Really Bare Bones Board (Arduino) Revision B & B2 Instructions revised 8/18/09

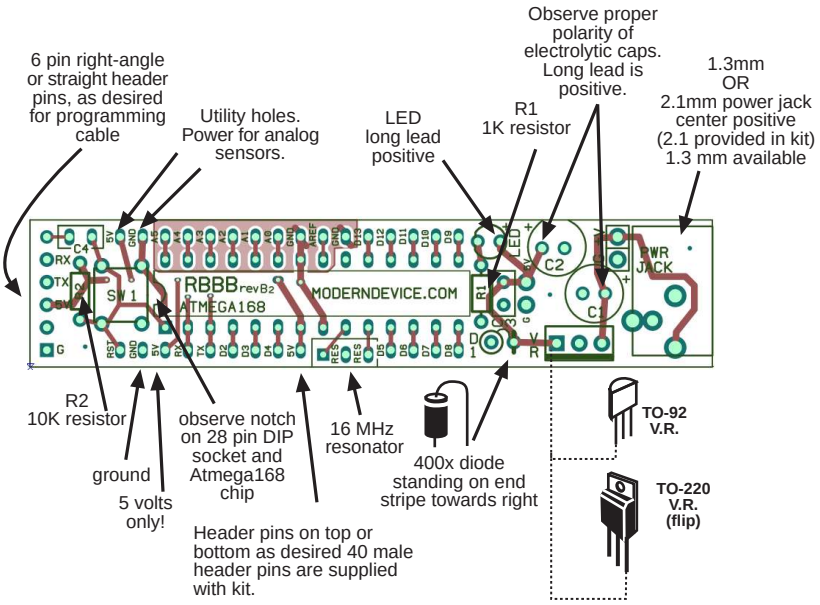


figure 1: BBB layout

Kit Parts List

Resistors

R1 1K 1/4 watt resistor (brown-black-red)
R2 10k 1/4 watt resistor for LED (brown-black-orange)

Capacitors

C1, C2 4.7-47 ufd electrolytic capacitors (polarized), 25V
C3, C4 .1ufd (104) ceramic, 25 V (not polarized)

Semiconductors

L4931CZ50LDO 5V, low-dropout voltage regulator, TO-92 package (provided in kit)
optional - LM2940 1A LDO regulator TO-220 package, LM7805 TO-220 package
D1 400x silicon diode
3mm LED
Atmega168 28 pin DIP package preprogrammed with Diecimila bootloader
16 MHZ ceramic resonator with built-in capacitors, three-terminal SIP package

Hardware

29 snap-off .100" center male header pins, or .100" female headers as desired. (40 pins in kit)
6 right-angle male-header pins .100" centers (see text)
1 momentary switch
1 28 pin (narrow .3") IC socket OR 2, 14 pin DIP sockets

Programming Connection

FTDI TTL-232R USB programming cable FTDIchip.com
or Modern Device P3/4 serial port programming adapter

Auxillary Power

5-12 Volt - 1A power adapter (optional)
Digikey T980-P5P-ND 6V/1A switching supply
9 volt battery, 4 AA batteries, 2 lithium button cells etc.

The Really Bare-Bones Board has been engineered to be a small, easy-to-construct Arduino-compatible (Freeduino), specially aimed at students and prototypers. The board can be plugged into a breadboard, or can be embedded in small or large projects such as toys, small appliances and wearable-computing projects.

The Really Bare-Bones Board includes all of the functionality of all other Arduino boards, except for the physical limitations of its form factor.

The boards can be built in a half hour by a beginner or less by someone who has built a few of them before.

Only a low-wattage soldering iron (15 - 25 Watts), diagonal cutters, and solder are required.

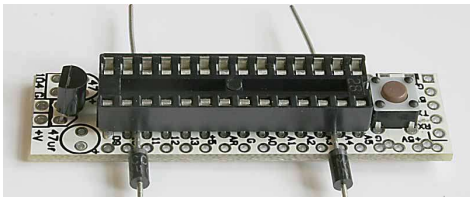
Assembly Instructions

For beginners that have never done electrical soldering before see the Soldering section before starting. There are a few tips for multiple-unit "power builders" too.

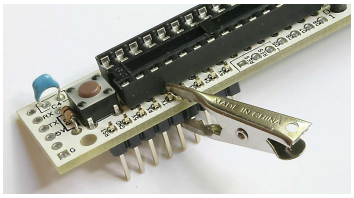
Start assembling the board with the smallest, lowest profile components first. When parts are inserted, and the board is flipped over to solder, components will tend to stay in the board better, if similar, low-profile components are inserted first.

