3:Physical Interfaces:Arduino

Interactive Installations F07

Dan Wilcox

C:Art:Media
Valand

Physical Interfaces

- A Few Notes ...
- Physical Interfaces
  - General IO: Keyboards / Mice
  - MIDI
  - Gamepads / Joysticks
  - Custom: Arduino, serial boards, etc

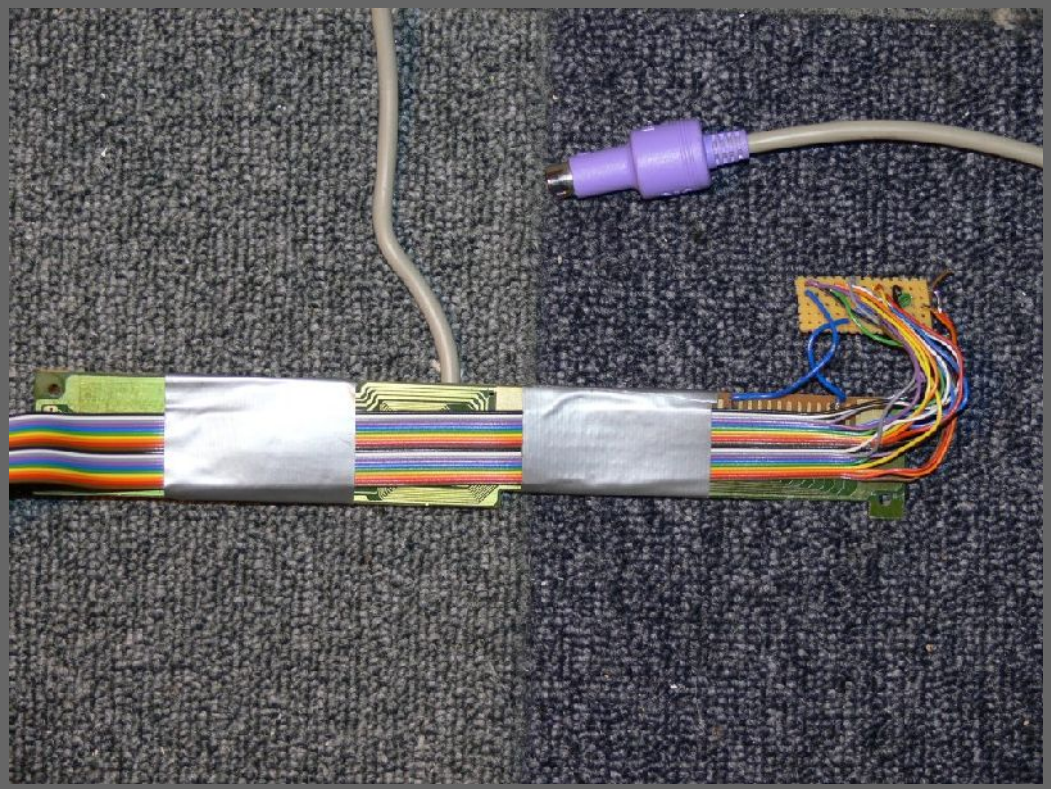
A Few Notes ...

- Before we examine some common computer physical interfaces it is important to note that:
  - you should not get lost in complexity, keep it simple!
  - you should refrain from using “amazing technology” for it's own sake
  - we're here to make art, not an overly expensive tech demo
  - choose what will work the best and the most reliably

General IO

- General IO: Keyboards / Mice
  - **Pros:**
    - cheap
    - easy to hack, user familiarity
  - **Cons:**
    - usually bound within OS (multiple mice/keyboards can only control 1 cursor/keymap)
    - key controller matrix must be reverse engineered

General IO



A hacked keyboard controller board
(keyboard minus case, keys, and key matrix)

