

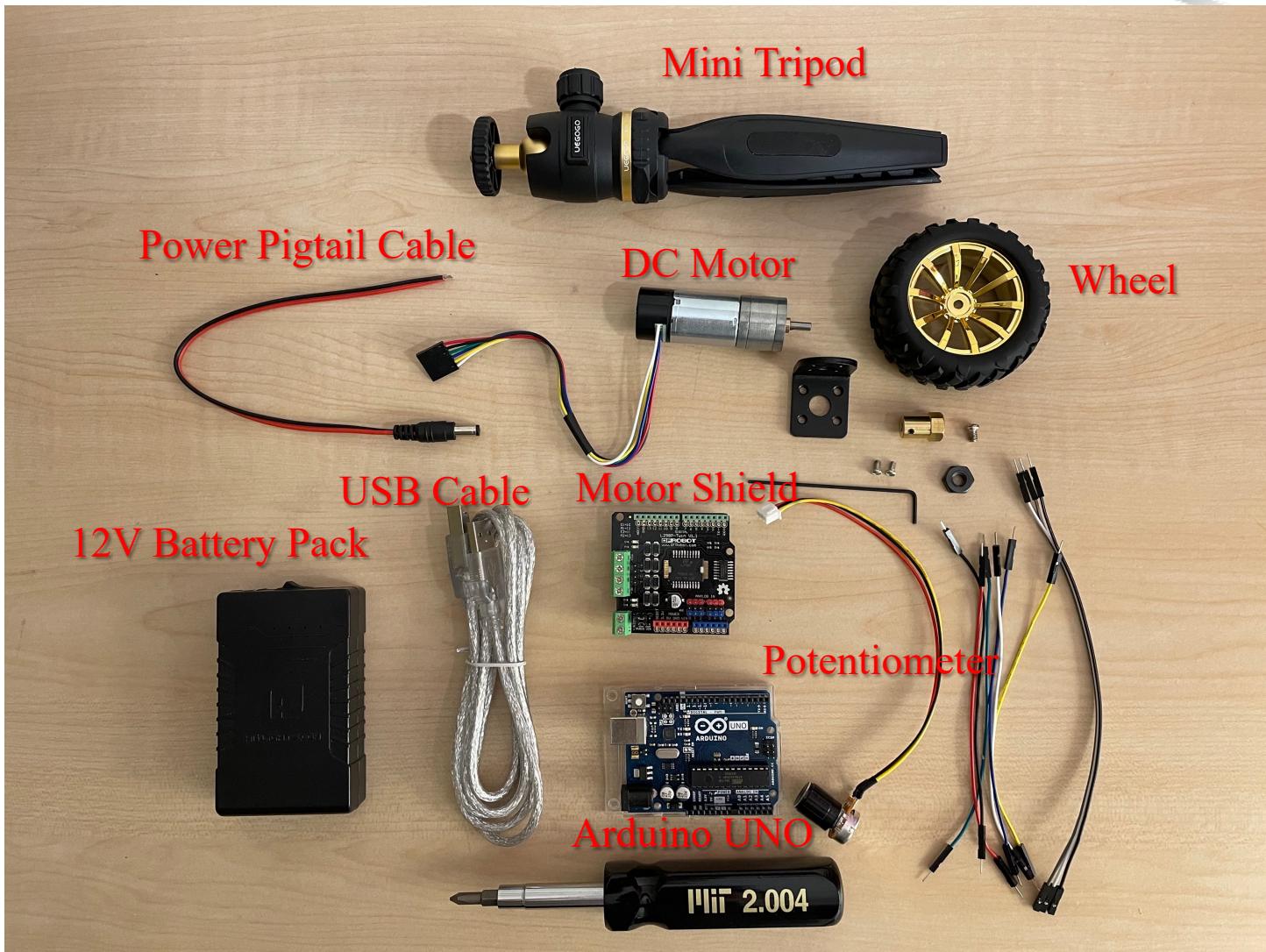
2.004 Dynamics and Control II

Lab Kit: DC Motor Plant Assembly Instructions & Arduino Intro

Fall, 2021

- The lab kit contains parts and components needed to perform lab experiments related to dynamic modeling and feedback controls.
- You will need to assemble the hardware, perform sensor calibration, estimate the model, and program an Arduino UNO microcontroller to implement various controllers in subsequent labs.
- Lab 2 will focus on characterizing the DC motor plant.
- Prelab 2:
 - Install Arduino software.