- Solder in the resistors, and small .1 (104) ceramic capacitors first. The resistors and smaller capacitors have no polarity and may be inserted in either direction.
- Next add the LED, reset switch, and voltage regulator. Make certain that polarized components, such as LED, diode, and voltage regulator are inserted in the correct orientation. The long lead on the LED is positive. Study the drawings and photos to get the v.r. oriented correctly. The diode mounts upright with the stripe oriented toward the hole on the right.
- Mount the two 47 uF electrolytic capacitors, carefully observing the polarities and orientation of each. C1 is mounted with positive lead (long lead) on the right, as shown. C2 is mounted with positive lead on the left.
- Insert the IC socket (not the chip) into the board, taking care to align the notch with the notch on the board silkscreen.
- On the older rev. B version of the board the labels were more legible is the socket was spaced off the board a mm or so with some diode leads under the socket. The newer rev. B2 board has labels between the pins which makes this unnecessary.

For larger and multi-pin components, such as the chip socket and header pins, there is a little trick that may be helpful to get them mounted neatly.
Solder in only the two diagonal corner pins. Then flip the board over to inspect it. If the component is not mounted tightly down on the board, simply put a little pressure on the component with your index finger while reheating the soldered pad(s) with the soldering iron, this will get the part



Spacing the socket off the board with diode leads helped to increase the label visibility on Rev B and earlier boards. The rev B2 board has smaller labels between the pins, so this isn't required.



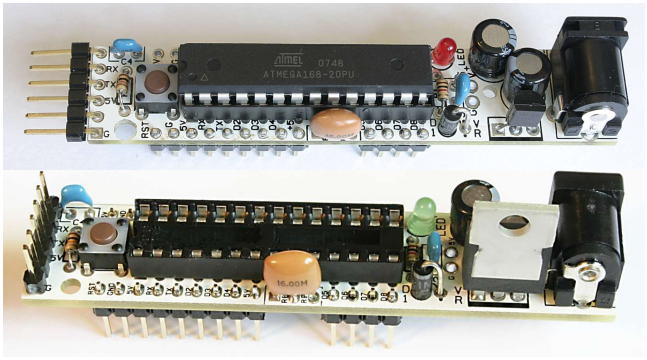
The small size Radio Shack alligator clips, with the noses pinched down, turn out to be ideal clamps to hold header pins for soldering.

mounted down flush before you solder in the other pins.

- Solder in the header pins. The board is made to be easy to customize for particular applications. The following instructions are for the most standard orientation for header pins, but feel free to mount (or omit them) as you wish. In some situations it may be more robust and reliable to solder wires directly to the board.
- For the side of the board with the resonator, snap off a nine pin section of header pins and a four pin section, mount them on the bottom of the board and solder them on the top For multiple kit builders, some small alligator clips, available at Radio Shack, are handy for holding header pins on while soldering. For a single kit just use the tip on the previous page, soldering in the end pins first, and checking the alignment.
- For the side of the board without the resonator, snap off a 16 pin section of header pins, mount them on the bottom of the board, and solder them on the top.
- Mount the 6 pin programming header. Feel free to use either the right-angle headers or straight headers, depending on how you wish the programming cable to be situated.
- Mount the resonator, it's not polarized but it's a good idea to mount it with the label showing.
- Solder in the 1.3 or 2.1mm (provided) power jack, if you plan to use it .

Clean The Board

- Hobbyists and prototypers often omit this step, but it's a good idea to clean the solder flux off your board with a toothbrush and isopropyl (not denatured) alcohol. Scrub and rinse with clean alcohol until the board looks flux-free. Dry it off a little with a paper towel or rag and have a good inspection of your solder joints. Make sure pins are not bridged (shorted) with solder, and that all the solder joints look shiny, smooth, and solder covers the pads completely. Questionable solder joints may be fixed simply by reheating with the soldering iron. Bridged pins may be cleared by using desoldering wick, available at Radio Shack.



The top image shows a rev. B RBBB with standard TO-92 regulator and right angle programming headers. The bottom image shows the current revB2 RBBB with an optional TO-220, LM2940, 1 amp regulator, and vertical header pins for the programming cable. The chip has yet to be inserted into the socket.

