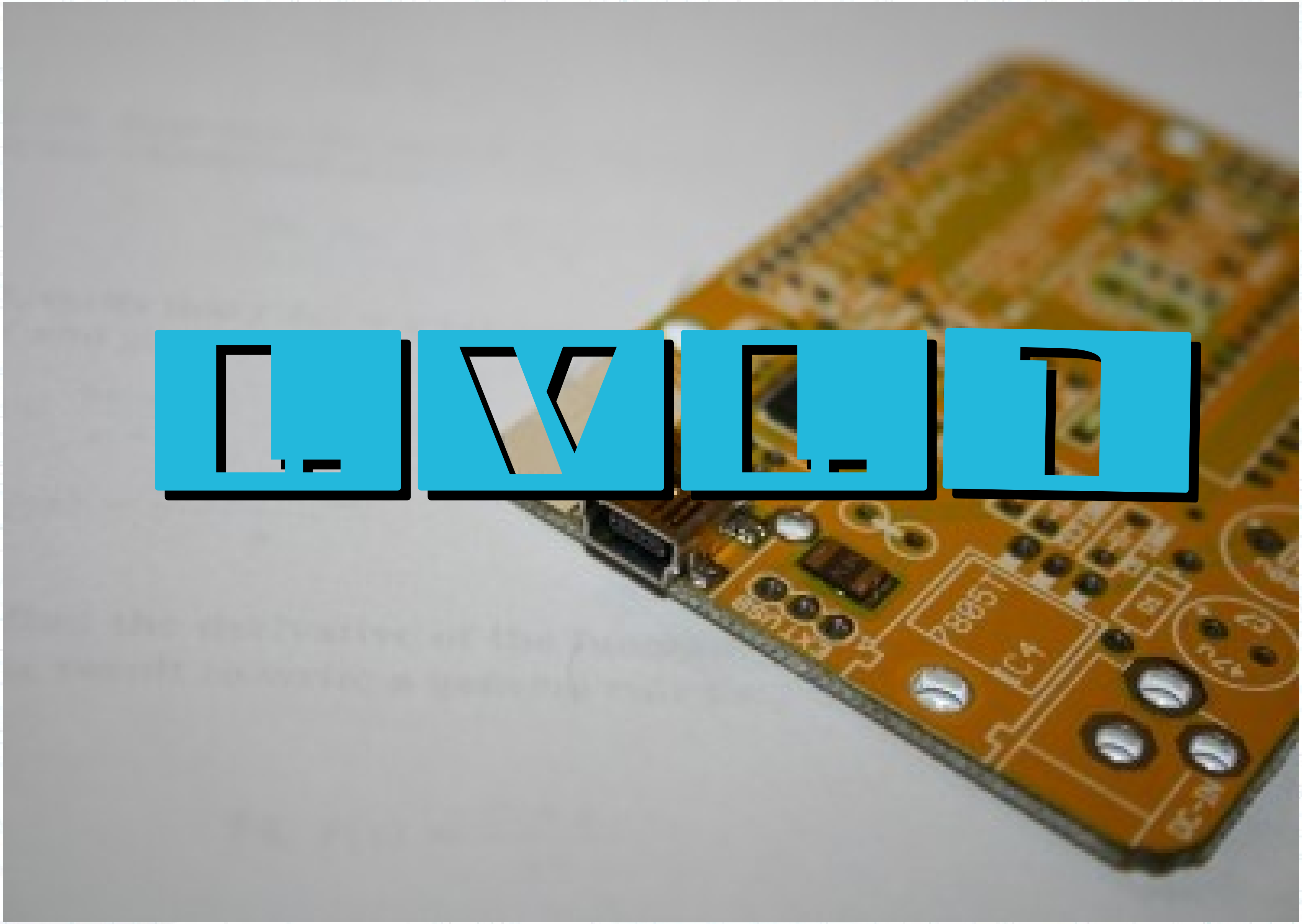
Solder Your Own Freeduino Workshop



Greetings!

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

- History of Arduino / Freeduino
- Soldering tools and technique
- Electronic component identification
- Overview functionality of the Freeduino

About Me

- My name is Chris Cprek and I'm the Micro Colonel (president) of LVL1.
- I work at the University of Louisville as an HPC Systems Adminstrator for the Cardinal Research Cluster supercomputer.
- I'm self-educated in electronics, build custom music synthesizers and play in a band called Nzambi.

