PAT 451/551 INTERACTIVE MEDIA DESIGN I

MOTORS

CONTROLLING HIGH-POWER DEVICES

GENERAL PRINCIPLE

POWER

to the device is supplied externally (NOT from the Arduino)

- For DC devices, from a wall-wart or other DC power supply

- For AC devices, from AC mains power outlet

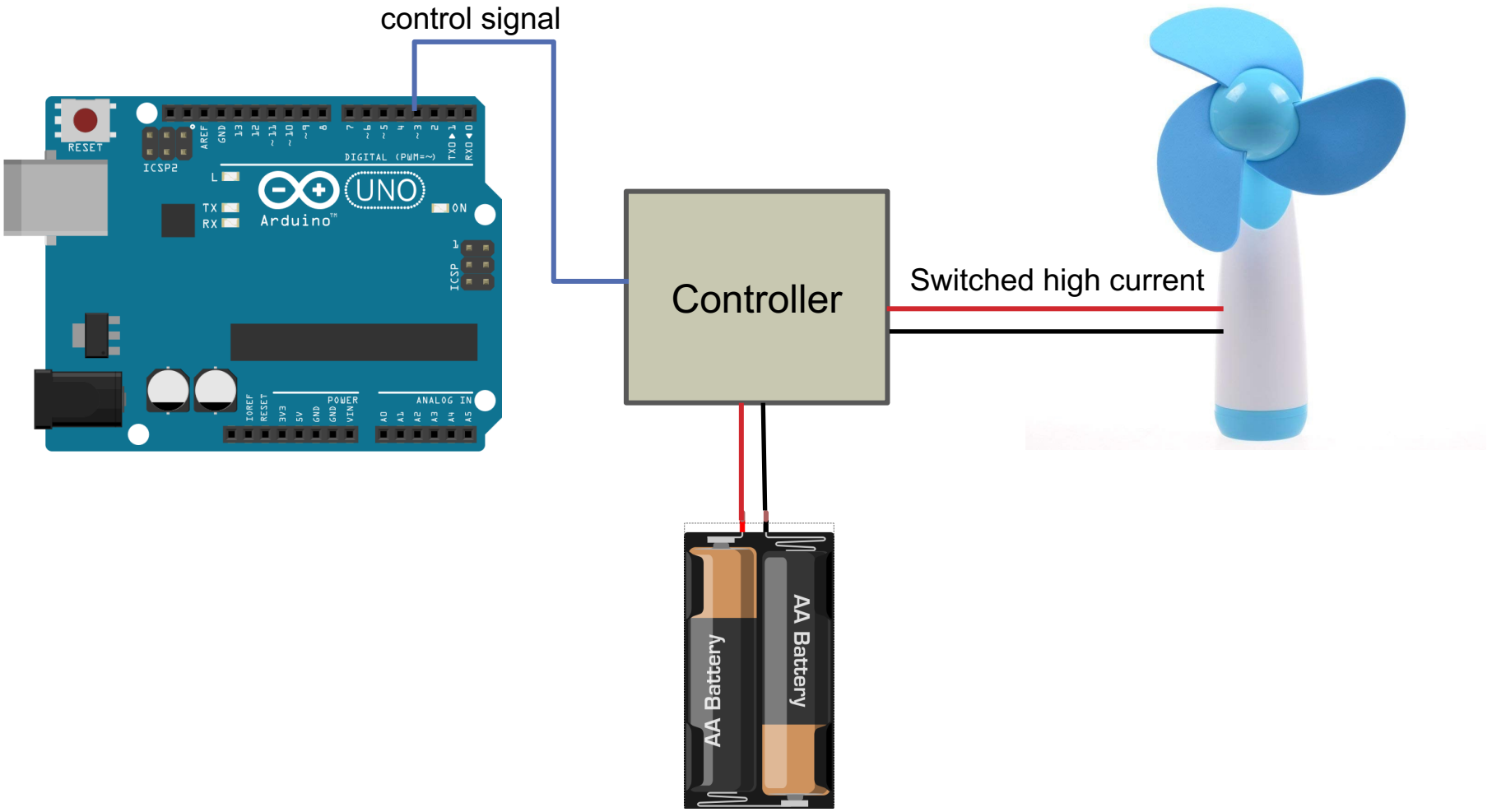
A chip or controller

allows the Arduino to switch on/off the current flow to that device using a low-voltage control signal

Why?

