Bare Bones Board Arduino Assembly Instructions, Rev. C

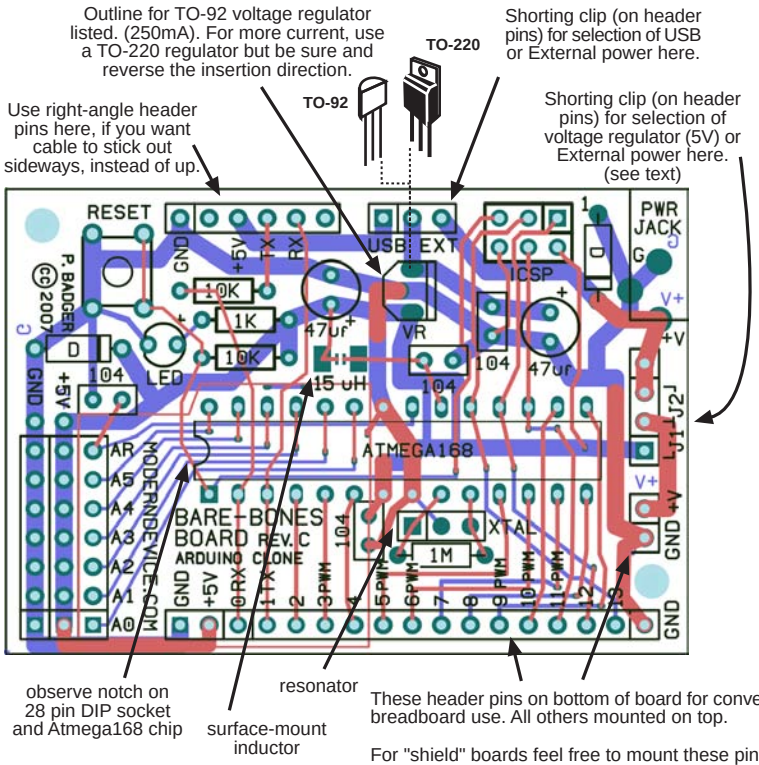


figure 1: BBB layout

Parts List

Inductor

1 15 uH surface-mount inductor

Axial 1/4W Resistors

2 10k (brown - black - orange)
1 1M (brown - black - green)
1 1k (brown - black - red)

Capacitors

4 .1ufd (104) ceramic, 25 V
2 4.7-47 ufd tantalum or electrolytic capacitors (polarized), 25V

Semiconductors

2 400x 1amp power diodes
1 L4931CZ50LDO 5V, low-dropout voltage regulator, TO-92 package
optional - LM2937 500 mA LDO regulator TO-220 package
1 3mm (T1) LED any color
1 Atmega168 28 pin DIP package preprogrammed with bootloader
1 16 MHZ ceramic resonator with built-in capacitors, three-terminal SIP package

Hardware

50 snap-off male header pins .100" centers, or .100" femail headers as desired.
6 right-angle male-header pins .100" centers
1 momentary switch
1 28 pin (narrow .3") IC socket OR 2, 14 pin DIP sockets
2 .100" header 2 position shorting clip (shunt)
1 1.3mm power jack, Digikey CP-2519-ND

FTDI TTL-232R programming cable FTDIchip.com
5 Volt - 1A power adapter (optional)
All Electronics CAT# PS-504

The Bare-Bones Board has been engineered to be a low-cost easy-to-construct Arduino, specially aimed at students and prototypers. The board plugs into a breadboard, and has several convenient features that make it easy to use for both beginning and advanced prototyping.

Despite its title, the Bare-Bones Board includes all of the functionality of the most robust Arduino boards, and even some recommended analog noise-reducing components not found on any of the Arduino reference designs.

The boards and kits, which have been engineered to be as flexible as possible for widely varied types of prototype designs, can be built in an hour by a beginner at soldering, or in a half-hour by someone who has built a few of them before.

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

Assembly Instructions

Start assembling the board with the smallest, lowest profile components first. That way taller components will not get in the way when trying to install smaller components. Also 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.

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.

