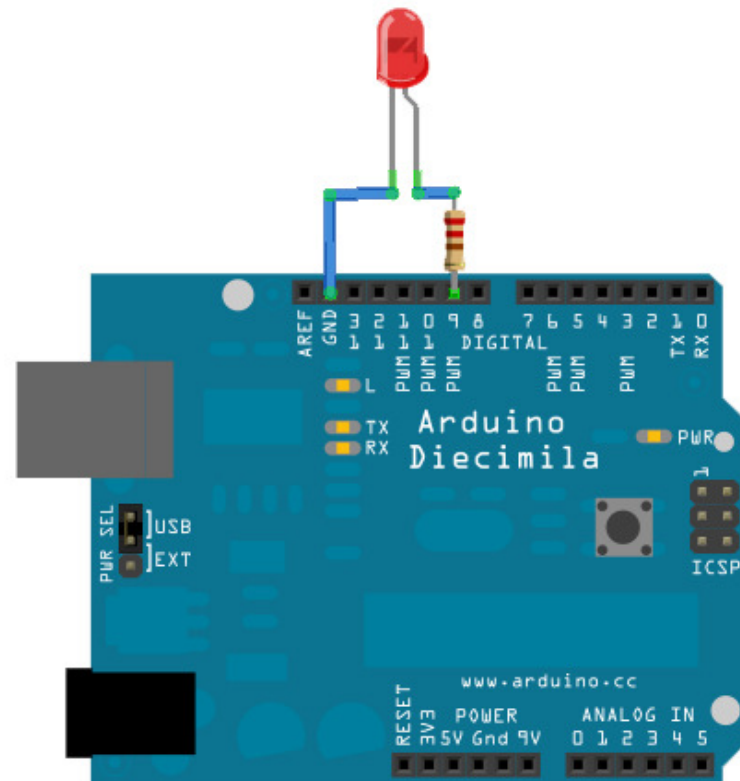
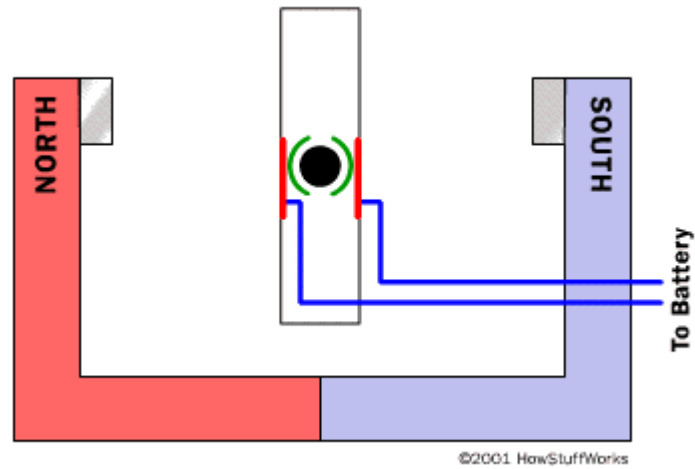


Last week...

# Exercises we did...

- Custom functions
- Step up / Step down
- Processing
  - Visual Dimmer
- Firmata
- Arduino vs Other





# MOTORS

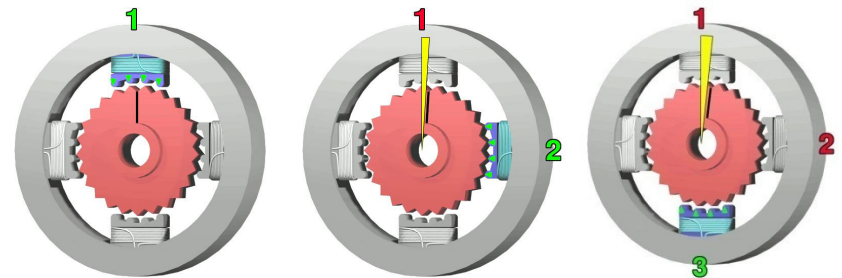
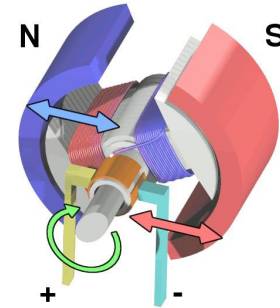
Servo

