Technote 8

April 2008

[Revised 2/2014]

Guidelines for Drawing Schematics

Tim J. Sobering
SDE Consulting
sdeconsulting@pobox.com

Guidelines for Drawing Schematics[1]

For the Electrical Engineer, electronics symbols are our "words" and the schematic is our "novel". And just as a poorly written document can cause confusion or disaster, "misreading" a poorly drawn schematic can cause a range of problems from a non-functional circuit to a fire, explosion, or electrocution.

There are a number of standards for schematic symbols and how to interconnect them. Some examples are:

IEEE Std 91, IEEE Standard Graphic Symbols for Logic Functions.

IEEE Std 315, IEEE Standard Graphic Symbols for Electrical and Electronics Diagrams (Including Reference Designation Letters).

IEEE Std 991, IEEE Standard for Logic Circuit Diagrams.

There are also a lot of web sites such as:

http://library.thinkquest.org/10784/circuit_symbols.html

http://en.wikipedia.org/wiki/Circuit_diagram

And, just to make this more confusing, some companies have their own unique rules for schematics or continue to use outdated symbols. Finally, symbols used internationally can be different than what we are used to in the U.S. For example, the resistor symbol commonly used in the U.S. is different than what is used internationally.
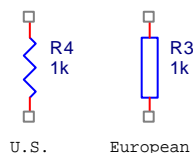
Figure 1. Standard resistor symbols

Reading standards can easily cure insomnia, but following them can be quite important especially when working in large, multi-company, or international project teams. And frankly, as a future engineer, you are obligated to learn how to do all of this correctly. Hey...you picked this field...I didn't make you enroll in it.

As students, only portions of the discussion below will apply depending on the purpose of your drawing and the requirements of individual faculty. Regardless, schematics should be neat and properly drawn for a given purpose. Properly drawn will be discussed more below. Neat means that all schematics are drawn and labeled either with a straight edge and templates, or preferably with a CAD system or other computed-based drawing package[2]. There are a lot of free software packages that can be downloaded for this purpose. But "neat" goes beyond simply having straight lines. Your schematic should organize and align

---

[1] While some of the information in this document comes from authoritative sources, much of the commentary is based on professional experience, and as such, represents the opinion of the Author. All information contained herein is subject to revision. However, credit does need to be given to Dwight W. Gordon, "Computer Engineering – Guidelines for Hardware Diagrams", Fall 1990, as well as to a number of the EECE Department Faculty. Another good reference is Horowitz and Hill, *The Art of Electronics, Second Edition*, Appendix E, 1989.
[2] There is nothing wrong with a hand drawn schematic – designers use them all the time to illustrate various points or to brainstorm. And while some very good designers hand drawn scribbles have become quite famous, you are not Bob Pease. Work that you turn in to your instructor or to your boss should look good.

components so that it is easily "read". Picking a uniform spacing between components also helps make the drawing look better.

As to defining "properly drawn", the ultimate purpose of the schematic needs to be taken into account as well as the intended audience. A schematic serves multiple purposes. First, and probably most important, is that the schematic accurately represent your design. Signals and voltages should be labeled, parts identified, pins properly connected, and unused pins marked (if unconnected) or "handled" properly (such as extra parts in a multi-part package). This way the schematic will provide a minimum representation of what you are trying to make. This level of detail is often sufficient for homework, initial design calculations, or as a starting point for a conceptual design review.

At the next level, a schematic should contain sufficient information to afford anyone a high probability of building the circuit successfully. After all, building a working circuit is usually the goal[3]. Often this is someone less technical such as the person doing the board layout or your technician[4]. In addition to having proper connectivity so that the netlist[5] is accurate, this means that you need to add ancillary information to the schematic such as "D3 must be thermally bonded to U7" or "All capacitors are 10%, 100V, X7R Ceramic unless otherwise noted". Most schematic CAD tools will also generate a parts list (or "Bill of Materials", BOM), provided you enter sufficient information into the tool. The BOM should describe every part in sufficient detail so that it can be specified and purchased, and also identified on the schematic and the PC board. Reference designators (R1, C10, D123) serve for identification. Other information like component tolerances, power dissipation, and the package type (footprint – the physical representation of the symbol) are included. Anyone who has tried to kludge a 2 watt resistor into a ¼ watt resistor footprint understands the important of this information being accurate. Imagine what would happen to your career if the Purchasing Department buys 200,000 parts in DIP packages when you actually designed with SOIC packages on your board. A lot of CAD packages will try to do this work for you by pre-linking symbols and footprints, but you still need to verify this information and provide much of it yourself as there are just too many possibilities to make a generic choices. I have also found errors in symbol pinouts from software vendors, so "Buyer Beware".

