

Using transistors as switches

by [Dan Morris](#)

Intro

A key aspect of proper hacking is the use of transistors for switching things on and off. A typical example is using a computer's parallel port to turn some external device on. I used to do this all the time, but I'm not an electrical engineer, and I don't claim to really remember anything I learned in the past about what's happening at the silicon level in a transistor. So every time I used a transistor circuit in a project, I would promptly forget the proper configuration for doing it, and everything I knew about which transistor type did what, and I've have to look it all up again next time.

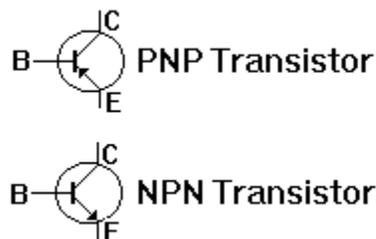
I assembled this document as a quick reference for myself, to avoid this painful lookup in the future... notice I say "assembled"; this is basically a "google composite", and I didn't write very much of it myself.

Basically what I want to know when I look at this document is:

- What transistor type do I need for my project?
 - What resistors do I need to use with this transistor?
 - How do I hook it up?
-

Notation

First some notation about transistor types and schematics. Transistors usually appear on schematics like this :



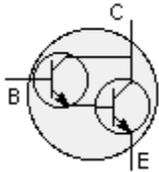
To keep emitter notation straight, you can think of a PNP's emitter "emitting" electrons, and an NPN's emitter "emitting" holes (positive charge). The arrow in a schematic is always the emitter. The collector then "collects" current carriers (holes or electrons).

The direction of the arrow on the emitter distinguishes the NPN from the PNP transistor. If the arrow points in, (Points iN) the transistor is a PNP. On the other hand if the arrow points out, the transistor is an NPN (Not Pointing iN).

Another point you should keep in mind is that the arrow always points in the direction of hole (positive charge) flow, or from the P to N sections, no matter whether the P section is the emitter or base.

Notation aside, the three pins – base, emitter, collector – are typically labeled on the data sheet for a transistor, or on the back of the box if you buy them at Radio Shack.

The Darlington transistor (I mention this because it's a term one comes across frequently, and because sometimes you actually need to use one for switching applications) is really two BJT's in one:



Darlington's can be used to yield very high amplification of a control current (since a Darlington's total gain is equal to the product of the gains of the two BJT transistors it is made from). These are generally used for high-current loads.

Applications

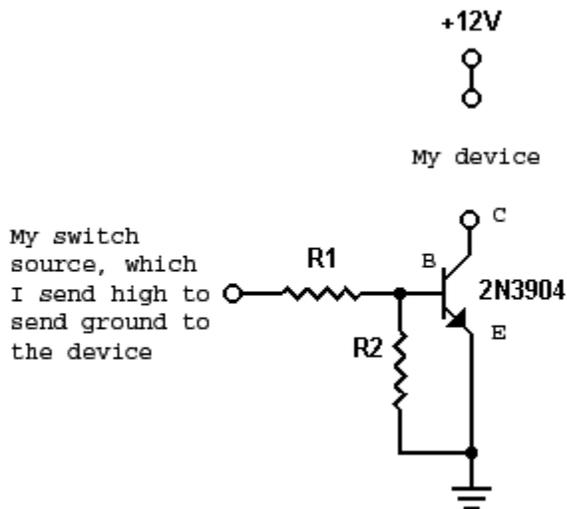
In my typical application, I want to turn some device on and off with some source of voltage (often the PC's 3.3V parallel port). I can do this in two ways... I can connect the device's ground to the world's ground all the time, and turn the device's power on and off. Conversely, I can connect power all the time, and switch the device's ground connection on and off. In practice, the latter is typically preferable. If I'm using a 3.3V source to switch on and off – for example – a 12V or 9V device, clearly I can't just power the device with my 3.3V source and expect it to turn on, so it's much easier to switch the device's ground than the device's power.

NPN transistors can be used to switch ground to a device. In this case, I would make the following connections:

- Device power to whatever power source I want to use
- Device ground to the collector on my NPN transistor
- Transistor emitter to “real” ground
- My “switch” – whatever line I am able to control from my button or my computer or whatever – to transistor base

If I make those connections, then connecting the base to *high voltage* (a little bit higher than ground) will switch *ground* to the device and start the current-a-flowing. This is intuitive to me, since on the side of my computer or whatever, a logical “1” means “yes, please let current pass”.