- The first component to deal with is the smallest, the 15 uH surface-mount inductor, marked "150". This inductor provides a small amount of analog (A/D) noise reduction, it is recommended in the Atmega168 datasheet, but is not included in other Arduino implementations. If you do not wish to deal with the small surface mount component you may just solder up (short circuit) the surface mount pads where it was to be mounted.

  If you do decide to mount the surface-mount inductor follow the instructions on page 4.

- Solder in the resistors, diodes, small ceramic capacitors, LED, reset switch, and voltage regulator next.
- The resistors and smaller capacitors have no polarity and may be inserted in either direction.
- Make certain that polarized components, such as diodes, LED, voltage regulator, and larger electrolytic caps are inserted in the correct direction. Note that the long lead on the LED is positive.
- Mount the two 47 uF electrolytic capacitors, carefully observing the polarities and orientation of each. The long leads on the capacitors are positive.
- Insert the IC socket (not the chip) into the board, taking care to align the notch with the notch on the board silkscreen.

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 one pin only, or in the case of the socket, 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 mounted down flush before you solder in the other pins.

- Solder in the power jack.
- 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 installations it may be more robust and reliable to solder wires directly to the board.
- The seventeen digital output header pins gets inserted into the bottom of the board and soldered on the top, as do the two pins, at front right, that are designed to power the breadboard. Mount these pins last after the others are mounted on top.
- All other header pins get inserted into the top of the board and soldered on the bottom. Using right-angle header pins for the USB-to-Serial cable connector makes the cable convenient to connect.
- Finally, mount the resonator (marked XTAL on board). It can be inserted either way but it's a good idea to insert it with the label showing.
- 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, 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.
- Add the shorting clip to the power selection header in the desired position, and the J1/J2 shunt (see below).

Testing your BB Arduino

- Put the USB / EXT shorting clip on the power selector pins. Power up the board, either with the programming cable or an external supply and make sure that the pilot light goes on. If not, disconnect the power right away, and consult the troubleshooting section. Do a check with a multimeter for 5 volts at the power pins to the left end of the digital breadboard pins.
- If all seems well, disconnect the power, and insert your Atmega 168 chip into the socket, taking care to align the notch on the chip with the notch on the socket and on the board silkscreen. Chips come from the factory with the legs splayed, and the chip will be much less fuss to insert if you perform the following operation. Hold the chip exactly perpendicular to a table and press down until the all the legs have a 90 degree angle to the chip top. Flip, repeat, and you're ready to put the chip into the socket.
- Next hook up a LED and a 1K series resistor on pin 13. Boot up the Arduino application and try downloading the blinking light program. Push the reset button on the board and click the download icon at about the same time.
- If your board doesn't seem to work, see the troubleshooting guide on page 6.

Powering the BBB Arduino.

There are two options for powering the Arduino Board. The USB cable can supply 5 volts to the board. How much current the FTDI 232R cable can provide is an open question. The USB standard calls for available current to be controlled by software and the cable's manufacturer quotes 50 mA as the amount of available current. (This limit is from the expected behavior of the USB interface and not any electrical limit from the cable.)

In practice, a Mac and a PC I tested were both able to provide 500 mA, which is as high as I went in my testing. Since I can't get access to any of the software interaction, it is impossible for me to know if the cable's software "asks" for higher current limits or the USB ports just generously provides more without being asked. I would guess the latter.

USB is a convenient option for powering your board because it eliminates other wires and batteries and it should be viable for many projects that only involve interfacing sensors, lighting LED's, or communicating with a laptop/desktop computer.

For circuit experiments and construction you are probably better off using a 5 - 9 volt power adapter. This will take the strain off your laptop battery and protect more expensive systems in the case of short circuits and the like. For powering small DC motors or solenoids especially, you will be much better off using an external power supply.

