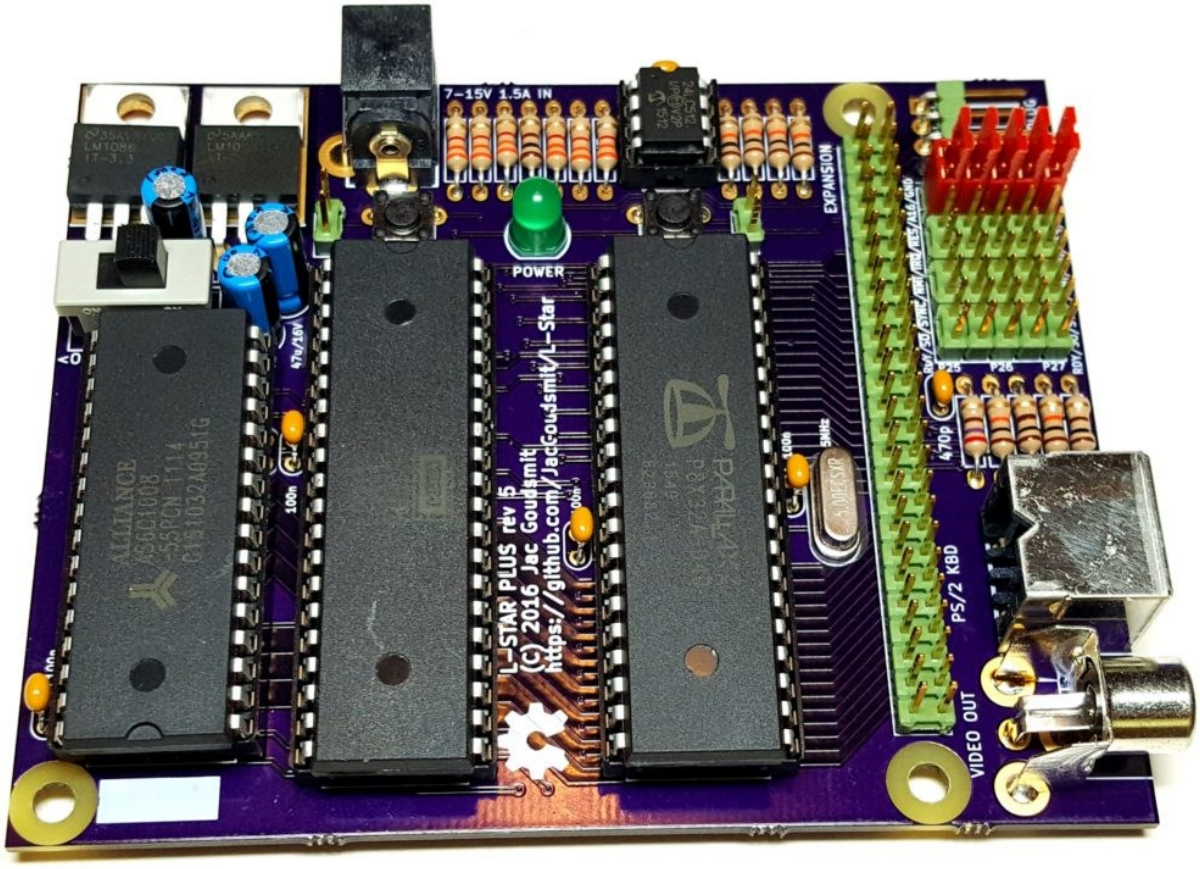
# L-Star Plus
# Building Instructions
# PCB Rev. 5






https://L-Star.org

# 1. Table of Contents

# 2. Introduction

Thank you for purchasing the L-Star Plus kit. It was a lot of fun to design the L-Star, and I hope it will be a lot of fun for you to build your own, and discover what you can do with a Software Defined Computer.

If you've already done plenty of soldering before, this document probably won't tell you many of things you don't already know. Nevertheless, you may want to glance over it while you build it, to catch some helpful hints.

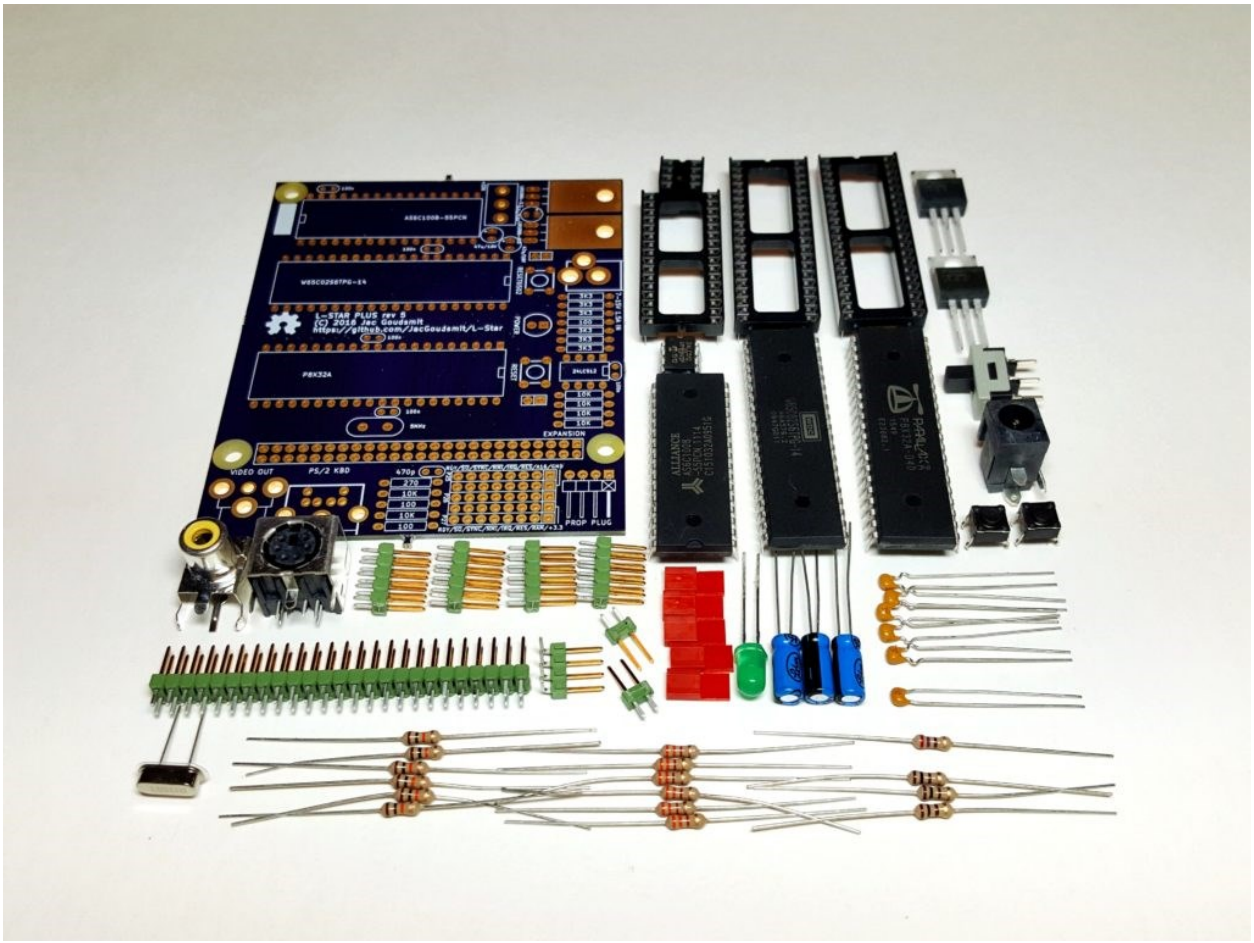
On the other hand, if you've never soldered anything before, I recommend that you ***don't build L-Star Plus as your first soldering project.*** I can't stop you of course, and L-Star is really not difficult to put together, but it will be a lot more fun to do if you already know how to avoid pitfalls such as:

- Using the wrong solder iron or the wrong tip or the wrong solder tin
- Overheating electronic parts as well as body parts
- Causing cold solder joints
- Causing short circuits
- etc.

I suggest you Google for "Learn how to solder" and learn some of the basics, even if you're only going to watch some pictures or online movies that show how to do it right. You can also get low-cost kits from many sources that will teach you all the details while you do it.

I tried to keep the instructions easy enough for a beginner to understand. Nevertheless, teaching you how to solder and how to recognize the components is outside the scope of this document. If you get stuck, please contact me via the Tindie website.

# 3. Let's Get Started



You should have received the following components. Please contact me if anything is missing:

| | |
|---|---|
| 1 | Printed Circuit Board |
| 1 | SRAM chip: Alliance AS6C1008-55PCN (128KB), with 32 pin socket |
| 1 | 65C02: Western Design Center W65C02S6TPG-14, with 40 pin socket |
| 1 | Propeller: Parallax P8X32A-D40, with 40 pin socket |
| 1 | EEPROM: Microchip 24LC512-I/P (64KB), with 8-pin socket |
| 1 | LDO Voltage Regulator 5V/1.5A: Texas Instruments LM1086-5V |
| 1 | LDO Voltage Regulator 3.3V/1.5A: Texas Instruments LM1086-3.3V |
| 1 | Crystal 5MHz with insulator |
| 1 | Green LED |
| 3 | Electrolytic Capacitor 47uF/16V |
| 5 | MLCC Capacitor 100nF |
| 1 | MLCC Capacitor 470pF |
| 3 | Resistor 100 Ohm (brown, black, brown, gold) |
| 1 | Resistor 270 Ohm (red, violet, brown, gold) |
| 6 | Resistor 3300 Ohm (orange, orange, red, gold) |
| 6 | Resistor 10k Ohm (brown, black, orange, gold) |
| 1 | RCA socket (monochrome video output) |
| 1 | Mini-din 6 socket (PS/2 keyboard) |
| 1 | Power socket (barrel connector 2.5mm) |
| 1 | Power switch |
| 2 | Tactile switch (push buttons) |
| 1 | Header 1x4 pins, right angle (Propeller plug) |
| 1 | Header 2x25 pins (Expansion port) |
| 4 | Header 2x5 pins (Jumpers) |
| 2 | Header 1x2 pins (Reset jumpers) |
| 5 | Shunt (for jumpers) |

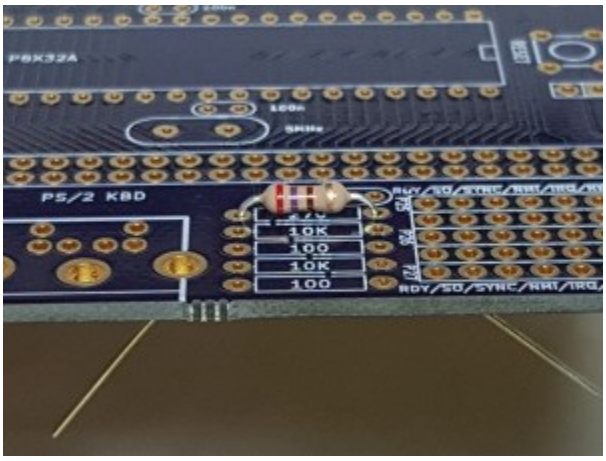
When you get the kit, the IC's are mounted on a piece of conductive foam to protect them from electrostatic discharge (ESD) during transport. ***Leave the chips on the conductive foam until you're ready to mount them on the board.*** The other parts are not static sensitive, but I may use the foam to protect other parts from getting damaged, or causing damage to other parts.

# 4. Resistors

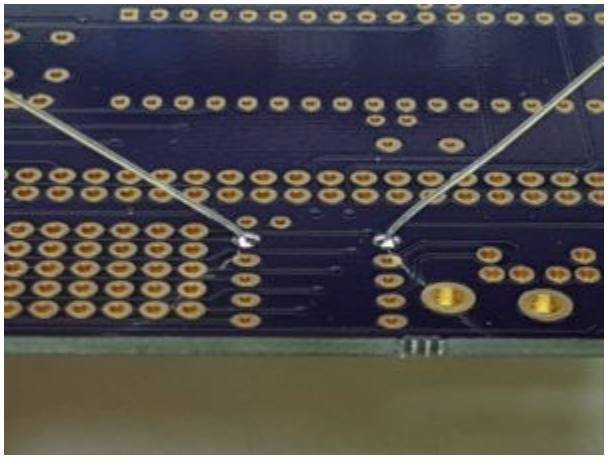
The first resistor you're going to solder onto the board is the 270 Ohm resistor. The rings are colored red-violet- brown-gold. It doesn't matter which way you solder it but I like to be consistent and solder all resistors in the same direction just to make the product look good.

Bend the leads as shown in the picture above, and insert the wires into the board on the top side, where it says "270".

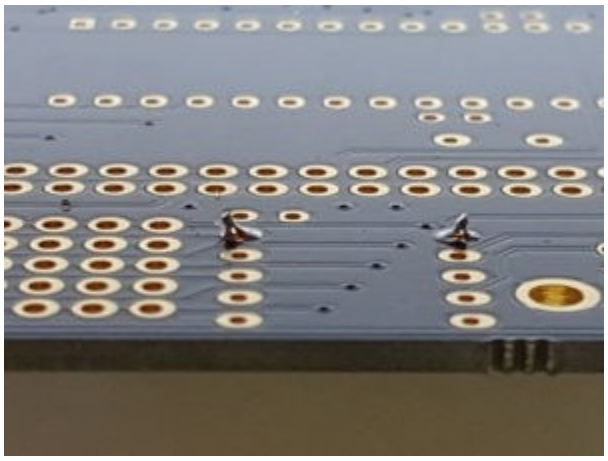
While you hold the resistor against the board, bend the wires on the other side so they're at 45 degree angles.