About LVL1

- LVL1 is a Louisville hackerspace.
- Hackerspaces are a relatively new phenomenon in the US. They are physical spaces that serve as a community workshop or lab, where technical and artisticly minded people can share knowledge freely and work together on projects.
- We host workshops, interest groups, technical meetings, geek social events and more.

LVL1 Vision

LVL1’s mission is to provide a shared space for collaborative technical and artistic experimentation in the Louisville area. We are a non-profit organization following the hackerspace model currently flourishing in cities all over the world. Our goal is to create an open workspace where engineers, artists and educators can meet and collaborate. LVL1 is where we begin making amazing things.

LVL1 Projects and Groups

- White Star Balloon (transatlantic balloon)
- Microcontroller Mondays
- Louisville Soundbuilders
- KYOSS
- KIPCUG
- LAN Parties
- LVL1 Store (cheaper than Radioshack!)
- And more!

LVL1 – How to get involved

- Come to a Tuesday meeting at 8pm!
- Http://www.lvl1.org (blog, calendar, news)
- Http://wiki.lvl1.org (info, equipment, space)
- Join the Google Groups (discussion)
- Twitter @LVL1HackerSpace (announce)
- Become a member-keyholder for $50/mo!
- Show your support with a donation of $13.37/mo
- Donate to our makership program.

History of Arduino

- Arduino is an open source electronics prototyping platform.
- Founded in Ivrea, Italy (2005) by Massimo Banzi and David Cuartielles.
- Originally designed to be an inexpensive interaction design prototyping system for students.
- Arduino is named after a local bar.
- Arduino means ”strong friend”.

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.

Freeduino Hardware

- Based around an 8-bit Atmel ATMega328 AVR microcontroller.
  - 32KB on-chip Flash memory
  - 1KB EEPROM and 2KB SRAM
  - 5 10-bit Analog to Digital converter
  - 6 PWM outputs
  - 13 I/O pins
  - USB Serial connectivity
- Shield format for easy modularity.
- Easily programmable via USB/Serial.

Freeduino Software

- Cross-platform Arduino IDE derived from Processing
- Programming an Arduino is based on C/C++ with Wiring libraries for ease of development.
- Language Reference
- Arduino Libraries
- Interface with other software languages

Arduino Community

- Open source design has created a vast ecology of Arduino devices, Freeduino devices, shields and learning tutorials.
- Arduino playground is a community edited wiki.
- Freeduino Index of Knowledge is an encyclopedic resource of all things Arduino/Freeduino.
- Fritzing for visual prototyping circuits, schematics and PCB design.

What you need to solder

- You need the following:
  - Components to be soldered
  - Soldering iron
  - Solder
  - Good lighting
  - Clean work surface
- Optional:
  - Helping hands
  - Desoldering braid

Soldering Iron

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

www.HVWTech.com

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.

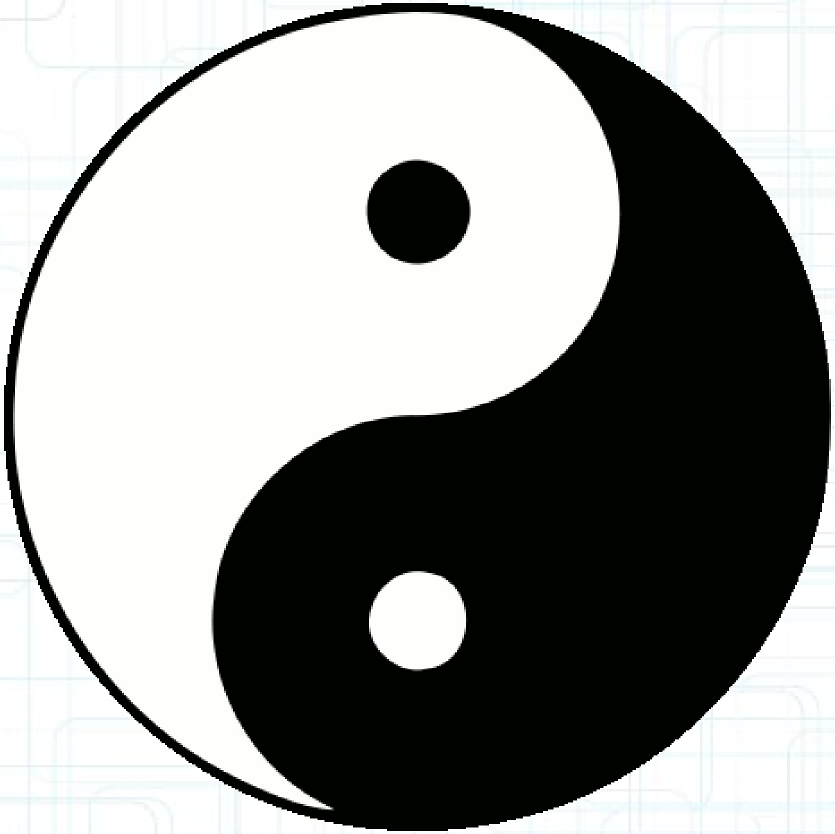
Soldering is a Manual Skill and you are not Neo