- After cleaning and inspecting the board, prepare to insert the chip. The pins on integrated circuits (IC's) come from the factory slightly splayed, so you will fist have to put the chip legs down on the bench and push on the body of the chip to bend each set of legs perpendicular with the top of the chip. Very slightly bowed in is even better than perpendicular.

  Matching the notch on the chip with the notch on the circuit board legend (especially check the board legend as it is possible the socket is inserted incorrectly - this won't hurt anything if the chip is always inserted to match the board, not the socket.)

Testing your RBBB rev B

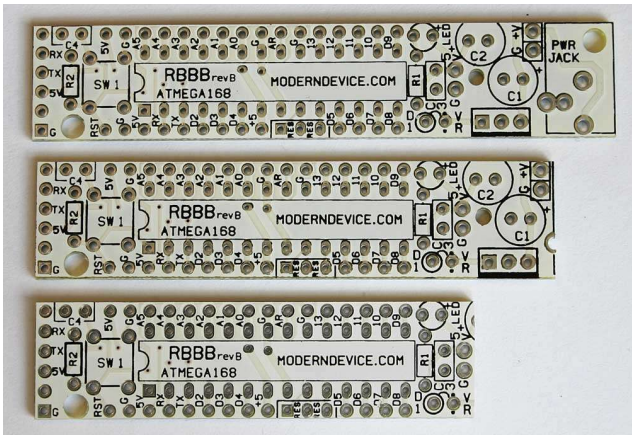
- Push the completed RBBB into a solderless breadboard.
- Put the long lead of an LED into pin 13 and the short end into a free row. Put a 1K resistor into the same free row with the other end going to ground, so that the LED and resistor are in series, with pin 13 on one end, and ground on the other.
- Hook up an USB-BUB or FTDI USB-TTL serial cable. Alternately you may use the P4 programming adapter from a host computer's RS232 serial port, and a power supply.
- Check to see if the pilot light comes on immediately. If not disconnect the cable or power supply right away and check to see if the LED and D1 are both inserted incorrectly.
- If the diode is correctly inserted, set your multimeter on an appropriate voltage range and check for 5 volts at any of the 5 volt and ground pins. (Except the holes near the power jack - they will show the input voltage if you are using batteries or a wall adapter.)
- Boot up the Arduino application.
- Check the Tools->Serial Port menu item in Arduino, to make sure you have the correct serial port selected.

• Try uploading the sketch "Sketchbook->Examples-> Digital->Blink. Don't push the reset button if using the Diecimila bootloader. The Arduino IDE should reset the board and upload the sketch automatically.

Windows users, listen up!

If you are using a FTDI cable, as oposed to a USB-BUB or P4 programming adapter, there is a serial port setting that you will need to adjust:
Device Manager -> Comm Ports -> USB Serial Port -> Port Settings -> Advanced button -> Set RTS On Close.

• If your board doesn't seem to work, see the troubleshooting guide on page 6.



The RBB board approximately life size and two customizations of the board to make it smaller. Size specs are below.

Full board	.6" x 3.0"
Without power jack	.6" x 2.65"
Without power jack and regulator	.6" x 2.25"

Possible board modifications

• The board can be shortened, as shown above, in several different manners. If you don't need the power jack, just snip off the end of the board as in the middle image.

• If you don't need the power jack or regulator just snip off both outlines. The easiest way to cut the board is with a large pair of tin snips, but hack saws, jewelers saws and band saws will also work well.

• Note that an LED and a 1k resistor is supplied as a pilot light in the kit. If you are trying to build as low-powered a device as possible, you may wish to use omit the pilot light, or blink an unused pin on the RBBB instead of the steady current draw of a pilot light.

• Along the same lines, with low-powered design you may wish to omit the voltage regulator and use three or four AA batteries, or two lithium coin cells. The Atmega chip will tolerate 6 volts, and run on as low as 2.5 volts.

Powering the RBBB Arduino.

Because the RBBB was designed to be small and simple, the power system is also simple. There are several options for powering your RBBB.

Option 1) Power from the USB cable.

The Modern Device USB BUB or FTDI RS232 to TTL cable is a convenient way to power your RBBB from a laptop or desktop computer. The USB port can provide 500 mA of current to the RBBB or other circuits.