The next purpose of the schematic is for debugging your circuit. This (combined with a board layout drawing) is your roadmap for tracing signals and checking voltages and finding out what is wrong[6]. Do the signals on the schematic flow and can they be followed easily? Or do they cross over and loop back to where it is a challenge to figure out which pin connects to which other pin? There are a couple schools of thought here. Technicians, particularly those with graying hair, prefer to have the schematic for the entire circuit or

---

[3] In reality, there are "construction" schematics, "simulation" schematics, and "simplified" schematics we use to illustrate a point. You should work to be neat and organized regardless of the final purpose.

[4] Never underestimate a Technician's skill. They often know more than the Engineers involved, particularly when it comes to making everything work "in real life". But also remember they cannot read you mind, so they can only work with the information they give you and their experience.

[5] A Netlist is a description of the design that is used in the transition from the schematic to a PC board layout. It can be as simple as a list which says "U1 pin 1 connects to R3 pin 2", but in reality most are not formatted to be readable by the operator. A netlist also contains information as to the package for each component (8-pin DIP, 16-pin SOIC, etc.)

[6] Nothing is wrong? Then clearly you were not the one who designed it, or you are not testing it right. A fundamental fact of engineering and design is that everything goes wrong. Nothing works right the first time. It is the good engineer who knows how to make it work (especially when it is someone else's design).

subsystem on a single page. They don't care if it is a D-size drawing. They want to see every signal on the PC board and not have to flip through multiple pages and worry about which signals go where. I do not like this style of drawing although I do acknowledge the logic in the approach. If the design exceeds a single B size page[7], my preference is to use multiple pages and divide the schematic into logical subsystems. I discuss this more below.

To me, "Is it easy to read?" is the final and most important aspect of a schematic. Everything discussed above comes down to this statement. If the schematic is drawn with these purposes in mind, it will be useful and look professional. And as the schematic is the circuit designer's ultimate form of communication, this will reflect on you[8].

Here are some guidelines that I have used over the years that will help get you to a starting point. Remember, you will likely need to adapt to your companies specific requirements once you are employed. Or perhaps, you can crusade to change their culture. Not all of the conventions below are covered in a standard and some may deviate from the strictest interpretations. Some are simply picky, but I have a job where I review schematics drawn by many other people, so to make that process efficient I try and make all of the designers follow the same basic rules so that I can quickly understand what they draw.
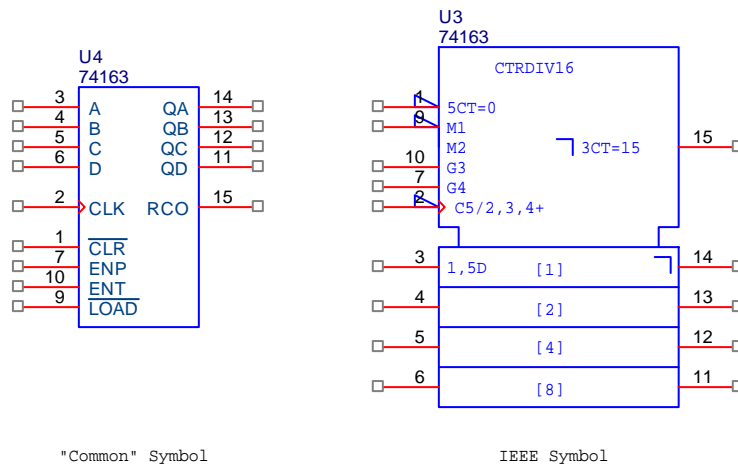
Figure 2. Common vs. IEEE symbols

1. Symbols are important. Some are quite esoteric and rarely used outside specific disciplines. It goes without saying that you should use the right symbol, but a common example where most organizations deviate from this rule is with logic symbols. IEEE has well defined standards for logic symbols, but it seems Texas Instruments is one of the few companies that use them. Most CAD packages have what we might think of as more common symbols, but some of the higher end packages offer the IEEE symbols in a separate library.

   One area where you should be very careful is with capacitor symbols. There are "polarized" and "non-polarized" capacitors. It is very important to distinguish them as polarized capacitors (Aluminum or Tantalum Electrolytics) are damaged when reverse biased. The damage will cause the capacitor to becoming "noisy" and eventually fail.