Arduino digital I/O pins can only sink/source 40mA. Most devices like motors, lights, solenoids require much more power than that.

GENERAL PRINCIPLE



CONTROLLER TYPES

Transistor

- Simplest for digital/binary on/off control
- DC devices only, current/load limited
- Fast switching, can be held on/off indefinitely
- Potentially noisy for continuous control
- No direction control for motors

H-Bridge

- Designed for motor control
- Usually used on a separate controller board
- Most can power 2 motors, direction and speed control
- DC or Stepper motors

Relay

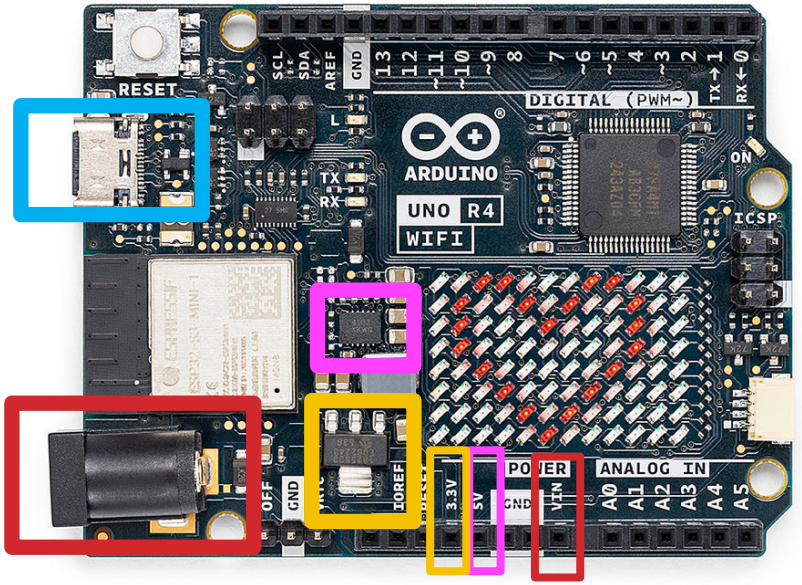
- For intermittent switching; can't be held indefinitely
- Can handle higher loads/current than transistors
- AC or DC versions
- Electromechanical or Transistor-based control