Then, turn the board over and solder the wires.



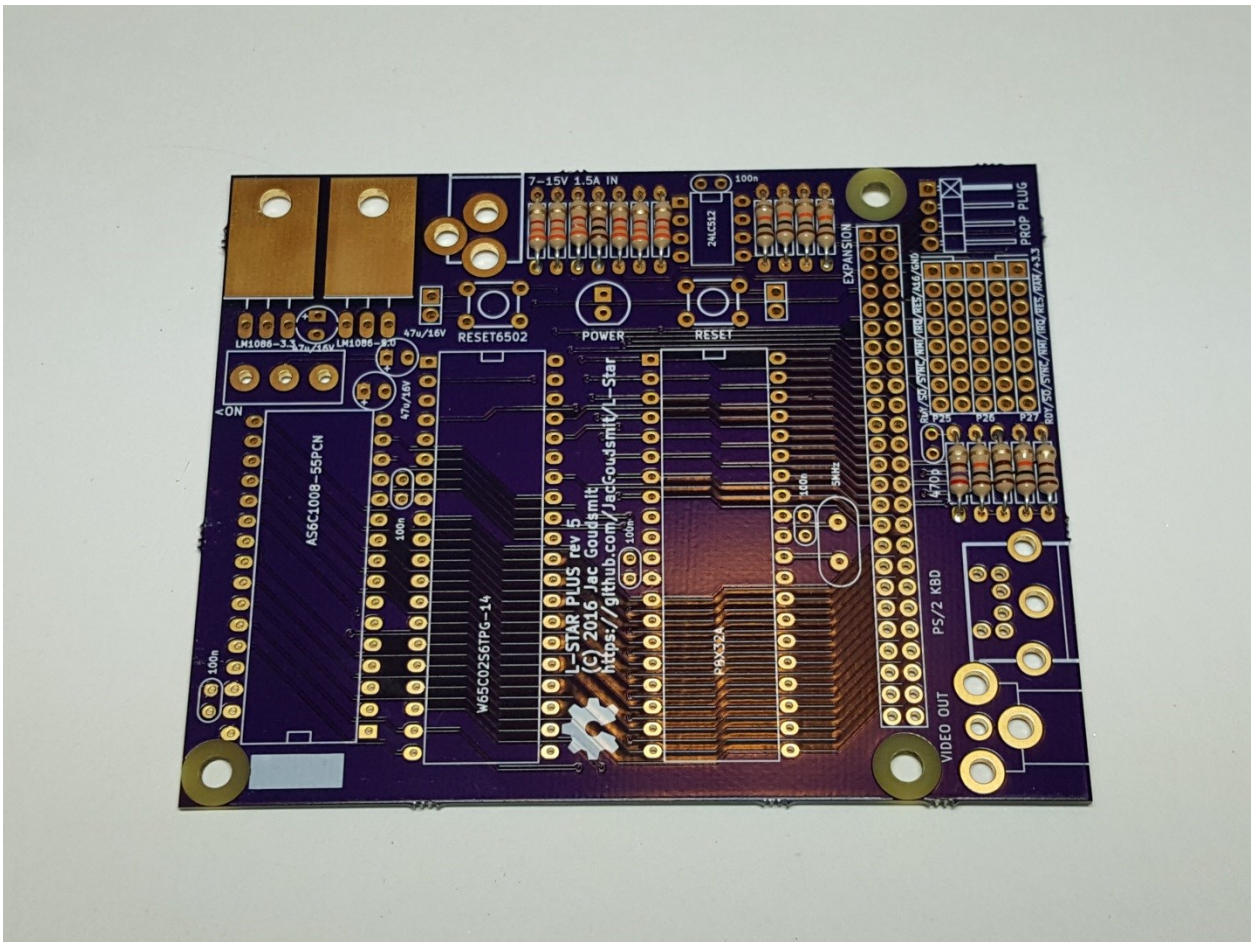
Then use side cutters to cut the wires as close to the board as you can.



Repeat this for the other resistors. These are the values, their color codes and their quantities. **I recommend soldering them in this order, to prevent confusion between similar values**:

| Quantity | Value | Color code |
|---|---|---|
| 1 | 270 Ohms | Red-Violet-Brown-Gold |
| 6 | 10K | Brown-Black-Orange-Gold |
| 3 | 100 Ohms | Brown-Black-Brown-Gold |
| 6 | 3K3 | Orange-Orange-Red-Gold |

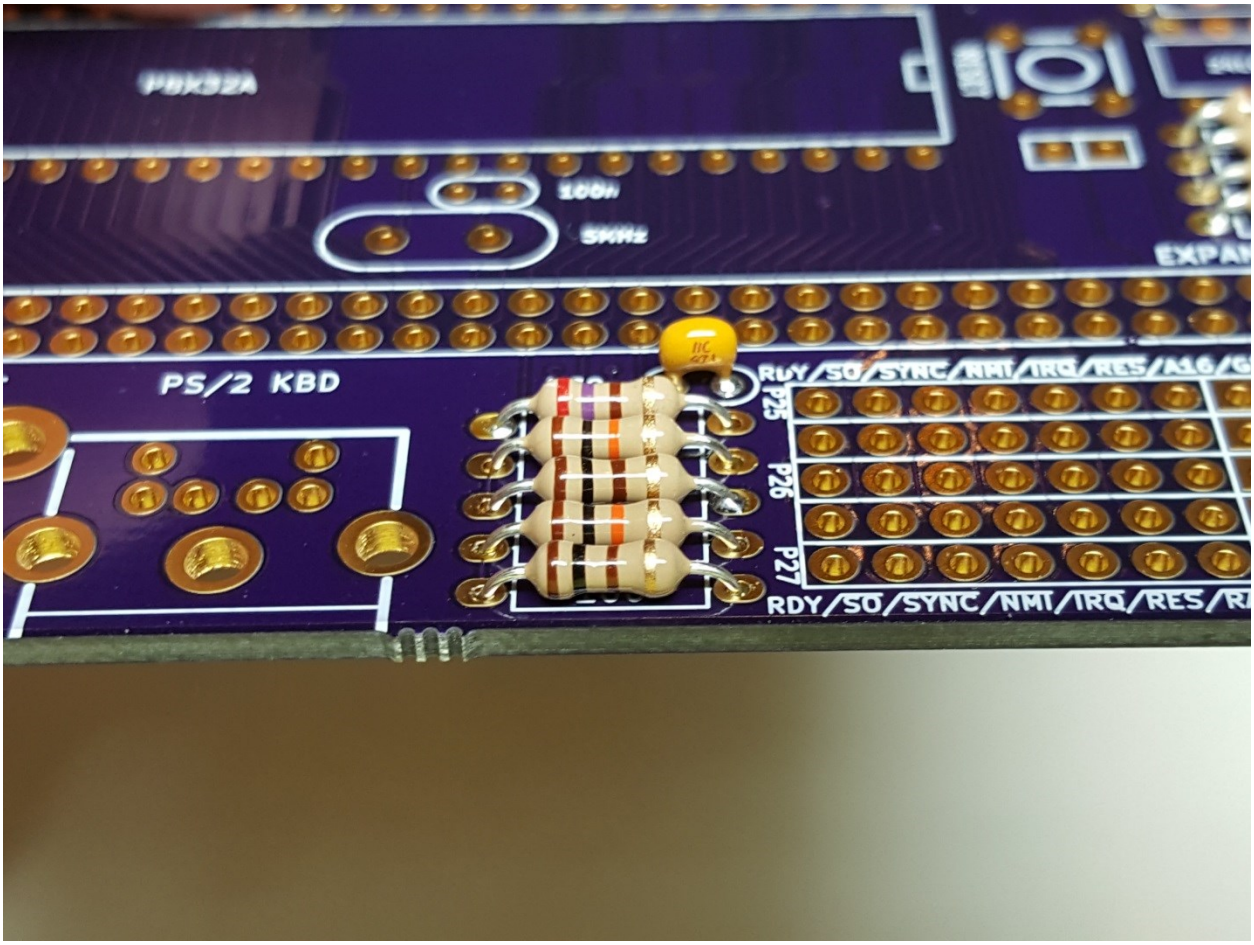
The following picture shows the PCB with all resistors soldered onto it:



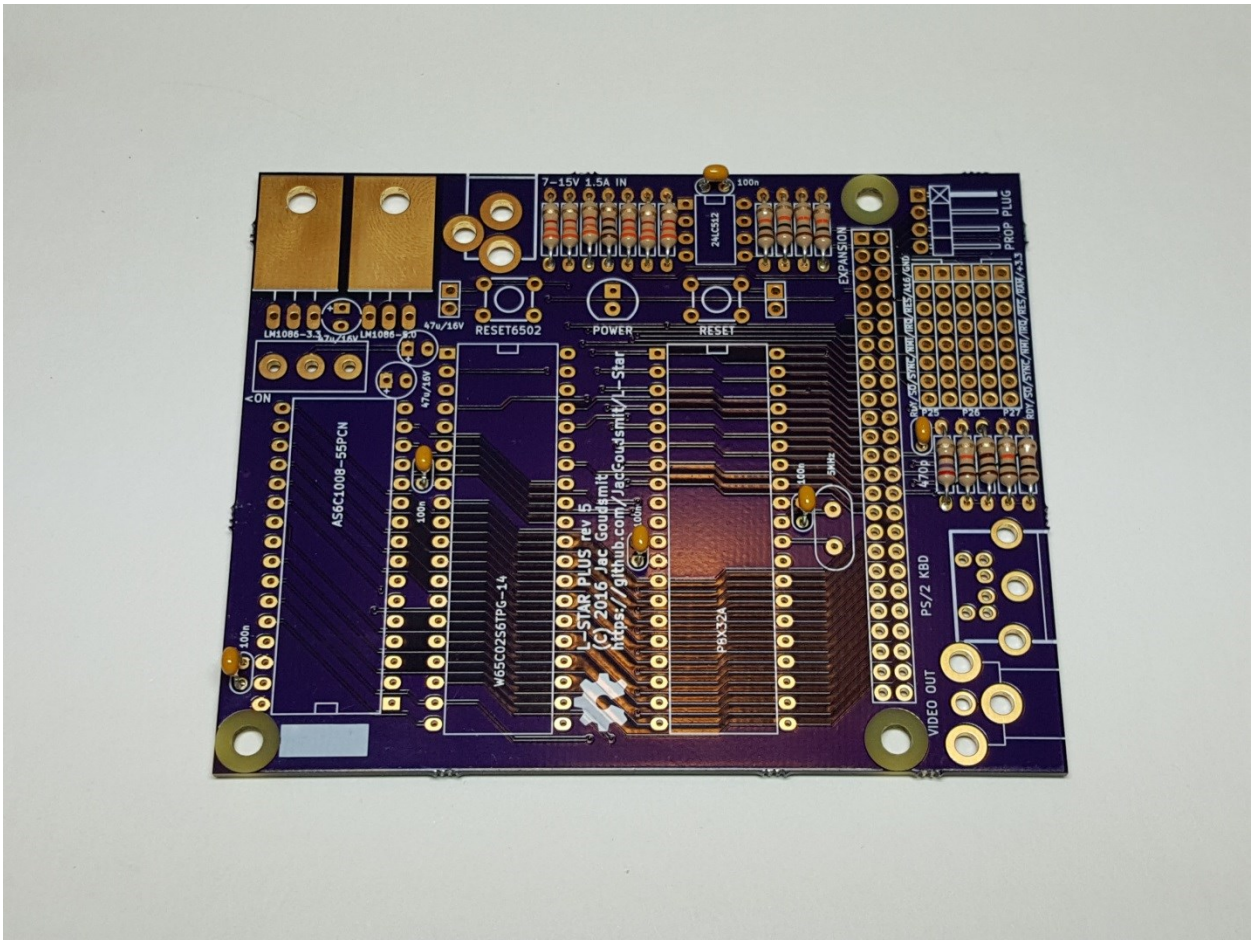
# 5. Capacitors

Start by soldering the 470 pF MLCC capacitor near the 270 Ohm resistor. The value on the MLCC capacitors may be hard to read and the capacitors look very similar, but ***the 470 pF capacitor has straight leads*** whereas the other MLCC capacitors have a little "curl" in them, as shown in the following picture:

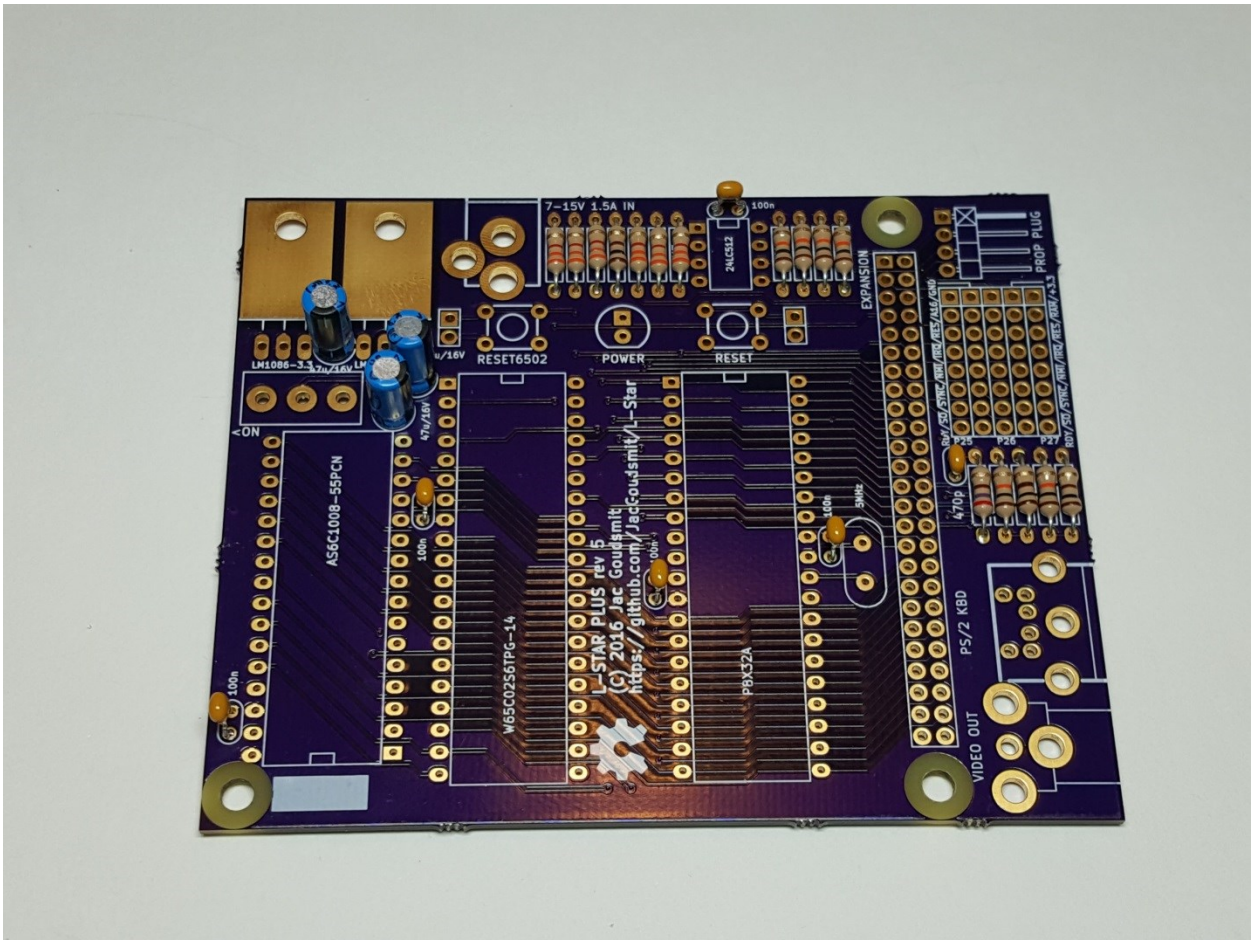
The next photo shows where that single 470pF capacitor (the one with the straight legs) goes:



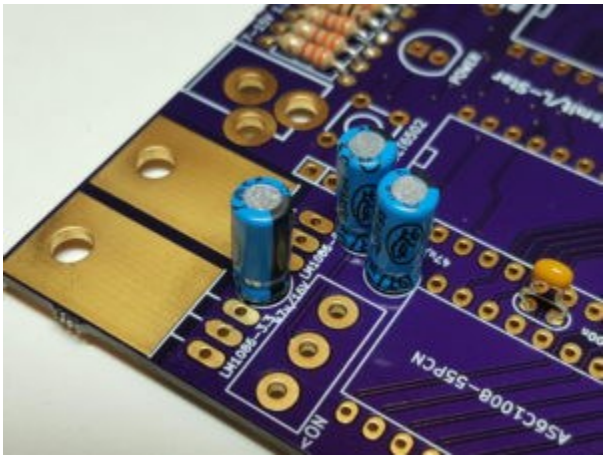
The other five MLCC capacitors are all 100 nF, and this picture shows where they go. There are two capacitors on each side of the Propeller, and there's one capacitor near each of the other chips.



Finally, solder the electrolytic capacitors. ***These are polarized, make sure you solder the '+' lead to the square pad, and the '-' lead (usually shorter, and indicated by a black band) to the round pad.***

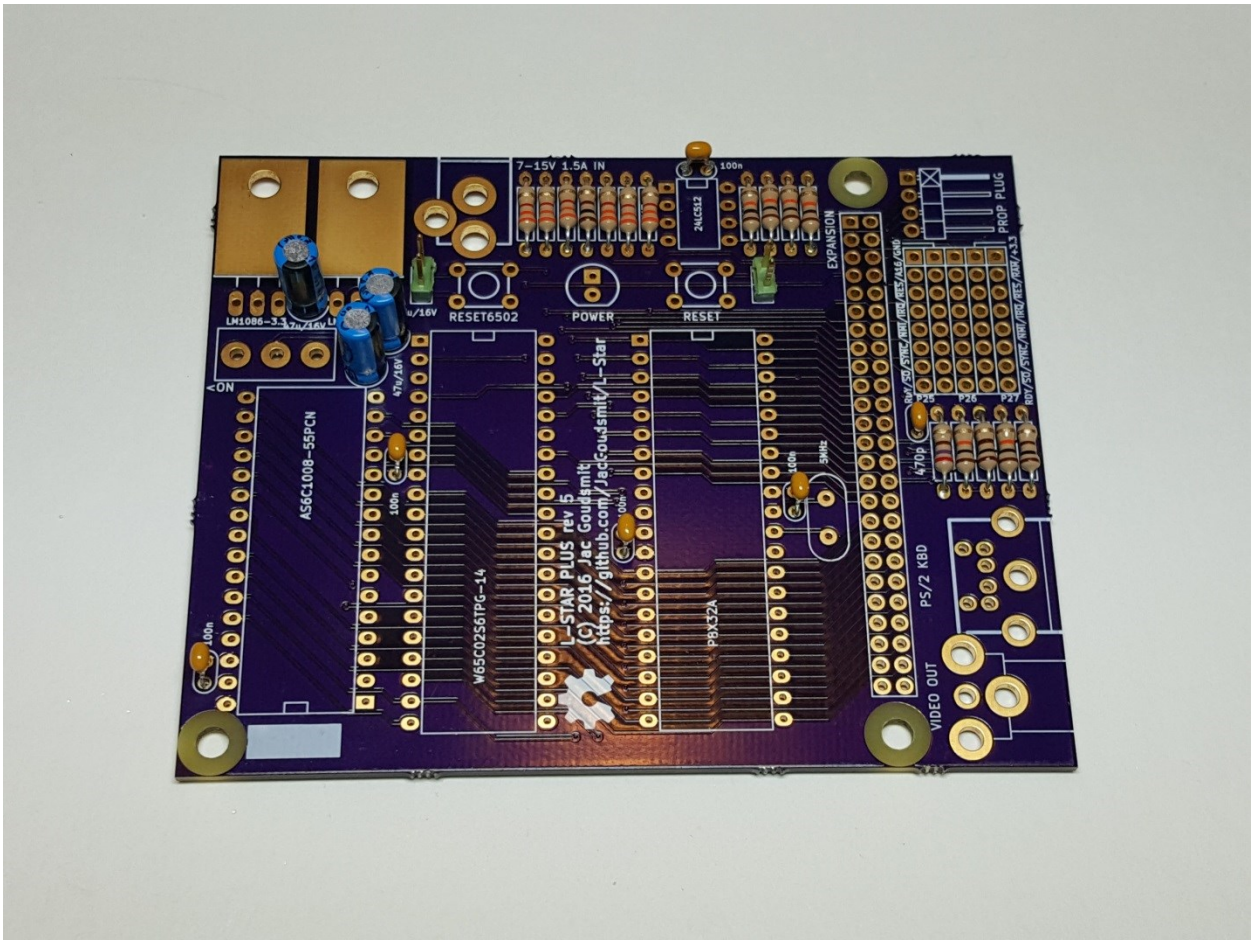



Here's a detail picture that shows you which way the electrolytic caps should be soldered. Pay attention to the black band (or '-' signs) on the packaging.



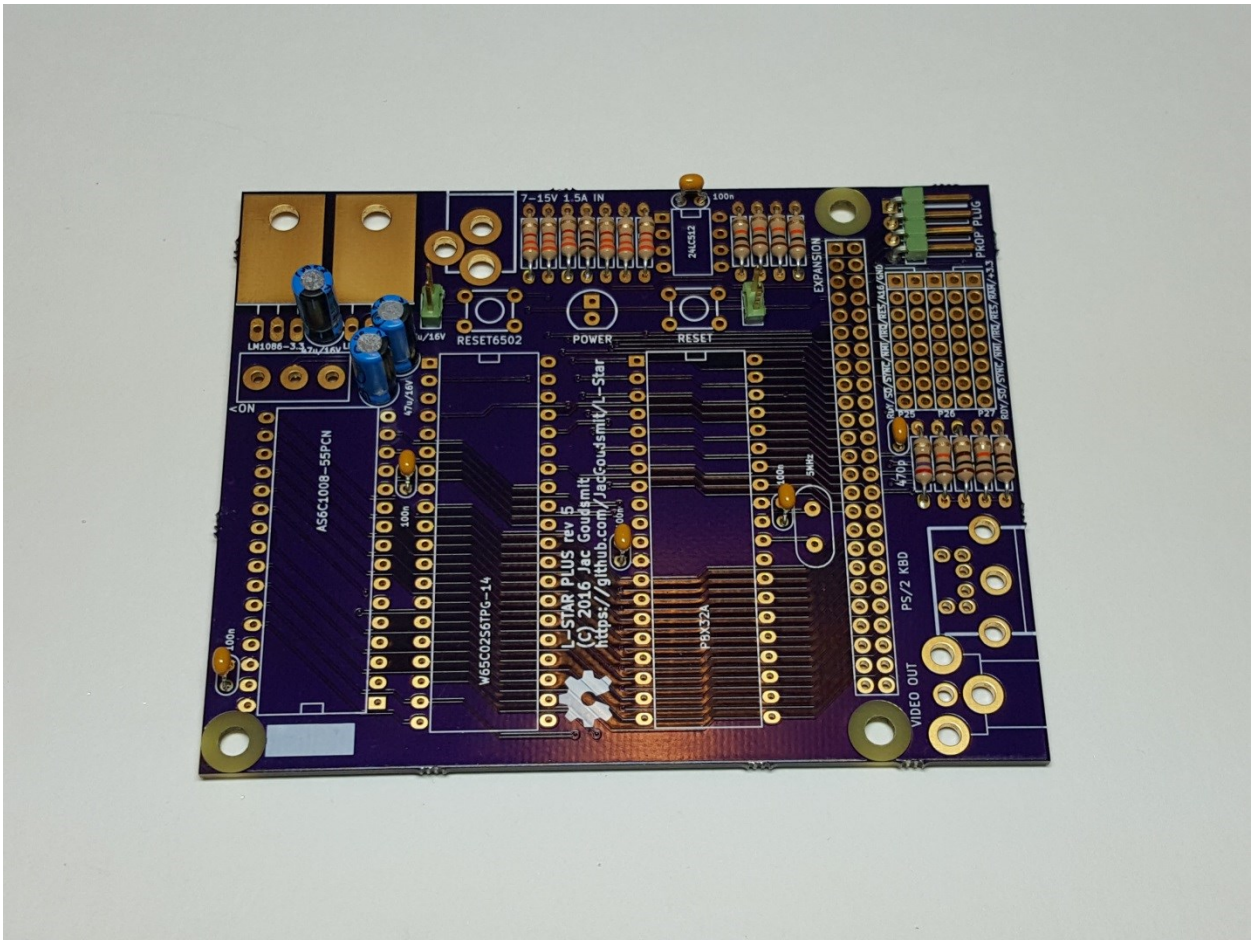
# 6. Headers

The project has two push buttons that we're going to solder later, to reset the 65C02 and the Propeller, but in case you're going to put the project into an enclosure, you're going to want to put some push buttons with normally-open momentary switches in that enclosure. They can be attached to the two-pin headers that are next to where the push buttons are going to be on the PCB. The kit includes a couple of 2-pin headers in case you want to use connectors, but you don't have to solder these headers. This is what it looks like if you do:



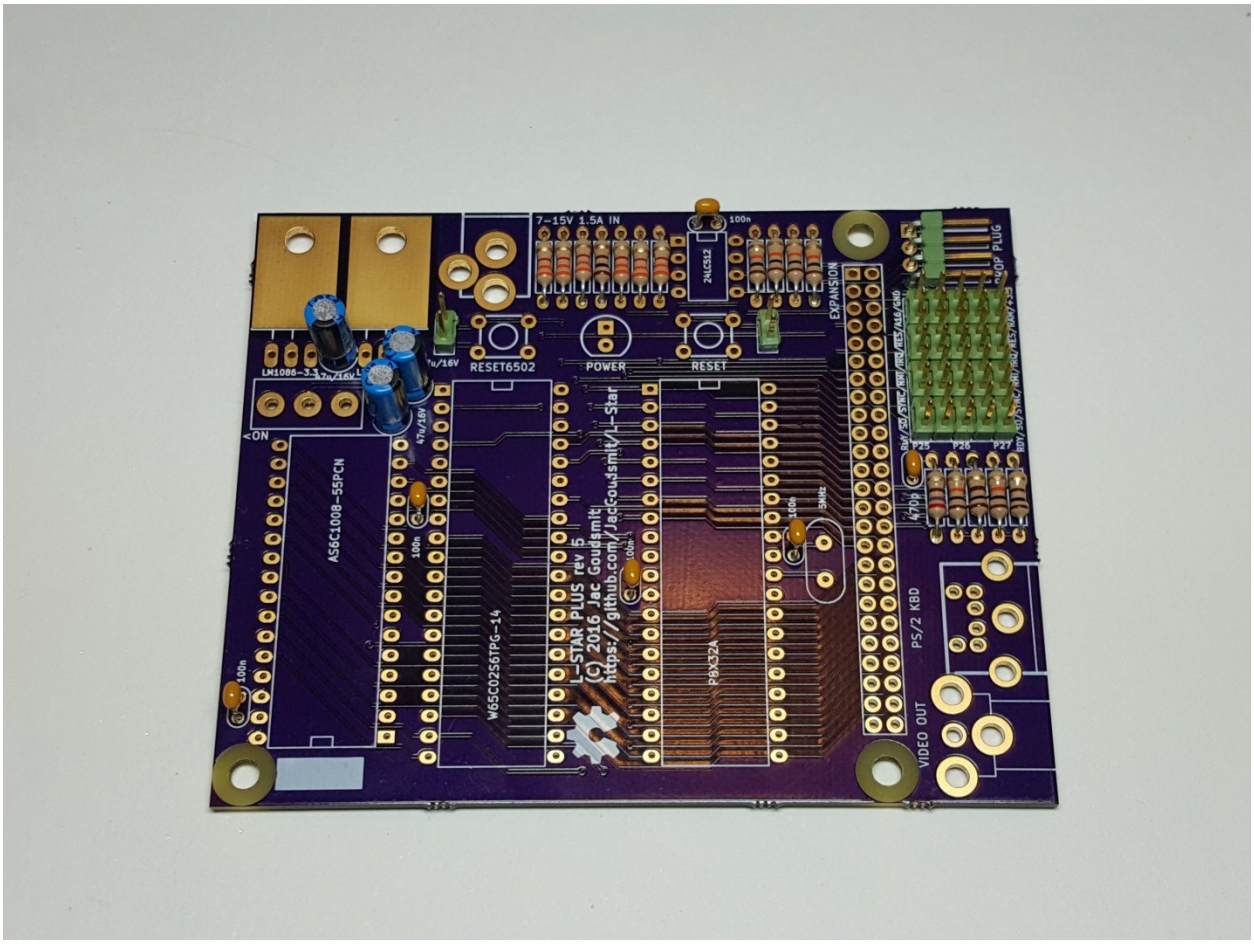
One header that you will definitely need is the one for the Propeller Plug. This is used to connect the L-Star project to a PC. The Propeller plug contains an FTDI chip (which are not available in through-hole), and a small resistor-capacitor-transistor circuit that pulses the reset line of the Propeller. ***You will need a Prop Plug to use the L-Star project***.

The Prop Plug header is at a right-angle, and the short leads of the header (the ones that have the right angle) are the ones that go through the circuit board. The long (straight) side of the leads will point towards the edge of the circuit board.

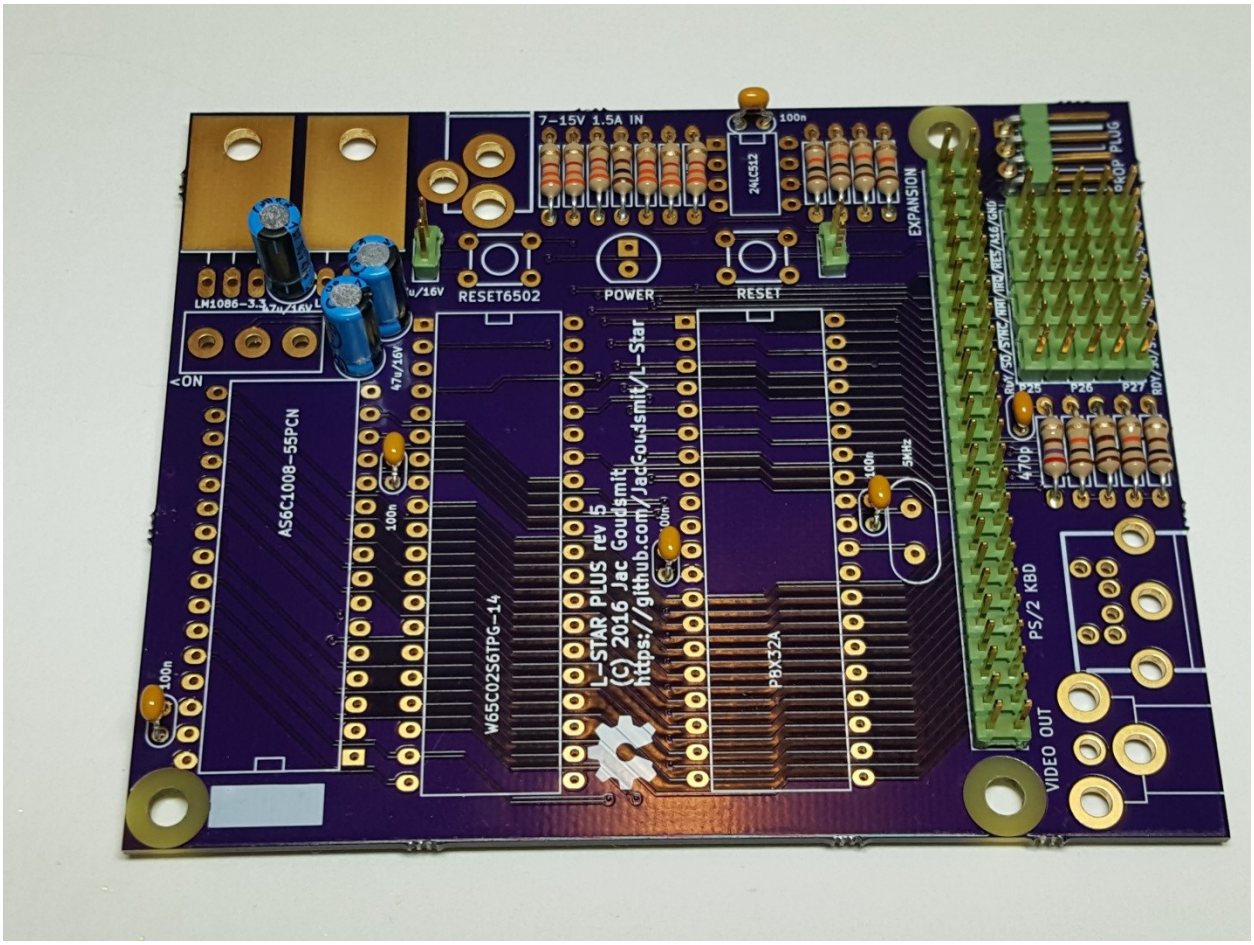


Next, solder the pins of the jumper block. You should have four headers of 5x2 pins each. ***The headers for the jumper block are mounted perpendicular to the lines of the drawing on the silk screen***: In the previous picture you could see the lines in the jumper block go from front to back, but in the picture below you can see that the actual headers are mounted from left to right. The reason I did it this way is that it's much easier to solder and align two-row headers than single-row headers. And here's a little secret: all the headers except the one for the Prop Plug start out as one long dual row header which I cut down to the right size for you. This reduces the cost of the kit a little.

Make sure your headers are all the way down on the board and line up perfectly; you may want to ***start by soldering just one pin at one end of the header***, then inspect your work on both sides (desolder the pin and correct the alignment if necessary) before you solder all the pins to the board.



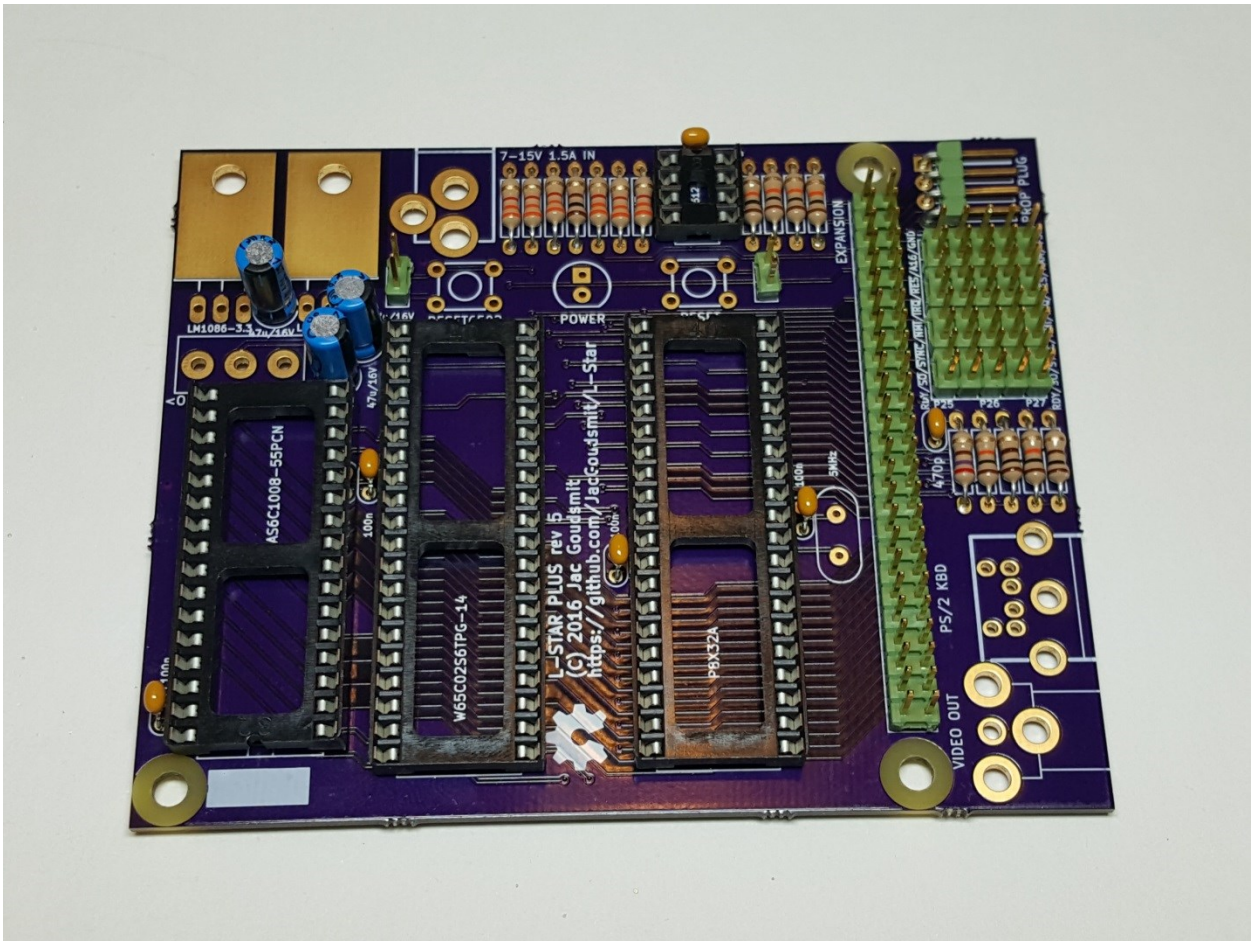
The expansion header is the last header to solder, and it's 50 pins. It's not needed for all projects, but you may want to solder it anyway in case you want to buy an expansion board later, or design your own. Again, I recommend you start by soldering a single pin at the end of the connector, then inspect your work and make sure the entire header is perpendicular to the board (make corrections if necessary by desoldering the one pin), and when you're absolutely satisfied that the header is in the right place, solder all the other pins.



# 7. IC Sockets

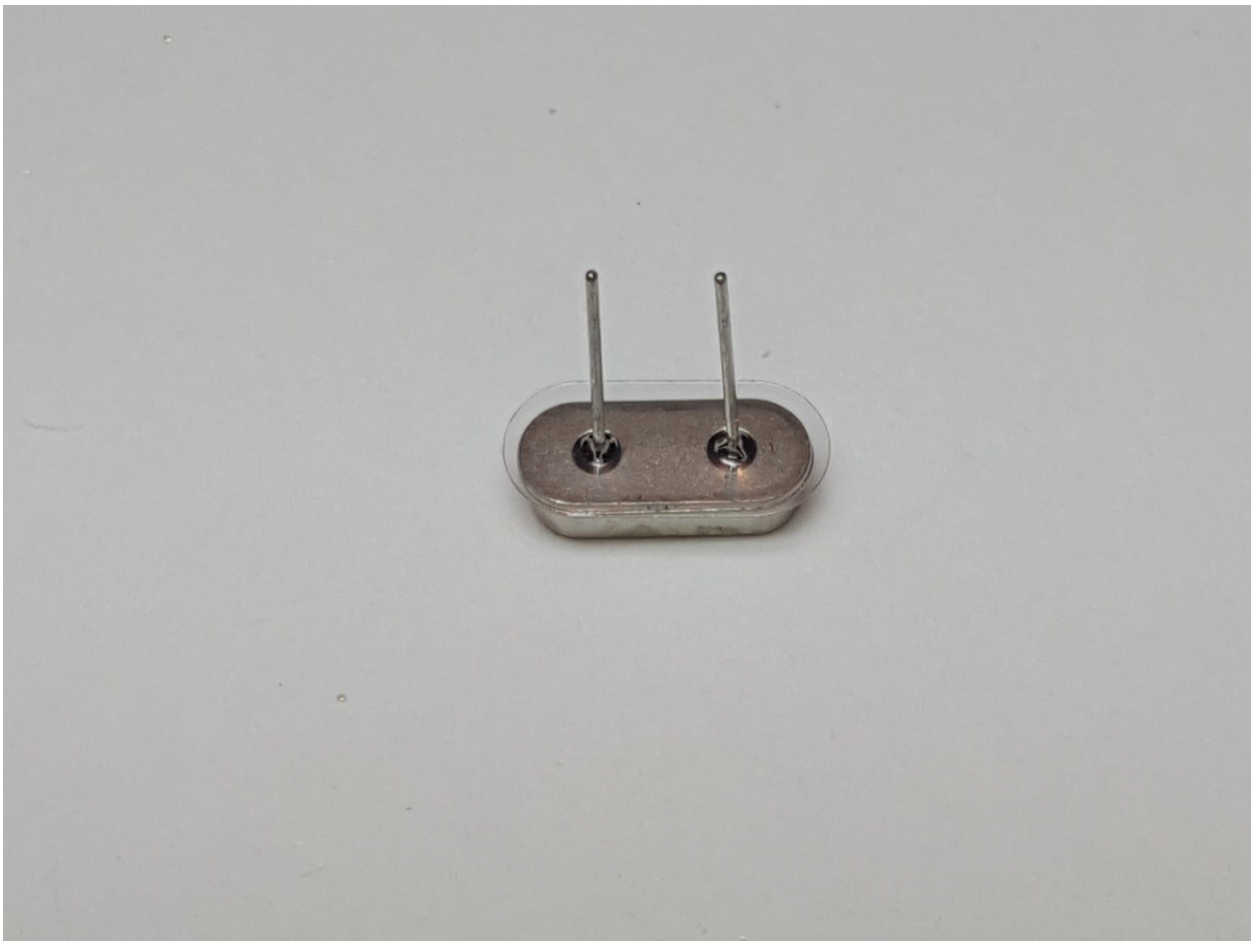
Soldering the sockets is tedious, but it needs to be done. Just like with the other big parts, I recommend starting with soldering a single pin on a corner, then verifying that the entire socket is completely in the right place, then soldering the other pins.