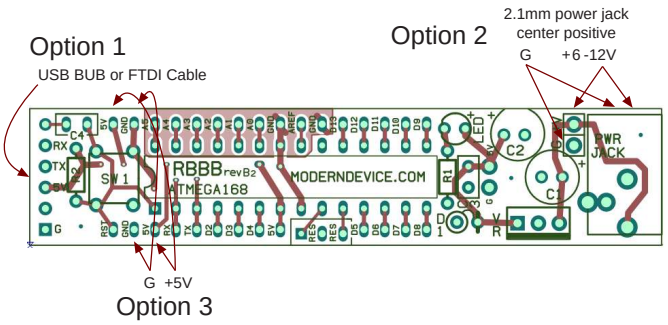
It is not recommend to plug in both a USB adapter and a wall adapter at the same time, because both supplies will be shorted together. We've done this though with several projects while programming the board, and have seen no problems with this arrangement. The adapter and the voltage regulator seem to play together well. Again, this is not recommended for as a long–term power supply scheme however.

Option 2) Power from a wall adapter or batteries supplied to the power jack.

5 to 12 volts may be supplied to these power jack, which connect to the input of the voltage regulator. A convenient battery powered option would be a 9 volt battery connected to these inputs, or 4 AA batteries, or 2 lithium cells. These could be connected either through the 2.1mm power jack, or through wires soldered into the holes adjacent to the power jack.

Option 3) Power (5 volts) supplied from any other source to any of the 5 volt pins on the board.

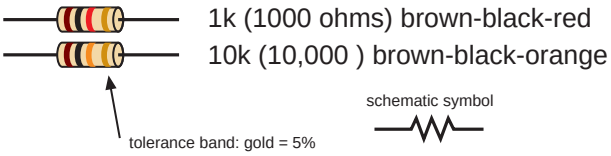
If you have some kind of 5 volt power on your breadboard, you can power the RBBB from the 5 volt breadboard rails. These pads and the ones on the top left of the board are connected electrically to the output of the voltage regulator. This means that the voltage regulator is not really in the circuit and will not reduce a voltage greater than 6 volts, which will, incidentally, damage the Atmega chip.



Powering options for the RBBB

Resistors

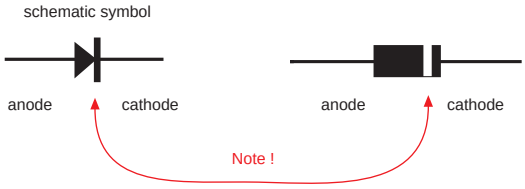
Resistors are components that limit the flow of current in a circuit. On the RBBB, R1 is used to limit the flow of current in the pilot light LED. R2 is a pullup resistor that holds the reset line high, under normal conditions.



resistors are not polarized, meaning it doesn't matter which way they are inserted into the printed circuit board

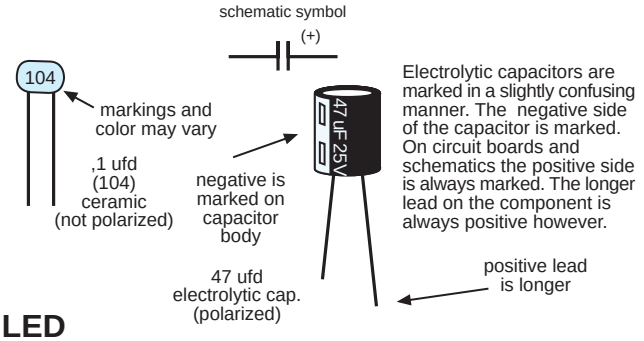
Diode

A diode is an electrical one-way valve. In the RBBB the diode provides reverse polarity protection on the 5 volt power rail. Under normal circumstances the diode does nothing. If the power is connected with polarity reversed, the diode shorts out the power supply. This protects the Atmega 168 from exposure to the reverse polarity voltage which would damage it.



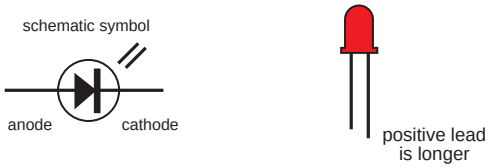
Capacitors

Capacitors are components that store electrical energy (charge). There are several different technologies that are used to construct capacitors. The BBB contains ceramic capacitors, that are not polarized, and electrolytic capacitors, that are polarized, and must be inserted with the correct orientation.