Hanging ball maze
Switches “press a key” when the ball falls through a hole

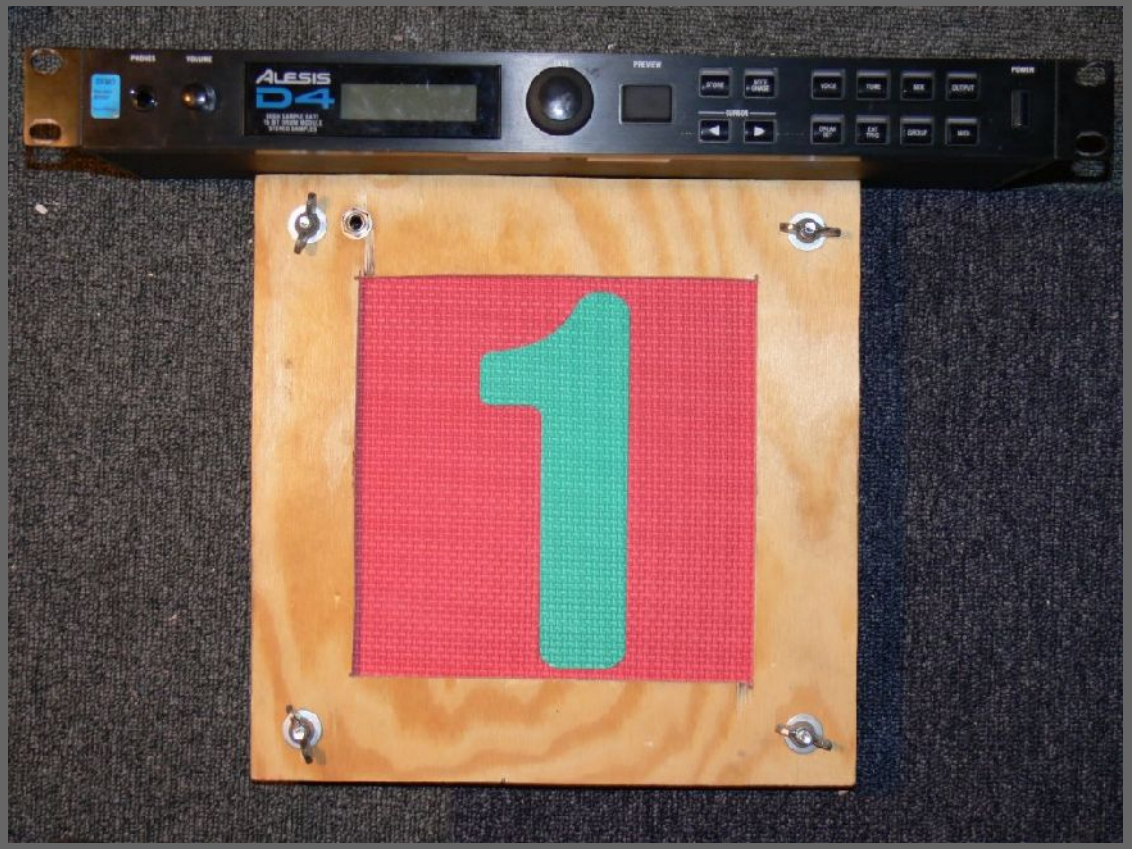
General IO

- Interface to PD using the following Objects:
  - Keyboard
    - [key], [keyup], [keydown]
    - Browser->examples/keyboardkeys/keyboard_main
  - Mouse
    - [MouseState]

MIDI

- Musical Instrument Digital Interface
  - **Pros:**
    - lots of user friendly, mappable devices available
    - nearly indestructible: most are built for stage use
  - **Cons:**
    - more expensive, harder to hack, need MIDI interface
    - less resolution (only 5 bit: 0-127)
    - requires knowledge of the MIDI protocol