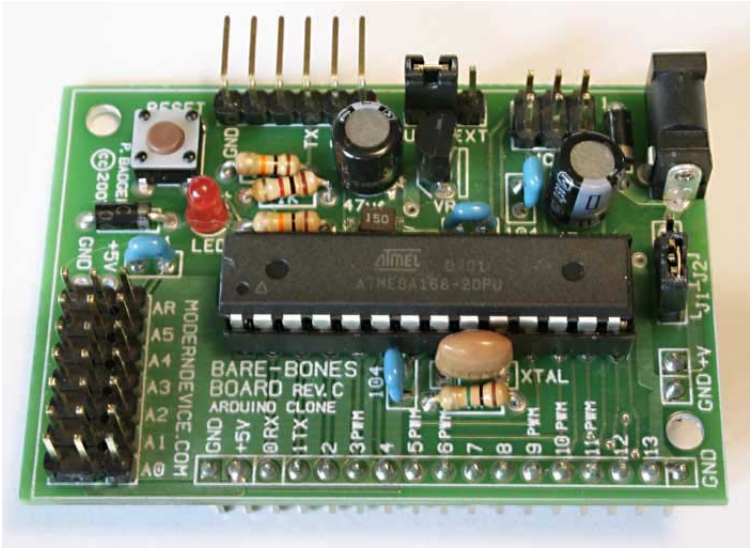
The board contains a 5 volt low-dropout regulator. In the case of an accidental short circuit, the voltage regulator on the board will limit the current draw to about 300 mA. This should protect the power supply although the regulator will get very hot. The tipoff to a short circuit will be the LED pilot light going off, and of course, a hot regulator, if you put your finger on it.

J1 / J2 Shunt or jumper

J1 or J2 is an option that sets how the BB Arduino is connected to the two breadboard bus power pins on the right side of the board.

Use J1 in these cases:

J1 jumper is used to power for the breadboard power-rail bus comes through the Arduino LDO voltage regulator. Use this if you want to power your breadboard from your Arduino adapter, and the adapter is between 5-9 volts. Most users will probably want this option, unless you have DC motors running on the breadboard.



Bare-Bones Arduino module with header pins set up for breadboard use. There are several options for the header pins depending on the project requirements.

Use J2 in these cases:

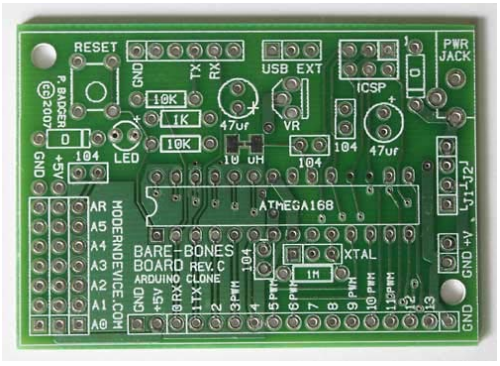
J2 is used to connect power to (or from) the breadboard power-rail bus, directly from the Arduino power jack. In most cases you probably don't want a higher voltage than 5 volts on the breadboard power rail so use this option only with a regulated 5 V adapter such as the All Electronics unit in the parts list.

J2 can also be used if you have a powered breadboard, with regulated 5 volts available, and you want to run your BB Arduino from the breadboard rails. It is also possible to use J1 when powering the BBB off 5V breadboard rails - in that case, the regulator would not be in the circuit.

jumpered voltage regulator

Options, Parts You Perhaps Don't Need

The Bare-Bones Arduino was engineered to be a small, versatile prototyping board, so depending on your circumstances, there may be several parts you can do without.



A BBB pcb board shown actual size, for comparison purposes. The production run boards however, are white, not green.

If your BB Arduino is tethered to a laptop or desktop computer as an IO device you may just want to run off USB. In this case the low-dropout regulator is redundant and you could just solder a jumper from one outside pin to the other.

If you are powering the board from a 5 volt, regulated adapter, such as the All Electronics model listed on the parts list, you could also leave out the regulator and one of the electrolytic capacitors that are associated with the regulator.

The pilot light and its associated resistor is a useful feedback signal, but if you are building a battery powered device which requires minimal power draw, for example, leave out the LED and add an LED (with series resistor) connected to an Arduino digital pin and blink the LED every 3 or four seconds, to save power.