---

[7] In my experience, a schematic drawn on a B-size page (11" x 17") can be printed on an A-size page (8.5" x 11") and still be very readable. So if you do not have a large format printer available, limit your schematic page size to a "B".

[8] Look closely at your schematic. A good schematic will take a little extra time to draw, but it should look good.

These failures can be dramatic, including smoke, fire, or even explosion. If you accidentally reverse bias a polarized cap (and are fortunate enough to avoid an explosion) throw it in the trash. It is useless. DO NOT INTENTIONALLY REVERSE BIAS A CAPACITOR "JUST FOR FUN". IT IS DANGEROUS, CAN CAUSE A FIRE, USUALLY PRODUCES SHRAPNEL OR TOXIC GASSES, AND CAN CAUSE INJURY TO YOURSELF OR OTHERS[9].

I see a lot of variation in the symbols designers use for capacitors, and this can often cause errors. The figure below shows the capacitor symbols from IEEE Std. 315. Style 1 is preferred because of the straight lines. Style 2 is probably more recognizable to you.
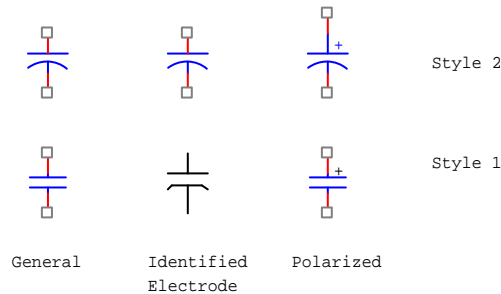
Figure 3. Standard symbols for capacitors

The problem I have with these symbols is that is you photocopy a schematic a few times, the little "+" sign can vanish. This can cause significant problems. So I have always used the bottom left symbol only for a non-polarized cap and the top right symbol for a polarized cap. This way, even if the "+" doesn't print clearly or the photocopy doesn't show it clearly, my designs are always clear, at least to me.

2. Being able to distinguish which nets connect and which cross over is critical. The "dot convention" has been developed to clarify connections. The figure below shows the accepted and preferred methods of connecting or crossing nets. The little arc in the upper left is the archaic method, so don't use it (your CAD tool likely will not draw it anyway). And in recognition of the photocopying problem, I have started forcing myself to use only the "preferred" connections. This way, if the "dot" disappears, you can still tell which nets cross and which connect.
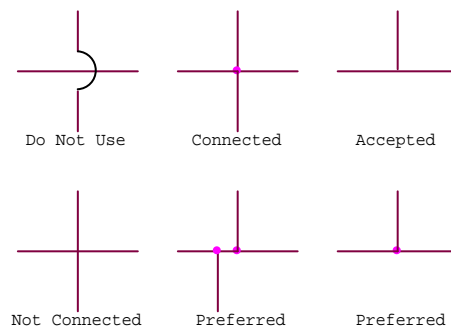
Figure 4. Conventions for crossing or connecting nets

---

[9] In my course, the failure to follow this restriction was grounds for disciplinary action under the Student Code of Conduct and a withdraw-failing grade for the course.

3. While I am not aware of a specific standard, when a net exits a symbol pin, I prefer that the connection advance at least one grid point straight out before making a direction change, as shown below. There was actually a time when the style used on the left, while appearing to be a connection in the schematic, would not show up as a connection in the netlist. It also helps with making a schematic neater as you are less inclined to pack parts too closely.
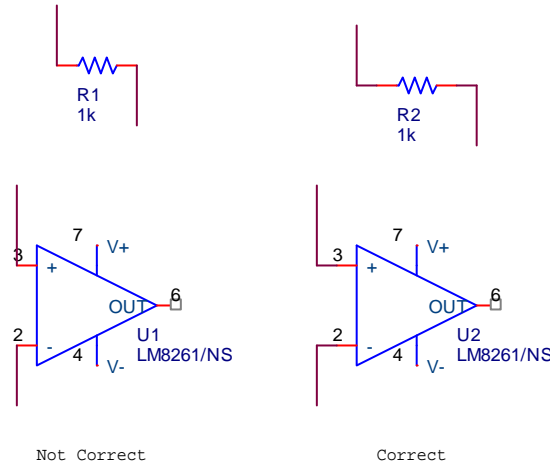
Figure 5. Connecting nets to pins

4. Another very important part of the schematic requiring attention are Reference Designators and the associated values. Part "references" are the letter/number combinations used to indicate the type of component (see Table 1) and to uniquely identify each one (R1, R2, R3, etc.) in your design. References are frequently printed on a PC board "silkscreen" to permit easy location of components. References should be located next to the component with which they are associated. You should develop a pattern and stick to it for the entire design (above or below horizontal components, left or right of vertical components. Refer to Figure 6 for examples of correct and incorrect placement. I have seen students remove all references from their schematic and move them off to the side. This makes talking about the design very difficult, especially by phone. It is a lot easier to say "I think you have the wrong value for R1" than to say "you see the third resistor from the left of that diode...". Don't make your life more difficult than it needs to be.