DC Motor Kit



Assemble the DC Motor Plant

1. Attach the motor bracket to the mini tripod using the hex nut. (Note: you may need to remove the rubber pad on the tripod to tighten the nut.)

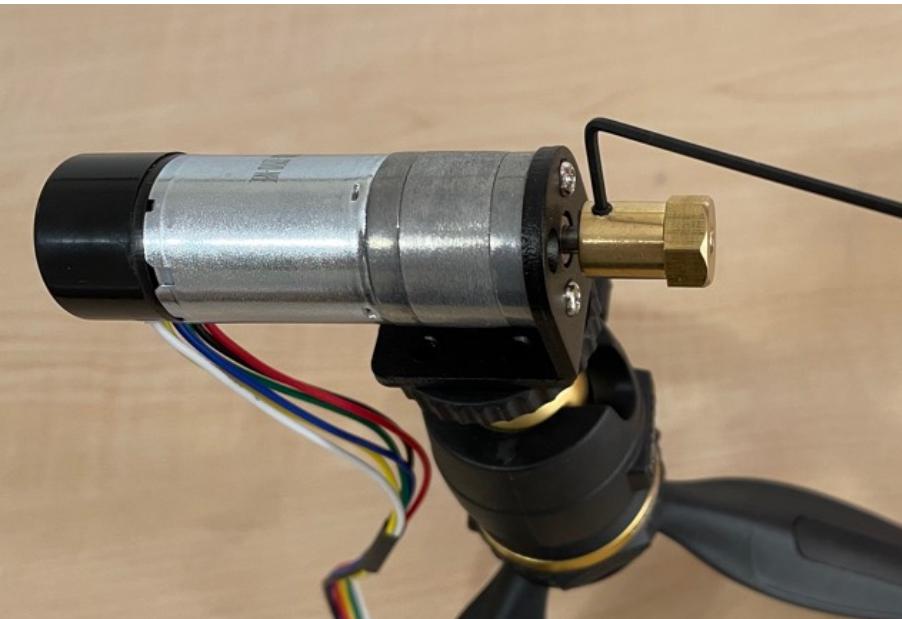


2. Mount the motor to the bracket. Make sure to tighten the screws.



Assemble the DC Motor Plant

3. Connect the shaft coupler to the motor output shaft with two setscrews. Use the hex key to tighten the setscrews. Leave about 1 to 2mm gap between the motor and the coupler to avoid rubbing.