Any of the header pins can be left out, or soldered on the top or bottom of the board, or can be replaced by female headers, (for mating with shield boards, for example). One flexible option might be to use longer, male, header pins that protrude both above, and below, the board. This would make it possible to use your BB Arduino with either a breadboard, or a shield board. See Digikey part # S1031E-36-ND for example.

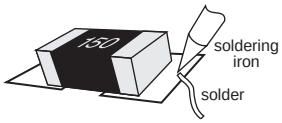
You could even leave out the 16 Mhz resonator, and program the Atmega168 chip to run on the internal RC oscillator, at 8 Mhz. This would require reprogramming the Atmega168 fuses. This is not rocket science, but not for those impatient with trial & error, in my experience, see http://www.arduino.cc/playground/Learning/Atmega83-3V.

Please let us know if there are features you think would be handy or should be built into the next version. Corrections and suggestions for this documentation are also highly valued and appreciated. Most will be implemented immediately.

Inductor

schematic symbol

Inductors (coils) act to oppose a change in current. In the BBB circuit the inductor's role is to reduce fast-changing power-supply noise. The 15 mh (millihenry) inductor is the only surface-mount component on the board. Here's how to mount it.



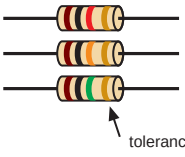
Place the inductor on the pad, hold it down by laying an Exacto knife or needlenose pliers on top of it, and heat the pad, (do not touch the inductor with the soldeing iron). Solder will rapidly flow onto the pad and inductor, lift the soldering iron immediately. This will hold the inductor down, touch the other pad for a second with some solder and your soldering iron, and you're almost done. Just reheat the first side quickly to insure a good solder joint.

Don't worry if the inductor is not on perfectly straight, the electricity can't tell the difference. Don't overheat it either, it's small and will solder quickly. If it's really crooked, you can position it by quickly and alternately heating opposite ends and pushing gently with the soldering

solder pool to short pads if not using inductor

If you don't want to fuss with the tiny inductor, just leave it out and solder up the pads as shown above. No other Arduino boards include it although the Atmega 168 datasheet recommends it, and the reduction in analog noise that it provides is fairly minor.

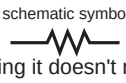
Resistors



1k (1000 ohms) brown-black-red
10k (10,000 Ω) brown-black-orange
1M (1,000,000 Ω) brown-black-green

tolerance band: gold = 5%

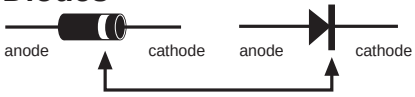
schematic symbol



resistors are not polarized, meaning it doesn't matter which end goes where

Diodes

schematic symbol

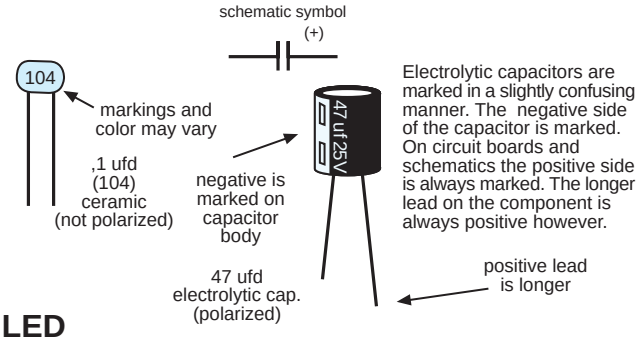


Diodes act as electrical "one-way" valves. Electricity flows in the direction of the arrow, but not the other way. They are polarized. Stripe is negative (cathode end). Think of the stripe on the diode as the stripe in the schematic. Get the stripe oriented the correct way on the board, or the project is almost guaranteed not to work.

Prepare all the resistors and diodes for inserting into pcb by bending their leads at right angles, adjacent to the component body.

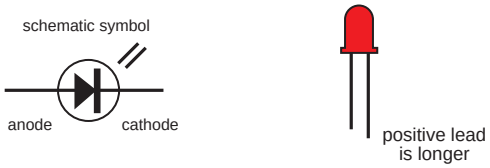
Capacitors

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.