Table 1. Common reference designators

| Code	Package	Code	Package
C	Capacitor	P	Connector, Plug
D or CR	Diode	PS	Power Supply
D or VR	Zener or Breakdown Diode	Q	BJT, SCR, SCS
D	LED	R	Resistor
F	Fuse	S	Switch
J	JFET, Connector, Jack, Jumper	T	Transformer
K	Relay	TP	Testpoint
L	Inductor	U or IC	IC
M	MOSFET	X or Y	Crystal
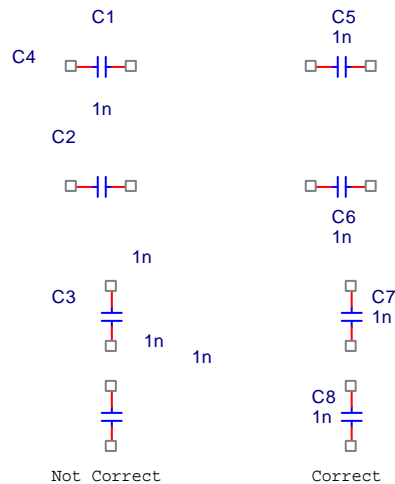
Figure 6. Positioning reference designators

On a related note, you should only use standard component values in a schematic. Students get used to using a simulation program that is happy with any value you put into the design. But while you can simulate with a 1.02376 kΩ resistor, you'll never find one in the parts bin. So unless you are doing something fairly unique, you only reveal your inexperience when you stray from standard values. And use an appropriate number of digits. Yes, a 10 kΩ resistor is the same as a 10.0 kΩ resistor, but if you have specified 1% parts, 10.0 kΩ is not ambiguous. If you don't know the value for a component yet, or if it needs to be determined during testing, give it a unique value. This way, the design can still go to layout (as long as you specify the footprint) but everyone knows more information is needed. A company I worked for used the term "select", for parts that needed to be selected by the engineer. For parts like these I change the "value" field to RSEL or CSEL to signal my technician that I don't know what I need yet[10].

5. Draw your schematics so that signals "flow" from left to right. With amplifier circuits this is usually very easy. With larger digital designs it can be more challenging. The goal is readability, so if following this rule makes the design less readable, don't do it.

6. Use busses when possible. All good CAD tools have the capability of grouping signals into a bus. Many designers don't think about this outside of digital circuit design. But if you have a 10-channel preamp, it could make sense to use a bus to connect those signals to the next stage. Avoid crossing busses if possible as this can be ambiguous. Again, the goal is readability.

7. Learn to use "global connectors" to minimize clutter. One global connector (or symbol) that we all use but probably don't think of in this way is the ground symbol. Some people will actually draw a schematic with a single ground symbol and a mass of wires going everywhere. Or they will have the power net across the top of a schematic and the ground net across the bottom. While this approach can work with smaller schematics and is often seen with transistor amplifiers, it often just makes the schematic less

---

[10] I always see a lot of 1nF caps and 1k resistors in student designs as these are the default values in PSpice. I always wonder if the student really wanted a 1k resistor, or if it just happened to work.