***Pay close attention to the orientation of each socket!*** *The SRAM chip on the left of the pictures (the one with 32 pins) is the only one that has its notch pointed to the bottom of the board, all the other ICs point the other way, with their notches towards the top of the board. It's* ***very*** *important to get this right, so you don't make mistakes when you insert the IC's later on.*

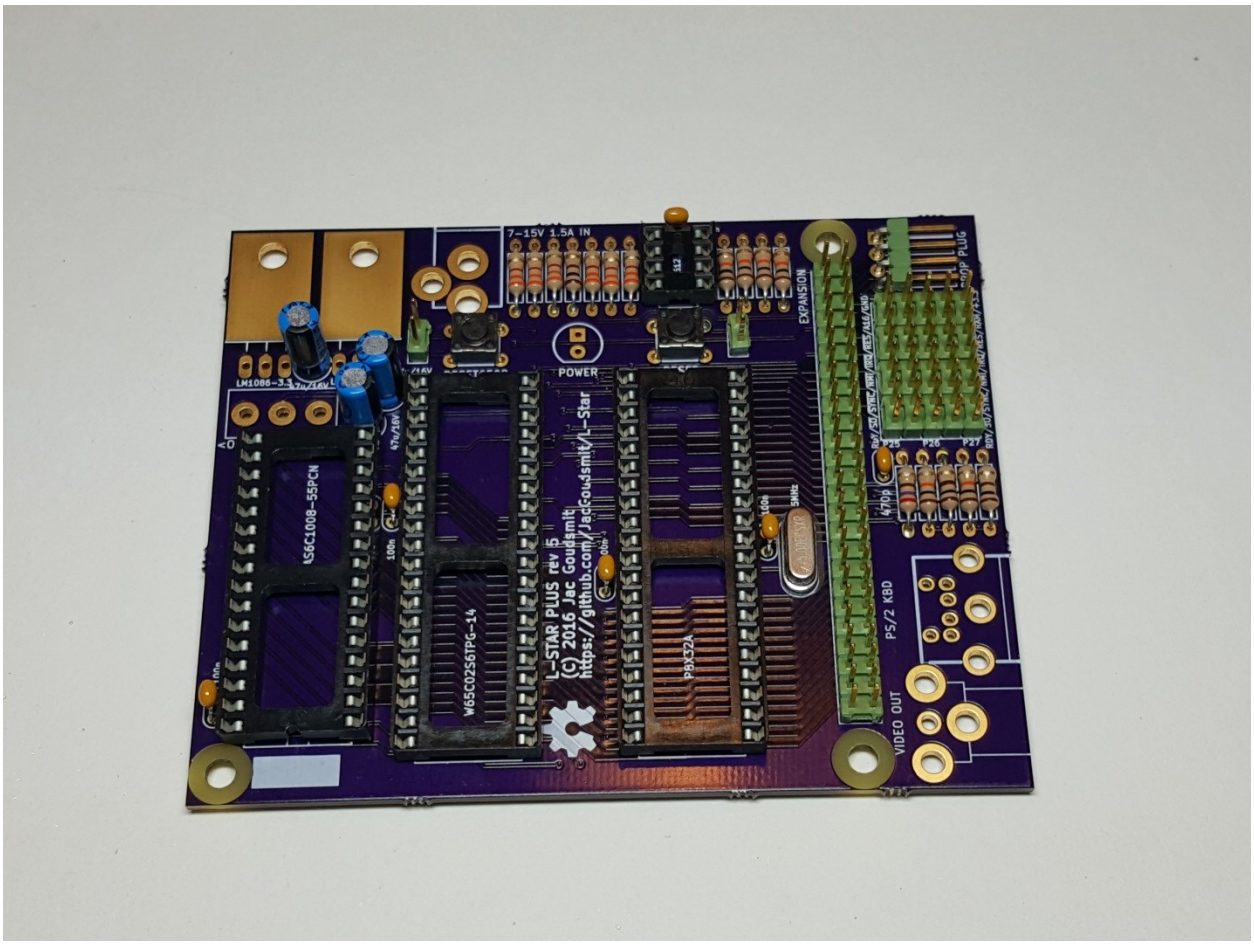


# 8. Just a Few More Parts...

The crystal should have a **transparent plastic insulator** on it, that keeps it from causing a short-circuit. Without it, the metal housing of the crystal might touch the soldering pads, which would make the circuit stop working. ***Do not remove the insulator***. If the insulator gets lost, replace it with a piece of thin cardboard, thick paper, or plastic.



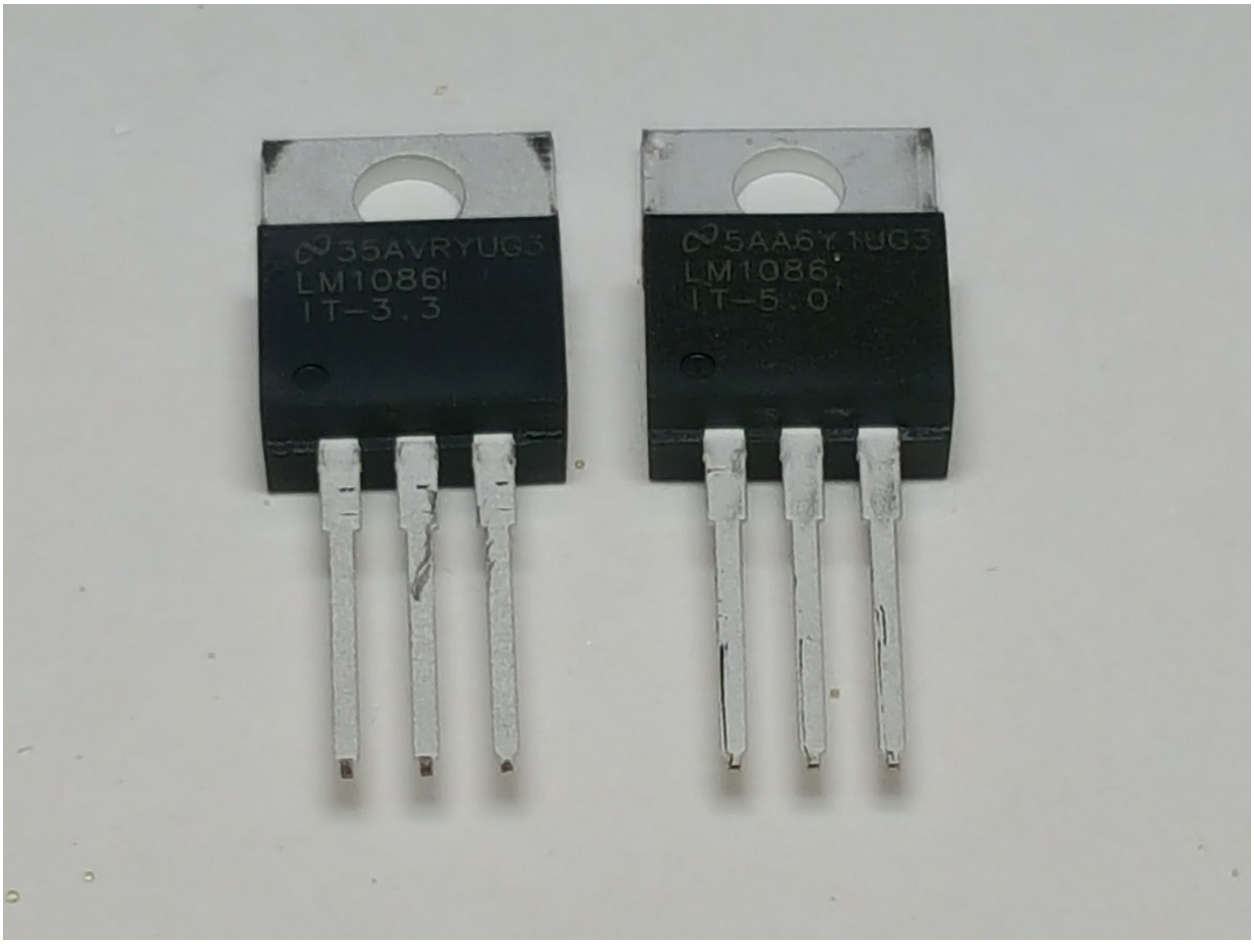
The two tactile switches should be easy: they're even designed in such a way that their legs "grab on" to the circuit board when you put them in. The ones you got in your kit may look a bit different from the picture:



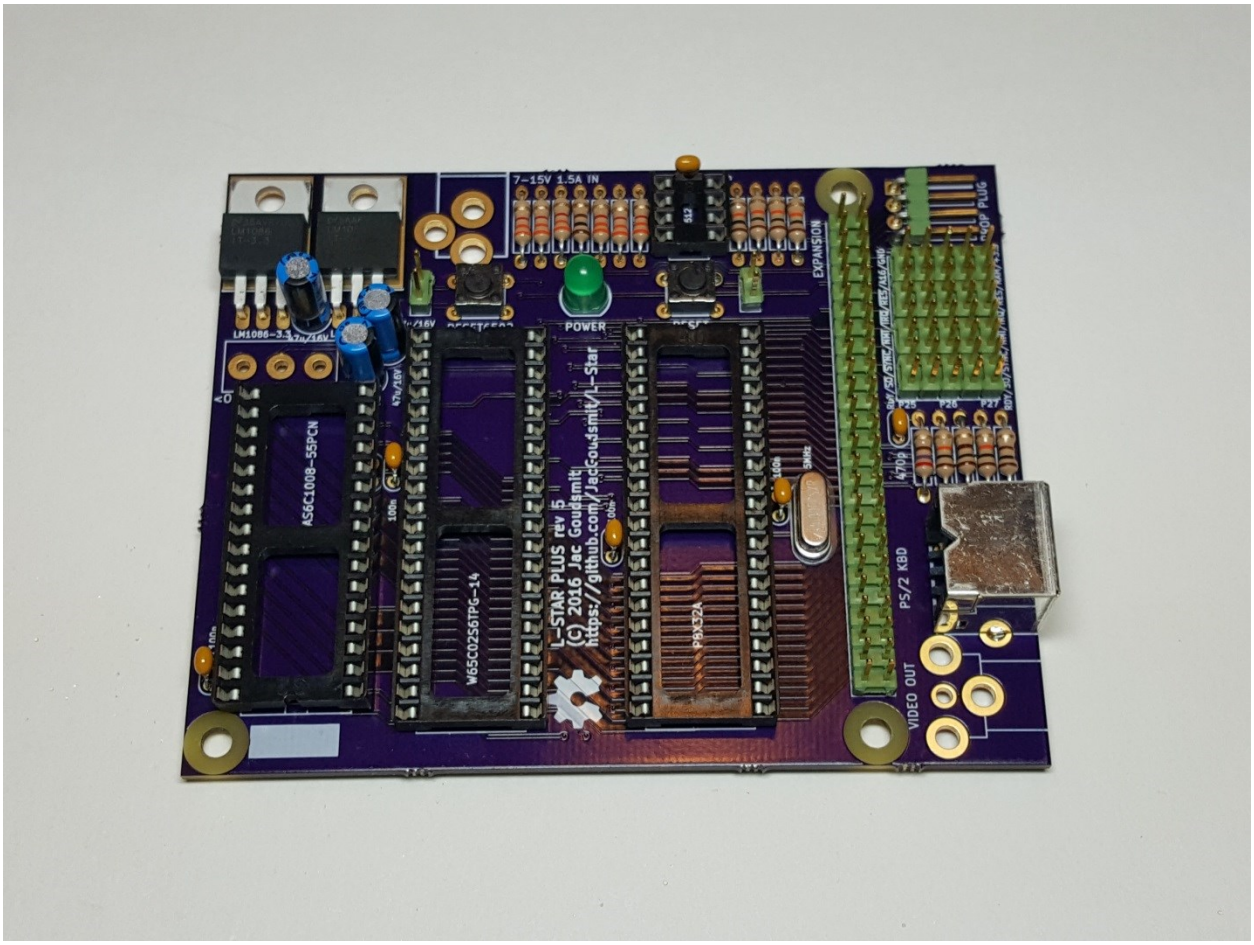
Next, let's do the power LED. It will only work if you mount it the right way, so pay attention to the orientation. The anode goes in the hole with the rectangular pad, and the cathode goes in the hole with the round pad. The anode of the LED has a slightly longer wire on the outside, and the body of the LED is flattened on the cathode side; the silk screen on the PCB shows where the flattened side goes. ***In the picture below, the anode is on the left and the cathode is on the right***. If you look inside the translucent housing, you'll also see that the cathode is always a little bigger: the little crystal that lights up inside the LED is mounted on top of the cathode.