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.

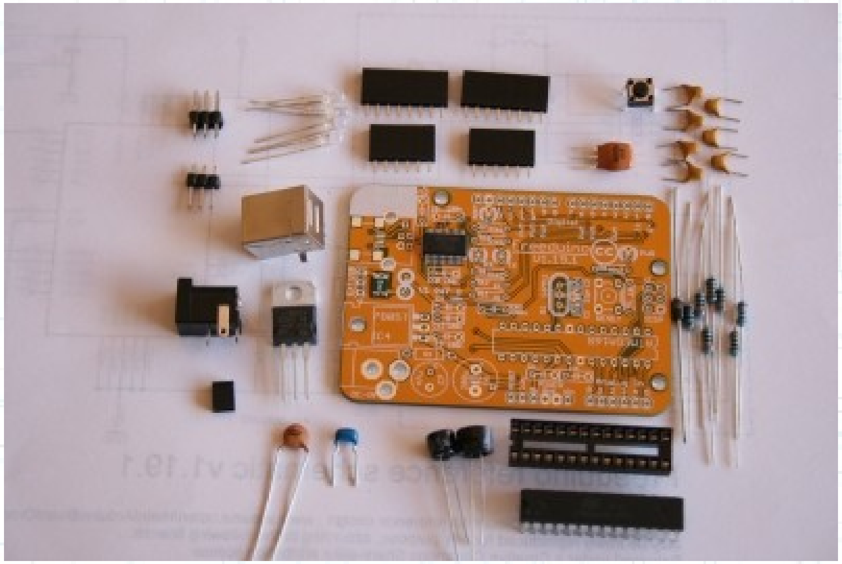


Desoldering

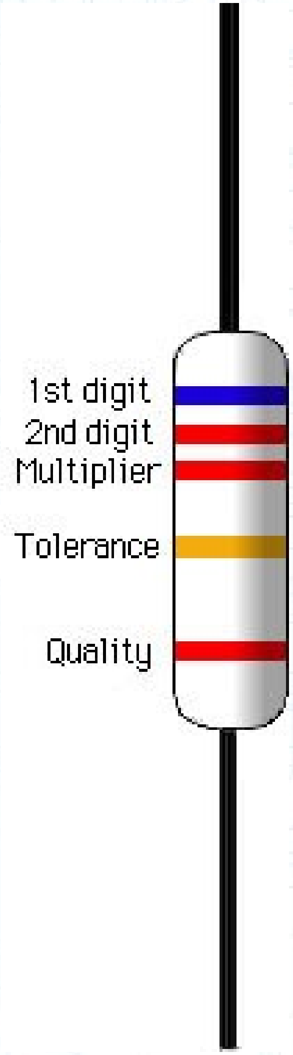
- Soldering wick does what it says.
- There are also soldering suction devices that will remove solder.