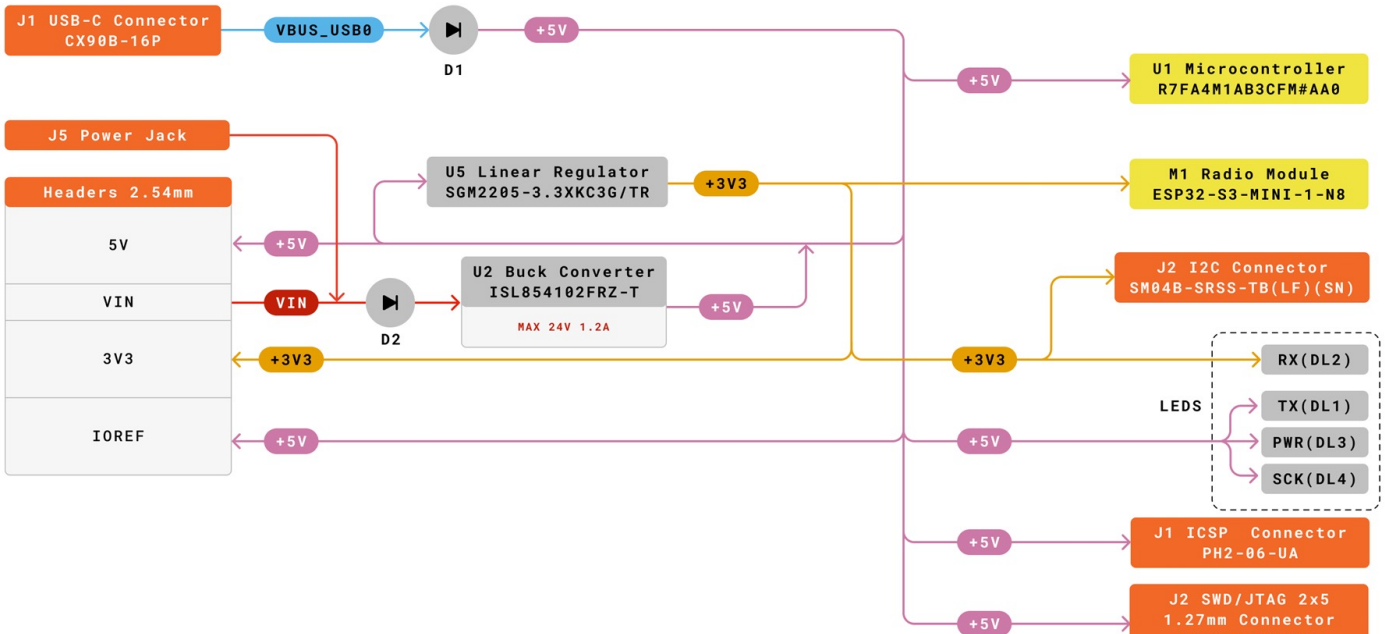
ARDUINO POWER ARCHITECTURE

V_in pin carries whatever voltage is supplied on the **barrel jack connector (red)**

