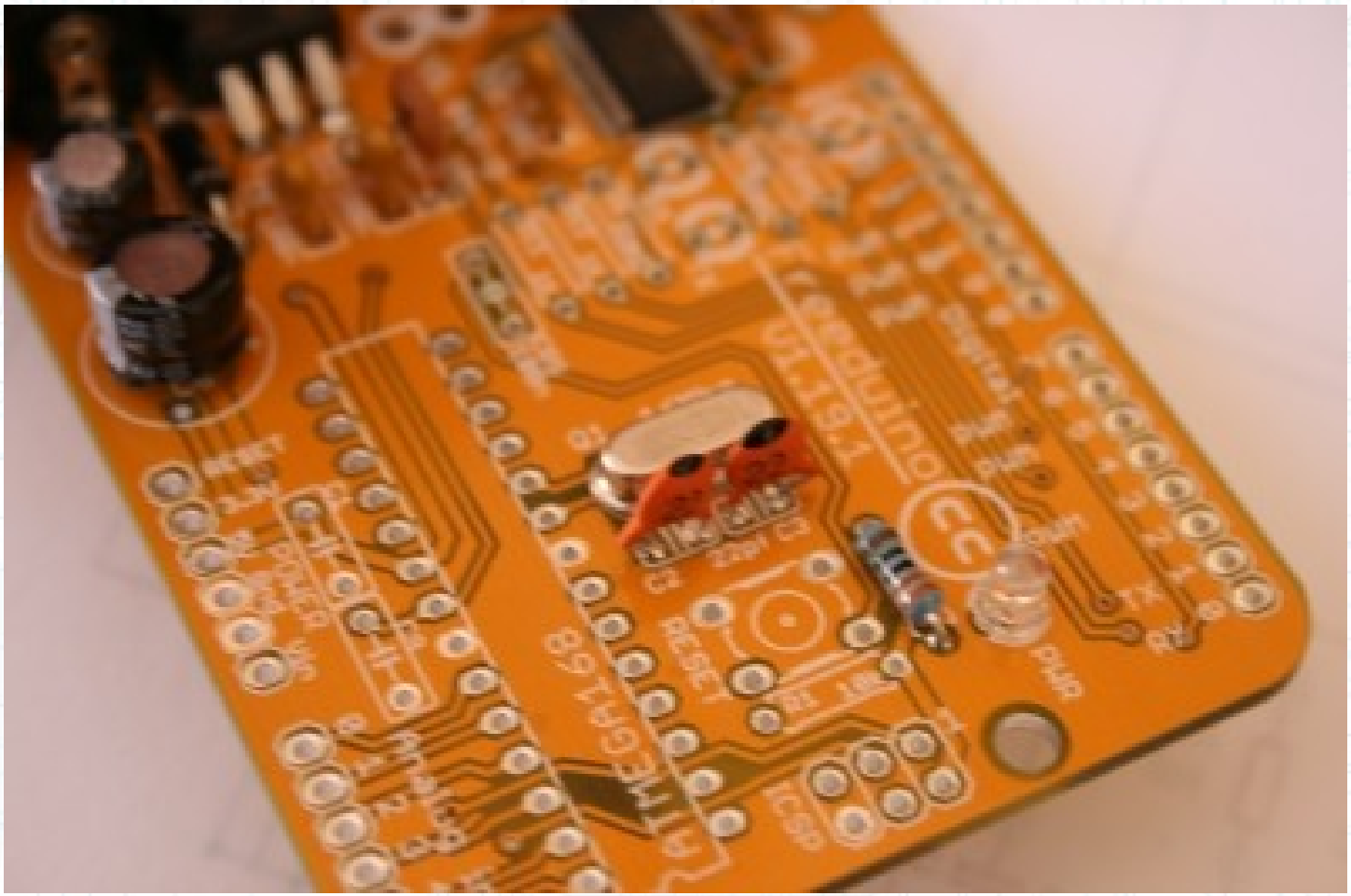
Make sure to match the polarities of the electrolytic capacitors and diode to the board! The voltage regulator will be bent over after it's been soldered to sit flat against the board.