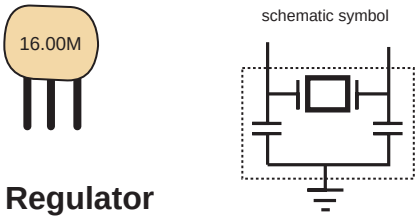
LED

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.

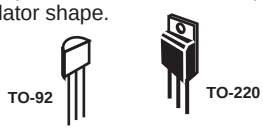


Voltage Regulator

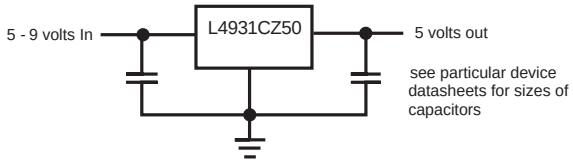
The voltage regulator is an integrated circuit which will limit higher input voltages to 5 volts. It will also limit the current flow in case of short circuits. The electronics industry calls the physical form an IC is packaged in a "package" or "case", the actual IC is always a small chip embedded somewhere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

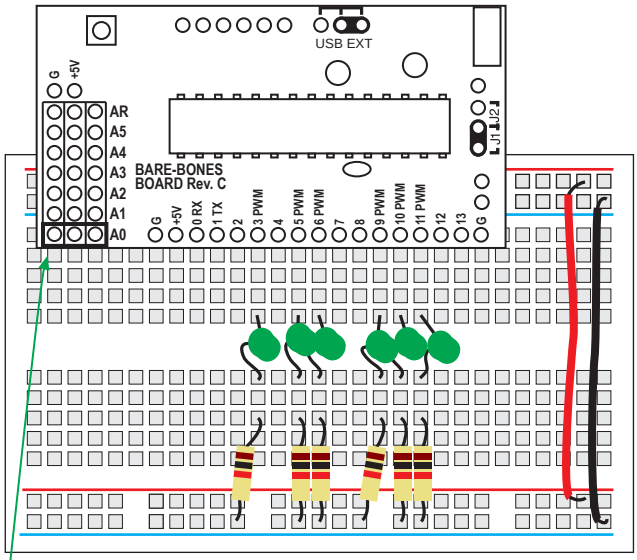
If you need to have more regulated power, to power a lot of high powered LED's on a breadboard, for example, then, your board will accommodate a TO-220 package regulator, such as the LM2937 listed in the parts list, or the popular 7805. Just remember to insert it backwards as shown in figure 1.

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

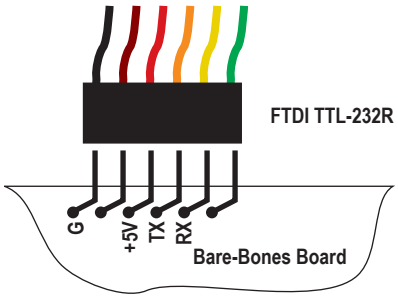


typical 3 terminal regulator schematic symbol



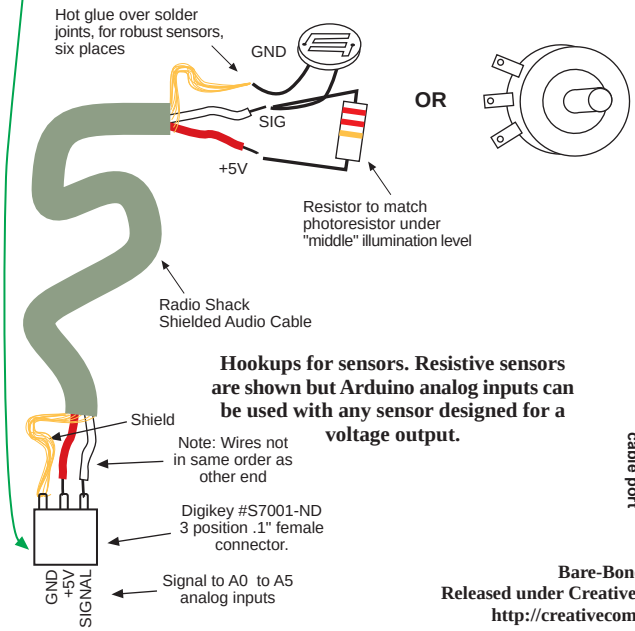


An Arduino Board set up on a solderless breadboard with six LED's ready for dimming with the PWM outputs.