Here’s a schematic:



Notice there are a couple of resistors also. Resistor R1 controls the amount of switching current that goes to the device. To compute a good value for R1, you need to know:

- The current you intend to send through your load
- The voltage you’ll be using to switch your transistor
- The H_{FE} value for your transistor, which you typically get from the data sheet or the box your transistors came in

Skip to equation (6) if you just want to know the magic formula, read on if you want just a tad more intuition.

H_{FE} is defined as (load current / base current), (remember, a transistor is really a current amplifier), where load current is the amount of current flowing through my device, and base current is the amount of current flowing from my switching line to ground. So if I look at the circuit between my switching voltage, my R1, and ground, I have:

$$(1) \quad V = IR$$

...which really means:

$$(2) \quad \text{switching voltage} = \text{base current} * R1$$

Now I know that $H_{FE} = \text{load current} / \text{base current}$, so I have:

$$(3) \quad \text{base current} = \text{load current} / H_{FE}$$

...which I can substitute into (1) to get:

$$(4) \quad \text{switching voltage} = (\text{load current} / H_{FE}) * R1$$

...solving for R1 – which is what I wanted to find – I get:

$$(5) \quad R1 = \text{switching voltage} * H_{FE} / \text{load current}$$

In practice, we want to allow for a little extra load current (at the expense of a little extra base current), just in case, so we make it:

$$(6) \quad R1 = \text{switching voltage} * H_{FE} / (1.3 * \text{load current})$$

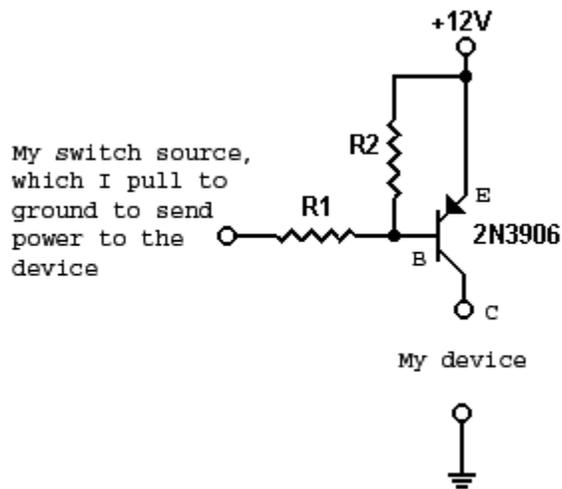
R2 is not that important here; it has to do with stabilizing the base and preventing it from going slightly negative when you turn the device on and off. The web says it should be about $100 * R1$, which is good enough for me.

I found that a good test circuit for the above diagram uses a standard LED in series with a 470Ω resistor as the “device”, and a 470Ω resistor for R1 (although the exact values don’t matter much here). I use a 5V power source for both the load power and the switch (I just plug R1 into the power source to turn the LED on).

Sometimes I want to switch power to a device instead of ground, for which I typically use a PNP transistor. All the computations are the same... the differences are:

- My device gets connected to the power side of the transistor instead of the ground side
- To turn the device on, I send *ground* to the base, instead of *high*

Here’s a schematic:



Switching power with a PNP transistor – instead of switching ground with an NPN transistor – is “The Right Thing” when your switching pin (the thing you’re going to connect to the transistor’s base) is “good” at pulling its output to ground. By that I mean that a device may have its connection to “high” go through a pullup, so the “high” state isn’t actually very good at pulling the pin high against a load, whereas ground may be a *direct* connection to ground (an “open collector” device). So in this case you’re really letting the pin float when you send it high, and the PNP schematic above may be more robust, assuming your switching device uses the same voltage as your supply.

On the other hand, some devices will have internal pull-down resistors to ground and a direct connection to a logical high. Only your device’s documentation can tell you this.

In either case, if your device is having trouble getting the base either high or low, a small-ish pull-up or pull-down resistor to “help” the device get to whichever state it’s “not good at” will usually fix the problem.