readable. Referring to the drawing below, the "bubbles" for +V and –V, and the "bars" for VIN and VOUT, are global connectors.
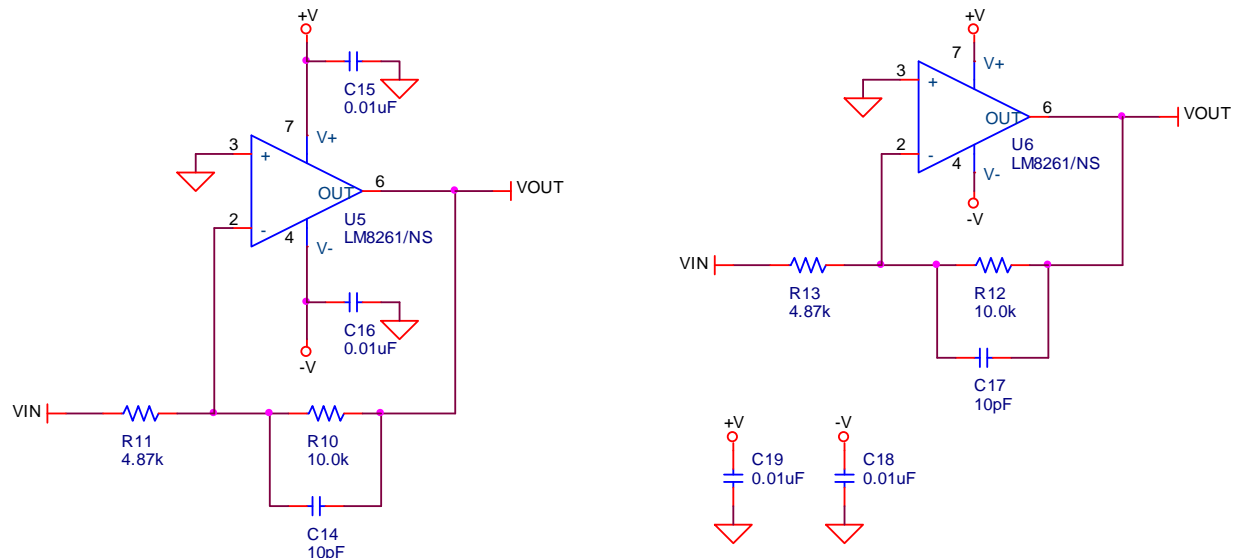
*Figure 7. Using global connectors in a design*

The difference between the two schematics is that the schematic on the right uses additional global connectors to move less important components[11], the bypass capacitors, away from the signal path to reduce clutter. While in this simple example it doesn't make a big difference, imagine a schematic with a few hundred I.C.'s. Globals will reduce clutter.

The downside of globals is that you can lose track of where a signal goes. If you are not careful you may not recognize that VOUT goes to several different locations, perhaps to several different pages of the schematic. As the designer you might recognize this, but as a technician, or as someone helping to debug someone else's design, it can cause a problem. For single page schematics, this can be handled by limiting your use of globals to logical connections, such as power, ground, or connectors. Another good rule is to have all of your inputs aligned on the left side of the page and all of your outputs aligned on the right. This works well with schematics spanning a few pages because you can find a signal name by simply scanning down the edge of the page. A signal from another page that appears in the middle of a drawing is easily overlooked. Naming signals logically and clearly will help avoid these problems.

For schematics that span many pages, a hierarchical design is best. With a design hierarchy, "ports" are used in place of the signal global connectors on the schematic pages. These ports are then used to develop a top level drawing that shows the interconnection of all of the signals between the pages of the schematic, as shown in the figure below.

---

[11] I can already hear the screams. Bypass capacitors are arguably some of the most important components in a design. However, it's safe to assume that any "good designer" would never leave them out, right? So since they are ubiquitous, they are often placed separately with instructions as to how to physically associate them with I.C.'s on the PC board.
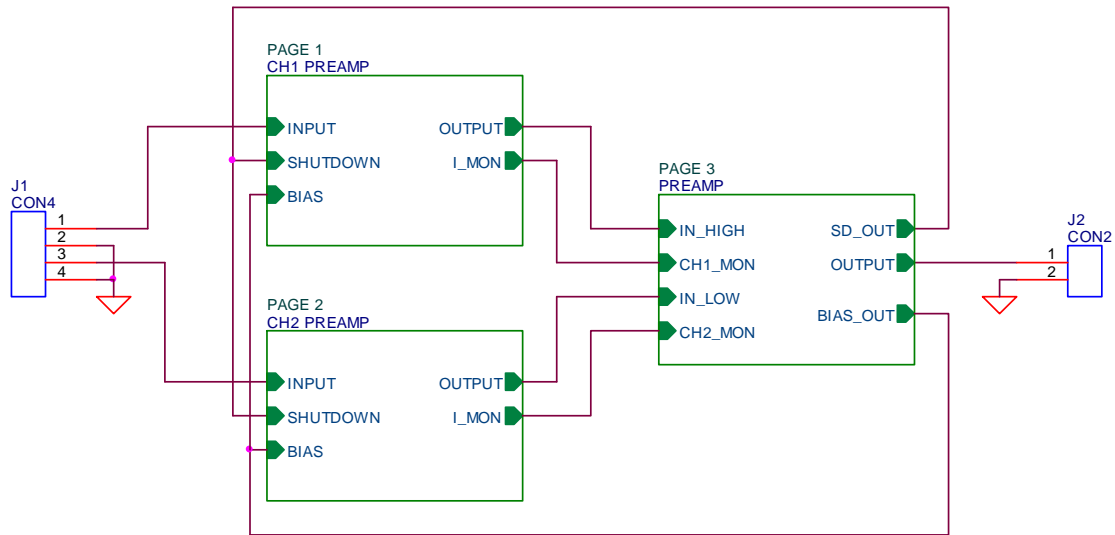
Figure 8. Top level of a hierarchical design