MIDI



MIDI trigger interface / drum machine
+ custom piezo trigger pad

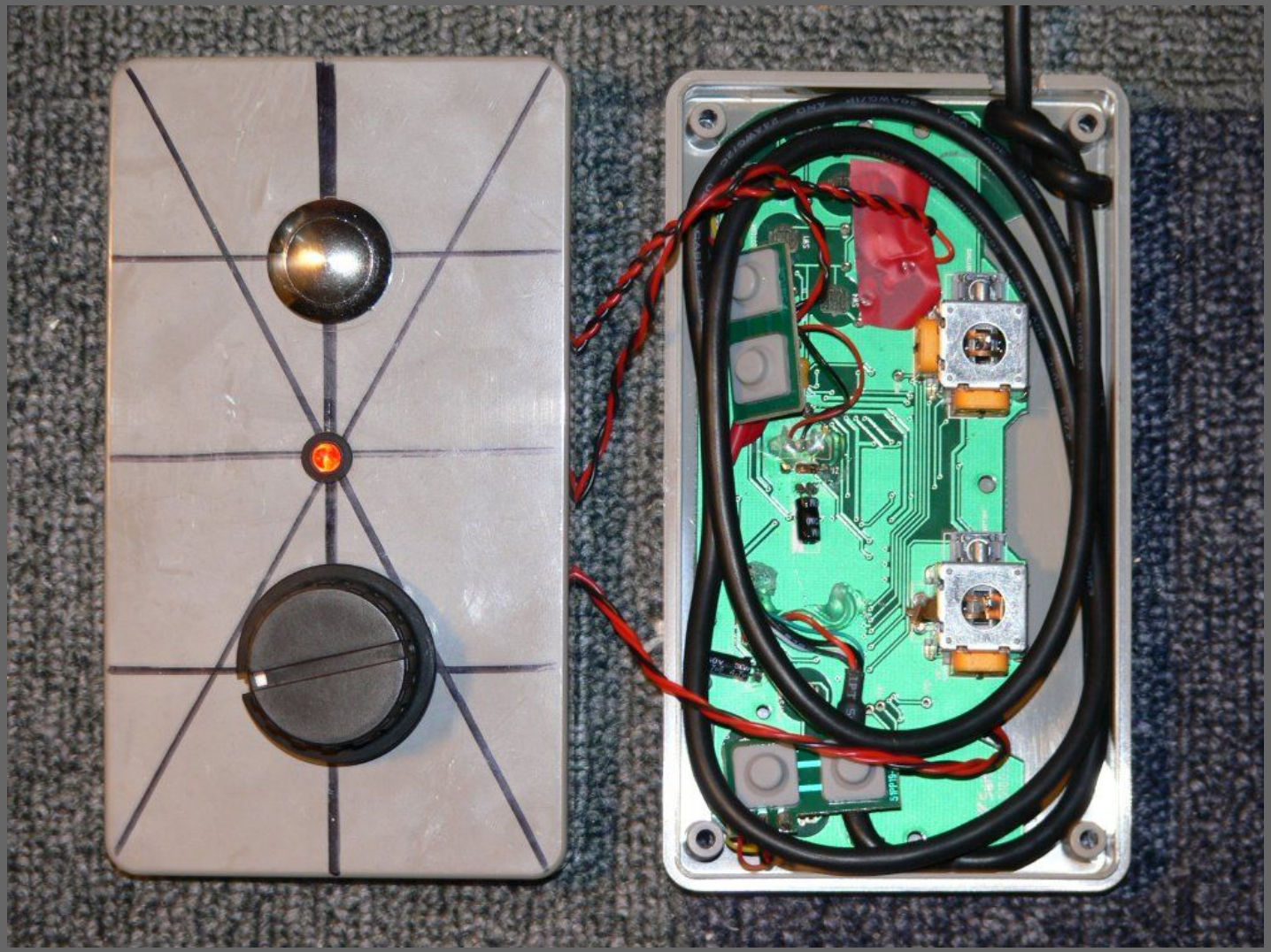
MIDI

- Interface to PD using the following Objects:
  - MIDI in
    - [notein], [ctlin], [pgmin], [midiin],etc
  - MIDI out
    - [noteout], [ctlout], [pgmout], etc
  - Browser->5.reference/other_midi_objects

Gamepads / Joysticks

- HIDs (Human Interface Devices)
  - **Pros:**
    - cheap, easy to find
    - easy to hack and build into custom cases / interfaces
    - lots of inputs: (~16 buttons, 4 adcs) and decent resolution (USB is 12 bit)
  - **Cons:**
    - requires a software driver interface