Step 2 – Power Components

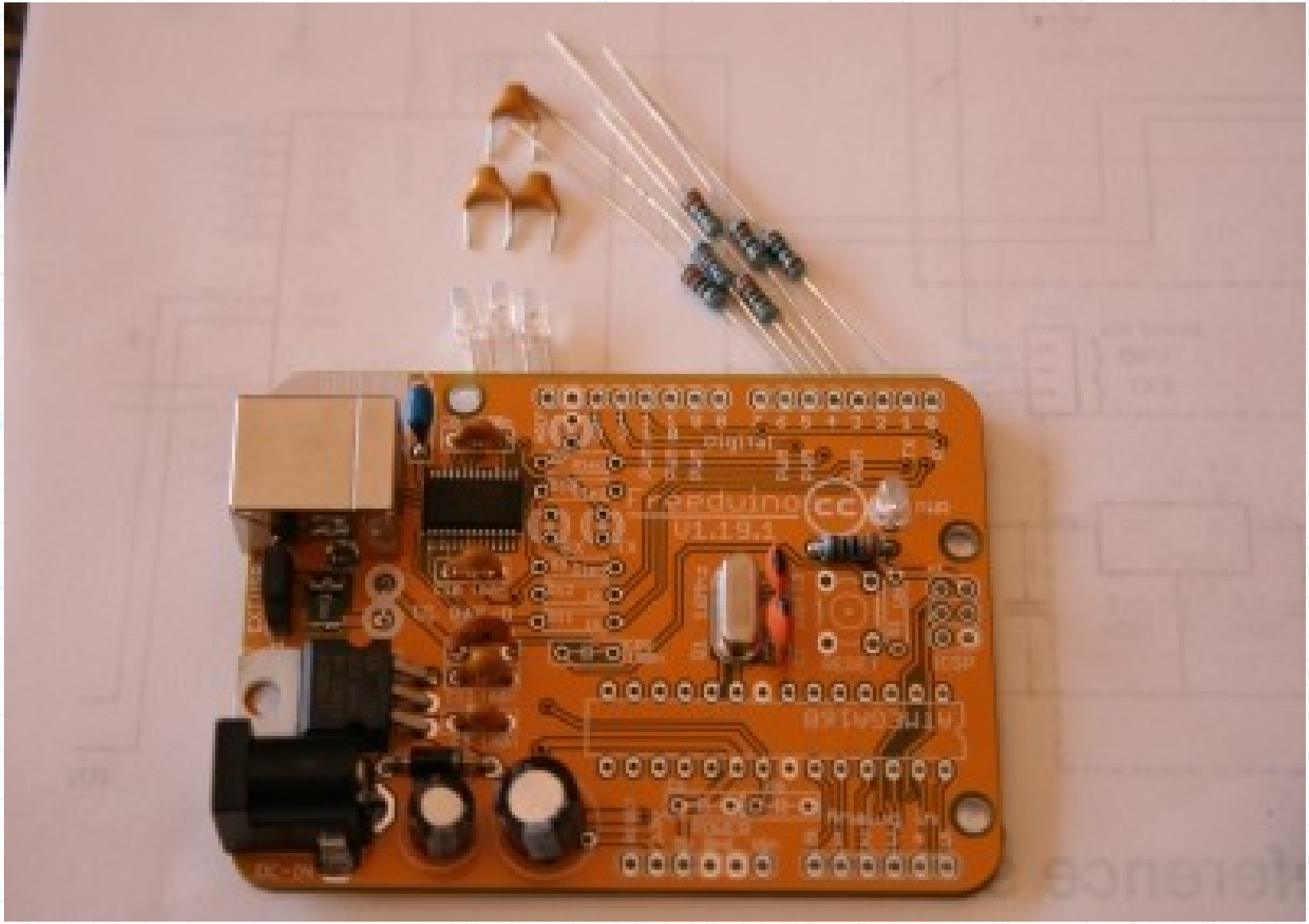


Step 3 – Crystal and Caps

- Q1 – 16Mhz Crystal (marked 16.000)
- C2, C3 – 22pF ceramic capacitor (marked **104**)