The better schematic packages permit you to select the hierarchical block and descend the hierarchy to the schematic page below. Usually there is no effective limit to the number of levels in a hierarchy, and most CAD tools permit a schematic to be re-used, i.e. if your design has multiple identical circuits (or channels) you only need to draw the schematic once and by placing multiple hierarchical blocks all referencing the same underlying schematic, you can build up your design quickly. This way, if the design of the channel changes, you only need to edit one drawing and the change is propagated through all of the channels in the design. Hierarchical designs can be a little confusing at first, but they are very powerful.

8. Another useful but less used global is the use of explicit Net Names. Net Names are placed directly on a "wire" connection. They are global, so they can be used for connectivity (as long as you don't misspell the name!). They are often used to name the signals on connector pins. Even if the signal doesn't go anywhere else in the design, they can be very useful for clarifying the purpose of a signal. Some CAD tools also let you place a "bar" above a name to indicate a digital signal is active low. However, a more common convention is to append a \ or a * to the name as some netlist generators cannot handle non-standard characters. Another use is when the design transitions to PC board layout. Rather than having some cryptic net name assigned by the software (N00237), you will see the text of the net name in the layout (IN_27). This helps in identifying nets that require critical routing.
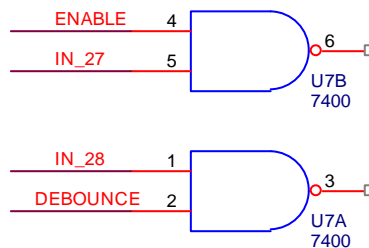
Figure 9. Net names

9. Most Op Amp symbols are drawn with the non-inverting input above the inverting input (see Figure 7). Because this is "upside down" from the way students see Op Amps drawn in class, they often flip the Op Amp orientation to match what they are used to. However, if you do this, your positive power pin will point down and the negative point up. The convention is that positive power is up and negative is down, so it is likely that someone will miswire your design if you start flipping parts. Do not flip parts without a good reason and always draw schematics with positive up and negative down. This goes for ground symbols as well. Ground should not point up, left, or right; only down.

10. The three columns of Figure 10 show the ground symbols available in the Capture CAD tool used in the department. Below each symbol is the associated net name, which is accessible if you double-click on the symbol. The "0" net is a required node in PSpice and all signals are references to this node. "GND_EARTH" would typically be used for a chassis ground or safety ground. I have never used "GND_FIELD SIGNAL" but I'm assuming it has something to do with RF design. That leaves "GND", "GND_POWER", and "GND_SIGNAL. There are two issues here. First is the physical shape of the symbol and what that shape means to most engineers. Second is the name of the net associated with the symbol and also that you can edit and change the name.

| 0 | GND | GND_EARTH |
| GND_FIELD SIGNAL | GND_POWER | GND_SIGNAL |
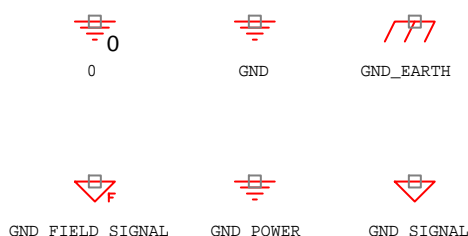
Figure 10. Orcad (Cadence) Capture ground symbols

In reality you could use any shape you want for a ground net symbol. You could even use one of the "bar" or "bubble" globals discussed above and just name it whatever you like. This would be a little silly, but it is important to understand that "ground" is no different than other nets as far as the CAD tool is concerned. What should be evident from looking at Figure 10 is that the symbol used for "GND" and "GND_POWER" are identical, yet the net name (which is not displayed in the schematic editor) is different. So if you are careless and use both of these symbols in your design, visually they all appear to be a single net, but the CAD tool will treat them as two separate nets that are not connected to each other. This is a huge problem.