Stepper

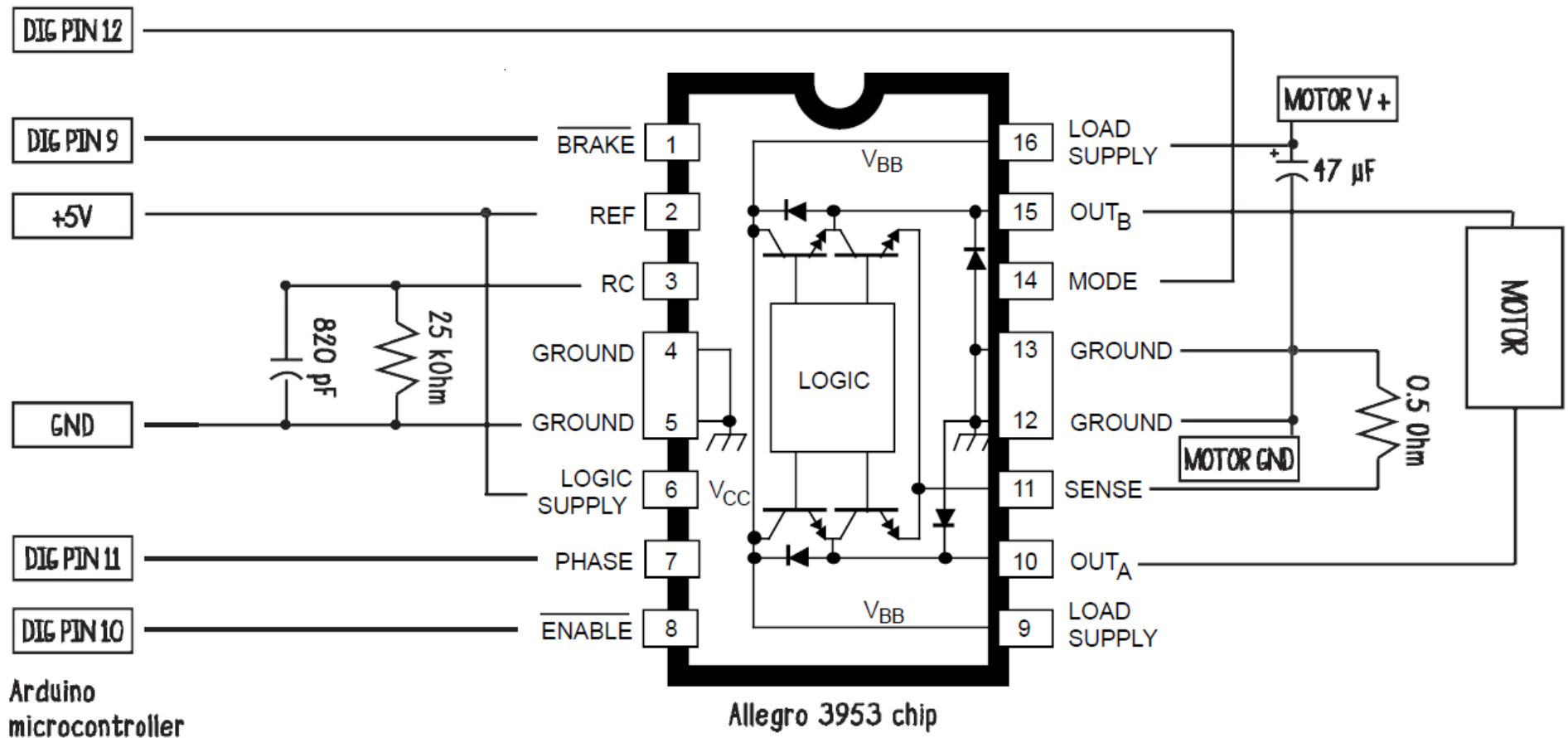
DC

# Types

- DC Motors
  - Continuous rotation
- Stepper Motor
  - Stepped Movement
- Servo Motor
  - Tracked stepped movement



