Extreme Surface Mount Soldering

by **doctek** on January 30, 2009

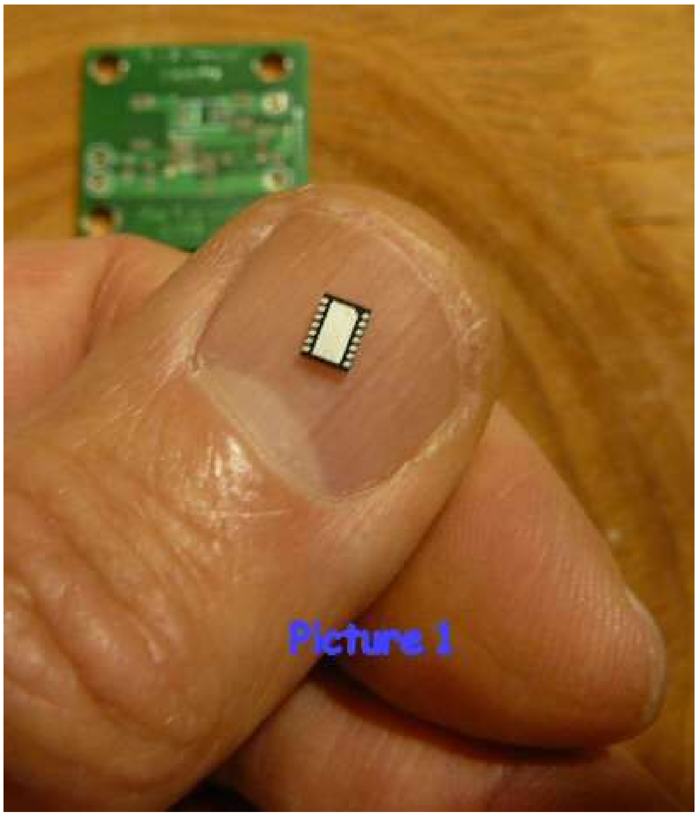
Table of Contents

Intro: Extreme Surface Mount Soldering

So you're thinking of building a super-widget and wanted to use the latest QFN / MLF (micro lead frame) package parts. But then you actually LOOKED at one, realized how tiny it really is (Picture 1), and decided no way can I solder that! Think again! This instructable will show you how to design and build a circuit using truly tiny SMT devices. Not only will I show you how to build your own Hot Plate Soldering System capable of soldering the tiniest surface mount components using lead-free solder, you will learn to design the footprints, stencil on solder paste, and solder the components. I'll also point you to cheap sources for design software, PC boards, solder paste, and stencils.

This instructable is about two things: How to build and use a Hot Plate Soldering System, and general guidance on how to design circuit boards using surface mount parts. All the information you need for successful surface mount design and construction. Throughout this Instructable, many sources will be cited: like Newton, I truly stand on the shoulders of giants! (That's another way of saying others have already worked most of this out.) What I'll do is explain the methods I've used successfully to build tiny surface mount circuits. The focus is on specifics, not generalities. I believe the method and tools I'll describe are the most economical available which will yield satisfactory results.

So let's get started. Successful design and construction with those Extremely Tiny devices you've been dying to use is just an Instructable away!



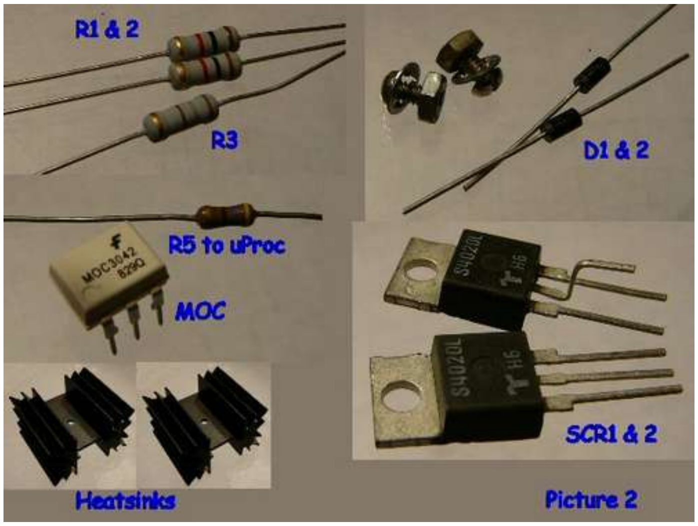
step 1: Order Parts

Order the parts to build the Hot Plate Soldering System. Here is the Parts List with Digikey Part descriptions and numbers:

Parts for the Pulse Width Regulation Unit: (If you plan to use a Ghetto Development System, then you already have U1.)
U1 *IC MCU AVR 2K FLASH 20DIP* **ATTINY2313-20PU-ND**
U2, U3 *IC I2C I/O EXPANDER* **568-4236-5-ND**
SW1, 2, 3, 4 *SWITCH DIP EXT RCKR UNSEALD 4POS* **GH7170-ND**

Parts for the AC Control Unit:
MOC *OPTOCOUPLER TRIAC-OUT ZC 6-DIP* **MOC3042M-ND**
SCR1, SCR2 *NON-SENSITIVE GATE 20.0A 400V* **S4020L-ND**
R3 *RES 180 OHM 1W 5% METAL OXIDE* **180W-1-ND**
R1, R2 *RES 1.0K OHM 1W 5% METAL OXIDE* **1.0KW-1-ND**
D1, D2 *DIODE GEN PURPOSE 50V 1A DO41* **1N4001FSCT-ND**
Heat Sink *HEATSINK TO-220 W/PINS 1.5"TALL* **HS350-ND**

For the Pulse Width Regulation Unit you'll also need LED1 and LED2, resistors R3 and R4 (470 Ohm), pull-up resistors R1 and R2 (3.3K Ohm) for the I2C bus, and R5 (470 Ohm) between the MOC and the Pulse Width Regulation Unit. These can be from the junk box or Radio Shack and the values aren't real critical. You'll also need some perf board. Get the kind without any metal on it for building the AC Control Unit. One other item is some silicone heat sink grease also from Radio Shack if you don't have any. Finally, you need S1, a momentary contact push button, and you should have C1, a 10uF cap.



step 2: The Hot Plate Soldering System

Lots has been said about surface mount soldering (SparkFun tutorials ; Scott Driscoll's web site and article in Make 16) but I think two things are critical that these sources don't address adequately (IMHO). Eventually, you'll want to solder really small SMT parts (QFN) and there's no reason you shouldn't. They're just incredibly tiny and their pads are underneath the part body making them virtually impossible to get to with a soldering iron. The technique I'll show you handles these with no problem. They also usually have a large central pad. Soldering it is **critical** for a lot of ICs - especially switching regulators. Since these are some of the most useful ICs, it's crucial to learn to solder them. Amazingly, the technique I describe solves both these issues: soldering tiny parts and soldering the central paddle.

Although both the Make article and the SparkFun tutorials say you **can** solder the pads on lots of SMT parts with a very fine tip soldering iron and then clean up shorts with desoldering braid, this seems mostly unnecessary to me and demands a level of patience (and perhaps skill?) that I just don't have. If forced to I would try this method, but I think my way is better. Putting solder on with a stencil, placing all the parts, and soldering everything in one, 10 minute pass on a hot plate just seems a whole lot simpler!

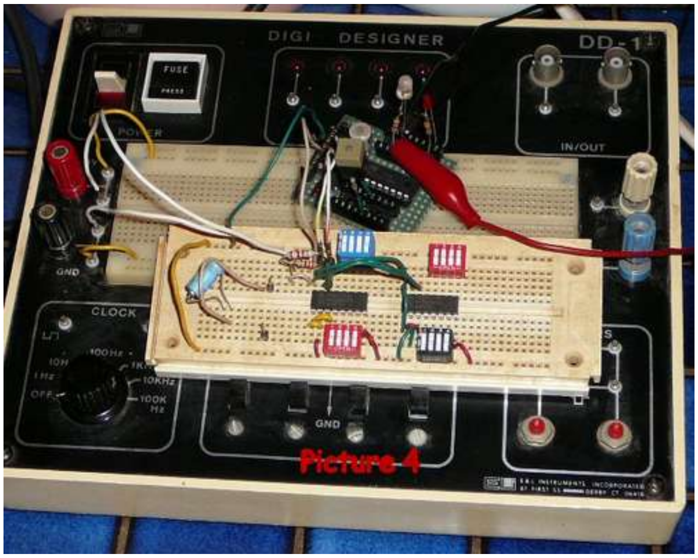
Even the Schmartboard method is a lot of work and doesn't really address how to build a board with multiple SMT components and keep it small. It's fine for prototyping and testing new parts quickly, but it won't yield really small boards.

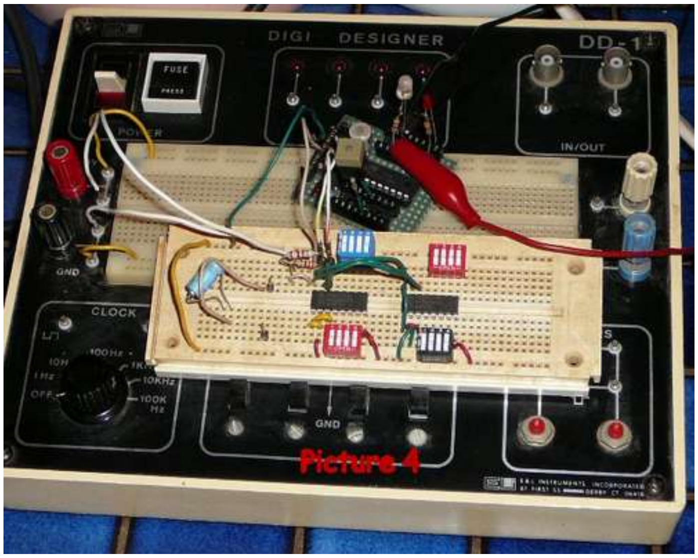
SMT soldering using an electric skillet is a technique described in a tutorial by SparkFun. When I first read about this, it struck me as a great way to go. I had two concerns. First, I was using lead free solder which melts at a higher temperature than solder with lead. (420F minimum; temperatures up to 500F are used in the process. These are at the very end, or just beyond, the range of most electric skillets.) Also, the temperature profile for heating and soldering seemed marginal using the skillet and would be even more marginal with lead free solder. A hot plate would solve both these problems since it could get a lot hotter a lot quicker.

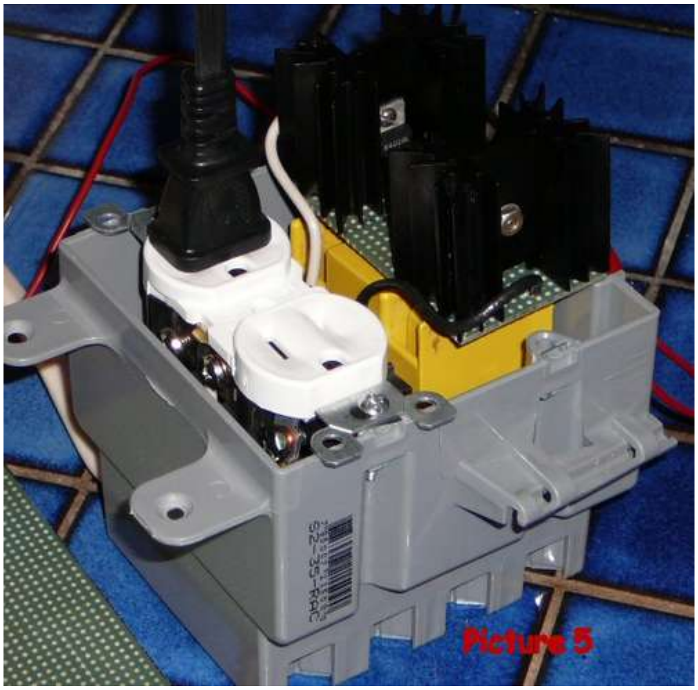
So a few dollars changed hands at the local department store (pick your favorite) and I was attempting to solder with a hot plate. (Picture 3) Although the unit itself had more than enough heating power to melt the solder, the heat control system was totally inadequate. After severely burning one test board I realized that a better controller was needed. Some alternatives were considered, but I quickly concluded that what was needed was a solid-state power control unit and Pulse Width Regulation to drive it. Creating and controlling the pulses was easy - I just used an Atmel ATtiny2313 and a couple of I2C Port Expanders, plus a few DIP switches. I'll describe this system in more detail and provide schematics and source code for it.

While Pulse Width Regulation is a great way to create control pulses, a little harder problem was how to use the 5V pulse output to control the 115VAC to the hot plate. More research turned up a simple solution for this also. I'll present all the details and explain how to build the AC Control Unit in the sections to follow. After doing a moderate amount of research, this seemed like the simplest and best approach. It has proven to work just as I hoped.

The Hot Plate Soldering System consists of three parts: the hot plate, the Pulse Width Regulation Unit (Picture 4), and the AC Control Unit (Picture 5). The latter two units form the Hot Plate Control System. I'll explain the details of the Hot Plate Control system and show you how to build the two units that comprise it. Finally, I'll detail the process of using this system for soldering tiny surface mount devices.







step 3: The Pulse Width Regulation Unit

An ATtiny2313 microcontroller is the heart of this unit and generates the pulses. Two PCA-8574A Port Expanders, connected using the I2C bus, read DIP switches to control the pulse widths.

A few words about the control technique are in order. Pulse Width Regulation is a variation of the technique better known as Pulse Width Modulation and is a way to control power by rapidly turning it on and off. I prefer the term Pulse Width Regulation since we're really not modulating the pulses, but rather changing their duration (duty cycle) and thereby regulating the power delivered to the load. If you're more comfortable with the term PWM, go ahead and call it that.