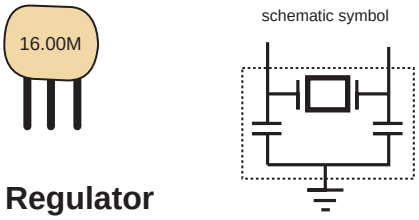
LED

LED's are diodes which emit light. They are polarized like diodes so insert them in the correct direction.



Resonator

The 16 MHz ceramic resonator acts as a calibrated oscillator for the BBB. As you can see from the schematic, it contains a crystal element and two small capacitors. It's symmetrical, so you can't put it in backwards, but it is a good idea to put it in so that you can read the label.

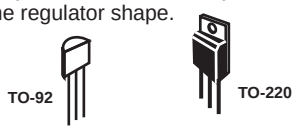


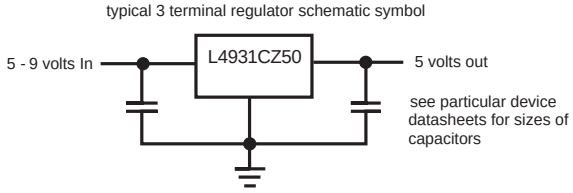
Voltage Regulator

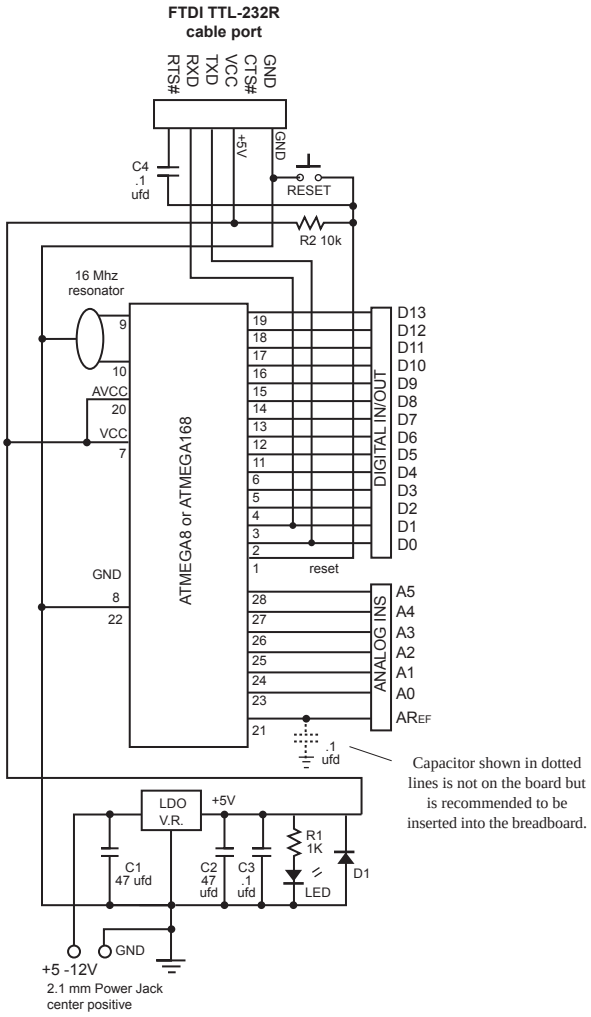
The voltage regulator is an integrated circuit which will limit higher input voltages to 5 volts output. It will also limit the current flow in case of short circuits. The electronics industry calls the physical form an IC is packaged in a "package" or "case". The actual IC is always a small chip embedded somewhere in the plastic. The voltage regulator provided may vary by number but is in a TO-92 case.

If you need to have more regulated power, to power a lot of high-powered LED's for example, your board will accommodate a TO-220 package regulator, such as the LM2940 listed in the parts list, or the popular 7805. Just remember to insert it backwards as shown in figure 1.

Make sure you get the TO-92 regulator inserted in the correct orientation. It is **not** symmetrical so carefully match the part outline on the board with the regulator shape.