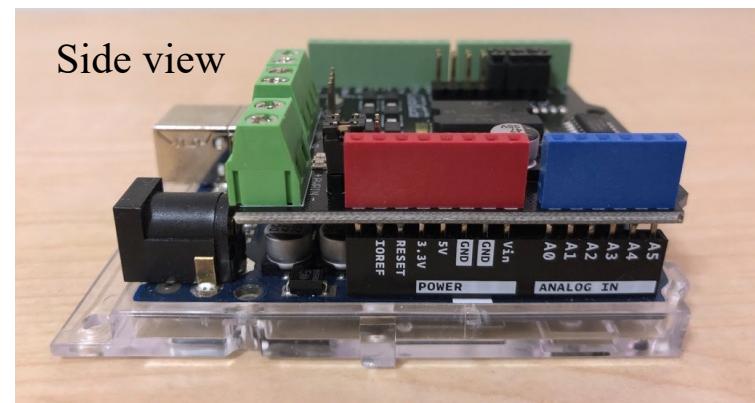
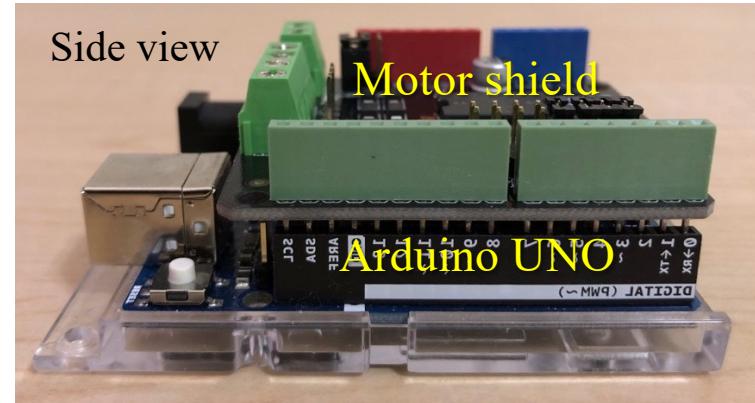
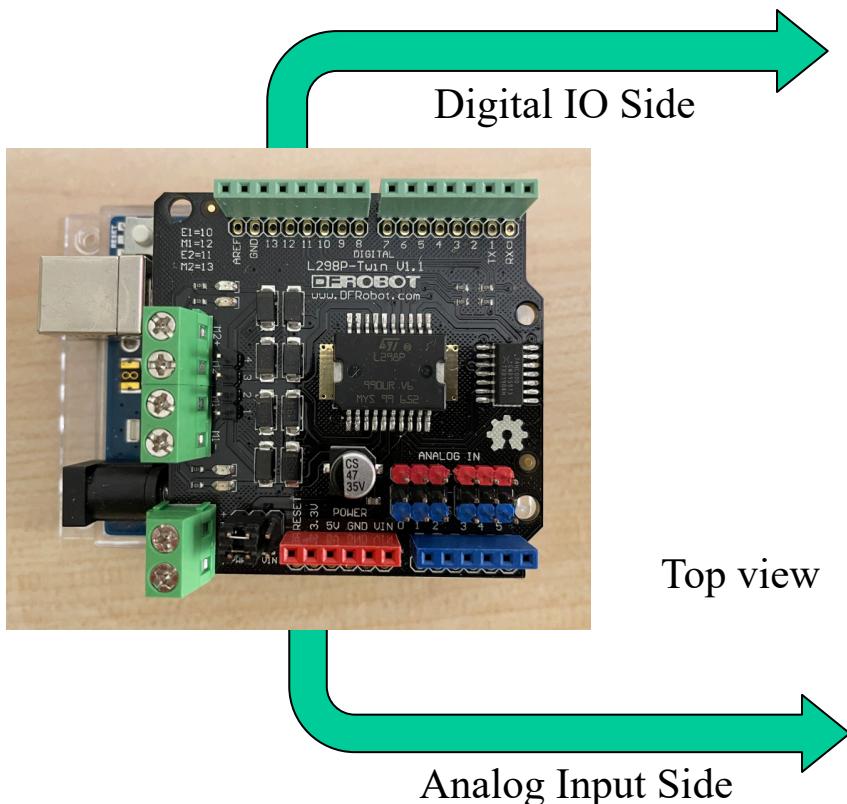