This is really pulse width regulation, not pulse width modulation. The pulse width is regulated to control the period of time that power is applied. For example, during each 10 second period the control output is on for 3/10ths of a second. This would be a 3% duty cycle and the width of the pulse would be that same 3/10ths second every 10 second cycle. Increasing the time when the output is on by making the pulse wider would turn the unit being controlled on for a longer time. Thus we regulate the pulse width to control the load; in our case the hot plate. By turning the hot plate on and off in a relatively rapid, but constant rate, fashion, we closely control the temperature.

Pulse width modulation, by contrast, varies the width of the pulse every cycle. By doing this, and filtering the resulting pulse train, complex waveforms can be produced. This is true PWM and isn't really what we're doing here. Pulse Width Regulation is sufficient for our needs. You can learn more about PWM here .

As stated, the Pulse Width Regulation Unit is controlled by an Atmel AVR ATtiny2313 microcontroller and two I2C Port Expanders. Go ahead and download the schematic and the source code for the controller now. The source code for the control program is a Zip file that will unzip into it's own directory named Control. Use the Ghetto Development System to program the microcontroller. It's not required, but you may want to have a look at this Instructable to learn more about the I2C bus. The I2C Instructable will show you how to modify your Ghetto Development System so that you can use the I2C Bus on it.

Use the parts you ordered back in Step 1 and the schematic you've downloaded. Picture 4 shows how to build the Pulse Width Regulation Unit. My Ghetto Development System is shown (built on perf board) along with the Port Expansion and the DIP switches (built on solderless breadboard). The larger DigiDesigner box is just used as a 5V power supply. I built it this way since I use the Ghetto Development Systems for lots of other experiments. You can use the same approach or build a dedicated system - your choice. In any case, the schematic and the software are the same.

Note that the DIP switches as shown on the schematic indicate the Port Expander pins to connect the switches to. The orientation of the switches should be as shown in Picture 6. SW1A and SW1B on the schematic are really one package of four DIP switches - SW1. Same for SW2, 3, and 4. Connect as shown in the picture to pins of U2, but ignore the pin numbers on SW1A, etc. that are shown on the schematic. Just plug them in so they connect directly to U2 and U3 on the proper pins, with the other side of the switch packages connected to ground. Picture 6 shows more detail.

To program your ATtiny2313, start the WinAVR environment, change to the directory with the files you unzipped in it, and use the Makefile to compile the software and program your part. The program is pretty straightforward. Timer/Counter 1 is used in Fast PWM mode to produce the pulses used to control the AC Power Control Unit. The pulses appear at PB4, and PD6 is used to flash LED1. The 1MHz default clock of the ATtiny2313 is used and is divided by 1024. So the resolution of the PWM is .001024 seconds. Since we're going to use steps of one tenth of a second, we multiply the DIP switch settings by 98 which yields steps of 0.1003 seconds - close enough.

Once you've built the Pulse Width Regulation Unit, you'll want to test it. Power it up by connecting a 5 volt power supply to it. If you've worked ahead and built the AC Power Control Unit don't connect it just yet. When you turn on the 5V power, LED1 should blink slowly twice to let you know it's ready to go. The DIP switches connected to Port Expander U2 are used to control the Cycle Length while the DIP switches connected to Port Expander U3 are used to control the duty cycle or On Time of the cycle. Both controls are set in units of 1/10 second and use binary code. Don't panic if you're not familiar with binary code, I'll explain it.

Binary code is best thought of as a doubling code. That is, the numeric value assigned to the first switch is 1 and the subsequent switches have numeric values that double. So the second switch has a value of 2, the third's value is 4, then 8 and so on. The value of the set of switches attached to either U2 or U3 is the sum of the numeric values of all the switches that are ON.

Here is a table showing the values of the individual switches on SW1 and SW2 which are connected to U2 for setting the Cycle Length. This assumes that you've built you circuit the same as mine and have the DIP switches oriented the same way as shown in Picture 6.
DIP Switch Number Pin Number Value when ON (HI)
SW1-1 U2-4 1
SW1-2 U2-5 2
SW1-3 U2-6 4
SW1-4 U2-7 8
SW2-4 U2-9 16
SW2-3 U2-10 32
SW2-2 U2-11 64
SW2-1 U2-12 128

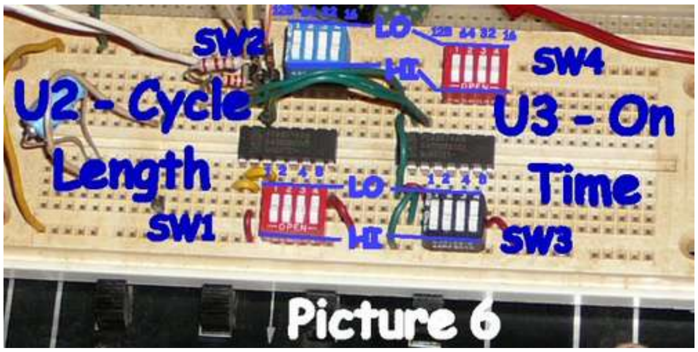
The same values apply to SW3 and SW4 on U3 for the On Time setting.

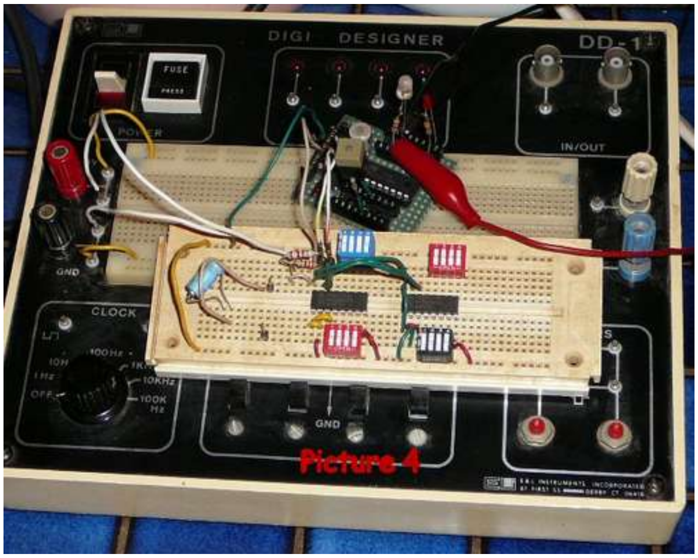
An example should clarify this binary stuff. To set the value of 100 for the Cycle Length, turn on switches SW2-2, SW2-1 and SW1-3. From the table, the numeric values are 64+32+4 = 100 which is the value we want. All the other switches on SW1 and SW 2 should be off. Using the software I've provided, the minimum time for Cycle Length and On Time is one tenth second, so a value of 100 represents 10 seconds. As a test of your understanding, figure out what switches to turn on to set the value of 50 (5 seconds) for the Cycle Length. I'll tell you the answer below, but you should work this out yourself first.

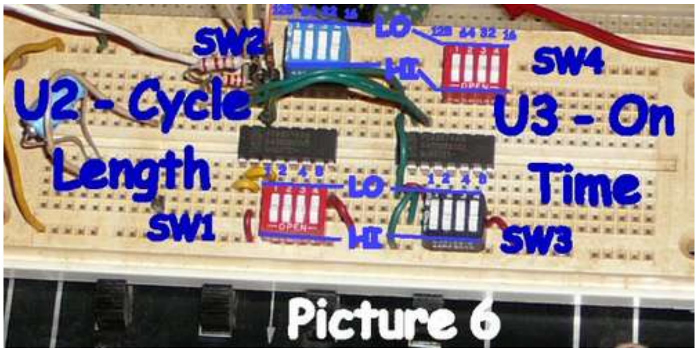
Continuing with our testing, LED2 will flash to show the pulses being generated. So set a couple of numbers on the DIP switches - say 50 for Cycle Length and 10 for On Time (Switches SW2-3, SW2-4 and SW1-2 ON = 50 for Cycle Length - and the answer to the question above; Switches SW3-4 and SW3-2 ON = 10 for On Time) - and press the button. Note that the DIP switches are connected to ground (lo) when, according to the marks on their packages, they are on, and are pulled up (hi) when they are off or open. This may seem backwards, but closing the switch connects it to ground and makes it read LO. Opening the switch allows it to be pulled up to HI. Picture 6 show how this works. Easy once you get used to it. Pressing the button causes the switches to be read and their setting used to program the pulse generation. You should see LED2 flashing for one second every 5 seconds. If you see any more flashes from LED1, then something is wrong with the I2C Bus and you should check your wiring carefully. Change the Cycle Length DIP switches to 100 (values shown above if you're not sure), push the button, and you should get a 1 second flash every 10 seconds. Play with this until you get bored. The push button is there to ensure that settings don't change until you're ready for them to change.

You may be wondering what happens if the On Time switch setting is the same or higher than the Cycle Length switch setting. Go ahead and try it. Now is a good time for those sort of experiments since you can't hurt anything. Probably better not to do this with the AC connected since the SCRs could be turned on constantly and your hot plate will heat up a lot!

You may also wonder why the minimum Cycle Length and On Time are 1/10 second. My research indicated that this value would avoid causing any disturbances to the AC line voltage and seemed like a good value to start with. This proved to give satisfactory control of the hot plate so I haven't had any reason to change it. Obviously, you can experiment with other values by modifying the control program.







File Downloads

Control.zip (10 KB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Control.zip']

PWM Controller.pdf ((612x792) 33 KB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'PWM Controller.pdf']

step 4: AC Control Unit

Caution: You are dealing with 115V AC! This is capable of giving you a lethal shock. You will be controlling an appliance that could burn your house down. You've been warned! Proceed at your own risk and with all due caution. Follow the directions carefully. Check your work twice! Don't take shortcuts! If you're not comfortable doing house wiring (115AC) get someone to help you.

Note that this unit works only for purely resistive loads. This means only standard (not high intensity) incandescent (not fluorescent) lights and heating units without fans (like hot plates or electric frying pans). Do not try to control inductive or capacitive loads like motors, relays, or fluorescent lights. There is no snubbing in the circuit and the SCRs could be destroyed if used on loads that are not purely resistive.

Download the Applications Note that contains the circuit schematic from the Fairchild site. The schematic we'll use is Figure 14 in that apps note, except that we'll use an MOC3042. Note that the 180 Ohm resistor is R3 in our parts list and SCR1 is closest to D1, R1 and the MOC, while SCR2 is closest to D2 and R2. I'll use these numbers consistently in drawings and figures to follow.

The MOC is an optocoupler that isolates our Pulse Width Regulation Unit from the AC line while allowing the pulses to control the SCRs. Another function of the MOC3042 is to ensure that the SCRs are switched on only when the AC voltage is at 0. This is known as zero-crossing switching and minimizes EMI and any disturbances to the AC power lines. Read more about it in the Fairchild apps note.

Gather the parts you need. In addition to the parts you ordered in Step 1 and shown in Picture 2, you'll also want to visit a hardware store and buy a sturdy plastic, double outlet box, a 15A rated receptacle, a 15A rated wall plug, a single outlet or switch box (that you'll cut up), a terminal block rated for 115AC (Radio Shack, or hardware store), and some 14 gauge, solid copper wire. Use 14 gauge copper wire for main conductors, 18 or 20 gauge for smaller conductors. These items are shown in Picture 7. I used 14 gauge, 3 wire (2 conductors plus ground) electrical cable to connect to the wall plug. You may want a wall plug with stranded (more flexible) wire. That's fine, just be sure it's rated for 15A service. Use plain perf board with no plating or plated holes on it to build the circuit. This makes sure that conductor spacing can be maintained. Our goal is to maintain 0.1" to 0.2" spacing between all lines carrying 115VA. If you follow my pictures, you'll be fine. Note that the SCRs specified have an isolated heat sink tab. That is, the heat sink is not electrically connected. This means that if you accidentally touch it, you won't have any risk of a shock. This is a good thing.

I've provided a number of pictures and diagrams to guide you. Picture 8 shows the completed circuit board with the heat sinks in place. Picture 9 is the top view of the circuit board without heat sinks and with the parts labeled. Picture 10 is the bottom view of the circuit board showing construction details. Picture 11 is the bottom view with parts and pins labeled. The AC Board Diagram is a schematic showing parts placement and wire routing. It is a bottom view corresponding to Picture 11; parts that go on top of the board are shown in pink. If you study these a bit, you should have no trouble building the unit following the directions below.

Start by cutting out the perf board to fit in one half of the double outlet box. The dimensions I used are 2.9" by 1.6". (If you're in doubt about part locations, just count the holes – they're 0.1" apart.)

Make the holes for the heat sinks in the locations shown (count the holes). They will be mounted by bolting them to the SCRs and soldering the SCRs in place.

Bend the middle SCR lead as shown in Picture 12. The bent leg should extend 0.2 inches in front of the other two legs. Bending the middle lead like this helps maintain the conductor spacing. Bolt the SCRs to the heat sinks and test fit them in place. You may want to tape the assemblies in place temporarily until you solder the leads. Once the SCRs are soldered in you can remove the heat sinks for more room to work.

Cut about 16 inches of 14 gauge wire and strip about 6 inches of it so you'll have some bare copper wire to work with. Cut two lengths and bend them to make the drain to source (K to A) connections to the SCRs. Look at the pictures!

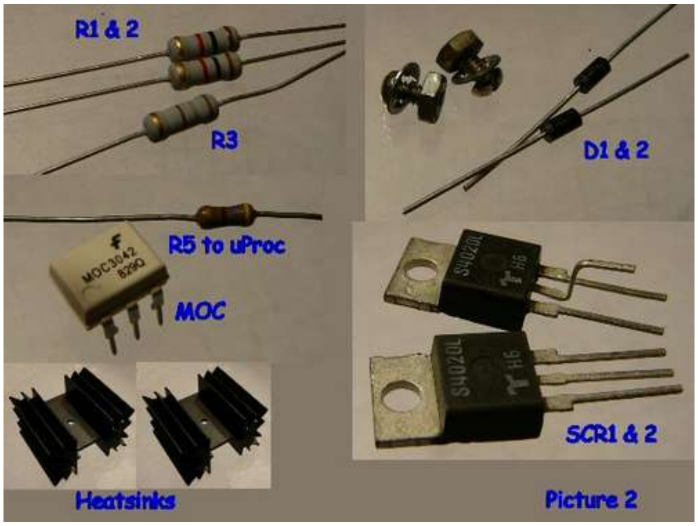
One end of each length of wire should come out of the perf board to connect to the terminal block. We'll bend that later to fit during the mechanical assembly.