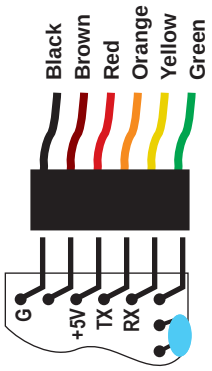






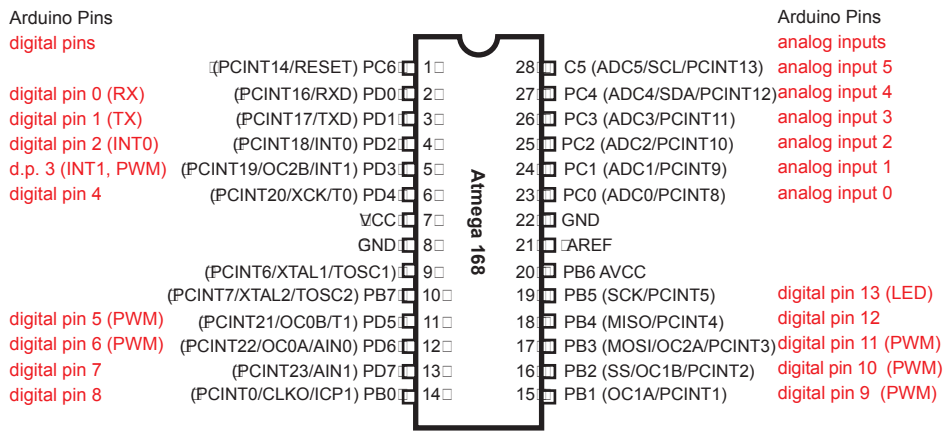
Real Bare-Bones Board / Board Schematic
RBBB
Arduino® - Compatible

RBBA/B Schematic
Unlike some other Modern Device products, this schematic and board design are in the Public Domain.
Paul Badger 10/2007



Programming cable connections between a RBBB and a FTDI TTL-232R USB to TTL serial cable.

Please note that the TX and RX labels refer to the cable's labeling, not the chip's. The TX label at the programming connector is electrically connected to the RBBB's RX pin (digital pin 0) and the cable's RX pin is connected to the RBBB's TX pin (digital pin 1).



Pin mapping of the Atmega168 chip to the Real Bare Bones Board

Troubleshooting

Symptom: No pilot light LED

No power connected

Polarity of wall transformer wrong. Should be center positive.

Cable connected backwards.

Bad solder joint - reheat suspicious solder joints

Procedure: check for 5 volts at power buses: at USB port, at power jack, near analog pins, at pins 7&8 of the Atmega168

If 5V is found at power bus pins, LED may be in backwards or poor solder joint. If low or incorrect voltage, check voltage regulator, solder joints, power supply

Symptom: Pilot light on but pin 13 LED doesn't flash.

Connect an LED and 1k series resistor to pin 13. The anode (LED long lead) should connect with the RBBB pin 13, the shorter LED lead should connect with the resistor, with the other resistor lead going to ground.

If the chip is programmed with a Diecimila bootloader, pin 13 should flash at about 1 Hz.

Causes:

pin 13 LED in backwards

Atmega168 in backwards

electrolytic capacitor in backwards

voltage regulator in backwards

bad solder joint, reheat some or all solder joints

Atmega168 in backwards or not seated properly (check for pins that have "escaped the socket"

Atmega168 not programmed with bootloader

resonator wrong value or missing

Symptom: Pilot light on, pin 13 flashes, but program won't upload to board

Hardware Causes:

cable not connected

problem with P4 programmer (if using P4)

FTDI drivers not installed: check Arduino->Tools->Serial Port

on PC - (Comm port higher than 2 should show - choose highest comm port)

on Mac - Check for driver with "FT" in title

bad solder joint at cable connector or chip pins 2 & 3 (check for shorts or bad (solder joints) on pins, reheat all solder joints

wrong resonator value

Procedure: if you have an oscilloscope, check for signals across resonator pins and on RX pin during download.

Software / PC side causes:

check for FTDI drivers installed (if using USB cable)

check for proper board (Atmega168) selected in Arduino->Tools->Microcontroller->Atmega168

If using Windows, the FTDI cable requires this serial port settings to be adjusted.

Device Manager -> Comm Ports -> USB Serial Port - Port Settings -> Advanced button -> Set RTS On Close

Don't push the reset button on upload, this should happen automatically.

The presence of FTDI drivers can be seen in Device Manager by checking to see if "USB Serial Port" appears under the "Comm Ports" heading. Drivers may be downloaded at FTDIchip.com and also come with the Arduino install package.

General "Cure-Alls" and troubleshooting:

Check orientation on all polarized parts, voltage regulator, caps, diodes, V.R., socket and chip.

Reheat all solder pads on bottom of board, look for bridges (shorts) on chip pins, or bad solder joints. Good solder joints should be shiny, smooth and cover the pad completely.