Gamepads / Joysticks



Custom control box built using a gamepad

Gamepads / Joysticks

- Luckily there is a driver in PD!
- Interface to PD using [hid]

Custom

- Arduino, Serial Boards, etc
  - **Pros:**
    - completely customizable, super flexible
    - powerful (external device can do some of the math)
    - lots of diy information online
  - **Cons:**
    - more expensive
    - typically more time consuming

Custom

Custom control / relay box built using custom circuit boards based on the Arduino platform

Custom

- Interface to PD using the following Objects:
  - Arduino
    - Pduino: [arduino]
    - http://at.or.at/hans/pd/objects.html
  - Serial Devices
    - [comport]

Arduino

- Arduino Start
- Pduino
- Simple interfaces
  - Potentiometers
  - Buttons
- Simple Outputs
  - LEDs
  - Relays

Arduino Start

- Let's start with a simple approach to the Arduino platform:
  - Pduino [arduino] PD object: turns Arduino into a sensor box that sends info to PD
  - Easy: wraps up serial communication for you
  - No need to write Arduino code, use the Firmata Arduino source!

Pduino

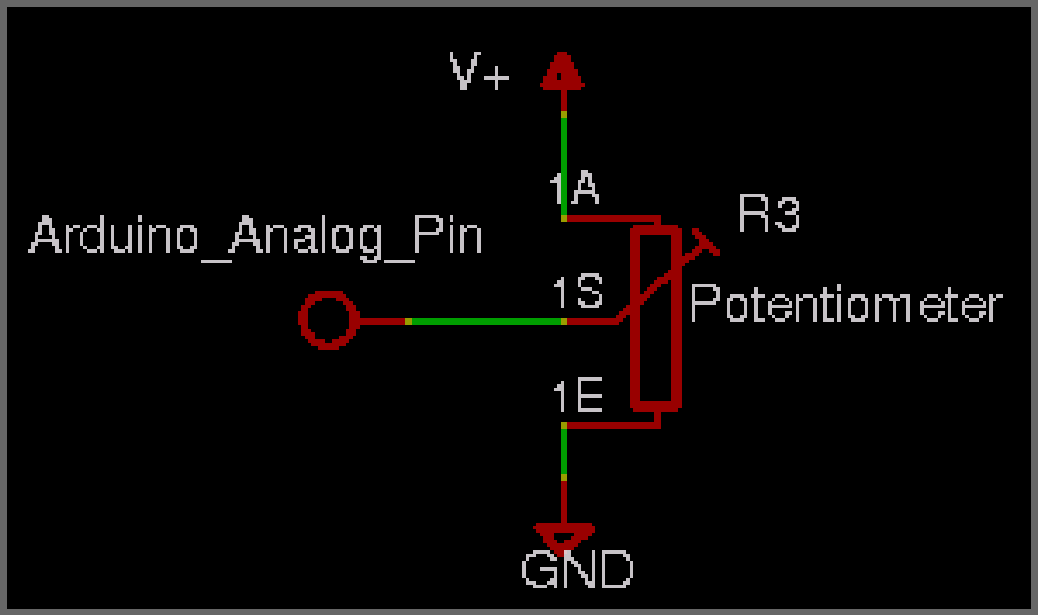
- Let's setup Pduino
  - Get the Arduino software from: http://www.arduino.cc/en/Main/Software
  - Download Pduino from: http://at.or.at/hans/pd/objects.html
  - Using the Arduino software, open the Firmata.pde file and upload it to the Arduino board
  - Open arduino-help.pd in PD and see if it works!