Bend the leads of the SCRs over the wires you've just fitted and solder them in place. Make these joints very sound mechanically and solder them carefully. This ain't 5 volts you're dealing with here! Picture 10 shows this clearly.

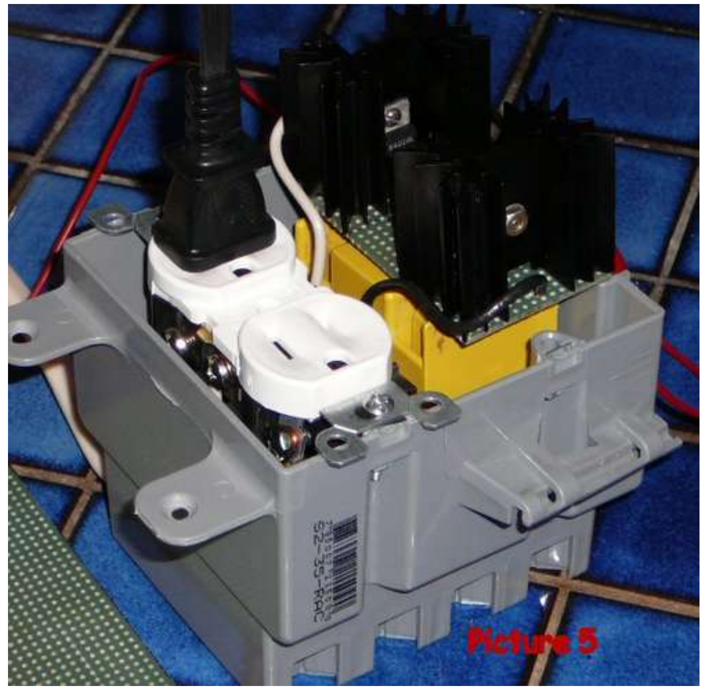
Mount the MOC3042 in place. Again, use some tape to hold it until it's soldered. Use an insulated piece of 18 gauge wire to connect from pin 6 of the MOC to the gate of SCR1. Picture 11 and the AC Board Diagram show this.

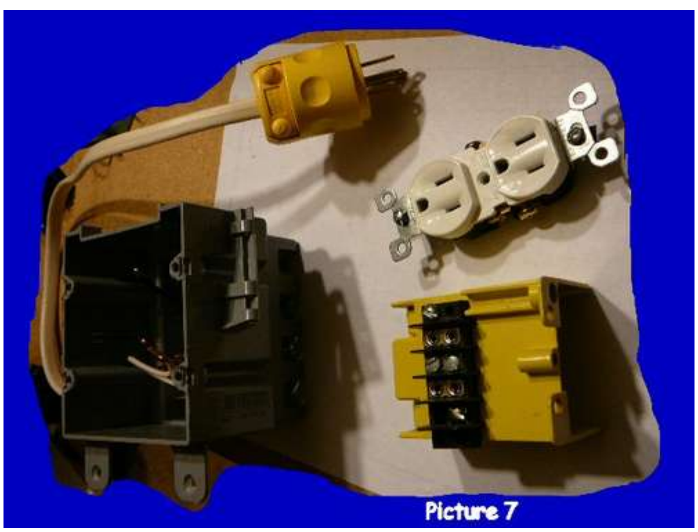
Route the leads for the resistors and diodes. Make solid mechanical connections and do a good job soldering. Picture 11 shows where I connected each component while Picture 9 shows the top view. These should be a useful guides.

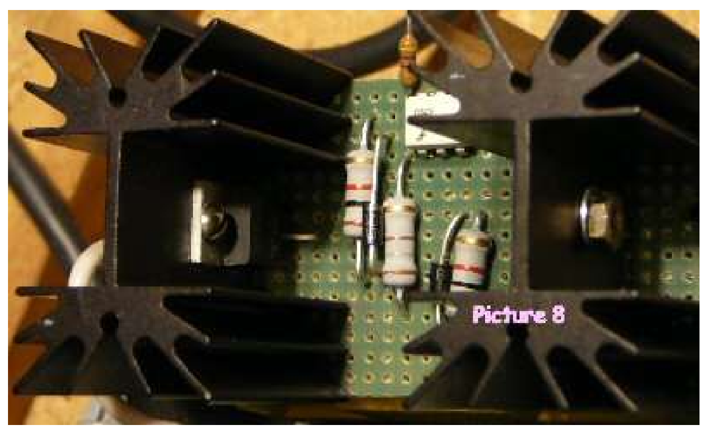
Once all the soldering is done, dismount the heatsinks from the SCRs and put a thin layer of heat sink grease (Radio Shack) on the back of the SCRs and remount the heatsinks. Tighten the screws firmly, but don't overdue it. You don't want to distort the tabs on the SCRs. Your circuit board should look like Picture 8. The resistor shown as "R5 (uProc Gnd)" is R5 on the Pulse Width Regulation Unit Schematic. Mount this where it is most convenient. I put it on the AC Control Unit since it's a convenient place to connect ground from the Pulse Width Regulation Unit. The control signal from PB4 on U1 (the uProcessor) connects to the wire to MOC pin 1.

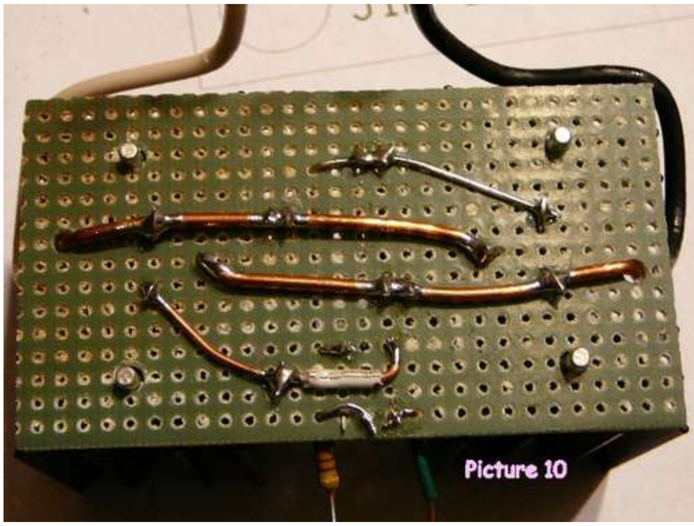


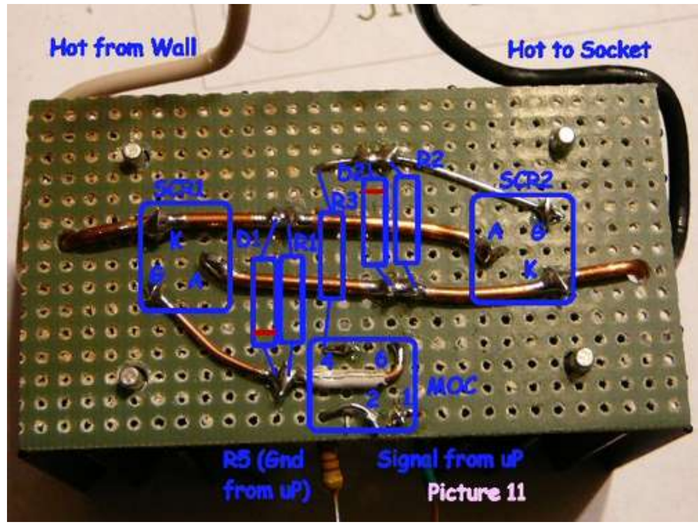
Picture 2

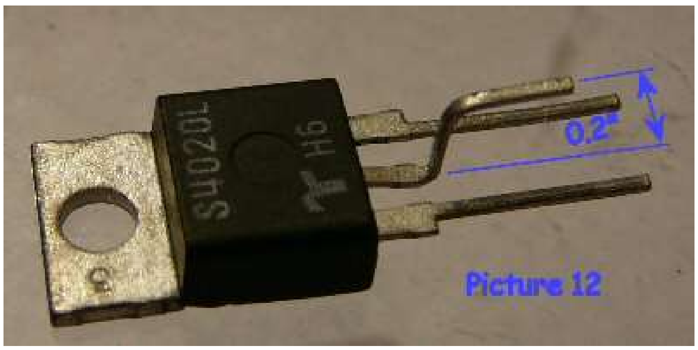

Picture 5

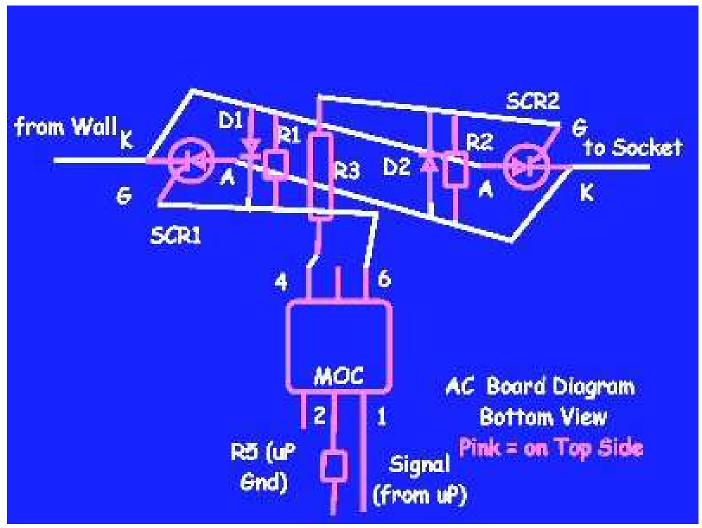

Picture 7

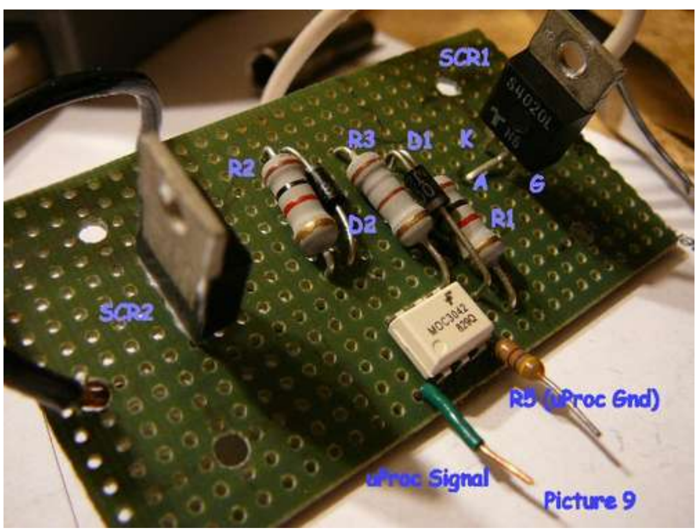

Picture 8


Picture 10


Hot from Wall
Hot to Socket
SCR1
K
G
A
D1
R1
R3
D2
R2
SCR2
A
G
K
4
6
2
1
MOC
R5 (Gnd from uP)
Signal from uP
Picture 11


S4020L
H6
0.2"
Picture 12


from Wall
K
G
A
D1
R1
R3
SCR1
D2
R2
SCR2
G
to Socket
A
K
4
6
MOC
2
1
R5 (uP Gnd)
Signal (from uP)
AC Board Diagram
Bottom View
Pink = on Top Side

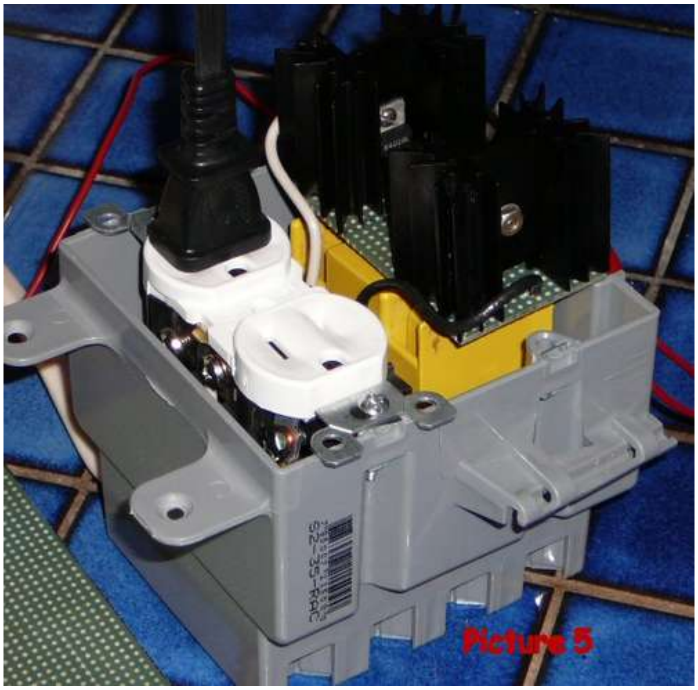

SCR1
S4020L
R2
R3
D1
K
A
G
D2
R1
SCR2
R5 (uProc Gnd)
uProc Signal
Picture 9

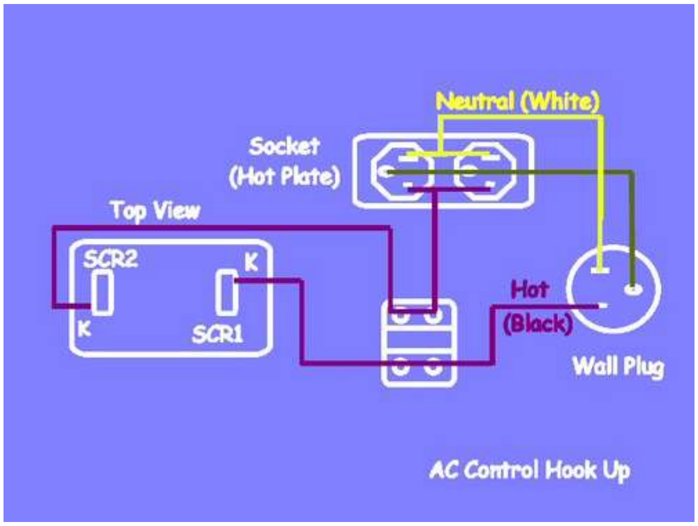
step 5: AC Control Unit - Mechanical Construction

Have another look at Picture 5 to see what the completed unit looks like. We'll use the single switch box as a support inside the double box. Cut it so that you have a piece about the width of the perf board the circuit is built on – about 1.6 inches. It's not critical, you just want this to support your circuit and hold the terminal block. Picture 13 shows this.