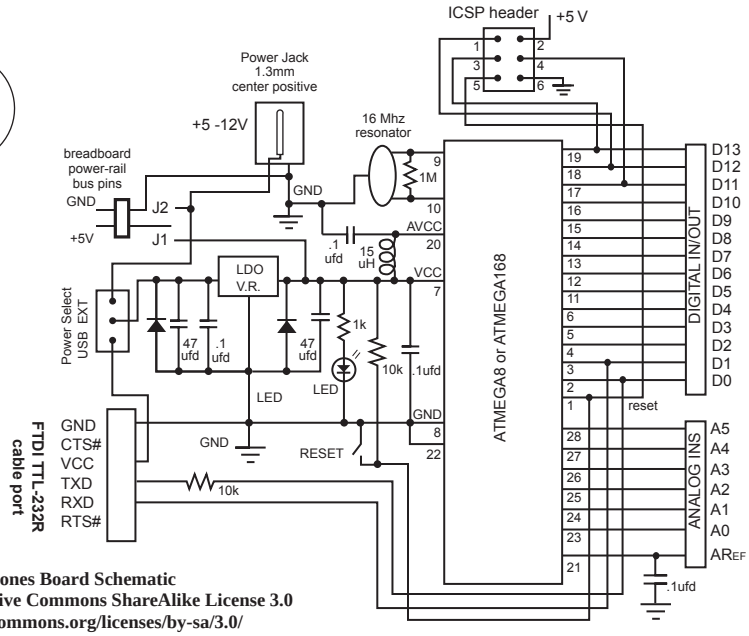


Programming cable connections between a BBB and a FTDI TTL-232R USB to TTL serial cable.

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 Arduino's RX pin (0) and the cable's RX pin is connected to the Arduino's TX pin.

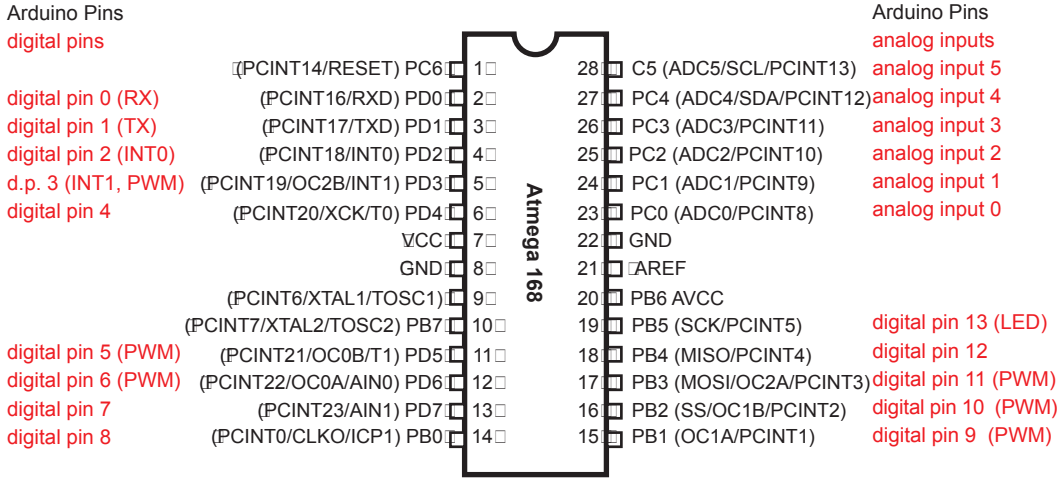


Hookups for sensors. Resistive sensors are shown but Arduino analog inputs can be used with any sensor designed for a voltage output.



Bare-Bones Board Schematic
Released under Creative Commons ShareAlike License 3.0
http://creativecommons.org/licenses/by-sa/3.0/

Bare-Bones Board Arduino Schematic



Pin mapping of the Atmega168 chip to the Arduino Board

Troubleshooting

Symptom: No pilot light.