Clean PCB with toothbrush and isopropyl alcohol

Swap cables, power supplies if possible, to troubleshoot

Use your meter to check voltages on the board and on the chip

Arduino is an open-source hardware and software initiative closely related to the Wiring and Processing open-source initiatives.

Arduino Home - http://arduino.cc

Wiring Home - http://wiring.org.co/

Processing Home - http://processing.org/

The Really Bare-Bones Board is an open-source hardware project of Paul Badger and Modern Device, moderndevice.com

Unlike other Modern Device Designs, the printed circuit board designs and schematics for the Really Bare-Bones Board are in the public domain. We hope you will use them only for good things.

We also hope you will contribute code, ideas and energy to the Arduino project at arduino.cc. Freeduino.org is another site where alternative Arduino/Freeduino development is taking place.

Electrical Soldering for Beginners

Equipment

Use a high-quality soldering iron with the sharpest point you can find. It should be rated between 15 and 25 watts.

Keep the soldering iron tinned (coated with solder) at all times. The tip should look slivery and shiny. On a new soldering iron, it is important to tin the tip as soon as the soldering iron gets hot, the first time it is used.

Wipe your soldering iron tip off on a wet sponge, or a copper "scrubbie", to keep it clean and shiny. Do this whenever the tip stops looking shiny or has too much solder buildup on it.

Use either leaded or "no-lead" solder but be aware no-lead solder is a little harder to use for beginners, and makes solder joints that are slightly less shiny than leaded solder. That being said, no-lead solder is a good idea. Radio Shack sells small quantity packages.

Work in a room with some ventilation. There is a tiny bit of lead in solder fumes but the flux fumes are more of a health-hazard than the lead. Jameco sells a nice soldering iron / carbon filter combination for under $100.

Procedure

Heat the pad for about a second, then apply solder to the heated pad or leads, not the soldering iron. After the solder melts and "grabs" the pad, continue heating for another second.

If you haven't gotten the solder to grab after about 4 seconds let the joint cool down before trying again. Too much heat can ruin electronic components, but most beginners err on the side of too little heat ("cold" solder joints). If the solder joint looks lumpy, or if the solder doesn't completely cover the pad, the solder joint needs more heat. Just reheat it again until you see the flux around it "simmer" a bit, and the solder grabs the pad and smoothes out.

Mistakes

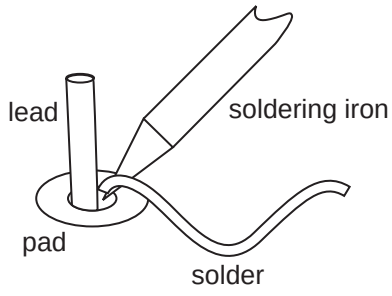
Mistakes in placement of components can be fixed with one of several methods. If the part to be removed is not expensive the easiest thing to do is to cut off the part leads at the board level. Solder can then be removed from the breadboard holes using desoldering braid.

Another useful desoldering technique is to heat the plugged hole/pad up with a soldering iron and the slap the board sharply against a workbench. The idea is to accelerate the liquid solder out of the hole.

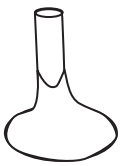
Cleaning

It is a good idea to clean the solder flux off your board with a toothbrush and isopropyl (not denatured) alcohol, when you have assembled your board. Most fluxes, when left on the board, will corrode pcb pads and traces. Additionally, fluxes are not perfect insulators, so can affect the electrical operation of your circuit.

Splash a little alcohol on the board and scrub with a toothbrush. Rinse with clean alcohol and repeat until the board looks, and feels, flux-free. Dry it off a little with a paper towel or rag and have a good inspection of your solder joints, to make sure pins are not bridged (shorted) with solder, and that all the solder joints look shiny, smooth, and cover the pads completely. Questionable solder joints may be fixed simply by reheating with the soldering iron.



cold solder joint
(not enough heat)



good solder joint

- smooth meniscus
- shiny
- covers pad

Document Revision History:

RBBB_Instructions_06.pdf 8/18/09

Fixed several small mistakes, including inserting the power jack. Thanks to Andrew Rapp.

RBBB_Instructions_05.pdf 1/15/08

lots or revisions, new photos, overhauled troubleshooting section, clarified power options. Added materials for Rev B, reset cap, voltage regulator, pilot light, power jack

RBBB_Instructions_04.pdf

fixed pictures and text for "Diecimila-compatible hack". Text used to indicate that cap be connected incorrectly