5V Pin carries a 5V, either from the **Buck Converter** or **USB**

3.3V Pin carries 3.3V from a **voltage regulator**





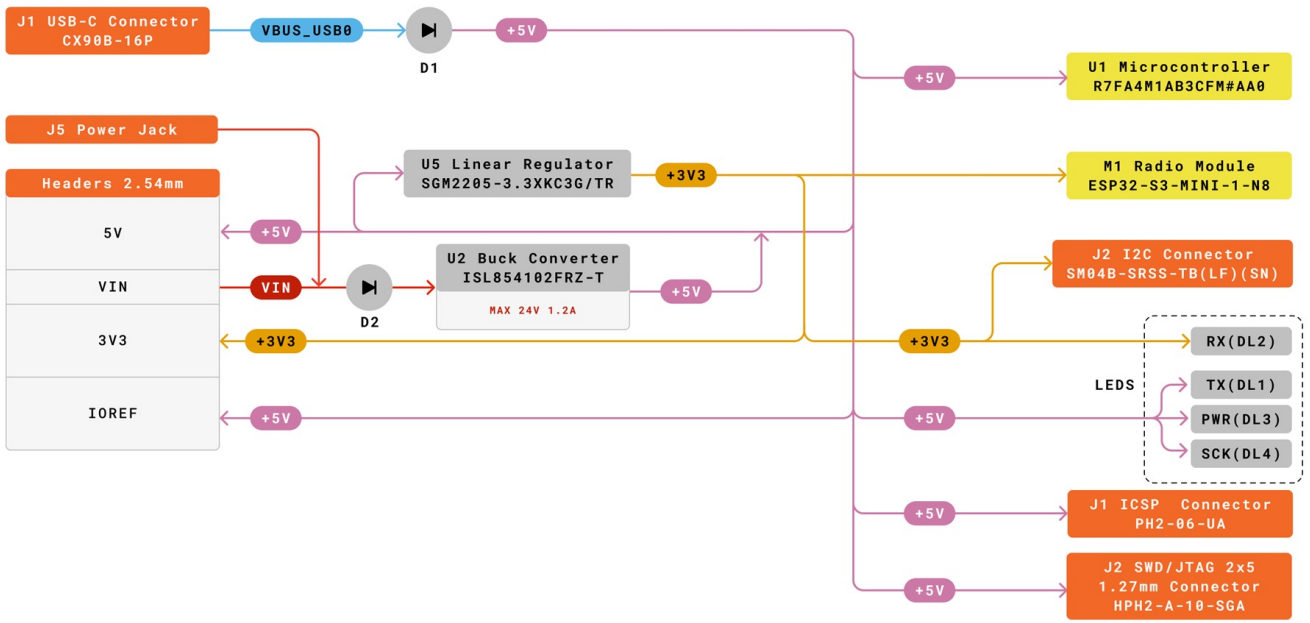
ARDUINO POWER ARCHITECTURE

Barrel jack connector:

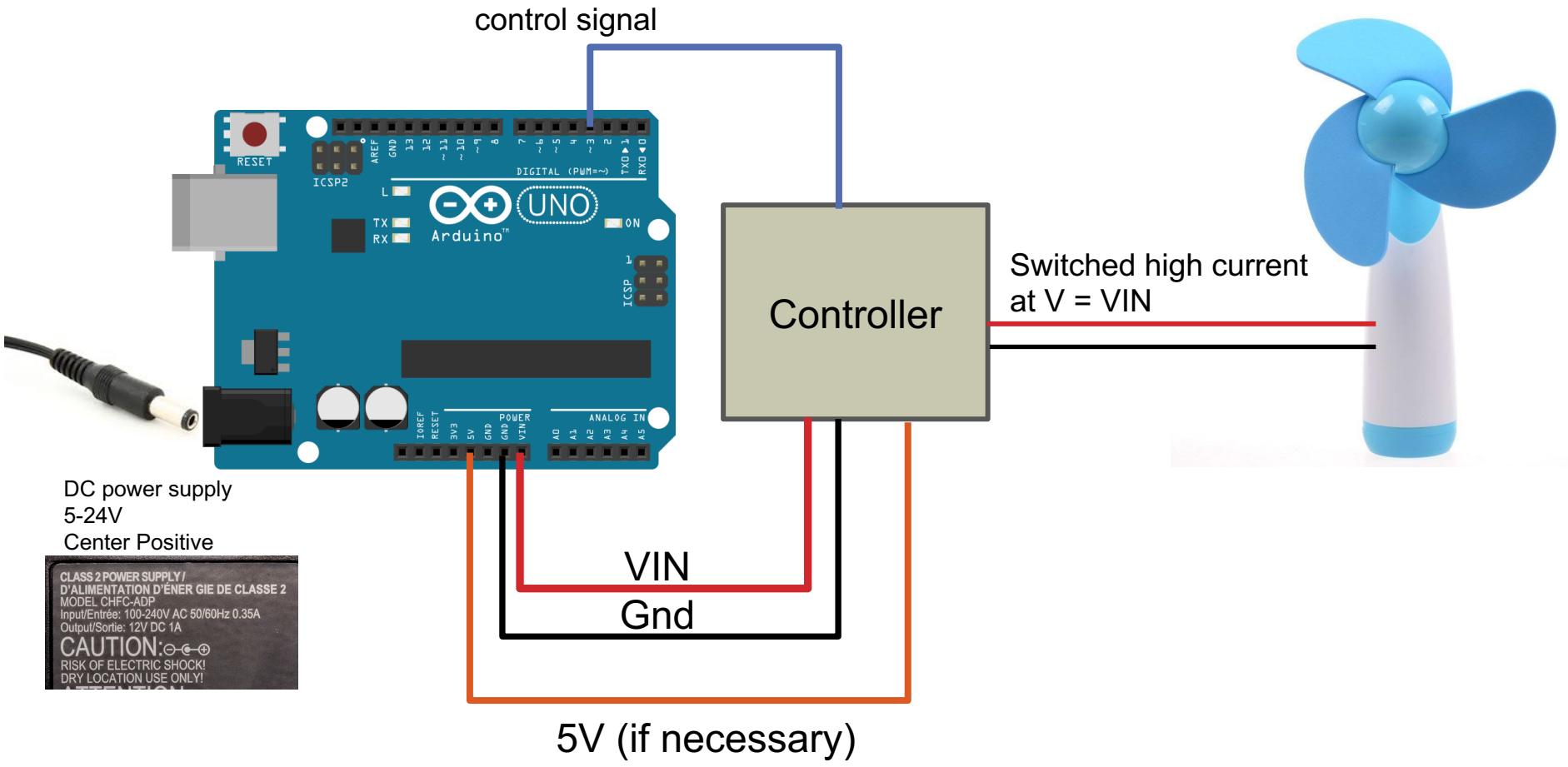
Must be 5-24V DC
center positive

If V_in > 6.6V, that voltage will be used to power the Arduino

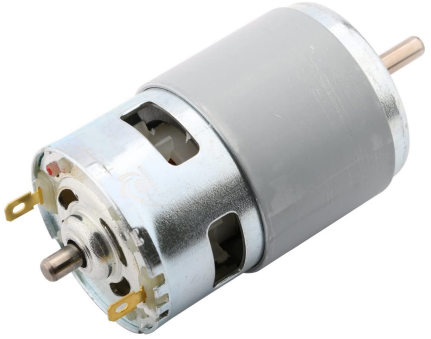
If V_in < 6.6V, the 5V USB power will be used to power the Arduino

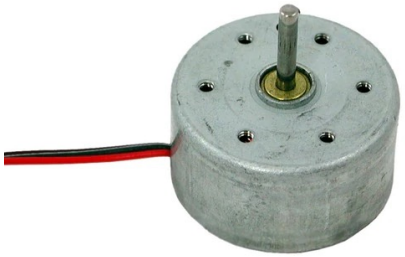


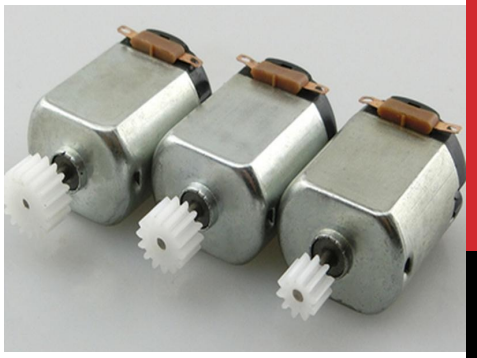
USING ARDUINO VIN



DC MOTORS







- DC: Direct Current
- Driven by some constant positive voltage (and a ground reference)
- 2 Wires
- Usually bidirectional: swap voltage and ground to turn the other direction
- Drive with a **PWM signal** instead of a constant voltage to control speed.
- Applications: robotics, remote/self-driving vehicles, haptic/vibration “buzzer, industrial automation, turntable, automotive (windshield wiper, windows)

DC MOTOR SPECS (1)

- **RPM (Revolutions per Minute):**

  Number of full 360° turns in a minute, i.e., speed. Rated at "no load", meaning nothing attached to the shaft. A "load" will decrease the effective RPM.

- **Voltage**

  The recommended voltage supplied to the motor. RPM is rated at this voltage. You can often go above the rated voltage by a little bit (3-5V), but you shouldn't drive it at full speed (100% PWM duty cycle) for very long. This will dramatically shorten the lifespan of the motor.

- **Maximum Current or "Stall Current"**

  If you see a current rating, this is the Maximum current a motor can handle. The motor will draw more current depending on the load. Practically, you want to make sure that your **power supply's current rating is greater than the motor's maximum current.**

DC MOTOR SPECS (2)