Mount the terminal block to it about midway up. To make this fit in the larger box you may need to do some additional cutting on one box or the other. Picture 13 and Picture 14 show details. Basically, your goal is to hold the parts firmly so they can't short out. The stiff, 14 gauge copper wire will help with this.

The diagram AC Control Hook Up and Picture 14 show how to hook things up. When you're done, the unit should look like Picture 5.





step 6: Testing

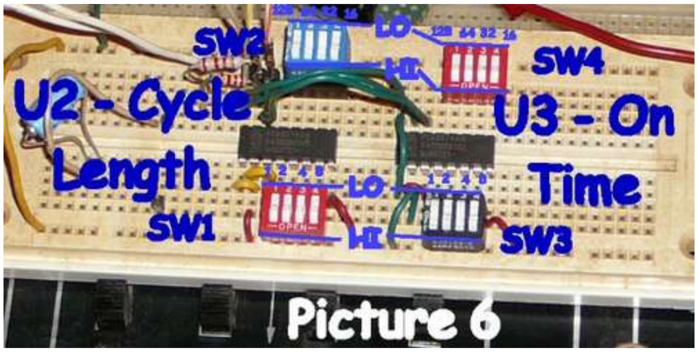
Now comes the exciting part! Before we try out a hot plate, let's make sure our Hot Plate Control System works by experimenting with a lamp. An ordinary table lamp will do.

Remember: Incandescent lights and heating appliances only! No fluorescents, high intensity lights, or motors!

Hook ground from the Pulse Width Regulation Unit to R5 which goes to pin 2 of the MOC3042 on the AC Control Unit. Hook the signal line from PB4 of the ATtiny2313 to Pin 1 of the MOC3042. The MOC3042 is an optoisolator and isolates the AC lines from the 5V Pulse Control Unit. Be sure you have a 470 Ohm Resistor (R5) on pin 2 of the MOC3042. Plug in both units, plug in your table lamp (be sure the lamp switch is on), and turn on the 5V power.

Refer to Picture 6 and the discussion in Step 3 to help you set the values of 100 for Cycle Length and 10 for On Time on the DIP switches and press the button. If all goes well, your table lamp should be blinking cheerfully for one second every 10 seconds. Works OK? Pat yourself on the back. Good job!! Play with the settings some - blinking a table lamp was never such fun!

Now it's time to try your hot plate. Turn everything off and unplug the table lamp. Plug in the hot plate, turn its control to about the middle of its range, and turn on the 5V again. Try a very conservative setting of 100 and 1, that is a ten second Cycle Length with 1/10th second On Time. In about 5 to 10 minutes, your hot plate should get barely warm to the touch. I can hold my hand on my hot plate indefinitely with this setting. Now try increasing the On Time to 2. The plate should get noticeably warmer over the next few minutes. Go ahead and experiment a little, but we'll come back to how to calibrate the hot plate for soldering soon.



step 7: The Surface Mount Design Process

With your hot plate controller ready to go all you need is a surface mount circuit board. That means you need a design that you want to realize using surface mount components. I'm not going to tell you what you should design. That's obviously up to you. If you haven't already given this some thought, you probably wouldn't be reading this Instructable. I'll assume you've already got a design in mind and were just waiting until you could figure out how to solder it before going ahead.

So while I'm not about to tell you **what** to design, I'm quite willing to offer some guidance on **how** to go about turning your design into a circuit board. Of course, I'll also complete the detailed instruction on soldering all the components onto it. As I stated at the beginning, most of this is pretty well worked out and the information is available. If it weren't for the fact that I see so many questions about how to do this and see so many expensive, complicated, or downright misleading answers, I wouldn't bother. But I do, so I am.

Great sources for new ideas and design guidance are IC manufacturer's web sites. Check out especially their application notes for the circuits you want to design. (That's where I got the circuit for the AC Control portion of the Hot Plate Control System.) There's also a ton of information available from other web sites. So getting good ideas is not a problem, but you still may not be sure your particular design will work. One way to help ease your mind is to simulate your circuit, and the best simulation tool I know of is this one. Although it was originally intended to help with switching power supply design, it's really a rather complete implementation of SPICE and provides powerful simulation capabilities for analog circuits. You can learn a lot more about SPICE using Google. A nice tutorial is here .

Whether or not you create your own circuit designs, you'll still need to draw a schematic and use it to guide your circuit board layout. As you do your circuit board, you'll need to be mindful of where you will have your actual circuit boards made. There are many circuit board manufacturers that cater to the hobbiest, but I'm going to recommend just one based on the combination of very reasonable prices and very solid support. That's BatchPCB which is the service that SparkFun uses and recommends (actually use Gold Phoenix to fab the boards). Besides price and support, there is another key reason for this recommendation. The circuit board manufacturing at BacthPCB is based on industry-standard Gerber files and is not tied to a proprietary design software. This means you can use whatever tools you like to design your boards and if you want to go to another circuit board vendor, you're free to do that without having to redesign your circuit boards. You just send the same Gerber files to the new vendor and you're good to go. BatchPCB also gives you solder mask on both sides and stencil (labeling) on the top side of your boards. Solder mask is essential for successful surface mount soldering - especially with very small components. I mention these things because not all circuit board manufacturers offer these at hobbiest prices.

The choice of a circuit board manufacturer that uses Gerber files is important because it means we should choose schematic and board design software that produces design output in the Gerber format. Again, I'm going to offer one solution based on price (free) and support (excellent). The free version of the EAGLE package will meet the needs of nearly every hobbiest. There are many excellent tutorials available from SparkFun and also as part of the EAGLE package itself. The software is mature and dependable with a large user base. EAGLE provides all the flexibility we need to create designs including the soldermask and stencil layers that are so important to successful surface mount design. EAGLE has extensive part libraries, including a growing one from SparkFun, to speed your circuit board design.

A possible alternative to EAGLE is KiCAD. While this package looks to be very capable and has a growing user base, it has nowhere near the support that EAGLE has. The main advantage of KiCAD is that there's no restriction on board size or number of layers. There's an independent evaluation of KiCAD here and atutorial here (courtesy of Scott Driscoll). Unless you are limited by EAGLE's restriction to just two signal layers, I'd suggest sticking to EAGLE. Multi-layer boards are significantly more expensive.

The SparkFun tutorial points to this Gerber viewer. Using the Gerber viewer to view each of our Gerber files is very important because we will see exactly what our circuit board vendor will use to make our boards. Errors can be easily overlooked when we're designing our circuits and using a Gerber viewer can help make sure we catch as many as possible.

The size and shape of the copper pads that surface mount devices are soldered to, the corresponding openings in the solder mask, and the openings in the solder paste stencil are collectively referred to as the footprints. Designing footprints for surface mount parts used to be pretty simple. For example, there are standard rules of thumb built into EAGLE that usually create satisfactory footprints based just on the copper pad dimensions and these work fine for most surface mount devices. Creating footprints using these is discussed in the SparkFun tutorials. However, when we want to use really tiny devices, such as QFN, DFN, and MLF, the footprint design becomes more complicated. Fortunately, the information you will need is readily available. Every IC manufacturer seems to have an applications note explaining the details for their parts. You should find these for the parts you use. If you can't find one for the specific parts you're using, then I recommend this one from Intersil. It seems to contain a very complete set of design guidelines along with detailed footprints for many standard parts. This one from Atmel has some pictures of stencils which show ways to do what Intersil recommends. Note the combination of paste type, size of opening, vias in pads, and thickness of stencil. These all interact - that's why you have to experiment. BTW, you'll have to use the EAGLE tutorials to learn how to create the solder mask and solder paste footprints since the SparkFun tutorials don't really get into those details. There are many other sources of information about proper footprints and soldering technique from Atmel , ON Semi , and Freescale .

While I highly recommend careful design of footprints for the copper pads, the solder mask, and the solder paste stencil, I'll also go out on a bit of a limb here and say everything doesn't have to be perfect. One particular area I've strayed is in the solder paste mask. Most recommendations are to use a metal stencil with etched openings having a prescribed side slope to it. If you do this, you'll find that your stencil will cost as much as or more than your boards and your parts all together!. A much cheaper alternative is to get your stencil from Pololu . The stencil will be made of 3 mil thick mylar with laser cut openings. For tiny openings (like on QFN devices), Pololu only guarantees that you'll get some paste on each pad. In my experience this works just fine and makes stenciling on solder paste a totally acceptable process for the hobbiest. Pololu just needs a Gerber file of your solder stencil design. Since EAGLE outputs that industry standard format, it's no problem!

You'll almost certainly have to order your solder paste since there don't seem to be local sources for it - even in fairly large cities. Stencils Unlimited seems to be very reasonably priced for solder paste. They'll have it shipped directly to you from their supplier. As far as solder paste itself, I recommend using lead-free, no clean flux. You can get this in a syringe and that should be about as much as you'll need for several small boards. Keep it sealed in two zip-lock bags in the refrigerator. Why not regular lead solder you ask. Well, aside from the toxic fumes, you'll find the shipping much more expensive and the refrigerator life much sorter. Since our Hot Plate Soldering System puts out plenty of heat, just use lead-free. (BTW, Stencils Unlimited also supplies prototype stencils. You'll find these to be much more expensive than the ones from Pololu, but I'm sure they're very nice and will work great if you want to spend the money.)

The Intersil apps note referred to above also discusses the soldering process and describes temperature profiles. While the documentation for the exact solder paste you chose will give more details, the Intersil explanation is very thorough and will tell you what you need to know. The Hot Plate Solder System you've built can come very close to the desired profiles. Certainly, it comes close enough to produce very satisfactory results based on my experience. We'll delve into solder profiles in a moment.

Here are a few additional hints that didn't seem to fit anywhere else:

Buy your components first and have them handy as you design the board. Make sure the footprints you use or create match your parts.

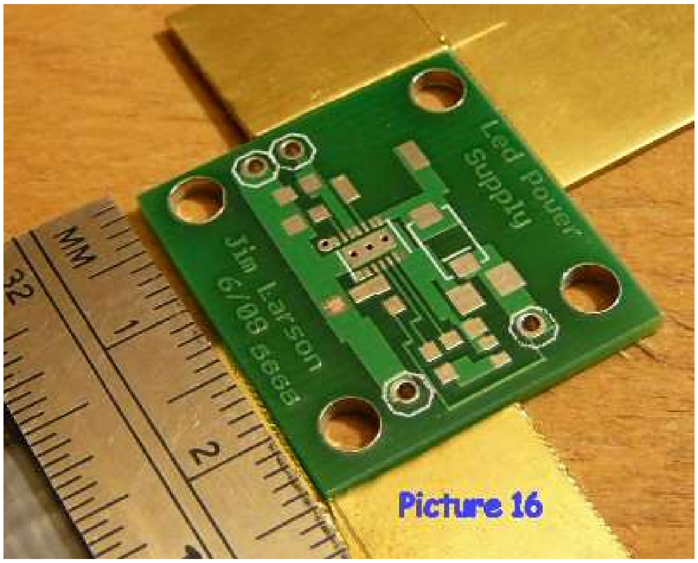
Print out the board design and place parts on it to be sure it all fits, that clearances are adequate, etc. Do this before you actually order the board.

Learn the limitations and restrictions of your circuit board manufacturer. These are known as Design Rules and you must know and observe them. The ones for BatchPCB are here . SparkFun provides an in-depth set and a great discussion of theirs here .

Professional PCB designers use checklists to be sure they don't miss any steps. You will find it helpful to create your own as you go along.

Read as much as you can - don't just use one info source. (Not even a great one like this Instructable. ;)}

Successful surface mount design requires some time and effort to learn. Start small, be patient, and practice. But it's not rocket science. Is it complex? Yes. Is it difficult? No, not really. Is it beyond a resourceful hobbiest? Not at all!



step 8: Preparing for Soldering

Now that we can design surface mount circuit boards we want to solder, let's move on with our Hot Plate Soldering System. Before we can to solder with it, we must stencil on solder paste and place parts on our circuit board. Let's go through the process with some pictures of an actual board I soldered. (I also created the board using the tools and process I just described.)

The board must be held securely while paste is applied. I use two pieces of thick brass shim stock with notches cut in them, and duct tape them onto a wooden base. See Picture 15 for details. The circuit board is held very firmly, but can still be removed without untaping the shims.

Place a board in position within the holder. Place the stencil over it and line up the openings as accurately as possible, then tape the stencil securely at the top. See Pictures 16 and 17. Picture 18 shows the holes in the stencil. Note the irregular opening on the DFN footprint – I made the openings in the stencil too small and had to hack it to get enough solder on. I don't recommend this!