Simple Interfaces

- The simplest interfaces using an Arduino are:
  - potentiometers: knobs, sliders, etc
  - switches:
    - momentary: push buttons, rotary encoders
    - toggle: wall switches, small toggles, etc

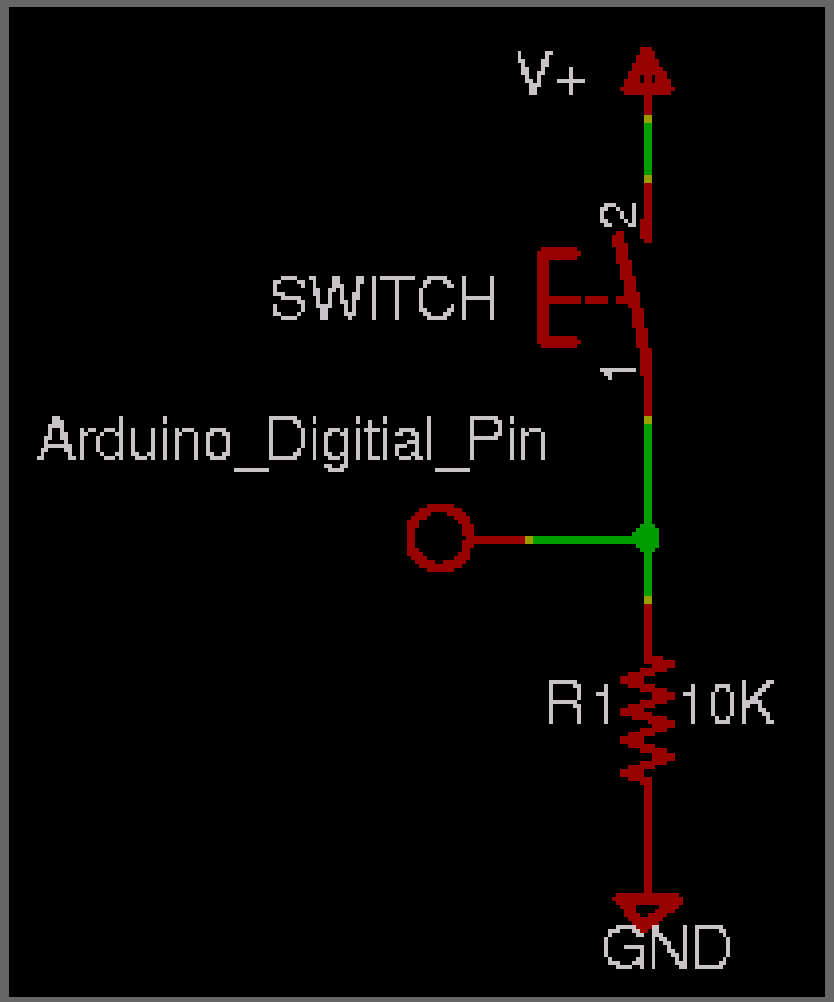
Potentiometers

- Variable resistors
  - analog
  - hook up to an analog input on the Arduino



Switches

- Close a connection
  - digital: on off
  - hook up to a digital input on the Arduino