# 9. Voltage Regulators

There are two voltage regulators on the board: one that generates 5 Volts, and one that generates 3.3 Volts. The 3.3V regulator goes on the corner, and the 5 V regulator goes next to it. ***Don't mix them up!*** The voltage is laser-etched in the regulator as part of the type, and can be hard to read.



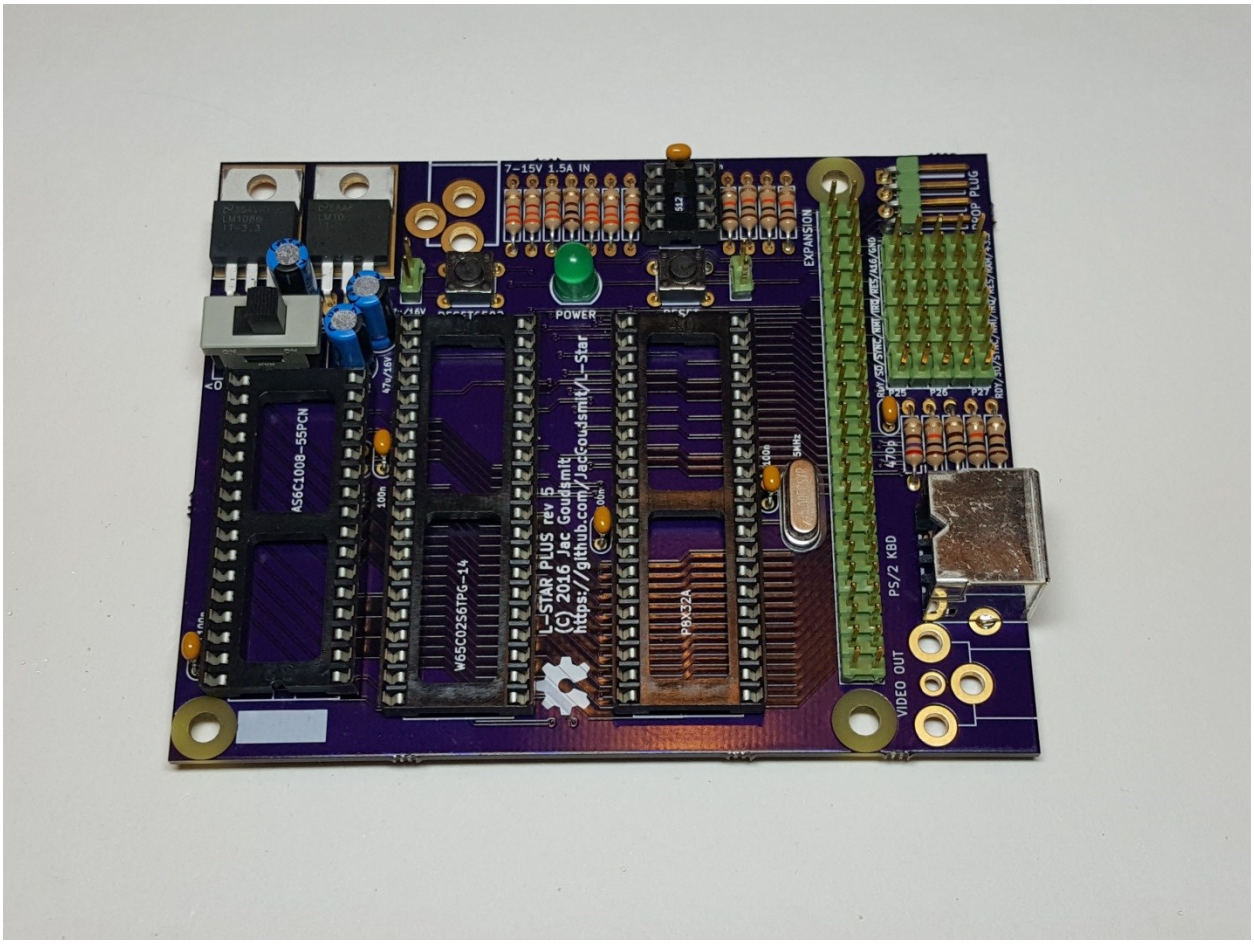

If you look closely at the leads of each regulator, you'll see that the legs change from narrow to wide about 1/4 inch (5mm) from the housing. Bend the leads down at a right angle at that point. Then solder the regulators straight onto the board. ***The 3.3V regulator is in the corner***, the 5V regulator goes towards the power inlet and resistors.



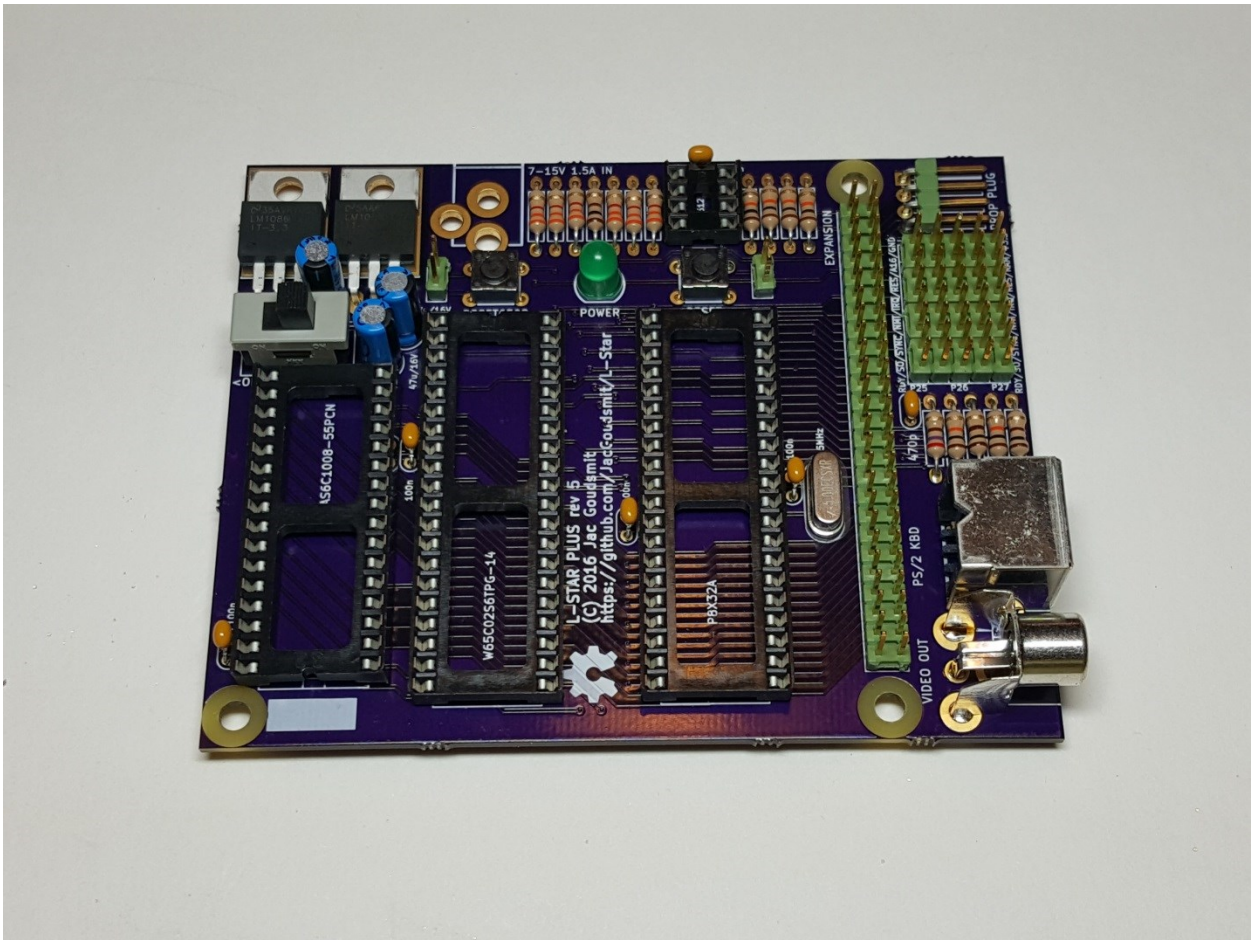

You may want to put a little solder on the top edges of the regulators, so the circuit board will work as a bit of a heat sink (it will also provide some extra mechanical support), but the regulators don't get very warm, so you can keep them as they are, too.

# 10. The Power Switch, and the Final Connectors

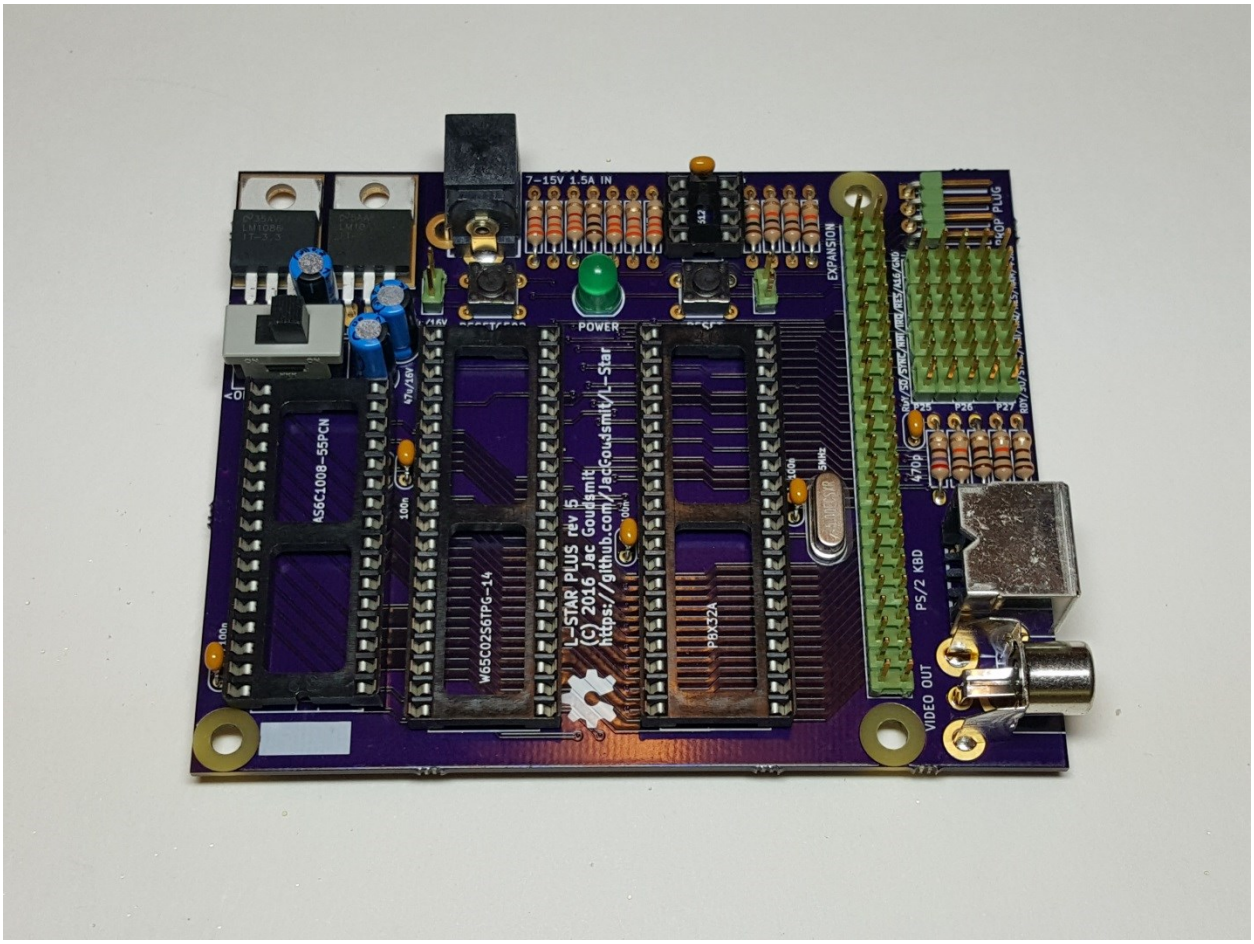
The power switch is easy. The direction doesn't matter. Once you've mounted it, slide it away from the edge of the PCB for now; that's the "off" position.



The RCA connector is also easy, it will hold itself in place while you solder it. Don't be stingy with the solder tin; on this connector it not only has a role as an electrical conductor but it's also used to transfer pulling and pushing forces to the circuit board. If you don't put enough tin, chances are that one day, you'll plug in or unplug something and the entire connector comes off the board. If your soldering iron has temperature control, you may want to turn it up a little for this to get nice evenly solidified joints.



The holes for the power socket are unfortunately a little too big (let's just say the data sheet is not very accurate on how big they should be), so aligning it before you solder it is a bit of a pain. So here too: solder one pin first, check the alignment (and re-solder the single pin if necessary), and once it's perfect, solder the other pins.



