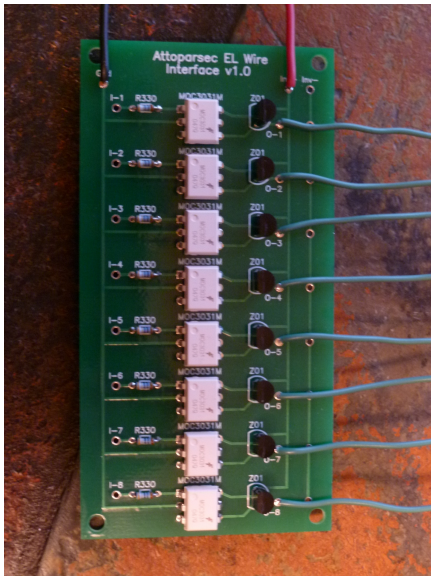
Attoparsec
Make. Stuff. Go.

8-Channel EL Wire Interface Kit

Introduction and orientation



Congratulations on your purchase of an Attoparsec 8-Channel EL Wire Interface kit! This product is designed to give you hundreds of megaseconds of reliable service. We don't know if it actually *will*, but that's what it was designed for.

In the kit you should find:

- 1 x printed circuit board
- 8 x 330 ohm resistors
- 8 x MOC3031M optoisolators
- 8 x Z01 triacs

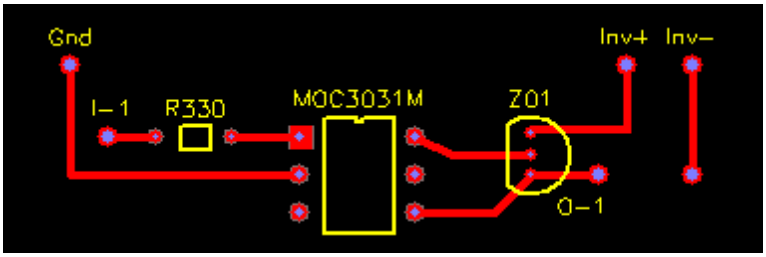
What you should not find:

- EL wire
- EL wire driver/inverter
- A microcontroller
- An echidna

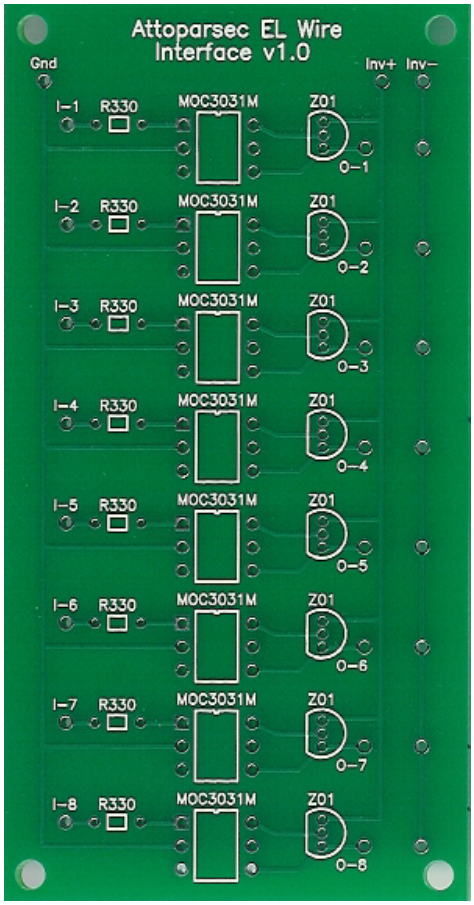
What, you might be asking, can the Attoparsec EL Wire Interface do for you? It can switch up to eight channels of EL wire (or any other high frequency, low power AC) using TTL level inputs. It is ideal for controlling moderately complicated EL wire displays from a microcontroller such as an Arduino. If you are of the bent, you could skip the microcontroller and hard wire a component sequencer, say a simple little 555 + binary counter circuit. The possibilities are only somewhat bounded!

Assembly

The interface card is simply the same circuit repeated eight times in parallel. Each consists of a resistor, an optoisolator and a triac.



For those interested in the design, the triac does the actual switching. Triacs are basically a form of transistor which can handle AC. Once opened they stay open until the source voltage drops. This happens twice a cycle with AC, so it only stays open as long as voltage is applied to the gate. In this case, the gate voltage is provided from the optoisolator. We *could* have directly connected it to the input, but that would mean your microcontroller would be a single triac failure away from 120VAC. With an optoisolator in the way, you can be assured that your precious magic smoke will stay locked away where it belongs. Finally the resistor serves to limit the current draw from your microcontroller output pin.



Assembly of the interface card is nothing more than the assembly of each circuit repeated 8 times. Bend the resistor wires down on each end flush with the resistor body, and insert through the holes in the board labeled R330. You might have to bend the wires out on the far side of the board to hold it in place. Match the notch in the optoisolator with the silkscreened notch on the board labeled MOC3031M and insert. These should hold themselves in place nicely, though it's sometimes a slight struggle to get the pins all through. The body of the triac has one flat side. Align this with the silkscreened outline labeled Z01 on the board and insert through the holes.

Once the components are inserted, flip the board and solder in place from the other side. Be careful not to bridge any of the leads! The Z01 leads are particularly close together, so take extra care there. A good soldering iron, circuit board clamp and bright lighting helps a lot.

You can do all the components at once, or one circuit at a time, or whatever you feel like. It really doesn't matter. Feel free to clip away the protruding excess leads as you go, if that gives you better access. You'll want to clip those off eventually anyway.

Implementation

So, all your components are soldered down. Now how do you use it?

The interface board has 3 global inputs and outputs (**gnd**, **inv+** and **inv-**) and 3 local ones for each circuit (**I-*n***, **O-*n*, inv-**).

Global:

- **gnd** - This connects to the microcontroller's ground. You can connect it directly to one of the gnd pins on an Arduino, for instance.
- **inv+** - This is the switched side of the EL wire driver. Calling it + is really a misnomer, as EL wire drivers are AC. Attach either side of the driver output here.
- **inv-** - The EL wire driver sink. Again, misnomer, deal with it. Attach the other side of the driver output here if you like. This is provided as a convenient way to complete the driver circuit, but if you would rather directly connect the return in some other way, feel free.

And then on each circuit:

- **I-*n*** - The control input for this channel. Connect this to the output pin of a microcontroller. Put the pin high to turn this line of EL wire on, put it low to turn it off.
- **O-*n*** - The output for this channel. Connect this to one side of the EL wire strand you want to control.
- **inv-** - This is the optional location to which the other side of the EL wire strand may be connected. If your design calls for a common sink closer to the strand in order to save on wiring, that works too. Just make sure that there is a complete circuit for each strand!

And that's pretty much it. You can now program the microcontroller to do whatever crazy display pattern you like. For gradual fade in/fade out effect it's easiest to use PWM output, but those pins are usually fairly limited. For larger displays you'll probably have to bit-bang it, or just use multiple microcontrollers. The EL wire jumpsuit featured on the Attoparsec page has to bit-bang the outputs, as it is using 3 boards for a total of 21 channels! Unfortunately this means it can't do a nice fade effect when in accelerometer mode, as the serial read delay is too high and induces flicker. There are always trade-offs.

Please let us know how the kit works for you and what you're doing with it. We'd love to put up a page of pictures and links to projects people have made with Attoparsec technology.