- **Stall Torque**

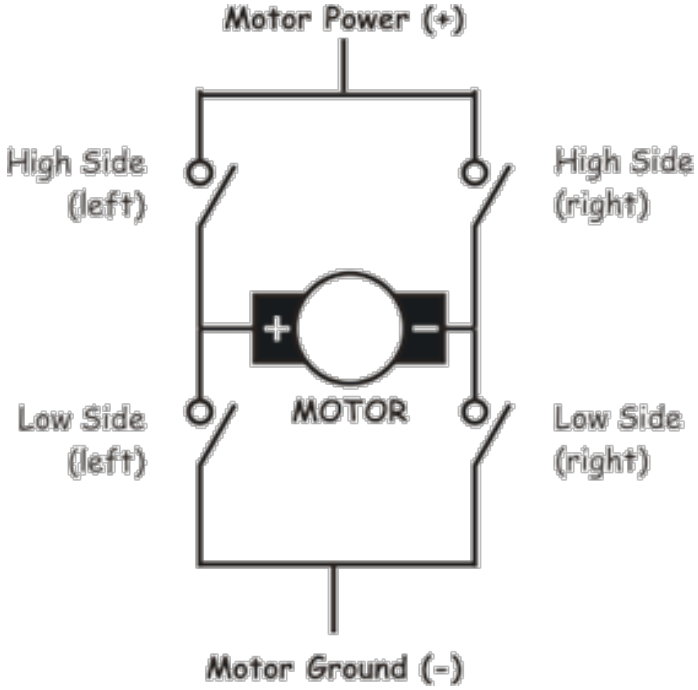
  This is the maximum torque (rotational force) that a motor can supply to a load where the shaft is not turning. Think of trying to lift or push something that is too heavy. The Stall Torque is the maximum force the motor can push to move a heavy load. The Maximum Current or Stall Current is the amount of current the motor is drawing under this condition.

H-BRIDGE

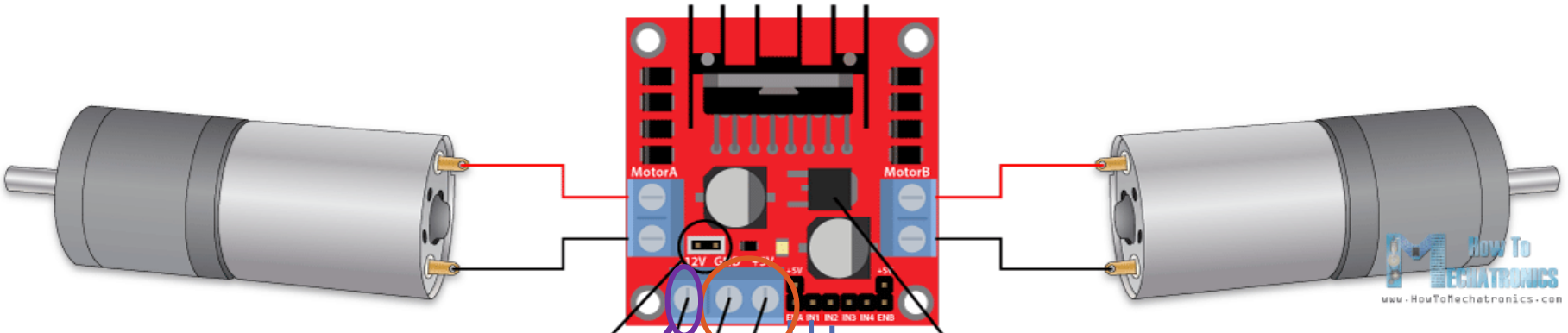
Allows you to use PWM to drive a DC motor