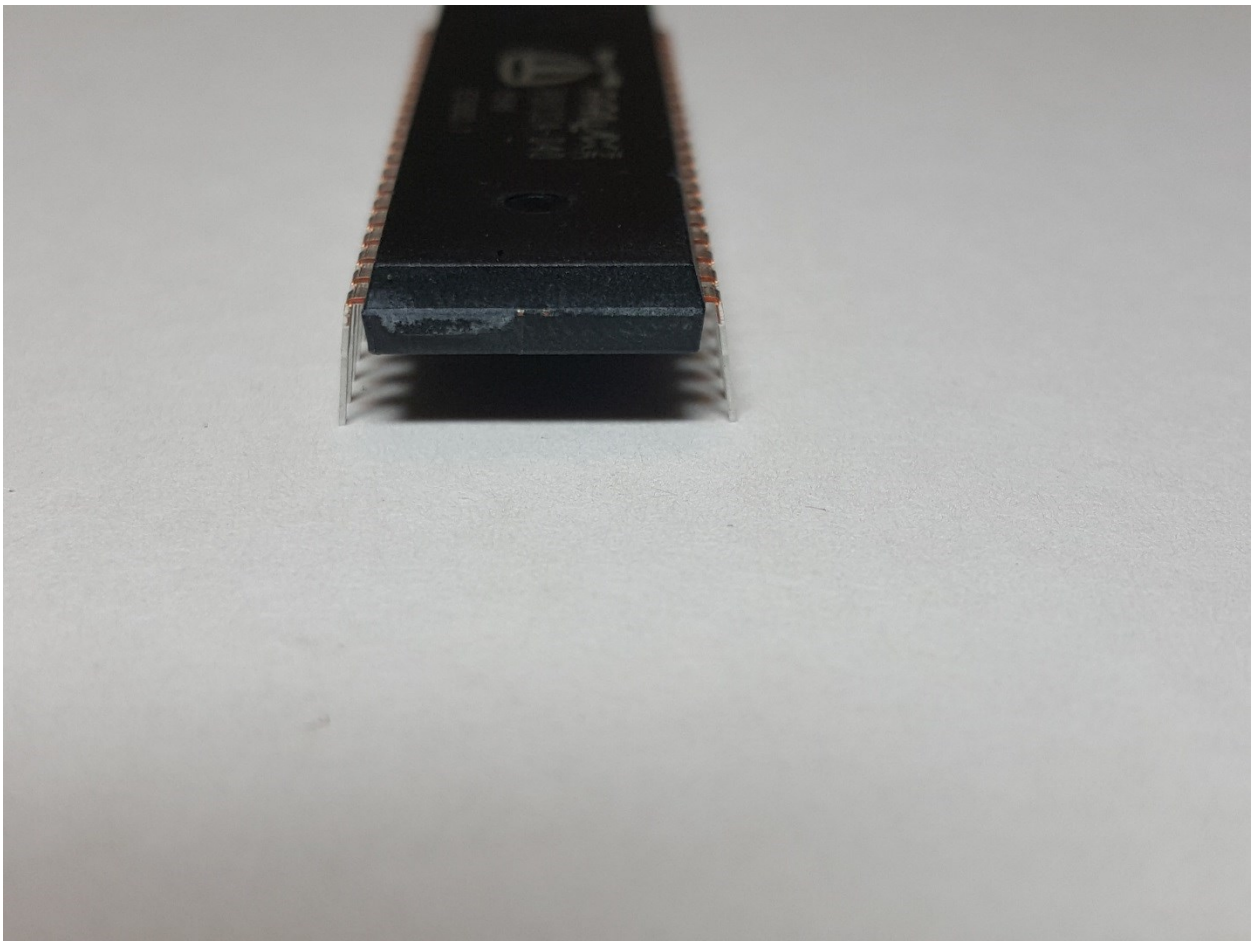
# 11. Power Supply and DIP Chips

Now that you're done soldering parts onto the board, it would be a good time to test and make sure that things are working. You will need a power supply with a 2.5mm barrel connector, between 7V and 15V, positive tip, negative ring (Don’t use your supply if it has a negative tip, positive ring; you will damage or destroy the voltage regulators). The L-Star Plus board by itself doesn’t need much power but I recommend using a supply that can source at least 1A (1000mA) in case you want to add expansions in the future. If you have an external hard disk, it probably has a beefy 12V supply which will work great with your L-Star Plus (I use one of those myself).

Lately, 5V “wall wart” power supplies have become very common too. If you have a 5V (minimal 1A) power supply (again, positive tip only!), you can short out pins 2 and 3 of the 5V power regulator on the L-Star Plus (the regulator that’s closest to the power socket, and the two pins that are closest to the power socket).

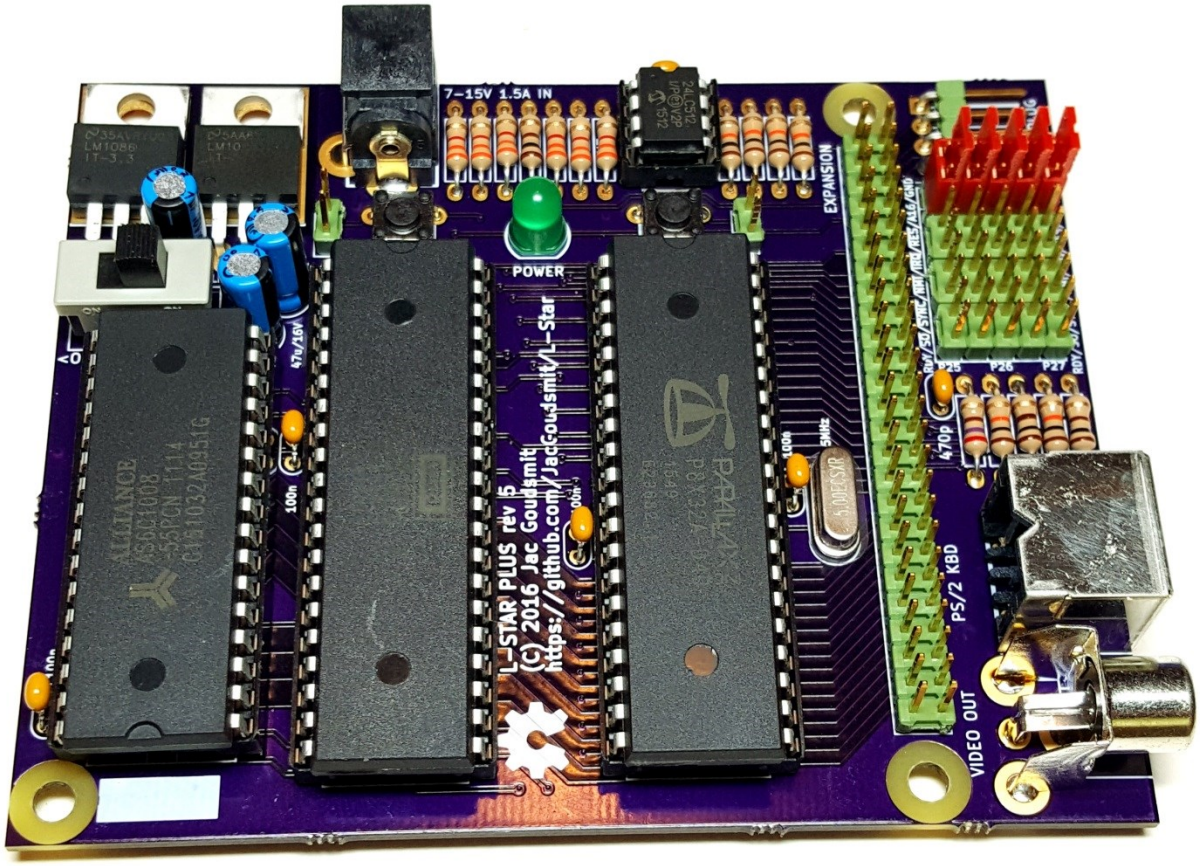
Connect the power supply to the power socket and slide the power switch towards the edge of the board to turn the system on. The LED should light up at a “normal” brightness. If it’s extra bright or it’s not lighting up at all, turn it off right away and check what’s wrong. You probably shorted something out that’s not supposed to be shorted out, or you didn’t connect something that should be connected. If you can’t find the problem, contact me and I’ll help you fix it.

If the LED turns on when you turn the power on, the power supply and the voltage regulators are working. Disconnect the power supply again for now and put the four integrated circuits onto their sockets. As you may be able to see in the picture, the pins are usually bent outwards a little when they come from the factory.



The IC's won't fit in the sockets correctly with the pins bent like that. The easiest way to fix this is to carefully hold the IC sideways on both ends with two hands, and gently press it down on your work surface to bend all the pins inward just a little bit. You can also do this in the socket if you have a knack for it, but try to avoid touching the pins either way, so you don't cause any problems with ESD.

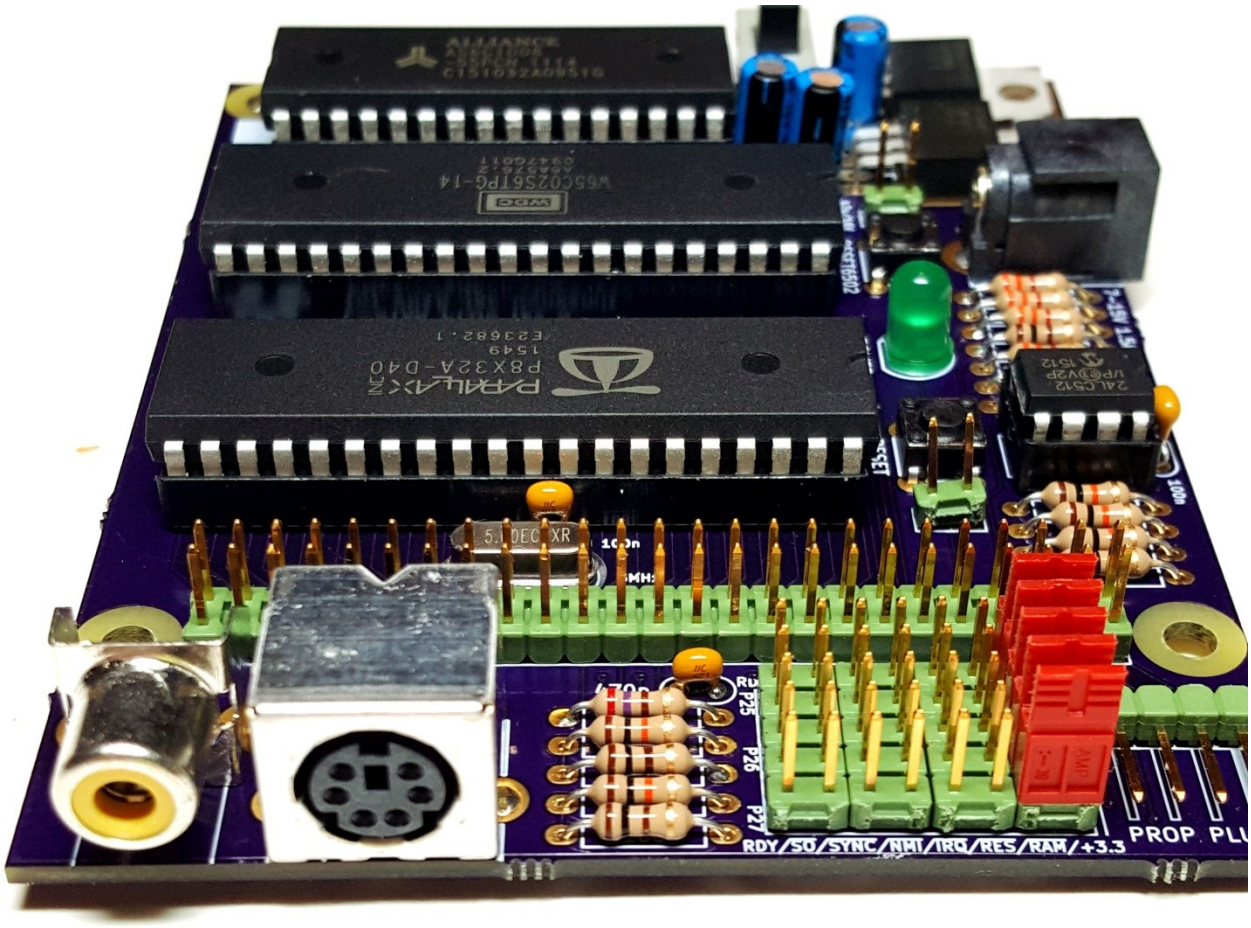
Pay attention to the markings on the chip; the Propeller and the 65C02 have the same number of pins but each has their own place on the board. if you swap them, the project will definitely not work, and you may fry them by doing that, too. The Parallax Propeller goes close to the crystal and the 8-pin EEPROM chip, and the WDC 65C02 goes near the 32-pin SRAM chip.




Pay attention to the direction: just like the sockets, all IC's have a notch that shows where pin 1 is, and all ICs except the Alliance AS6C1008 SRAM chip on the left side have their notch pointing to the edge of the PCB where the power socket is.