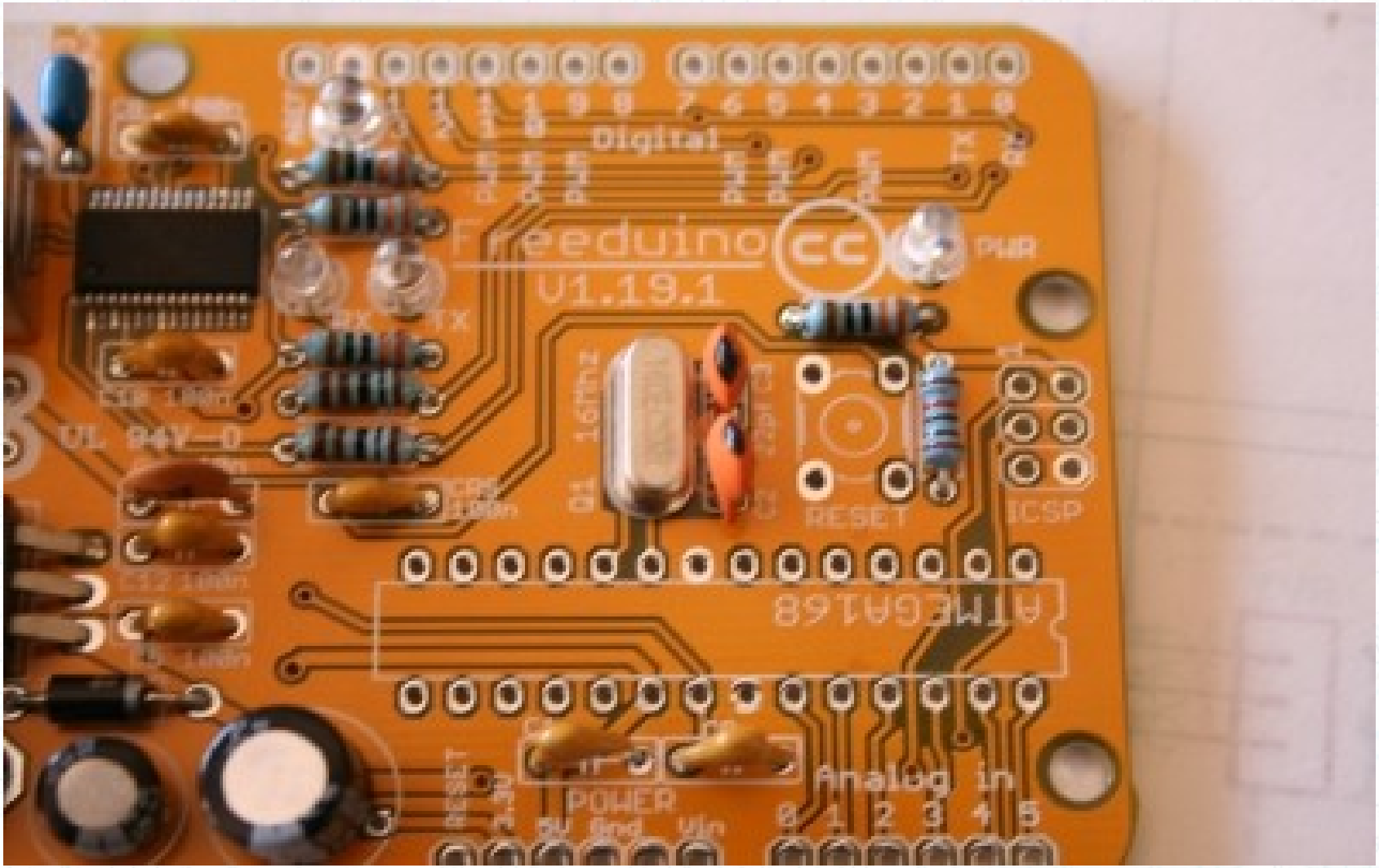
Step 4 – More Components



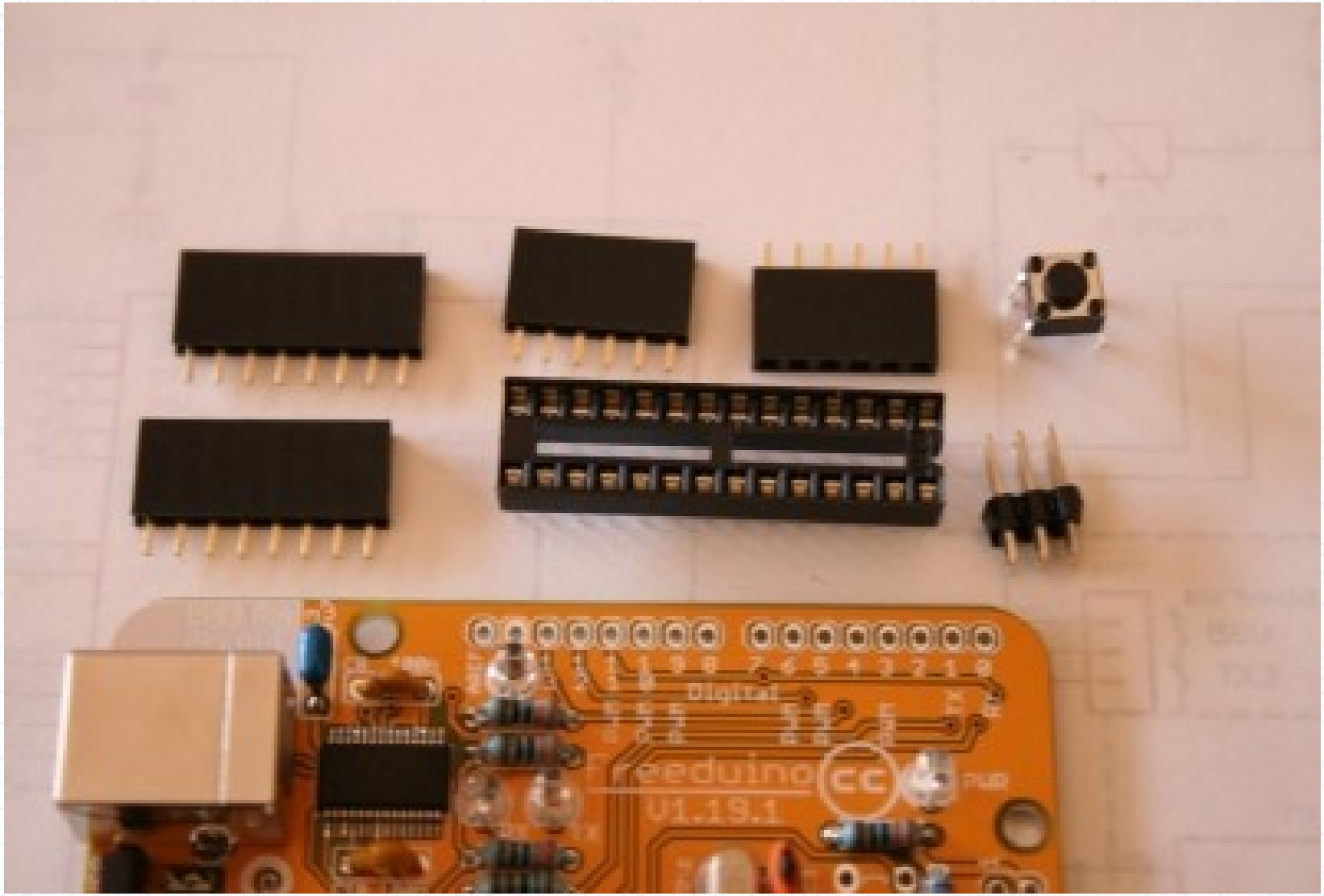
Stage 4 – More Components

- R1 – 10Kohm resistor (**brown black orange gold**)
- R11, R12 – 1Kohm resistors (**brown black red gold**)
- CRS, C1, C9 – 100nF ceramic capacitor (marked **104**)
- 13, RX, TX – 3mm LED
- R7, R9, R10 – 1Kohm resistors (**brown black red gold**)