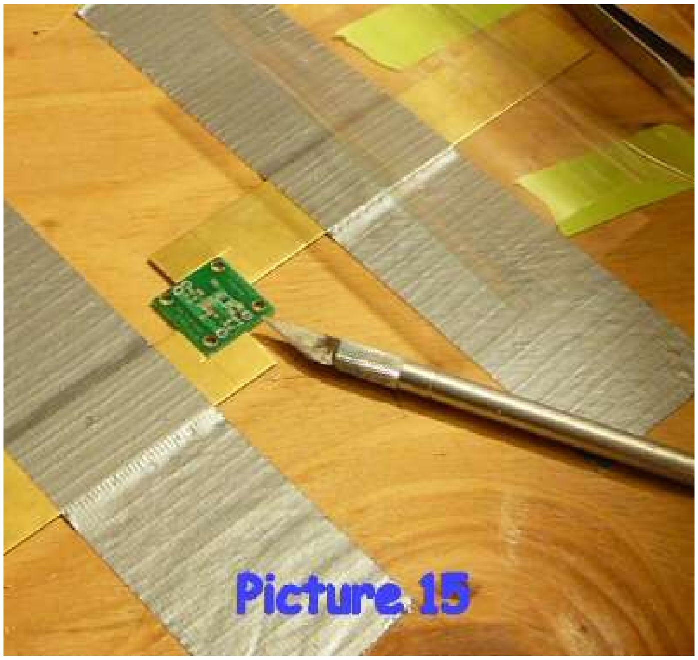
With the board in position and the stencil in place over it, apply a bead of solder paste along the top edge of board (Picture 19). The solder paste needs to be at room temperature for application, so be sure to take it out of the refrigerator at least 8 hours before you plan to use it. I'm using lead-free paste and it seems to be lasting just fine if I keep it refrigerated when I'm not using it. Just keep it tightly sealed and away from food – especially if you get paste with lead in it.

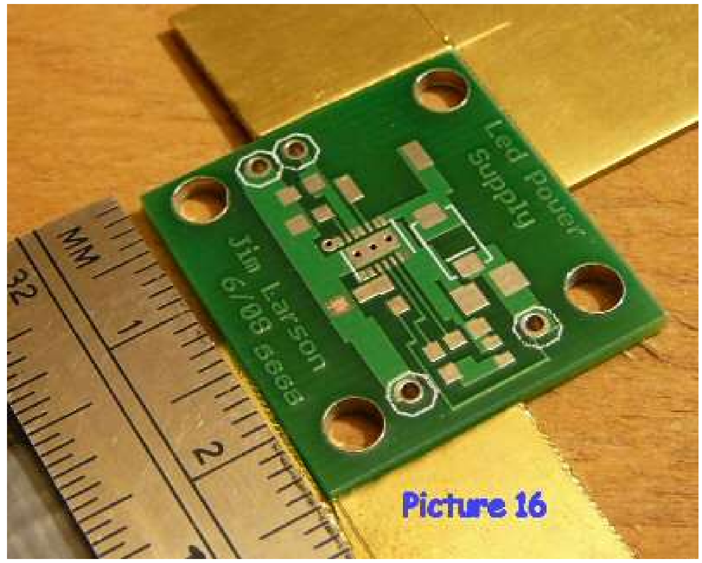
Use the razor blade as a squeegee and spread the solder paste. Ideally, you'll make just one pass across the board. Don't panic if you have to make an extra pass or two. That can cause a bit of smearing on your board, but that's not too critical in my experience. Picture 20 shows the stencil after the solder paste is applied.

Now lift the stencil off the board and tape it back so it can't flop back down while you're moving the circuit board. Use rubbing alcohol and a tissue to clean the paste off the stencil. Don't try to reuse leftover paste. Picture 21 shows the board after stenciling. You can also practice applying solder to the same board by simply removing the paste from the board with a tissue and some rubbing alcohol.

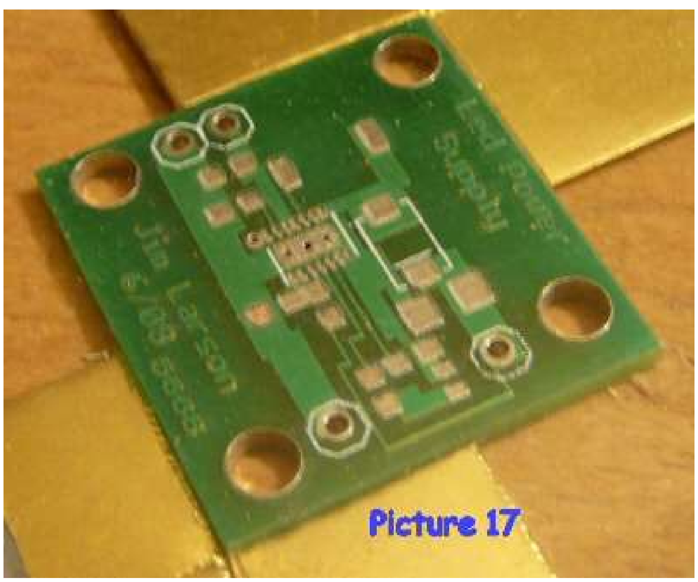
After applying paste it's time to place your parts. Try to arrange a static-free work space for parts placement. Static can destroy sensitive parts, so try to keep it minimized. If you're real concerned or if static is a problem in your house, then use a metal plate for a work surface and connect a wire from it to the ground pin of a nearby electrical socket. Be sure to touch the metal plate first before touching any parts to be placed. Create a parts placement diagram in your PCB design software and print it out. It's a good idea to enlarge this diagram for easy reading. One of mine is shown in Picture 22. Use a pair of tweezers – the parts are tiny! (See Picture 23.) Set them in place carefully with just a bit of force, but you don't have to mash them into place. If you're visibly squishing the solder paste, you're pushing too hard. Some parts are placed in Picture 24 and all of the parts are placed in Picture 25.

Once the parts are placed you're ready to solder! Handle your board carefully so you don't disturb the parts. I slide it onto a flat plate of some sort – metal, wood or perf board seem to work. Carry your board to your hot plate and slide it carefully onto the plate. The plate should be preheated and ready for the solder cycle, but I haven't told you how to do that yet! So let's turn our attention back to the hot plate and our control system.