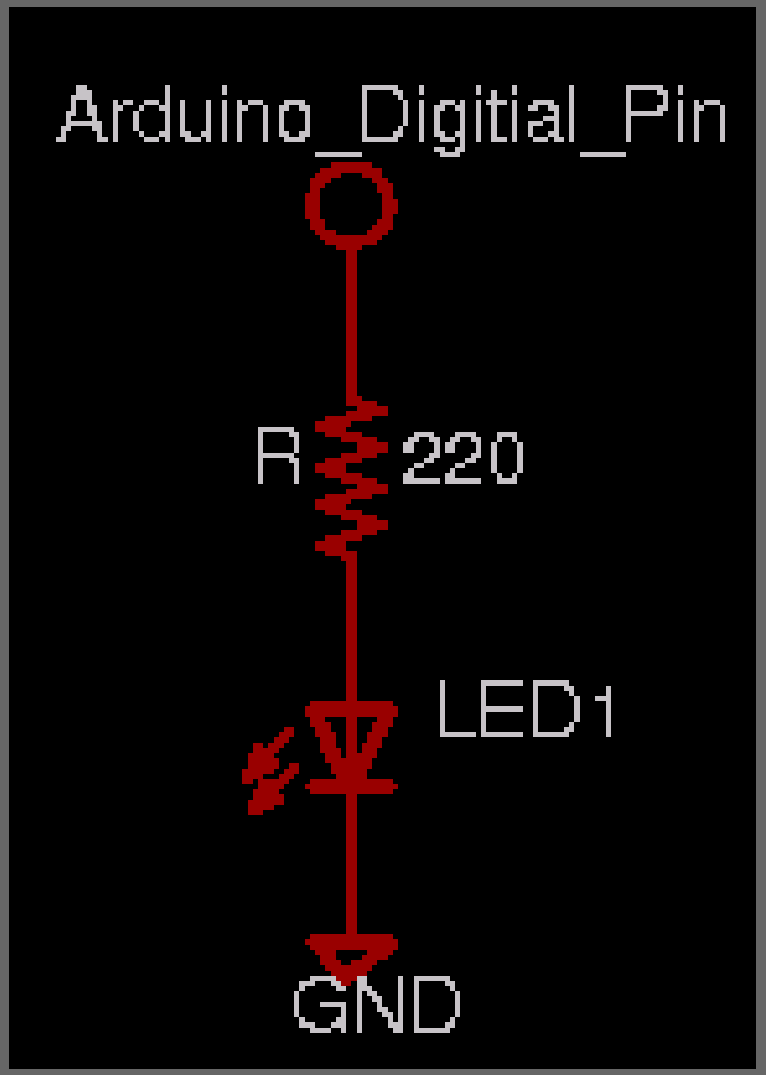


Simple Outputs

- The simplest outputs using the Arduino are:
  - LEDs (Light Emitting Diodes): colored lamps
  - Relays: controllable switches

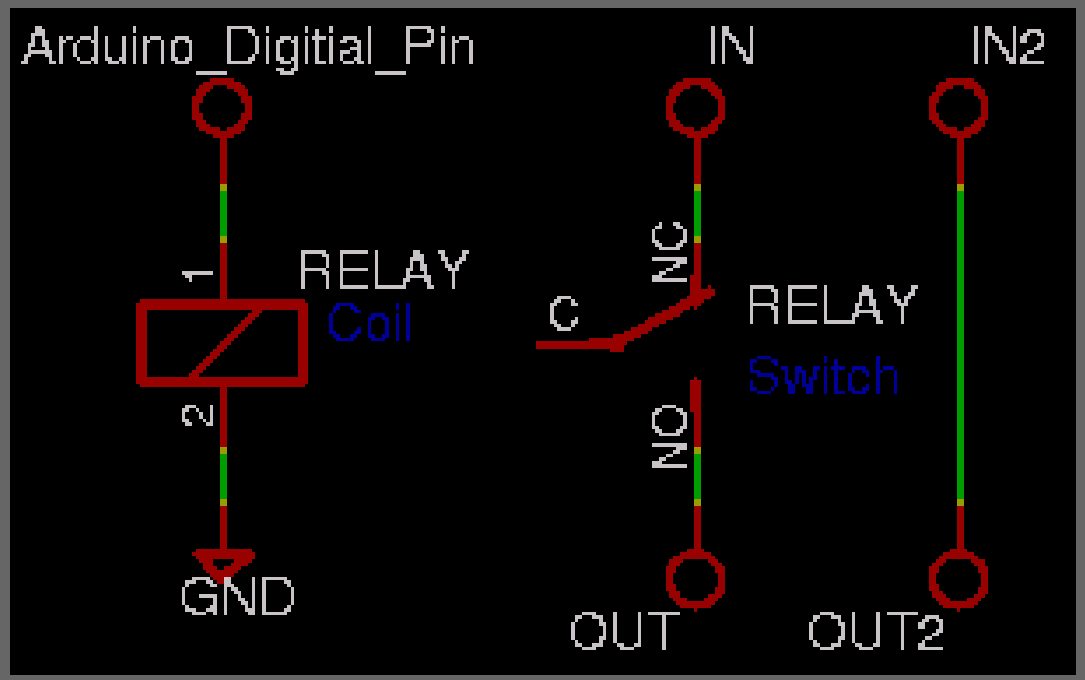
LEDs

- Small, colored lamps
  - digital: on off
  - hook up to a digital output on the Arduino
  - the long pin on the LED is the + side
  - R = 220 or 330 Ohms