# Motor Controller

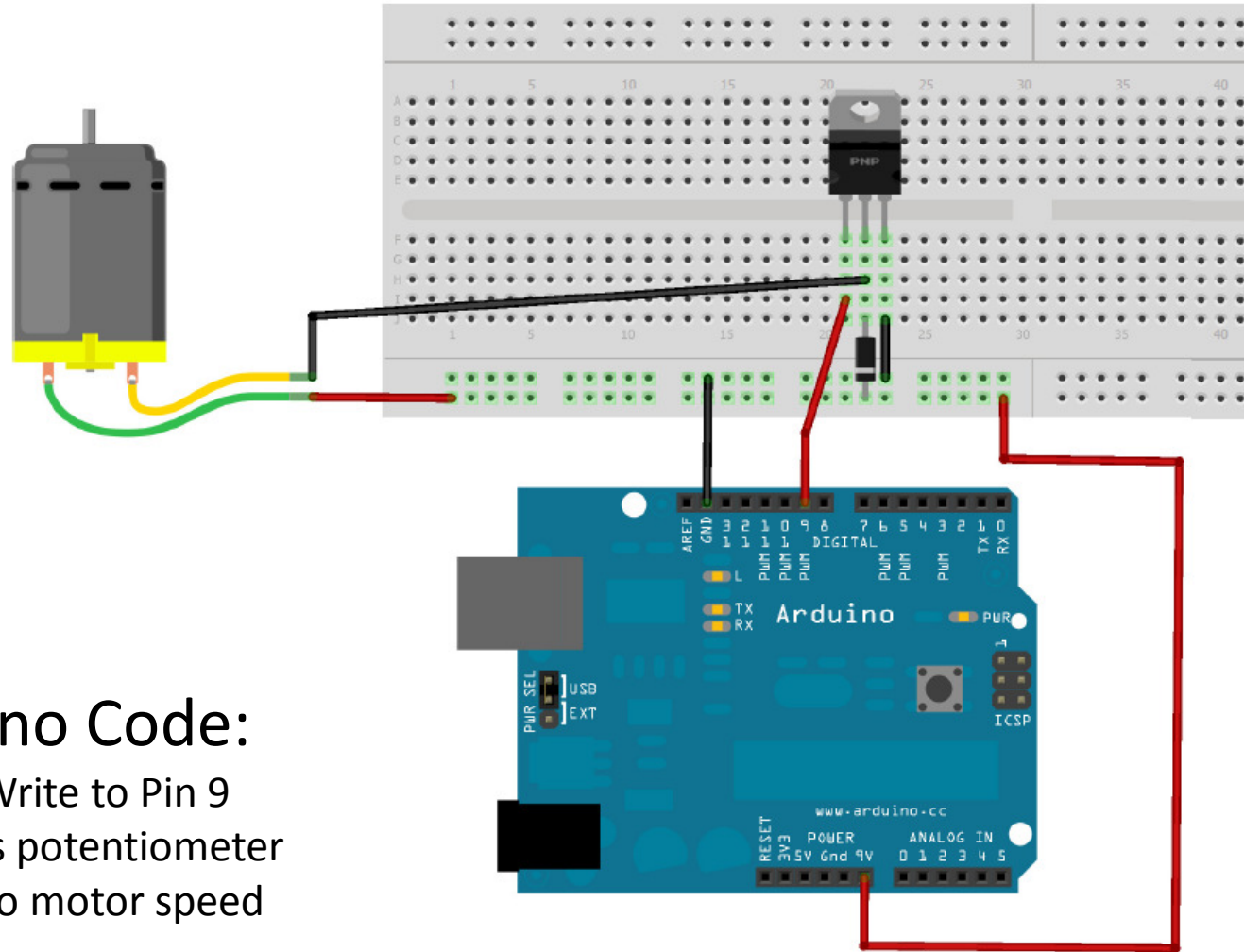


# Arduino code

```
/*
 * Motor Control
 *
 * A simple example of how to use the A3954 with an Arduino
 */
int brake = 9;
int enable = 10; // Must be a PWM pin
int phase = 11;
int mode = 12;
int mstatus;
void setup() // Set all of our pins as outputs
{ // Alternatively, we can address this
  pinMode(brake, OUTPUT); // bank of pins as a port, as shown
    at
  pinMode(enable, OUTPUT); // < http://www.arduino.cc/en/
  pinMode(phase, OUTPUT); // Reference/PortManipulation >
  pinMode(mode, OUTPUT); // This would be port B.
}
int forward(int mspeed, int mstat){
  if (mstat == -1){return 1;} // Do not change directions w/o a
    brake
  digitalWrite(brake, HIGH); // Disable the dynamic brake
  analogWrite(enable, mspeed); // Send a PWM signal to enable
    the motor
  digitalWrite(phase, HIGH); // Drive the motor forward
  digitalWrite(mode, HIGH); // Fast decay mode
  return 1;
}
```

```
int reverse(int mspeed, int mstat){
  if (mstat == 1){return -1;} // Do not change directions w/o a brake
  digitalWrite(brake, HIGH); // Disable the dynamic brake
  analogWrite(enable, mspeed); // Send a PWM signal to enable
    the motor
  digitalWrite(phase, LOW); // Drive the motor in REVERSE
  digitalWrite(mode, HIGH); // Fast decay mode
  return -1;
}
int stop(){
  digitalWrite(brake, LOW); // Turn on the dynamic brake
  digitalWrite(mode, LOW); // Use Phase pin for current protection
  // The remaining pins are irrelevant
  return 0;
}
void loop() // run over and over again
{
  for (int i = 0; i <= 100; i++){ // Drive the motor forward from 0-
    100%
    mstatus = forward(i, mstatus); // call the forward function
    delay(50); // wait between cycles to make this last 5 sec
    mstatus = stop(); // sink both motor leads low, making it stop
    delay(1000); // wait one second
    for (int i = 100; i >= 0; i--){ // Drive the motor backwards from 100-
      0%
      mstatus = forward(i, mstatus); // call the reverse function
      delay(50); // wait between cycles to make this last 5 sec
    }
    mstatus = stop(); // sink both motor leads low, making it stop
    delay(1000); // wait one second
  }
}
```