Motor power is external

Switches are transistors that are opened and closed by PWM signals



L298N MOTOR CONTROLLER



(disregard this) Jumper for the 5V Regulator

(from Arduino Vin) Motor VCC 5V - 35V

Ground

(from Arduino +5V) +5V In/ Out

Control Pins

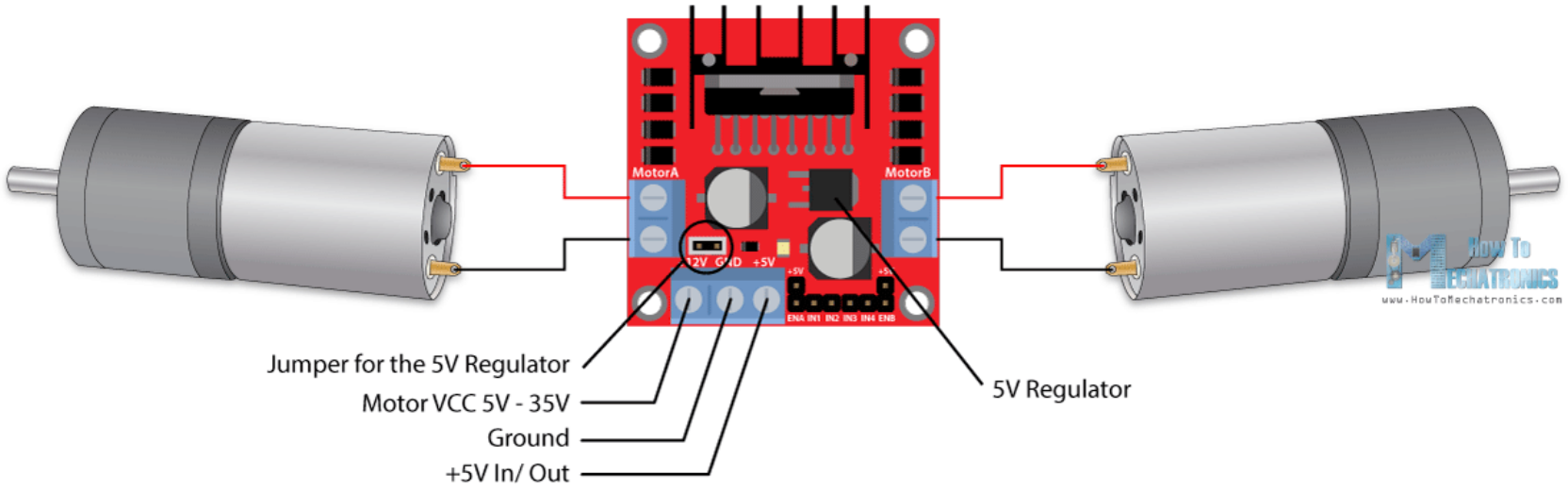
IN1: Direction 1
IN2: Direction 2
ENA: Motor Speed (PWM from analogWrite)

Clockwise
Direction 1: HIGH
Direction 2: LOW

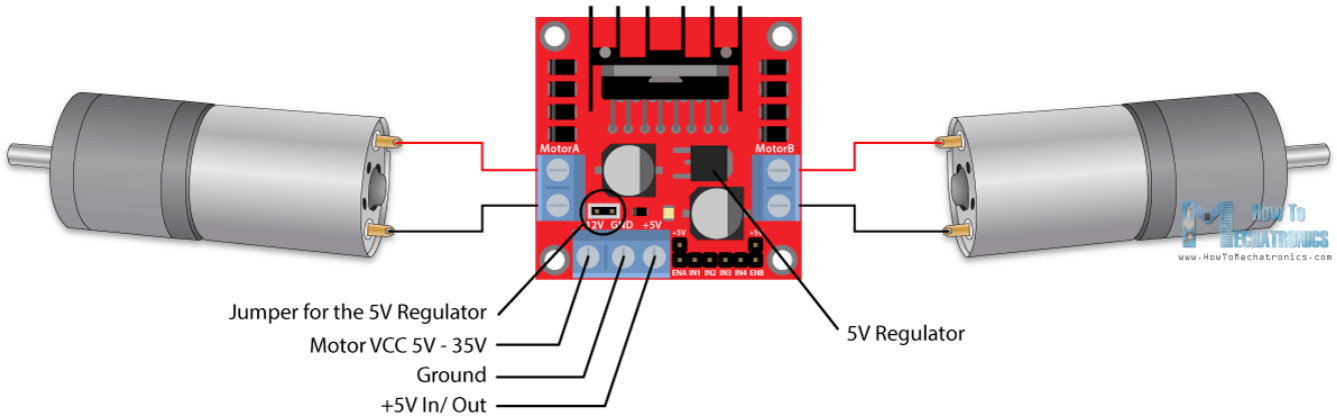
Counterclockwise
Direction 1: LOW
Direction 2: HIGH