Causes:

LED in backwards

electrolytic capacitor in backwards

voltage regulator in backwards

no power select shunt (shorting clip)

no power at external jack - check power adapter & polarity

bad solder connection - check power at power jack &

power-select pins

diode in backwards (disconnect power supply right away)

power supply connections reversed - check external supply with a multimeter

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 is in backwards or poor solder joint. If low or incorrect voltage, check diodes, voltage regulator, solder joints, power supply

Symptom: Pilot light on but program won't download to board

Hardware Causes:

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

Atmega168 not programmed with bootloader

bad cable

drivers not installed on PC - Check Arduino->Tools->Serial Port

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

wrong resonator value

wrong resistor across resonator (1M)

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

Software/PC side causes:

check for FTDI drivers installed (if using USB cable)

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

click RESET switch simultaneously with download attempt.

General "Cure-Alls":

check orientation on all polarized parts, V.R., caps, diodes, V.R., socket and chip.

check values of resistors

Reheat all solder pads on bottom of board, look for bridges (shorts) on chip pins

clean PCB with toothbrush and isopropyl alcohol

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

Electrical Soldering for Beginners

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. It is important to do this as soon as a new soldering iron gets hot.

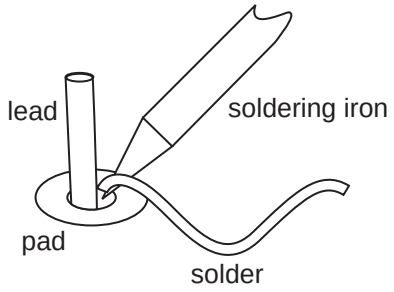
Wipe your soldering iron tip off on a wet sponge, or a copper "scrubbie", to keep it clean and shiny. Do this whenever the tip stops looking shiny or has too much solder buildup on it.

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.

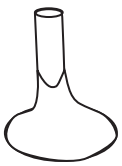
Work in a room with some ventilation. There is a tiny bit of lead in solder fumes but the flux fumes are more of a health-hazard than the lead. Jameco sells a nice soldering iron / carbon filter combination for under $100.

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.



cold solder joint
(not enough heat)



good solder joint

- smooth meniscus
- shiny
- covers pad

It is a good idea to clean the solder flux off your board with a toothbrush and isopropyl (not denatured) alcohol, when you are done with your board. Most fluxes when left on the board for extended periods of time, 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 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.

Power Soldering for Multiple-Kit Builders: or How to put together 10 BBB kits on Saturday and still have time to meet your friends.

Additional items required: Piece of foam rubber - antistatic pink foam is ideal, small alligator clips.

We insert parts in groups and don't bend any leads to hold parts in. Once the board is flipped for soldering, the foam holds the components agains the board. We put the boards together in 4 steps.

Step 1: mount the inductor, covered on page 4. You can get it straight by quickly and alternately heating either end and gently moving it.

Step 2: Insert the resistors, diodes, reset switch, LED and small (104) caps. Don't bend the leads. Cover the parts with the foam and flip the whole board, then solder it. If you are worried that a part may not be seated down against the board apply some pressure to the board while heating one pin.

Step 3: Cut the leads from the last step. Insert *all* the remaining parts *except* the 17 pin header and 2 pin header. If you have alligator clips, clip the programming header and the power jack to the board with them, you could also try this on the other headers if you have a lot of clips handy - the small ones are better .

Put the foam on top of the board and flip the whole mess so foam is now on bottom and board is upside down. Tack down one pin only on headers and socket. Solder in all pins on electrolytic caps, resonator and parts held in with clips. (You could also inspect them first for correct fit) Flip the board to inspect "fit" on headers and socket and adjust by heating with pressure from index finger. A thimble might be useful if you haven't burned out all the nerves in your index finger soldering, as we have. Flip and finish soldering.

Step 2: Insert the 17 pin and 2 pin header on bottom. Tack, inspect, straighten if necessary and solder.