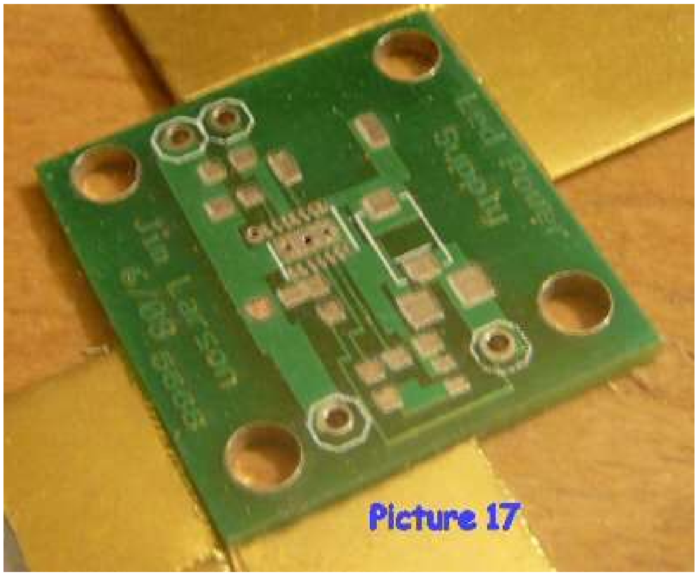

Picture 15



Picture 16

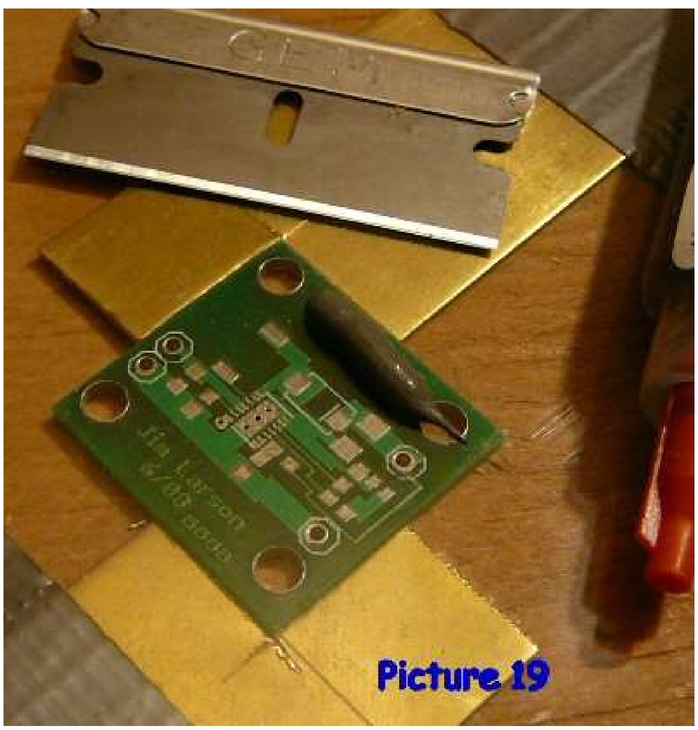


Picture 17

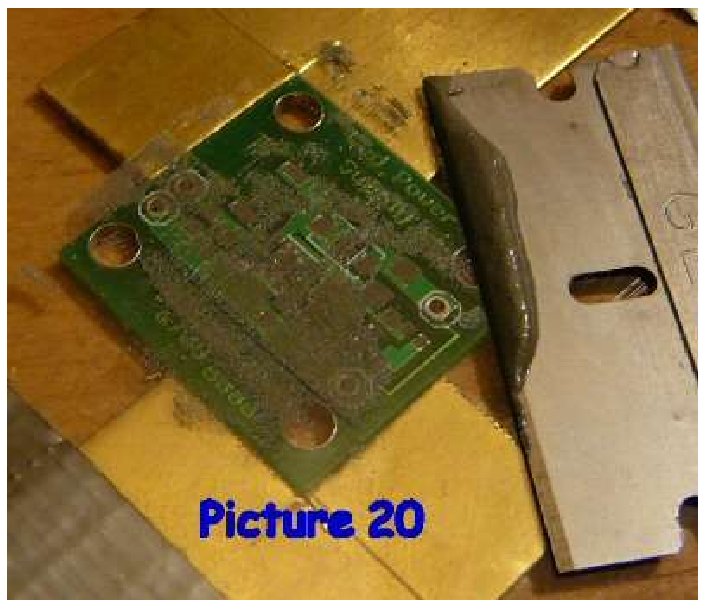


Picture 17

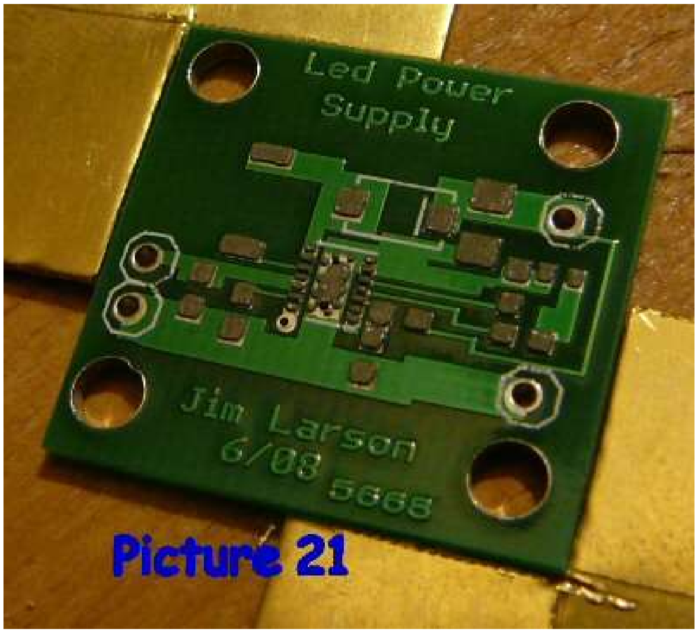
Picture 18



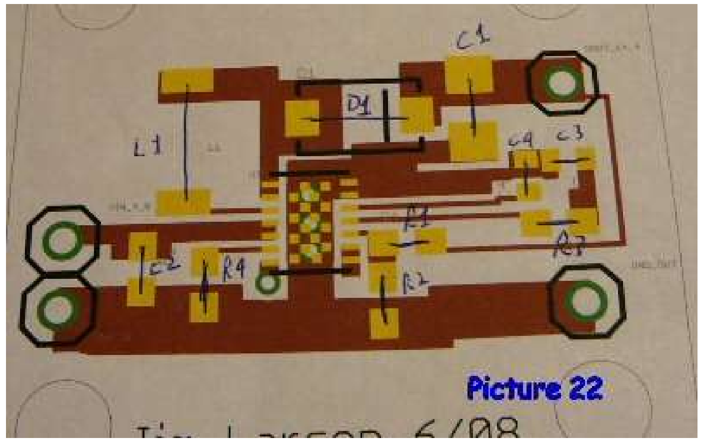
Picture 19



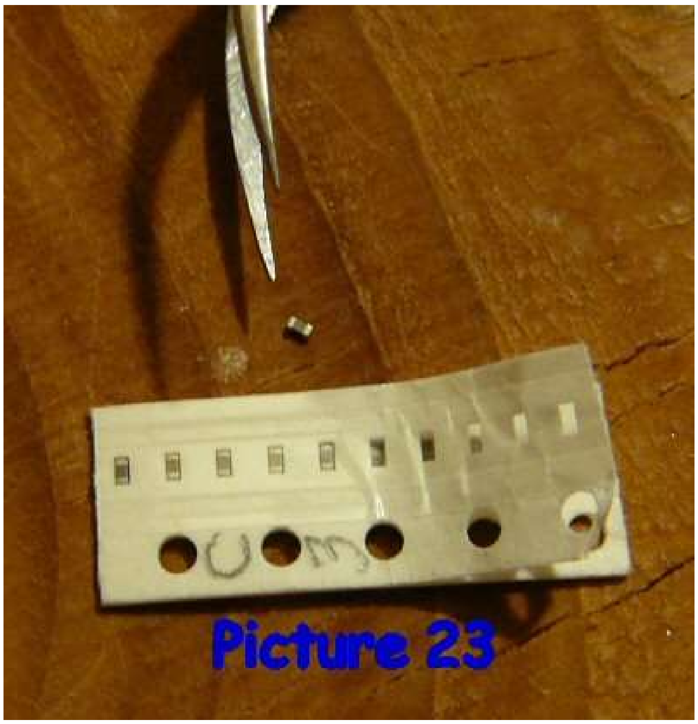
Picture 20



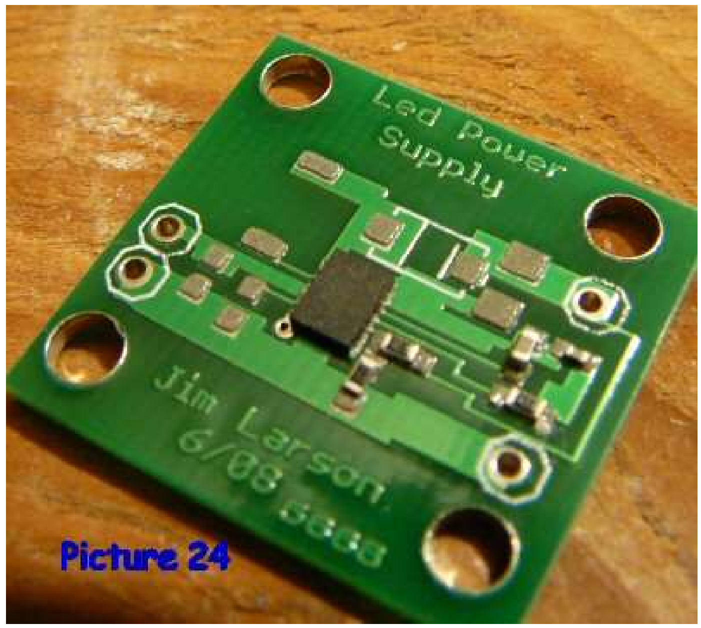
Picture 21



Picture 22



Picture 23



Picture 24

step 9: Using the Hot Plate Soldering System

The requirements for reflow soldering are fairly simple. Basically, the board and components must get hot enough to flow (melt) the solder. Along the way, the temperature must increase at the correct rate. Too hot and the flux boils and disturbs placed parts. Too slow and the solder oxidizes and won't flow correctly. The desired temperatures and the times associated with them constitute what is known as a Solder Temperature Profile. Have a look at Figure 3 and Table 2 in the Intersil apps note that you downloaded back in Step 7. These show a generic temperature profile. I'll show you how I interpreted this target profile and detail exactly the profile I used successfully. Remember, all my work is with lead-free solder paste. If you're using lead solder, you'll have to figure out the best profile yourself. Fortunately, it's not too difficult to provide a satisfactory profile. But first we have to have some way to know what temperatures our hot plate is reaching.

In this step, I will describe experiments that you will need to perform as you set up the Hot Plate Solder System. You should have several extra circuit boards that you don't care about for experiments. I had 8 boards and used 4 of them for experiments before I built a keeper. Two of my experimental boards had solder cleaned off them and were used a second time. Don't be afraid to experiment. It's the key to success.

Let's get started with the experiments. Hook up your hot plate and control system. In Step 6 we experimented a little with this, but now we'll proceed more methodically. I found that a cycle length of 10 seconds worked well, so we'll set that value for Cycle Length and not change it. We'll only change the On Time of the control pulse.

To control the soldering process we need to know what temperature the surface of the hot plate is reaching. The problem is that the hot plate doesn't have a temperature sensor, so we have to find one. The best sensor is probably a non-contact, infrared temperature gun . This can give a quick, accurate reading of the temperature of your hot plate. The SparkFun unit can only measure a maximum of 428F/220C which is not quite high enough (we'd like to measure up to 260C), but it's probably adequate. I managed to rent a unit locally with greater range, but two and half rentals would pay for the SparkFun unit.

A cheaper, but less accurate, approach is to use an oven thermometer (Picture 26) and bend the stand so it can lay flush on the hot plate. While this unit will not provide an accurate measurement of temperature, it will provide a repeatable reading and can be a useful reference. I'll explain how to do that. As you experiment, you'll want to record your settings, what temperature is indicated on the thermometer, and how long you wait between readings, for future reference. If you have an infrared sensor, record it's readings also. Don't put a PCB on just yet – that's the next step.

Bend the stand so the thermometer sits as flat as possible on the hot plate. Let it sit on your hot plate with the Hot Plate Control Unit's On Time set to 2. The temperature indicator will probably not quite reach the scale. Now increase the On Time to 4 and wait a few minutes until the temperature is stable and waiting longer doesn't show an increase. Reaching an equilibrium temperature took about 30 to 40 minutes on my hot plate.

Continue increasing the On Time in increments of 2, then taking readings after 10 minutes, 20 minutes, 30 minutes, and 40 minutes, then increasing the On Time by 2, taking readings again, until the indicated temperature reaches at least 200C on the thermometer or 280C on the infrared sensor. If you're using the SparkFun unit, you'll have to stop at 220C on the IR, but you'll have a good bit of data by then to guide you. You should get a good idea of two things: First, what is the difference in temperature readings between the oven thermometer and the infrared sensor? Second, how soon does the infrared sensor indicate that the hot plate is at equilibrium as opposed to when the thermometer reaches equilibrium? This data will be extremely helpful as you try to create a temperature profile for soldering.

The next step is to try melting solder on an actual circuit board. My experience is that I get more repeatable results if I let the hot plate cool down fully between soldering sessions. This takes about an hour, so you have plenty of time to stencil solder on to one of your spare boards. Don't put any parts on it just yet, we're just going to learn when the solder melts.

Now that the hot plate's cooled down, let's warm it up again. Based on my experience this warm up makes the rest of the cycle more repeatable. I use an On Time of 5 and wait 30 minutes for the warm up. The temperature on my oven thermometer reaches a nearly stable reading of 70C, the first mark on the temperature scale. Don't worry about the temperature profile just yet – that's later. First we need to know when the solder flows.

Place your board on the warm hot plate and let it warm up also, say 5 minutes or so. Then gradually increase the temperature. I found that my lead-free solder flowed when the thermometer read about 140C to 160C on the oven thermometer, so be careful as you approach this temperature range. Record the thermometer reading when your solder flows. You'll probably note that the solder paste will first spread out on your circuit board – mine actually spread out and touched between pads. Don't panic, this is OK. (This is also the reason I don't worry if I smear a little solder paste. Surface tension takes over when the solder flows and pulls the solder onto the pads.) Then you'll notice that the paste will turn a darker gray. Finally, it'll get shiny silver as it flows and beads up on the pads. That's the point when you want to record the temperature. Picture 27 shows a board with the solder just melted. Don't increase the temperature any further or you'll risk damaging (burning) your board. The temperature may climb a little as the hot plate overshoots. That's OK.

You should now have sufficient data to determine a satisfactory soldering temperature profile. Refer to the first paragraph in this section if you've forgotten what a soldering temperature profile is. The soldering profile I used is pretty simple. Here it is:
- Starting with the hot plate at room temperature, set the On Time to 4 and wait 30 minutes. The oven thermometer should read 70C or very close. The IR sensor will

probably read between 110C and 120C. If you're soldering a board, put it on the plate and wait two minutes for it to heat up.
- Set the On Time to 20 (remember to push the button) and wait until the oven thermometer reads 120C (six to six and a half minutes). The IR sensor will read about 220C.
- Set the On Time to 48 and wait until the oven thermometer reads 150C (eight to eight and a half minutes.) The IR sensor will read about 300C.
- Turn everything off. The oven thermometer will continue to increase, but shouldn't go over 160C. The IR gun could read as high as 315C. If you're soldering a board, wait about 30 seconds, no more than one minute, and slide the board off the hot plate onto the carrier to finish cooling.

Note: The temperature of the surface of the circuit board will be lower than the temperature of the hot plate. Based on when solder flowed, I estimate the temperature of the board to be about 20% lower than the IR temperature readings.

Don't worry if your IR sensor won't read over 220C. You should have enough comparison data between it and your oven thermometer to know if you're tracking my readings. If you are, then use my times and see what the results are. If you're readings don't track mine, you'll need to do some more experiments.

I've included results from the experiments I did. Please note that for the results shown I kept the hot plate on for nine minutes so the temperatures overshot what I would normally use (off after eight minutes). When I turn off at eight minutes, the shape of the ramp is nearly identical to that shown for nine minutes, the peak just stays below 260C (board temperature).

Look first at Temp Profiles. The top line is the IR sensor readings and the bottom line is the oven thermometer readings. Based on when the solder flows, I estimate the board temperature to be about 20% lower than the IR readings. That's the middle line. Again, I'd turn things off at eight minutes if I were actually soldering a board and the maximum IR temperature would not go over 315C.

Now let's compare what I measured to the target from the Intersil apps note. Look at Target Profiles. The upper and lower target profiles from Intersil are shown along with the board temperatures from my experiments. Making the adjustment for 8 minutes versus nine minutes, we see that the profile is pretty close to the target. Not bad!

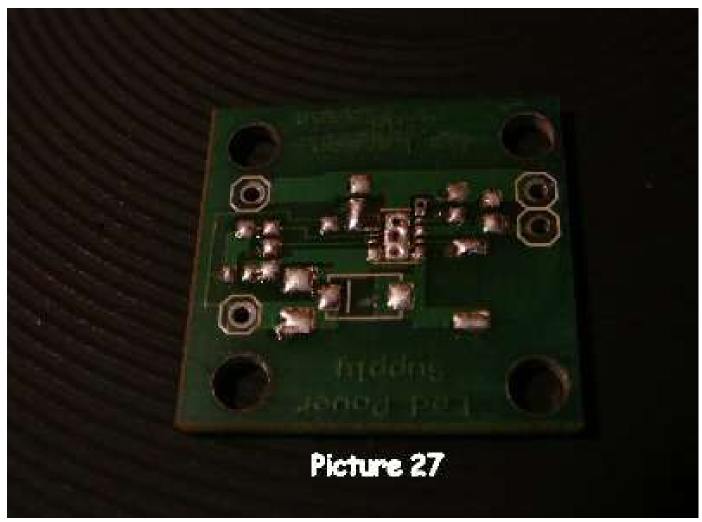
Two things should be pointed out. First, the cool down rate is kind of arbitrary. It's not clear to me what the impact of a rate considerably slower that Intersil suggests would be. But I probably don't care. What's most important is that letting the board cool briefly on the hot plate, then sliding it onto a carrier where it can finish cooling pretty quickly is probably just fine. Cooling too quickly could be bad, but putting the board on an insulator (like piece of wood or perf board) should be OK. Watch out for static electricity if you do this.

Second, the profile could be tuned more. Using a higher value for the initial On Time (say 24 instead of 20) would raise the initial rate of temperature increase. Since I've gotten satisfactory results, I haven't seen fit to do that. Please experiment to your heart's content.

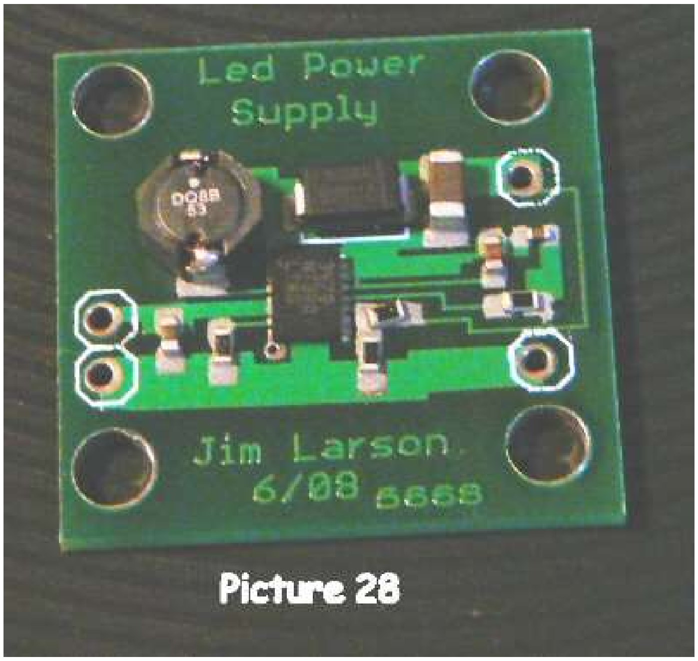
Once you have a profile that seems like it's close to the target, it's time to experiment by using it to solder. Stencil paste onto another board – still no parts. Use the profile you've determined to flow the solder on this test board and run your profile. Does solder flow when you expect it to (about when the oven thermometer reads 140C – about eight minutes)? If so, you're ready to solder some parts. If not, you need to run another experiment or two.

Now for the real thing. Stencil on the paste and place your parts. (If you're nervous and want to experiment a bit more, leave off any expensive or scarce parts the first time.) Be sure your hot plate has fully cooled, then warm it as discussed above. Put the board with parts onto the plate. Picture 28 shows my board with the paste heated. Picture 29 shows the board when the solder flowed. Run your solder cycle. When all the solder has flowed, turn off the hot plate, let it cool for 30 seconds to a minute (it'll still be mighty hot), and slide your circuit board off onto your carrier so it can finish cooling. How's it look? Great? OK, dance around the house some – you've just performed magic! (I still get a kick out of this process.) Picture 30 shows my final product.