Step 4 – More Components



Step 5 – Headers and Sockets

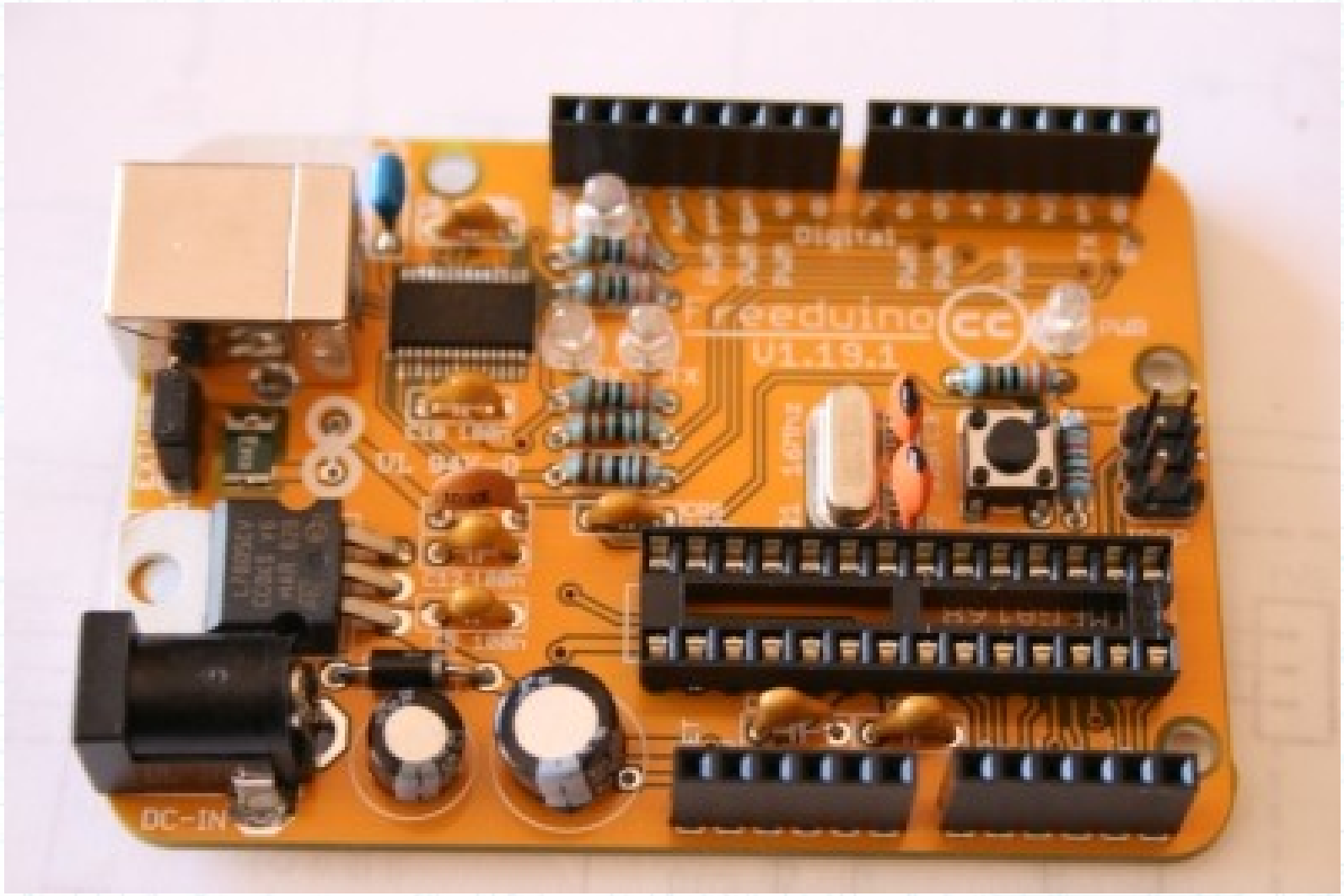


Step 5 – Headers and Sockets

- ICSP – 2x3 pin male header
- RESET – reset switch
- POWER, Analog In – 2x6-pin female header
- Digital – 2x8-pin female header
- ATMEGA168 – 28-pin DIP socket

Bend the corner leads of the IC socket to keep it in place. The female headers are easy if you use a shield as a jig. Male headers you can set down to solder one corner, then push in to set flush: I.e. You may burn yourself

Step 5 – Headers and Sockets



Step 6 – ATmega328 & Blink

- Straighten Atmega328 pins and insert into socket. It's helpful to use a flat surface to align them.
- Connect USB cable.
- Open Arduino Software.
- Check under 'Tools' → 'Boards' to make sure Arduino Duemilanove is selected.
- Open the 'Blink' sketch at 'File' → 'Examples' → 'Digital' → 'Blink'.
- Press upload and watch for any errors.

Bask in the amazing blinkiness!

- If everything is correct, the test LED should be blinking on and off.
- Try playing with the code to change the blink rate.
- Look at some of the other example code.
- Try the fading example under 'File' → 'Example' → 'Analog' → Fading... You will need to change the code from LED pin 9 to LED pin 13.