From Arduino VIN pin (**use external supply connected to Arduino Barrel Jack, 5-12V**)

From Arduino 5V and Gnd



- Can control 1 or 2 DC motors.
- Supply motor power from Vin pin and +5/Gnd from Arduino.
- **Arduino should be connected to external power supply**.** Power supply voltage will affect motor speed (higher V = higher speed).
- Motors I supplied are rated for 5V, so don't use more than 9V or you might fry the motor.
- **If not connected to an external supply, Vin will carry 5V from your computer's USB Bus. This appears to be sufficient to power the small 5V motor I supplied you. But it's potentially risky to draw too much current from your USB bus.



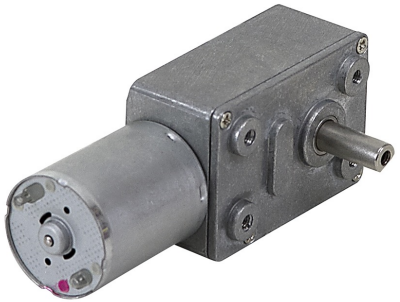
- Motor connects to OUT1 and OUT2. Polarity doesn't matter, just affects which direction it turns.
- ENA connects to an Arduino PWM pin for speed control
- IN1 and IN2 connect to Arduino digital outputs for direction control
- Set IN1 and IN2 pins with digitialWrite() from Arduino
  - IN1 == HIGH && IN2 == LOW: Clockwise
  - IN1 == LOW && IN2 == HIGH: Counter-Clockwise
- If IN1==HIGH && IN == HIGH, the motor will "brake." This is ok to stop it suddenly, but don't want to hold it like this for a long time.

(ENB, IN3, IN4 would be for a second motor using OUT3 and OUT4)

OTHER MOTOR TYPES (1)

- **DC Gear Motor**

  A DC motor with a gearing mechanism attached to the shaft to make another shaft rotate more slowly. It is difficult to make a very slow DC motor, so gears are more common. Used for slow-speed applications (windshield wipers, lifting jacks). You can often see the separate gearbox. No functional difference from a regular DC motor.



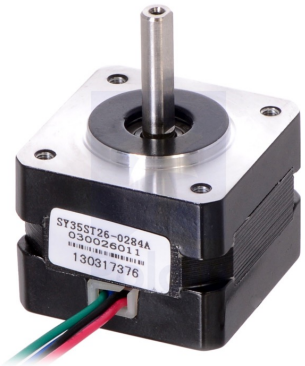
- **AC Motor**

  Driven by AC current: often main power from your wall outlet. But also could be from an AC power supply. Usually don't have fine-grained speed control, so not good for precision applications. Applications: household appliances (vacuum cleaner, washing machine, hair dryer); industrial.

OTHER MOTOR TYPES (2)

- **Stepper Motor**

  DC motors that move in discrete steps. Have a number of “phases” that are activated in sequence. Can be 4-400 steps per revolution. Used in precise positioning applications like printers, scanners, CNC machines, laser cutters. They power lots of “maker” tools.



Can be used in “open loop” position control—no feedback / position sensing is needed because we know how far it moves in each step. BUT you need to know where it is at the beginning. When you start up your printer, it has a calibration phase where it moves the motor to both ends of the range, where it’ll hit a “limit switch.”
See: https://learn.adafruit.com/all-about-stepper-motors/types-of-steppers

OTHER MOTOR TYPES (3)

- **Servo Motor**

  An AC or DC motor with an integrated position sensing mechanism. The position sensor could be a potentiometer or an optical encoder. Usually involve a “PID” (proportional–integral–derivative) control scheme. Often used in situations where you don’t need 360° rotation—Think about systems like the ‘flaps’ on an airplane, a robotic arm joint, or a rudder on a remote-controlled boat.

  See: https://www.jameco.com/jameco/workshop/howitworks/how-servo-motors-work.html