# Transformer Mosfet (IRF520)

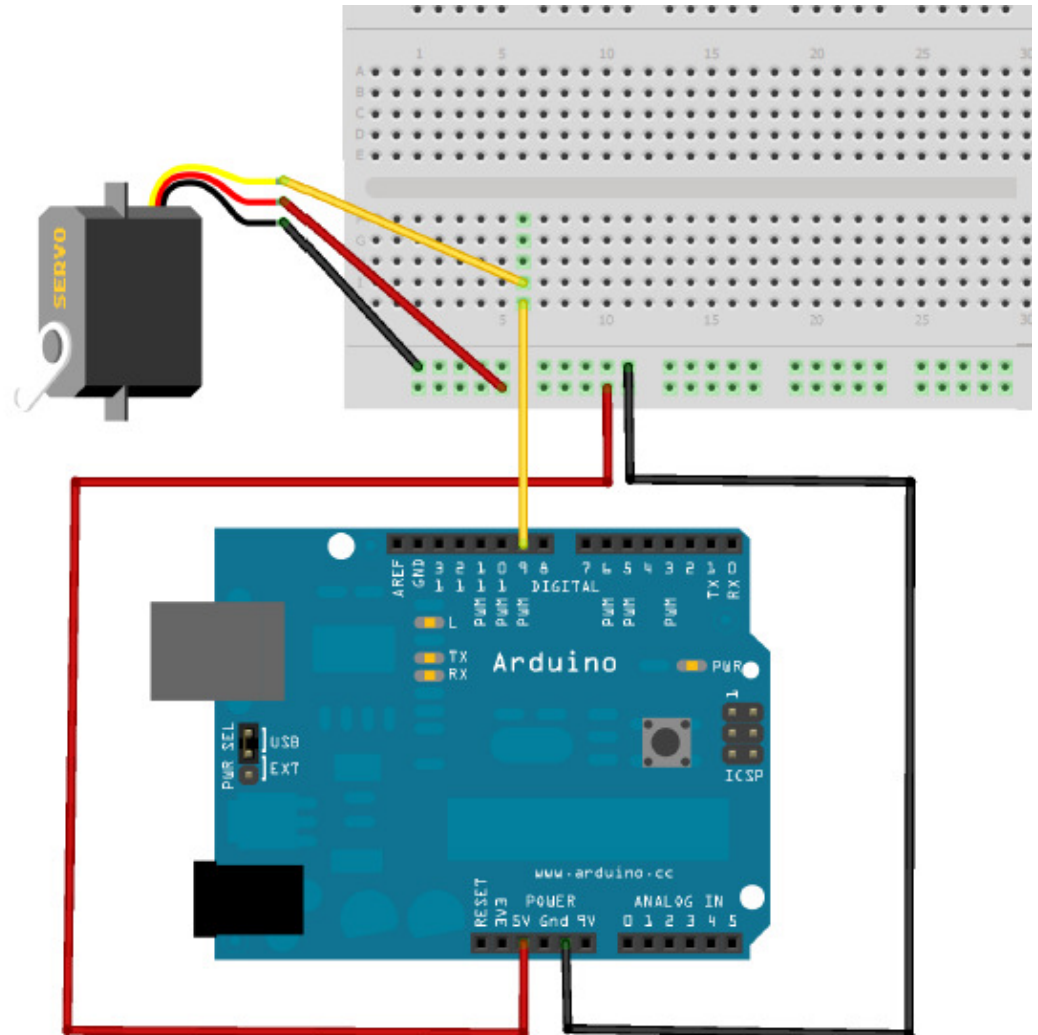


## Arduino Code:

Analog Write to Pin 9  
Works as potentiometer  
control to motor speed

# Servo Motor

- File > Examples > Servo> Sweep

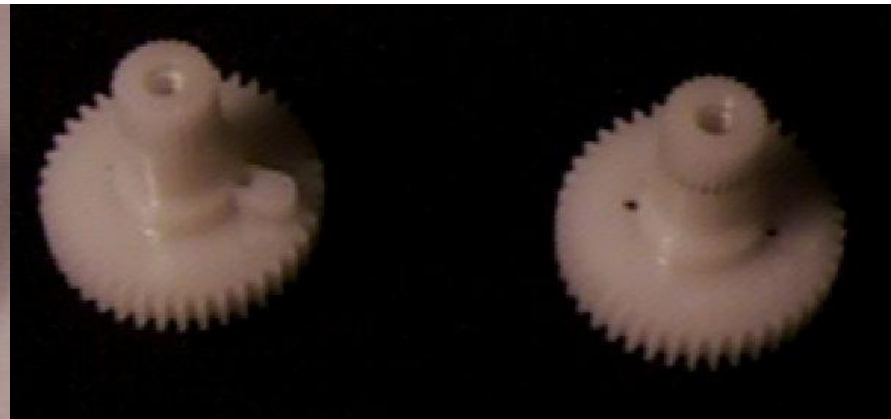
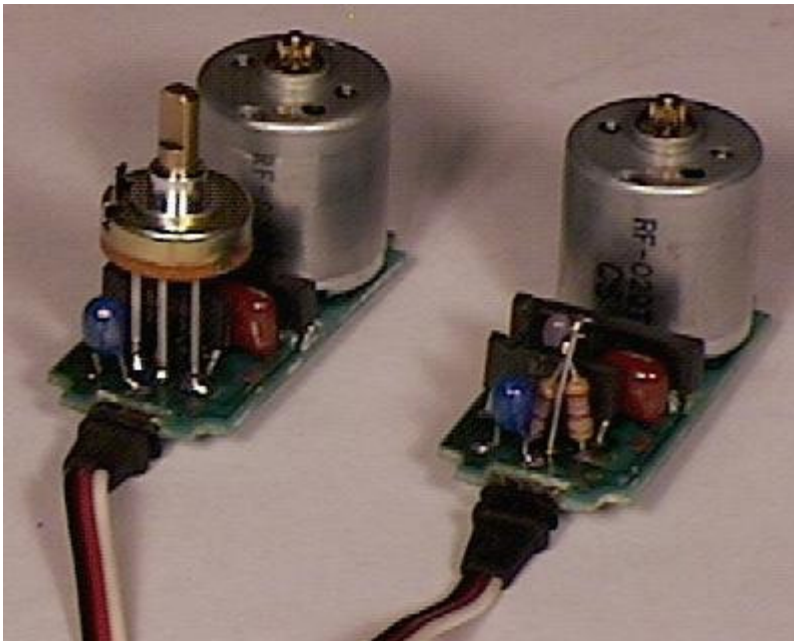




# Servo Motor (old version code)

```
/* Servo Example
 * Code based on AnalogInput by DojoDave <http://www.0j0.org>
 *
 */
int potPin = 2; // select the input pin for the potentiometer
int val = 0; // variable to store the value coming from the sensor
int servoPin = 7; // Control pin for servo motor
int pulseWidth = 0; // Amount to pulse the servo
long lastPulse = 0; // the time in millisecs of the last pulse
int refreshTime = 20; // the time in millisecs needed in between pulses
int minPulse = 600; // minimum pulse width
void setup() {
  pinMode(potPin, INPUT);
  pinMode(servoPin, OUTPUT);
}
void loop() {
  val = analogRead(potPin); // read the value from the sensor
  val = val * .175; // convert 10 bit val to derees (1024/180 = .175 )
  pulseWidth = (val * 10) + minPulse; // convert angle to microseconds
  updateServo(); // update servo position
  delay(10);
}
void updateServo() {
  // pulse the servo again if rhe refresh time (20 ms) have passed:
  if (millis() - lastPulse >= refreshTime) {
    digitalWrite(servoPin, HIGH); // Turn the motor on
    delayMicroseconds(pulseWidth); // Length of the pulse sets the motor position
    digitalWrite(servoPin, LOW); // Turn the motor off
    lastPulse = millis(); // save the time of the last pulse
  }
}
```

# Hacking Servo



# Stepper motors

<http://arduino.cc/en/Reference/StepperUnipolarCircuit>