Also, as mentioned above, it is possible to edit the net names associated with any of these symbols. I once had an engineer working for me who would do exactly that. Then, when I went in and edited one of his designs and had to insert additional ground symbols, we ended up with a real mess. Fortunately I realized what happened when we were laying out the PC board and corrected it before the board was built.

For these reasons, I established conventions that we use in my lab. First, never change the underlying net name associated with a ground symbol. Second, we use only the "GND", GND_EARTH", and "GND_SIGNAL" symbols in our designs. "GND_EARTH" is only used for chassis or safety grounds. "GND_SIGNAL" is used for analog ground and "GND" is used for digital ground. If a design does not need multiple grounds, either "GND" or "GND_SIGNAL" is acceptable.

There are times when another ground reference is required. Notice that the digital symbols in Figure 2 and Figure 9 do not show either ground or power connections. For reasons I've never looked into, logic symbols have hidden power pins. In Capture, these are typically named VDD or VCC and VSS or VEE. For connectivity, you need to specifically connect a global with that name to your power and ground nets, as shown in Figure 11. I typically do this where power enters my board so it is done and I don't forget. For the power connection, I use the common "bubble" global. For the ground connection, I use a global called "VCC_ARROW" and rotate it so it is pointing down. This serves several functions. First, it looks kind of like a ground symbol, so it makes sense. Second, it explicitly displays the name of the net, unlike the ground symbols in Figure 10 where the name is hidden. This is very useful because sometimes you need to have another, specifically dedicated "ground" in your circuit so that you can segregate a return current and treat it differently from the other grounds in your layout[12]. For example, you might have a thermoelectric cooler (TEC) that draws 5 amps. It could make sense to isolate that current from the (low current) grounds associated with the circuitry that measured and controls the TEC temperature, and connect them back where power enters your board. By using a separate but clearly defined symbol, your schematic can clearly illustrate this.
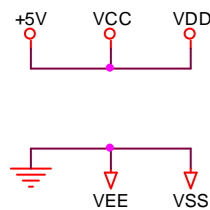
Figure 11. Method of connecting hidden power pins

11. Most CAD programs have tools for adding lines, boxes, arrows and text to a schematic. Use these tools to delineate circuit functions or add notes to be used when laying out, assembling, or testing the circuit. Anything that adds clarity is good.

12. Watch out for multi-part packages. Quad Op Amps or a hex inverter will show up in your schematic as U4A or something like that. Watch that all of the parts get "packaged" into the minimum number of IC packages required. It's unprofessional to use four gates from a package but have 20 unused gates because they are all in different packages. Also, <u>do something</u> with the unused devices as all unused inputs must be connected to an appropriate potential or you will likely have an oscillator or a large current sink on your board. For example, CMOS logic inputs can be tied directly to VDD or VSS. Some sources say that TTL inputs can be left open but I don't recommend it particularly for chips more complex than basic logic gates. TTL inputs should be tied to VCC through a 1kΩ to 5kΩ resistor (some sources say this is no longer necessary, so do your homework). Often you want to pick a state that has the lowest power

---

[12] While often neglected, "Ground" is a critical net in a circuit. Arguably the most critical. A good designer always thinks in terms of "return currents", i.e. how the current gets back to the source. This can make or break a design. If you allow a circuit that draws a large current to share a return path with a low current circuit, the parasitic resistance, capacitance, and inductance on the board will allow the high current circuit to influence the low current, likely destroying its accuracy or creating an unintended feedback path.

consumption. Unused Op Amps are best configured as a unity gain follower[13] with the non-inverting input tied to the middle of the power supply rails. Outputs can typically be left unconnected.

13. While perhaps not really a schematic issue, provisions for breaking up a design should in included particularly if you are working on a prototype. This can be as simple as including a jumper or a zero ohm resistor in between subcircuits so that each can be tested individually. Keep in mind however, that some analog circuits require specific load impedances, so if you disconnect the circuit from what it is driving you may want to include a provision to add the appropriate load. Also, testpoints (some means of connecting scope probes, signal sources, etc. should be added to your circuit (again, primarily for prototypes) to facilitate testing. This is a common error and nothing is more frustrating than discovering that a critical signal is hidden under a chip or on an inner layer and you cannot gain reliable access to it.