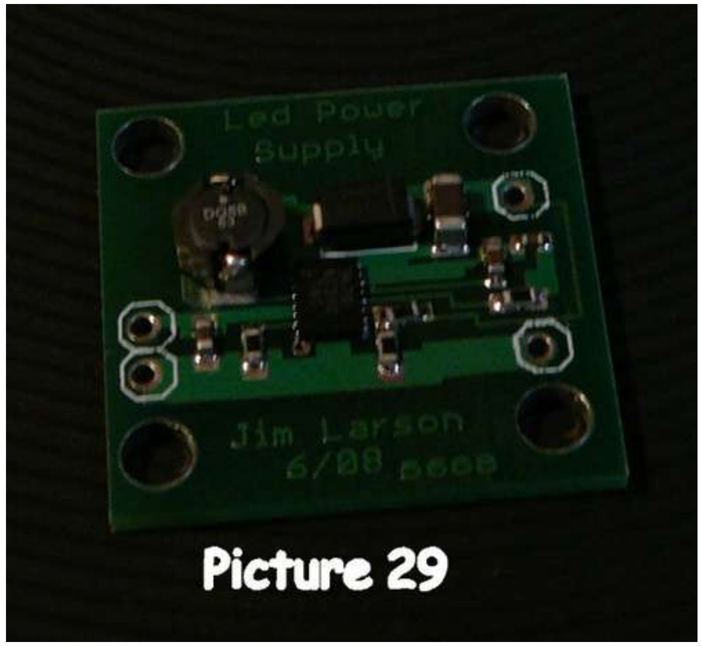
Picture 26

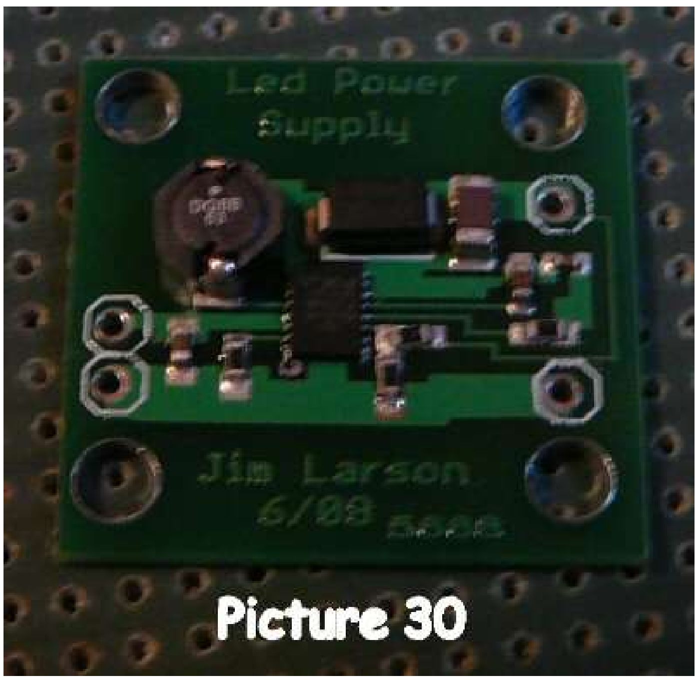


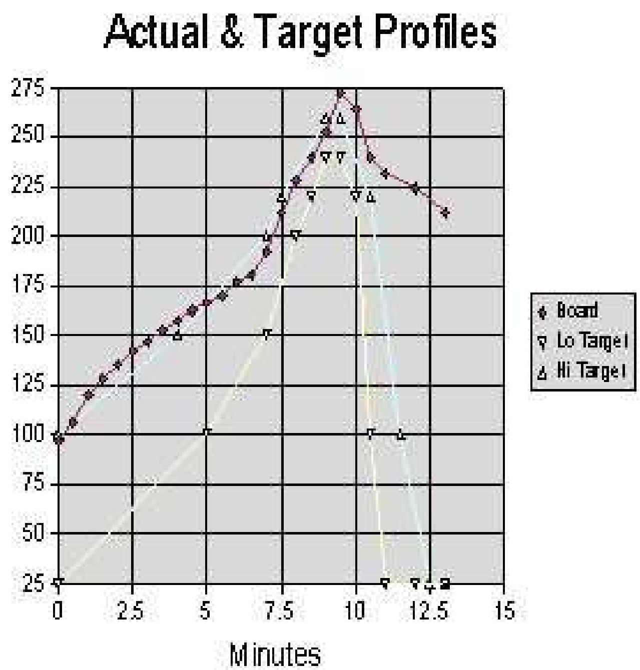
Picture 27

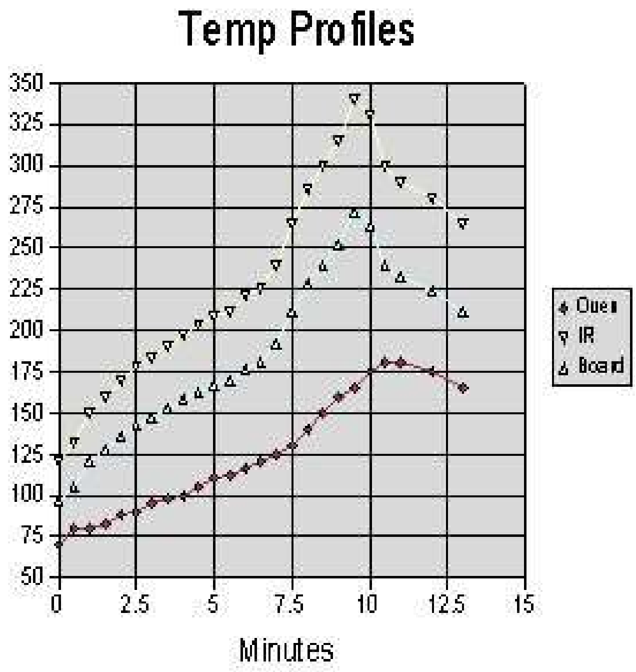


Picture 28


Picture 29


Picture 30

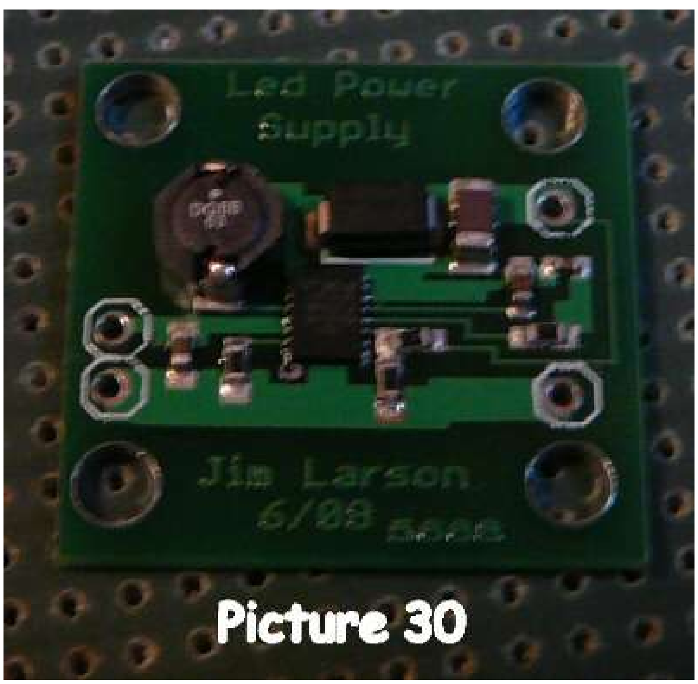




step 10: Final Thoughts and Conclusion

So let's say you don't plan to do your own designs, you just want to build the designs of others. How should you proceed? Scott Driscoll, in on his excellent web site, suggests applying solder paste with a 22 gauge syringe and discusses the technique. Although I haven't tried it, I think applying solder paste in this manner, placing your parts, and using the Hot Plate Soldering System to flow the solder should work great and much easier than trying to use a heat gun or a soldering iron, IMHO. For my own designs I still think the solder stencil is better and more reliable.

There you have it. A Hot Plate Soldering System to do surface mount soldering. Using this unit you will be able to solder the smallest surface mount parts Extreme Surface Mount! Your results should be suitable for prototyping and small production runs. Hopefully, you'll also find my suggestions for surface mount design helpful. Making your own gizmos using surface mount devices should no longer be a daunting challenge, just a satisfying experience.



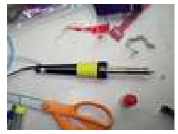
Related Instructables

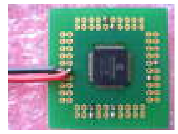

Easy SMT IC removal by electrotech

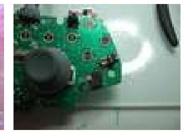

Quick helper for surface-mount soldering by bikeNomad


Making a fine tip for your solding iron for SMD soldering by MrZebra


Battery Powered Portable VU Meter by Vlorbschnat


DIY Hot Air Soldering Iron by charper


How to neatly solder (without loads of wires!) decoupling caps on SMT microcontrollers. by ste5442


Hacking the Xbox 360 controller by klee27x


How to hand-solder a PowerPad IC by endolith

Comments

50 comments Add Comment view all 76 comments

cgo says: Feb 23, 2009. 6:40 AM REPLY

Hi again. I completed your tutorial, it works great !
However, I'd like to point out this part: Sharp S116S02. It can replace the MOC+the SSRs+diodes+1W resistors, all in one $5 part. It needs a compatible heatsink, too.
Regards, Charles.

doctek says: Feb 23, 2009. 10:52 PM REPLY

Thanks for the suggestion. Looks like a real nice part! Here's two thoughts:
1 - Since it's a triac, it needs a snubber circuit as discussed in the Sharp apps note. One of the reasons I chose SCRs was to avoid needing a snubber. Not a big deal, just a couple of extra parts.
2 - The heat sink tab is not isolated, so you'll need an insulating washer on the external heat sink. Watch out for that.

The Digikey part number is 425-2399-5-ND and the price is $6.05.

musick7 says: Sep 5, 2009. 4:35 PM REPLY

Hi Doc!

This is Awesome! I would like to make my own and I was wondering if you had a YouTube Video of this in ACTION? I would love to see it working. Not that I don't believe you after all seems many have made one from your Tutorial in all the Reply's. I guess what I'm getting at is, How difficult is this to make? I am very confident in my Soldering Skills and would love to add this to my tools.
Is this expensive to make? I have quit a few different MicroChips, Resistors, Caps and so on... If I had to purchase all the Parts needed and I did Bargain shopping online getting the best price and giving each distributor a Max. of 10 days to receive parts ordered from them. What would be the total Cost?

And if you don't have a YouTube video where is another great demo of something just like this one? Does anyone else have there Plate on YouTube that they made from this Tutorial?

ewertz says: Aug 8, 2009. 2:22 AM REPLY

Can you safely pass on the snubber if you're sure that your load will only be resistive (like, for this?) ?

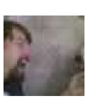
I'm asking because I've already got a small handful of triacs around...

Great write-up. Thanks!

doctek says: Aug 11, 2009. 8:15 AM REPLY

IF your load is purely resistive (like this hot plate), then you can skip the snubber.

This fact is discussed later in the project for the SCR circuitry used.


tundrawolf says: Aug 13, 2009. 6:34 PM REPLY

This is fantastic! I used to work with used test equipment and related equipment and wave soldering always fascinated me. I would like to see more steps and WAY more pictures. It is a lot for me to digest. But it is great nonetheless.


jimk3038 says: Feb 2, 2009. 9:15 AM REPLY

Hey,

Checkout this Non-Contact Laser Thermometer at Harbor Freight - it's only $30 bucks and it has a range up to 520 degrees C.

For $30 bucks you can't go wrong. Plus, I wonder if it could be hacked so that it could be connect to a PC.

Mine is already on order!

Jim

Polymorph says: Aug 5, 2009. 6:50 PM **REPLY**

It is important to know that most noncontact pyrometers are calibrated for 80% reflective objects. IE, grey to black. Just so you know.

Lighter objects read as a lower temperature than they actually are. A dark green PCB may read correctly or a little low. Tshirt pressers often stick a piece of black high temp tape (looks like masking tape) onto a white or natural aluminum press when testing the temperature.

---

ewertz says: Aug 2, 2009. 9:22 PM **REPLY**

Grrrrr. It's $60 now.

---

doctek says: Feb 3, 2009. 10:20 AM **REPLY**

Great find! Perfect if for this application.

Thanks for the pointer!

SMTstencils says: Jun 23, 2009. 6:46 PM **REPLY**

hello there,

Since you your discussion involves SMT stencils, I thought you may want to know that now Applied Electronics provide both Mylar and Kapton SMT stencils. Starting this coming Thursday (June 25th), we will take order for Kapton too. You can choose either Mylar or Kapton of size 8.5x11 inch (actual size for Kapton is 8.5x12) without any restriction on the area or number of components.

Mylar comes in thickness of 3 and 4 mil.

Kapton comes in thickness of 3 mil only.

Visit the following site for detail

http://www.applied-electronics.com

Regards
Applied Electronics

---

countable says: May 21, 2009. 12:52 PM **REPLY**

I do surface mount as part of my course, and I have to say it's very interesting. We have a machine that links to a PC and will automagically pick and place the components for you (assuming the solder paste is already there). For a laugh, i made a TDA2822D mini amplifier by placing all the components by hand. The only thing i can say about it is it helps to use a microscope at that kind of level!

Gilius says: Jan 31, 2009. 12:10 PM **REPLY**

Instead o using solder paste can I apply solder to the board instead? (like tinning)

The solder would then re-melt and secure the components to the board.

---

doctek says: Jan 31, 2009. 3:01 PM **REPLY**

Yes, I suppose you can do that. If you do, be sure to put additional flux on the board before your reflow it (Scott Driscoll discuss this on his site which is mentioned in the Instructable).

Now here's why I'm not keen on this approach.
- Getting solder on all those tiny pads is a **lot** of work. The whole point of using a stencil and solder paste is to avoid that work.
- Making the solder reasonably level is nearly impossible, so some pads may not start off touching. This may or may not be a problem, but if there's too much solder on a central pad, then the resulting pillow could keep the other pads from ever touching. Paste and the correct footprints avoids this problem.
- It's **really** easy to lift a pad from the board with a little too much heat when you're tinning them. Tinning the part itself is a better plan, but I still don't like it.

Hope that helps.

Chris2048 says: Apr 27, 2009. 6:18 PM **REPLY**
What about chemical tinning? I imagine it wouldn't have these problems.

doctek says: May 2, 2009. 3:17 PM **REPLY**
Not sure exactly what you mean here. Do you mean a process such as hot air leveling (HAL)? Generally, this isn't used for fine pitch surface mount due to poor planarity. Also, you still have to put on solder paste.

Perhaps there is another process of which I'm not aware (very possible!). Could you elaborate on what you have in mind?