Components

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



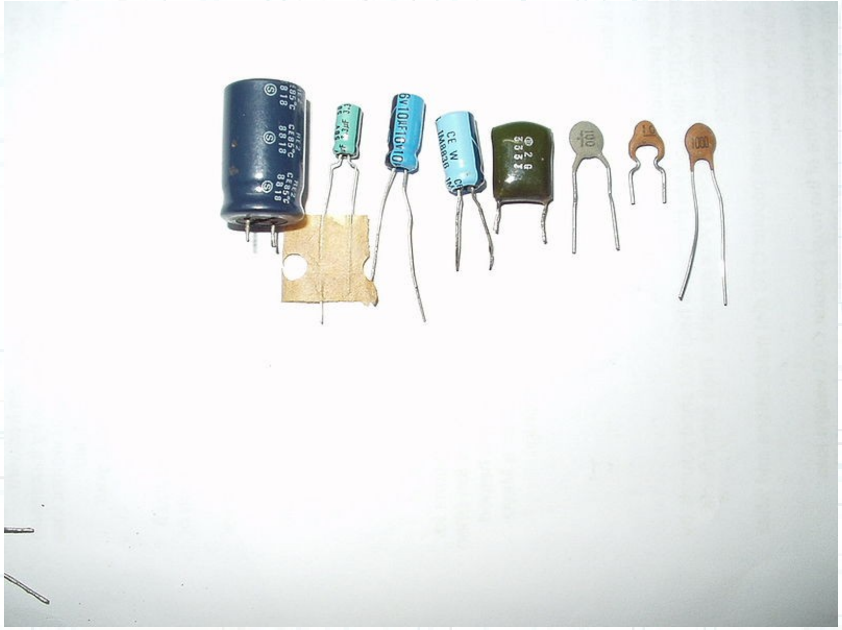
Resistors



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

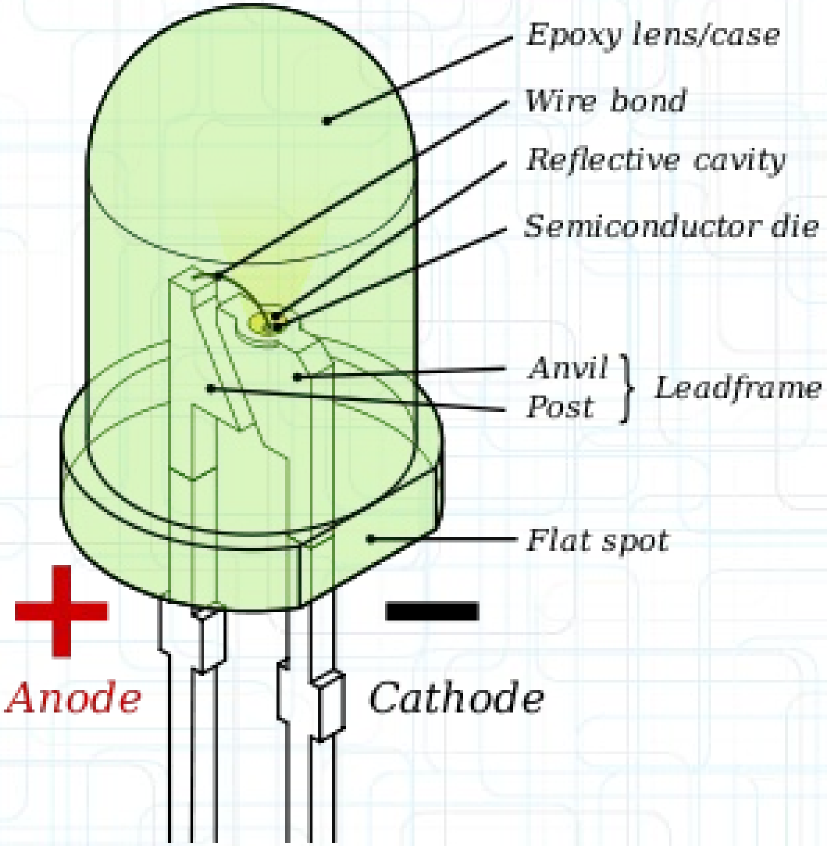
Capacitors

- 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.



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!

Stage 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.

Stage 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.

Stage 3 – Clock Components

- Q1 16Mhz – 16.000 crystal oscillator (silver oval)
- C2, C3 – 22pF ceramic capacitor (marked **22**)

These are the optional clock components. The AVR IC has an internal clock, but the ceramic oscillator provides faster and more stable timing.

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**)

Stage 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. Burn yourself

Stage 6 – ATmega328 & Blink

- Straighten Atmega328 pins and insert into socket.
- Upload the 'Blink' sketch to test