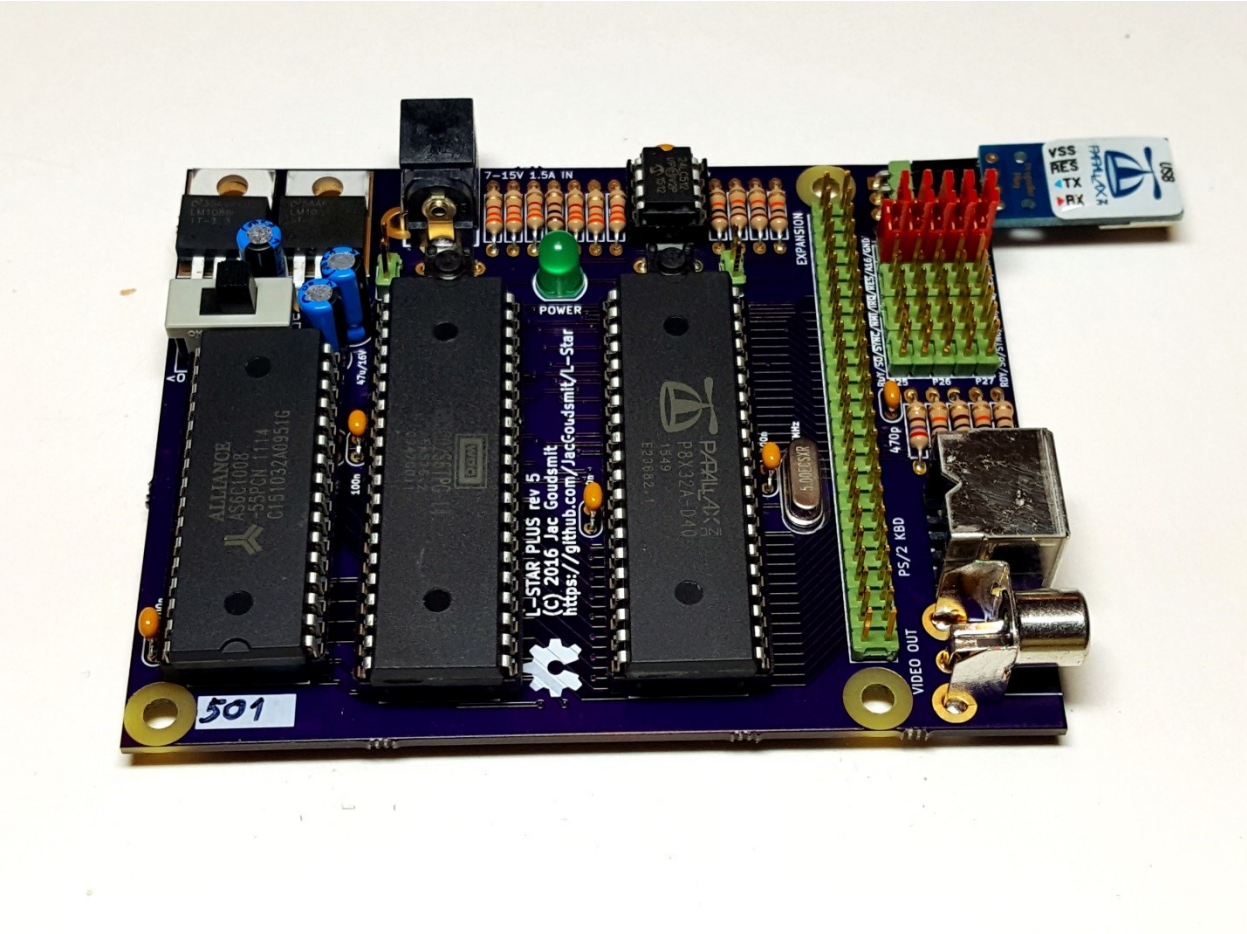
# 12. Shunts, Propeller Tool and Propeller Plug

You should have 5 shunts to configure how Propeller pins P25, P26 and P27 are connected. More information about this will appear on the website, but for now, just plug them in to connect the top 2 pins of each of the 5 columns on the jumper block. This is the default configuration for projects that don’t need the SRAM chip.



You will need to download and install the Parallax Propeller Tool from https://www.parallax.com/downloads/propeller-tool-software-windows. The “Propeller Tool” only runs on Windows, unfortunately. Parallax also distributes the multi-platform “Propeller IDE” tool (http://learn.parallax.com/tutorials/language/propeller-c/propeller-c-set-simpleide), but at the time that I started writing my firmware 5 years ago, it wasn’t stable enough for my project; things may have changed since then. I plan on converting my firmware from Spin to C/C++, and I’ll use the Propeller IDE in the future.

Here's one more picture showing how to connect the Propeller Plug (short: Prop Plug). The sticker side goes up; don't plug it into the connector the other way, you might break something. **Don’t connect the Prop Plug to the PC until after you install the Propeller Tool.**

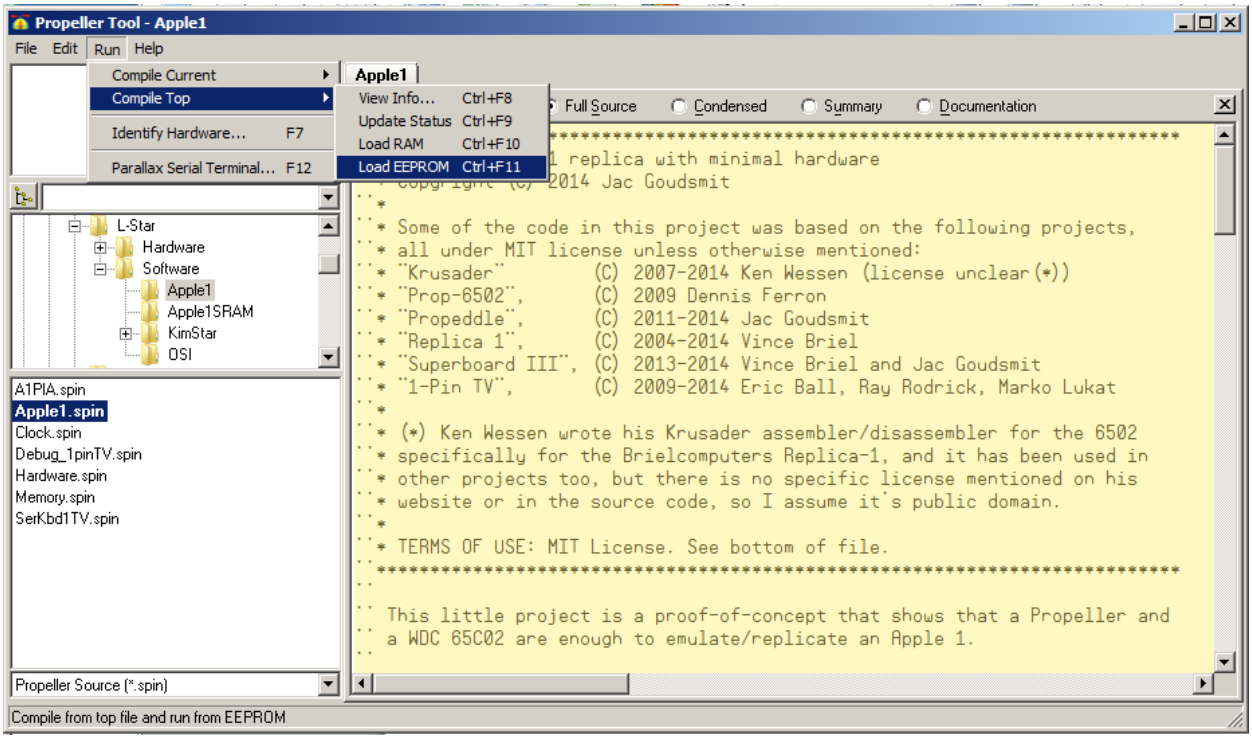



Connect the power supply again as well as the Prop Plug, and switch the power on. The LED should light up again. The voltage regulators may get a little warm, but shouldn’t get too hot.

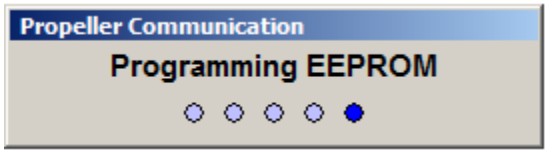
# 13. Apple 1 Replica Firmware

Download the L-Star source code ZIP file from https://github.com/jacgoudsmit/L-Star/releases. Unzip the L-Star-5.zip file. Then use the Propeller Tool to navigate to the Software\Apple1 directory and double-click the Apple1.spin file to open the source file. Then **right-click on the title tab of the Apple1.spin window, and click "Top Object File"**. This sets the file as the entry point of the Propeller program.

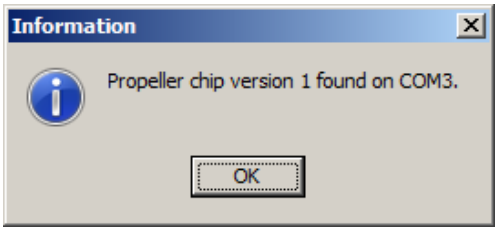
Connect the Prop Plug to any USB port if you haven't done so, and wait until the OS installs the FTDI drivers. Then select Run>Compile Top>Load EEPROM from the menu in the Propeller tool (or hit Ctrl+F11).



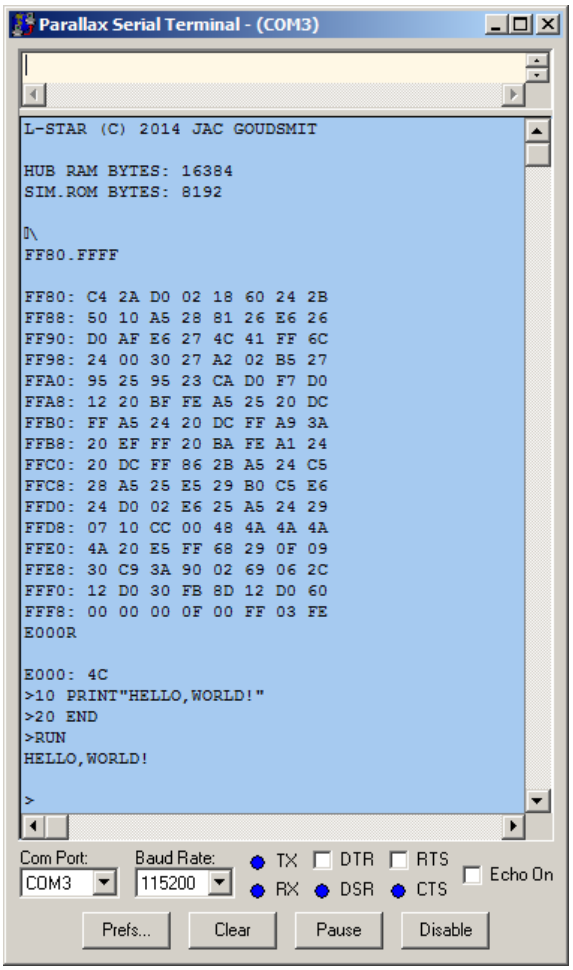

If everything is working correctly, the Propeller Tool will start downloading the firmware to the Propeller.



While the Propeller Tool is active, hit F7 to find out which COM port has the Prop Plug attached.




Hit F12 to start the Propeller Terminal, select the correct COM port, and select 115200 as baud rate. Click on the Enable button if the terminal is disabled, and then push the Reset button near the Propeller on the L-Star board. You may also need to push the Reset6502 button. Then you should see a message that L-Star is running, and you can type commands in the text box near the top (**CAPS-LOCK should be ON**; the Woz monitor only understand upper case).




The easiest way to send programs to the Apple 1 emulator is to use a full-featured terminal program such as TeraTerm. Basically, you log the output on the serial port to a file to save the content of your memory. Later on, you can send that file back to the serial port and the Woz monitor will think you're just typing it on a keyboard.

# 14. PS/2 Keyboard and TV Monitor

You can also connect a PS/2 keyboard and a video monitor. By default, the firmware is configured for NTSC frequencies, but if you open the “Debug_1pinTV.spin” module and change the PAL=0 to PAL=1 (at line 151, approximately), you can use a PAL tv. Here’s what the result might look like:

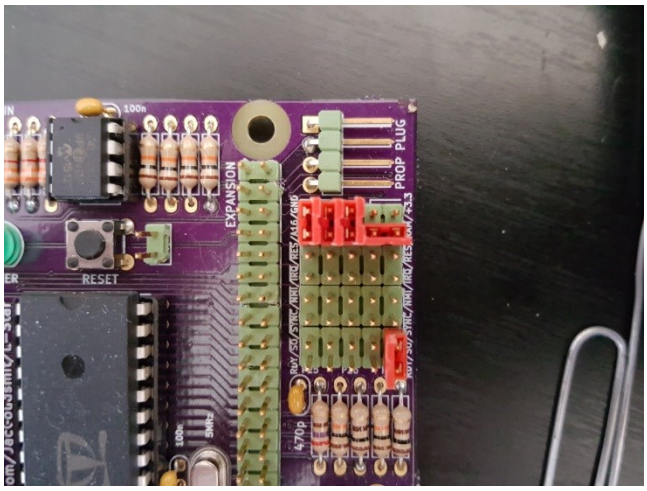

The 1-pin TV Output module generates a somewhat non-standard signal on the video output which may be a problem for some LCD TV’s and monitors. You may need to connect a CRT TV to make it work.

# 15. Using the SRAM Chip

The Apple 1 firmware that you tried out in the previous chapter doesn't use the SRAM chip: the Propeller emulated all the memory (RAM and ROM) in the 6502 address space. Because it used the 32KB Propeller Hub Memory for that, and the firmware is stored there too, you only got 16KB of (virtual) RAM.

There is another project in the ZIP file that **does** use the SRAM chip, so you have 32KB (or more) RAM available. Navigate to the Software\Apple1SRAM folder and open the Apple1.spin file from there. **Right-click the Apple1.spin file on the left and click "Top Object File" to make it the entry point of the firmware**. If you don't, the Propeller tool will keep using the firmware that you were using before.

You will have to change the jumper settings to wire P27 of the Propeller to the !RAMEN (RAM Enable) signal:




Also, because the firmware now has only one pin connected to the PS/2 keyboard, you have to push the space bar on that keyboard to get started. Of course that's a problem if no keyboard is attached; in that case you can change the line in Apple1.spin from:

```
term.Start(hw#pin_RX, hw#pin_TX, hw#pin_KBDATA, hw#pin_TV, BAUDRATE)
```

to:

```
term.Start(hw#pin_RX, hw#pin_TX, 0, hw#pin_TV, BAUDRATE)
```

Hit Ctrl+11 to download the Apple 1 SRAM project to the board.

And that concludes the build instructions. Refer to the website for more information (under construction). Thanks for watching, and I hope you will enjoy your fully assembled L-Star!

Jac Goudsmit, December 13, 2016