Gilius says: Feb 1, 2009. 4:08 AM **REPLY**
It's an alternative becuase I can't seem to get my hands on any paste. :(

doctek says: Feb 1, 2009. 12:25 PM **REPLY**
Paste is something you must order. Check out the link in the Instructable. You can just buy a syringe of it.I suggest getting lead-free paste for a few reasons:
- It's lead-free! (no fume dangers)
- It's less sensitive to a little bit of temperature during shipping so you don't have to go a real expensive route (with dry ice, etc.)
- If you use the hot plate method in the Instructable, you've got plenty of heat available.
- It's the wave of the future - lead will be gone completely in a few years.

So get lead-free, no-wash solder. Works great. If you live in Denver, you may be able to will-call it at a supplier, but most anywhere else, you'll have to order it. Nobody seems to stock it.

dagenius says: Apr 21, 2009. 9:32 AM **REPLY**
The fumes are actually from the flux boiling off of the solder, but in general I try to avoid lead, for various reasons, one of which is that if I touch it, THEN it may get into my system.

Gilius says: Feb 2, 2009. 2:07 AM **REPLY**
I'm not from the US. I'm in Romania and while we have large electronics suppliers here they refuse to sell to hobbyists because we're not companies. :(

However I prefer working with lead solder because it has lower melt temperature and it's less brittle than the ROHS stuff. Considering it's paste it's not a problem but I'll never normally use lead-free solder wire. I hate the stuff.

wierd idiot says: Feb 22, 2009. 2:43 PM **REPLY**
I agree lead free solder wire cracks very easily and it doesn't like to reheat either.

ste5442 says: Feb 2, 2009. 11:32 AM **REPLY**
I agree, lead-free solder doesn't flow like leaded and creates a dull joint (purely cosmetic I know!). For prototype and home jobs I use leaded.
The paste, when bought, is shipped sealed but once opened needs to be stored in the fridge (see the product datasheet) - none of the paste I have used is shipped in dry ice!

PCBPolice Electronics Forum

Chris2048 says: Apr 27, 2009. 6:19 PM **REPLY**
silver solder isn't that expensive, but I hear it flows well, and doesn't crystallize.

doctek says: Feb 3, 2009. 10:28 AM **REPLY**
It's easy to become "US-centric". I apologize for that.

If you can't get paste, then tinning is your best option. Tin the part not the board to avoid lifting pads. The Intersil apps note I reference provides good guidelines on how to tin parts in the Rework section. Can't blame you for preferring leaded to lead-free solder!

(I've seen dry ice specified for shipping lead-containing solder paste - not for lead-free.)

Gilius says: Feb 3, 2009. 11:15 AM **REPLY**
Don't worry I'm not upset or anything. :)

I'm going to try tinning when I get back to my projects next month. I'm also looking into something called a tinning brush and I've located a supplier for it. It might prove interesting.

I'll drop a note and perhaps a video of SMD soldering with the tinning method.

ste5442 says: Feb 3, 2009. 2:23 PM REPLY
If you make a short video I would appreciate it if you could post it on my (very low on users) forum?
The forum is new but we have a number of actual HW engineers ready to answer you questions - its just a bit slow to take off (its only a few days old).

PCBPolice Electronics Forum - we need some users....please!

Gilius says: Feb 3, 2009. 11:58 PM REPLY
Sure. But as I said I won't be doing any projects till march.

Polymorph says: Aug 5, 2009. 6:54 PM REPLY
We'd be honored if you'd drop a note at the Homebrew_PCBs list if you get around to making some videos. We've got over 4500 members.

Homebrew_PCBs

dagenius says: Apr 21, 2009. 7:27 AM REPLY
Cool 'ible, but it is very long, pictures are sparse, and not to the point. Just try to improve on that in future instructables.

ecson486 says: Apr 20, 2009. 6:28 PM REPLY
Awesome

cgo says: Feb 12, 2009. 12:33 AM REPLY
Hi, great Instructable, thank you. According to the datasheet, I believe the SCRs will do for 220V too, right ?

doctek says: Feb 12, 2009. 9:46 AM REPLY
You are correct on the SCRs (I like to over-design a bit), but the resistors would have to change. Look at figure 15 in the MOC Data sheet / apps note for details.

cgo says: Feb 13, 2009. 7:13 AM REPLY
Thanks for confirming me about the SCRs.
I will build your design with a little extension: with a dual hot plate, it will be possible to keep one plate at warm-up temperature, and the other switch between the two higher temperatures. So you won't have to wait one hour for the plate to cool down again. (since I have dozens of boards to assemble, time matters to me :))
Cheers

doctek says: Feb 14, 2009. 12:04 PM REPLY
It's not exactly clear what you plan here. Do you intend to control two hot plates with the same controller? Check power requirements first. Two would draw (potentially) twice as much current and exceed the capabilities of the components.

If you want to run two hot plates, then I would suggest using one controller and one plate to run thru the solder cycle. Then unplug that plate, plug in the second plate and run thru the cycle on it. You'll have to check the temperature that the unplugged plate cools down to - if it's significantly higher than about 80 to 100C, you may need additional waiting time. Maybe a third plate? If you're doing as many units as you say, then another $15 to $20 for a third plate is pretty cheap.

cgo says: Feb 16, 2009. 4:17 AM REPLY
Hi,
your idea of round-robin between plates sound good !
What I had in mind was having a warmup plate and a reflow plate. But it would require the reflow plate to cool down to 110-120°C anyway...
Your idea looks better. Thank you for your time.

scootyre says: Feb 8, 2009. 10:06 PM REPLY
Excellent Instructable. I can add a couple points:
It's the flux in the solder paste that requires refridgeration (or dry ice shipment). Most lead bearing solders use a more active flux, it needs to stay cold to keep from cooking itself off in the syringe or jar before it is used. No clean fluxes are usually not as active, and usually don't have as high a solids content.
The solder paste supplier is probably your best choice for heating profiles. The flux (plus the rest of the magic goo) in the paste will dictate how fast you should heat it up, and how long you can hold it above the solder melting point. The alloy in the paste will determine what temperature the solder goes liquid (Sn/Pb is 183C for electronics solder, Sn/Ag/Cu is 217C) and the peak temperature you need to achieve for good wetting (Sn/Pb 205-220C, Sn/Ag/Cu 230-250C).
Cooling rate of the solder joint determines grain size - slow cooling will create larger grains and less reliable (over time) solder joints. Cooling too fast will damage ceramic capacitors. Heating too fast will crack them, too.
Do not to heat faster than 2 degrees C/sec, don't cool faster than 4 degrees C/sec and the caps will be fine. Many solder pastes will work fine with the heat ramp, the flux won't outgas and pop...
A 3 mil stencil seems awfully thin - I run 5 or 6 mil stencils at work. I would suggest making your stencil aperture slightly larger than the pad for the caps and resistors to get enough solder volume into the joint.
I would try Techni-Tool as a source for solder paste in North America. I have used Kester brand solder at work for years, check out their website for profile

information.

doctek says: Feb 12, 2009. 9:59 AM **REPLY**

Regarding the 3 mil stencil versus 5 or 6 mil: 3 mil is the thickest that Pololu offers for the tiny features on QFN/DFN/MLF packages, so that's what I used. Seemed to work fine, but I understand your concern. Certainly you could get 5 or 6 mil stencils, but you'll pay a lot more. Using a slightly larger opening is a good idea - hand rework would be possible on large Rs and Cs as well (although I hate that idea).

Thanks for the excellent input.

a2retro says: Feb 9, 2009. 8:35 PM **REPLY**

Hi nice job. Thanks for the info. Couple of questions for you. The box that you cut in half to mount the terminal strip to, is it screwed to the bottom of the double receptacle box? Is the perf board mounted to the top of that somehow or is it just supported but the wires coming from the perf board?

I was wondering it it would be acceptable to make the perfboard large enough to use the normal receptacle mounting holes and then use marrets to join the wall pull and socket to the SCR wires?

If I make this I am thinking about also getting a triple receptacle box and putting the uC controller circuit on a third perf board so it's all in one.
Would that be okay?

doctek says: Feb 12, 2009. 9:55 AM **REPLY**

The half box is not screwed down, nor is the circuit board mounted in place. Both these could be done (and not a bad idea), but I found the stiffness of the wiring more than sufficient to hold things in place.

Making the perf board larger and using the receptacle holes is totally acceptable. When I made mine I was undecided on exactly where I wanted to mount the perf board, so I kept it small for flexibility in mounting. Your idea was something I considered and should work great. Using marrets (wire nuts) instead of the connector block would work fine also since you wouldn't need the half box if you mount the perf board to the box.

A triple box with the uC on another perf board would make a compact, complete system. I kept the uC separate since I use it for other things when I'm not soldering.

Nice suggestions, thanks.

przemek says: Feb 5, 2009. 7:24 AM **REPLY**

BE CAREFUL WITH LEAD-BASED SOLDER PASTE

while regular solder is fairly safe because you can't get significant quantities of it into your body (ignorant people used to chew on lead because of the slightly sweet taste of lead oxides), the soldering paste contains finely ground metal particles; if you get them into your digestive tract, there might be enough surface area for the body to absorb dangerous quantities. If you keep it in your refrigerator, and someone grabs it and makes a sandwich with it, or even if you get it in your mouth because you didn't wash your hands after soldering, you could get lead poisoning which is no fun. Avoid the fate of the Roman Empire and the Shackleton expedition---two tragedies that are blamed partly on effects of Pb poisoning.

doctek says: Feb 8, 2009. 10:29 PM **REPLY**

A very important caution. Thank you!

Another reason to use lead-free paste, even if it's a bit harder to work with.

snarfer says: Feb 6, 2009. 2:40 PM **REPLY**

Great instructable!

I like to get my stencils from Ohararp.com, supposedly kapton can be laser cut slightly more accurately than mylar.

Also, might want to note that when you need to replace those QFN parts you're going to need some sort of rework station with a hot air gun, unless you want to put the entire board back on the hot plate...

I got a nice one for 90 dollars, but there were cheaper ones available for half that.

Finally, just wonder if you've had any trouble with uneven heating on your hot plate, any ideas on how to address that. I was soldering one board recently that was kind of long and thin, the ends never melted but the middle overheated. Maybe I will try putting a piece of aluminum underneath next time.

doctek says: Feb 8, 2009. 2:32 PM **REPLY**

Great input! Thanks for adding to the discussion.

Think I'll try Ohararp.com next stencil. Sounds like a good product at a fair price. Pololu has been very good however.

Yes, for rework you need a hot air gun. In fact, some places (like curiousinventor aka Scott Driscoll) suggest two of them, one front and one back. I mostly have avoided rework by using cheap parts and small designs. I know I'll have to deal with this someday.

The plate I have seems to give even heat, but I've used only small designs. If you're trying a new design for the first time, then plan to get at least one or two extra boards to experiment with.

The aluminum plate idea is one I've seen mentioned in other places. I think it's a good one, but you'll have to work out the correct solder profile if you use one on your hot plate.

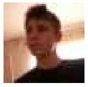
Again, thanks for the input.

fredob says: Feb 5, 2009. 9:24 AM **REPLY**

Wonderful kit !!! Be careful not to "cycle" the solder paste too often between fridge & room temp. - this leads to absorption of water, making the paste "spit" when trying to solder. Don't forget that water boils way before solder melts. I have seen it spit over 10mm away from a pad.

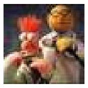
BlackoutXIII says: Feb 5, 2009. 7:51 AM **REPLY**

Great instructable. Very well-done.



alex-sharetskiy says: Feb 5, 2009. 7:00 AM **REPLY**

i don't like SMD anything, because you need a circuit board to use it...



ste5442 says: Feb 2, 2009. 11:43 AM **REPLY**

Nice instructable but normally you would heat both sides simultaneously or at least pre-heat the underside so as not to warp the PCB.
The hotplate approach is a neat (and cheap) trick for home use though! beats spending a fortune on a rework machine or vapour phase!!
Nice work!

PCBPolice Electronics Forum

doctek says: Feb 3, 2009. 10:46 AM **REPLY**

The profile I discuss includes time to allow the circuit board to warm up (pre-heat) before starting to ramp up to reflow. I haven't had problems with warping and apparently neither has SparkFun. But it's something to be aware of and affirms the importance of pre-heating.

Glad you enjoyed it. Coming from someone with your experience (and scars) that means a lot. Thanks,

yzf600 says: Jan 31, 2009. 6:01 PM **REPLY**

Would you perhaps give a little detail on what you made your solder stencil out of and how you made it? I'm thinking I can make my own out of mylar using the Epilog laser at work. What layer is used for the stencils? The tStop layer?

I've made a layout in Eagle using MLF44 and a QFN32 packages. My tStop layer has a solid rectangle over the pads on each of the 4 sides. I would think it would be better to have holes for individual pads like your stencil. That may not be a realistic expectation since the space between the pads is only 0.2mm.

doctek says: Feb 1, 2009. 12:40 PM **REPLY**

The layer you care about is the "tCream" (Top Cream) layer. Basically, this layer will have just the surface mount device pads - no pads around vias or holes. You should rework the opening for the center pad (exposed paddle) so that you only cover 50% to 80% of it with paste. Have a look at the On Semi or Atmel apps notes I have provided links to. They show some ways to do this.

Pololu (see link I provide) uses 3 mil mylar for the stencils they make. They basically promise to "get some paste" on every pad. Not precise, but it worked great. If I was making the stencil, I'd aim to keep the openings no larger than the actual pads on the board. Go smaller if you must in order to be sure you have individual openings for each pad.

Another thought: The fact that your tStop layer has a solid rectangle over your QFN/MLF pad areas means you won't have any solder mask between pads. With 8mil line/space rules and allowing some tolerance, you would risk getting solder mask on the pads if you tried to put mask between them. I tend to accept the risk in order to have mask between the pads, but your pads are spaced even closer than mine were. Please let us know what happens when you try solder these tiny pads with no solder mask between them.

Good luck!

view all 76 comments