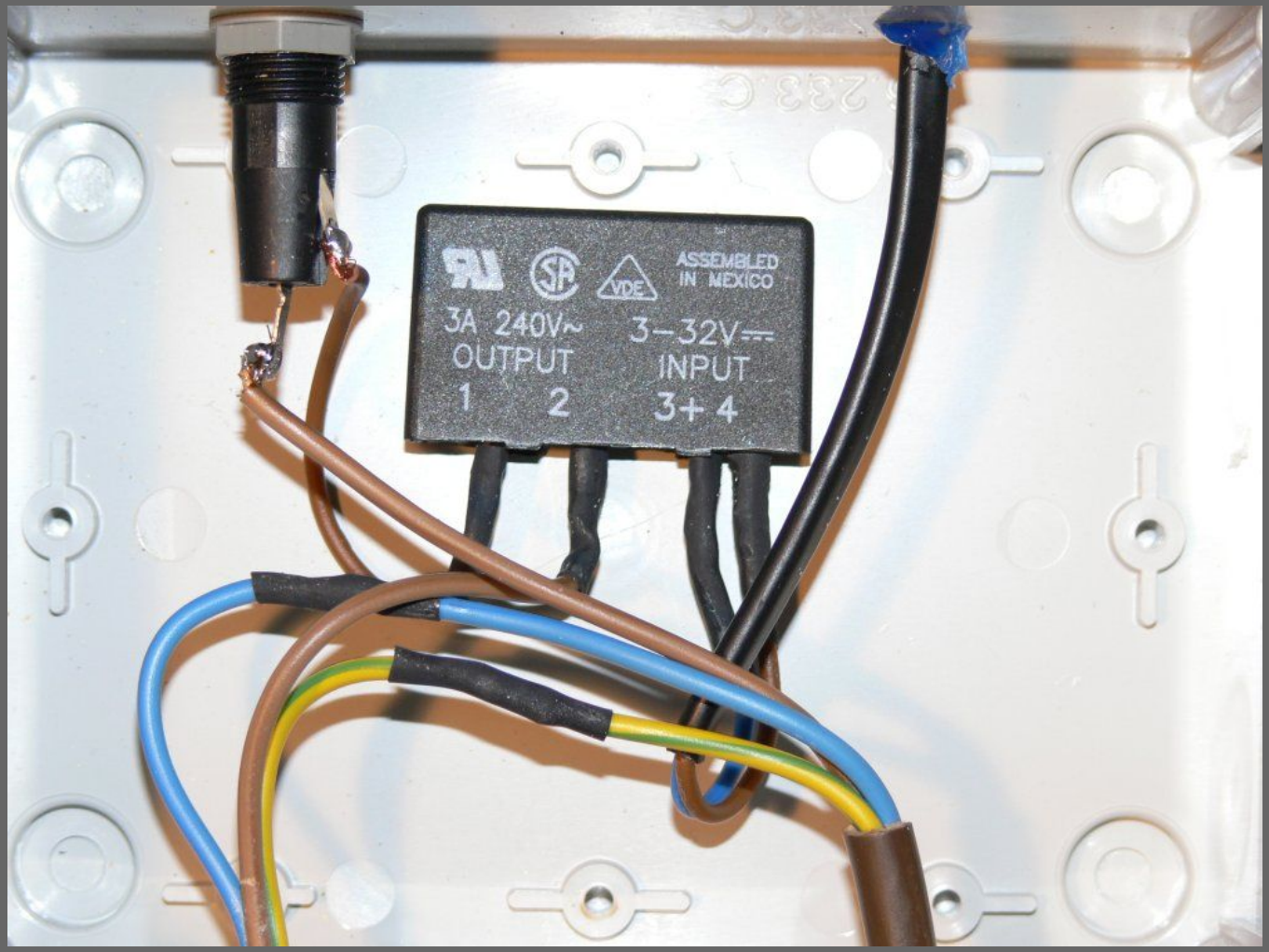


Relays

- Close or open a switch
  - digital: on off
  - hook up to a digital output on the Arduino



Relays



An 240V AC relay switchable by a 3 – 32 V DC signal