14. When you draw a schematic you should get used to adding connector symbols for all offboard signals and for power. Power has to get to your circuit somehow and while you may end up simply building the circuit on a "white board", it still makes sense to document all of the inputs and outputs. Keep in mind that all power and signal I/O should be twisted with a ground for low inductance and reduced susceptibility to EMI.

15. My last piece of advice is to use the Title Block. Name your circuit and put your name on the circuit. It makes the drawing look more professional. The figure below is an example of a basic schematic that should be produced by a student. It contains the minimum detail required for someone to build the circuit[14].

---

[13] This assumes the amplifier is compensated for unity gain stability. If it is not, you have more work to do to prevent it from oscillating.

[14] Some of the values are missing because the students in my class need to compute them. The schematic is also missing some details as to the footprints. This is OK because they are provided a parts kit for the project.
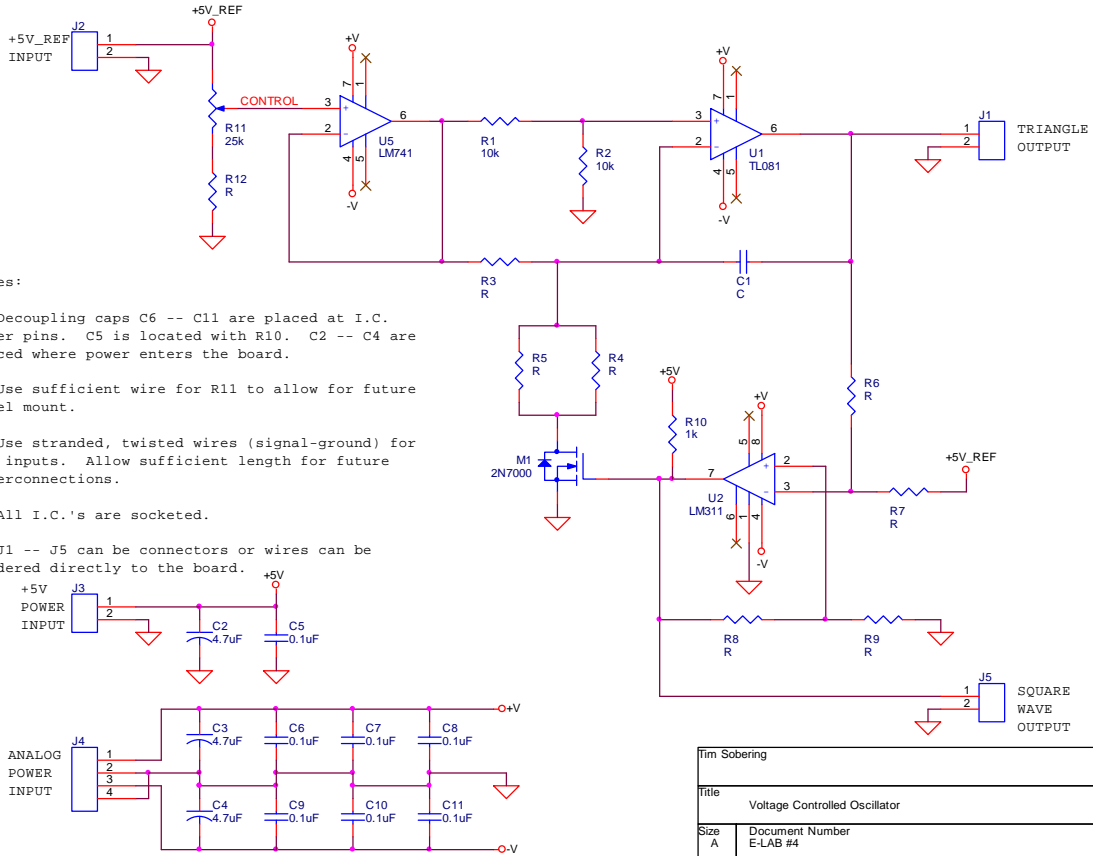
Notes:

1. Decoupling caps C6 -- C11 are placed at I.C. power pins. C5 is located with R10. C2 -- C4 are placed where power enters the board.

2. Use sufficient wire for R11 to allow for future panel mount.

3. Use stranded, twisted wires (signal-ground) for all inputs. Allow sufficient length for future interconnections.

4. All I.C.'s are socketed.

5. J1 -- J5 can be connectors or wires can be soldered directly to the board.

Tim Sobering			
Title	Voltage Controlled Oscillator		
Size A	Document Number E-LAB #4		Rev 1.0
Date:	Monday, March 03, 2008	Sheet 1 of 1	