4. Mount the wheel to the coupler. Tighten the wheel with the screw. Attach a sticker to the wheel as shown.



Arduino UNO and Motor Shield

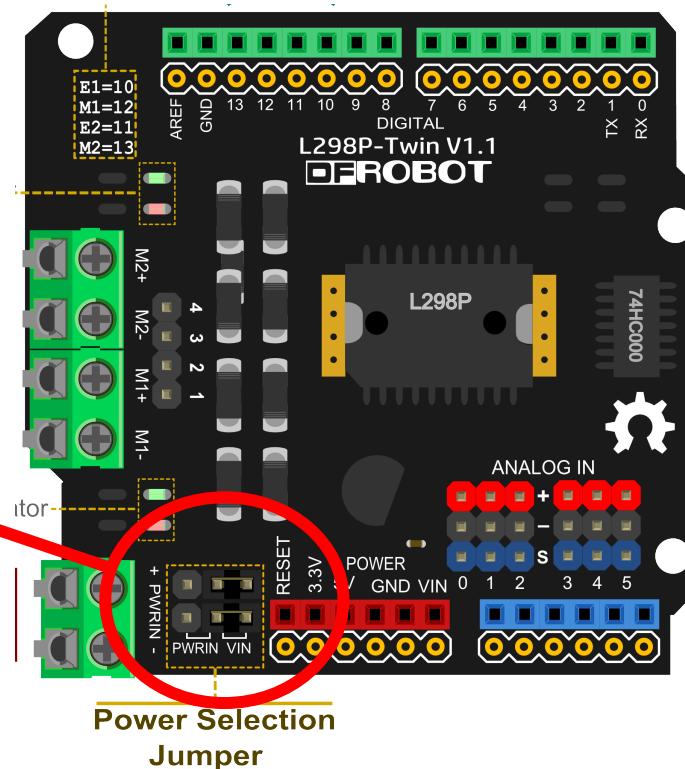
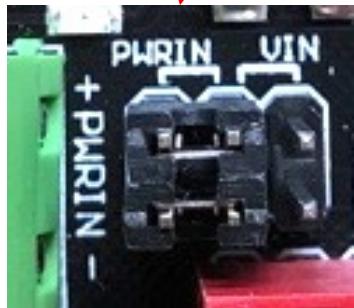
5. Attach the motor shield to the top of the Arduino board. Make sure pins are properly aligned.



Change Power Jumper Location

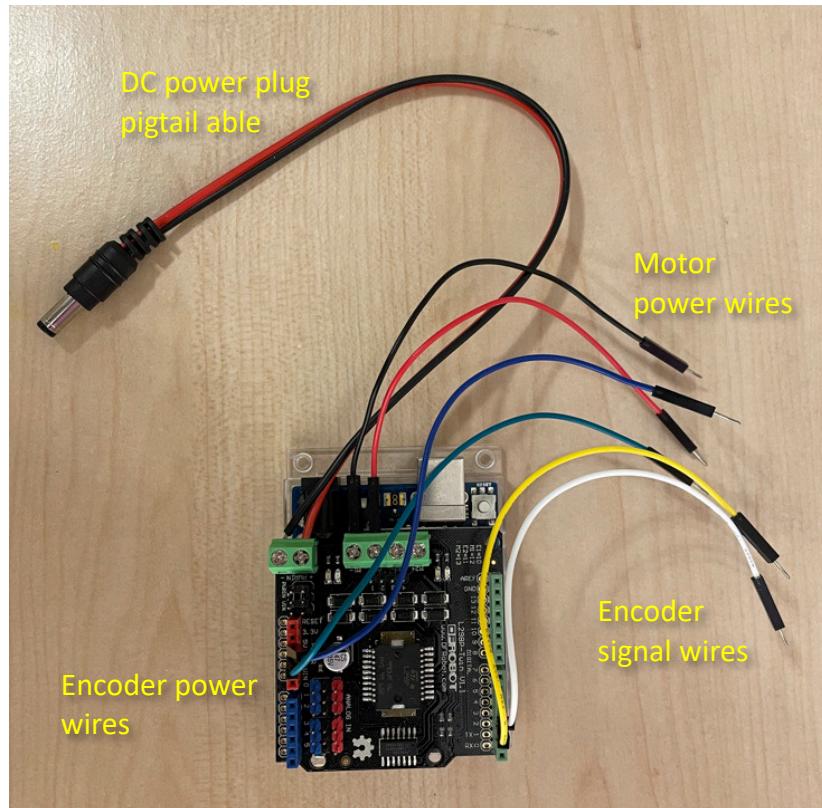


6. Change both jumper blocks to “PWRIN”



Motor Wiring

7. Attach jumper wires based on diagram shown on the next page.

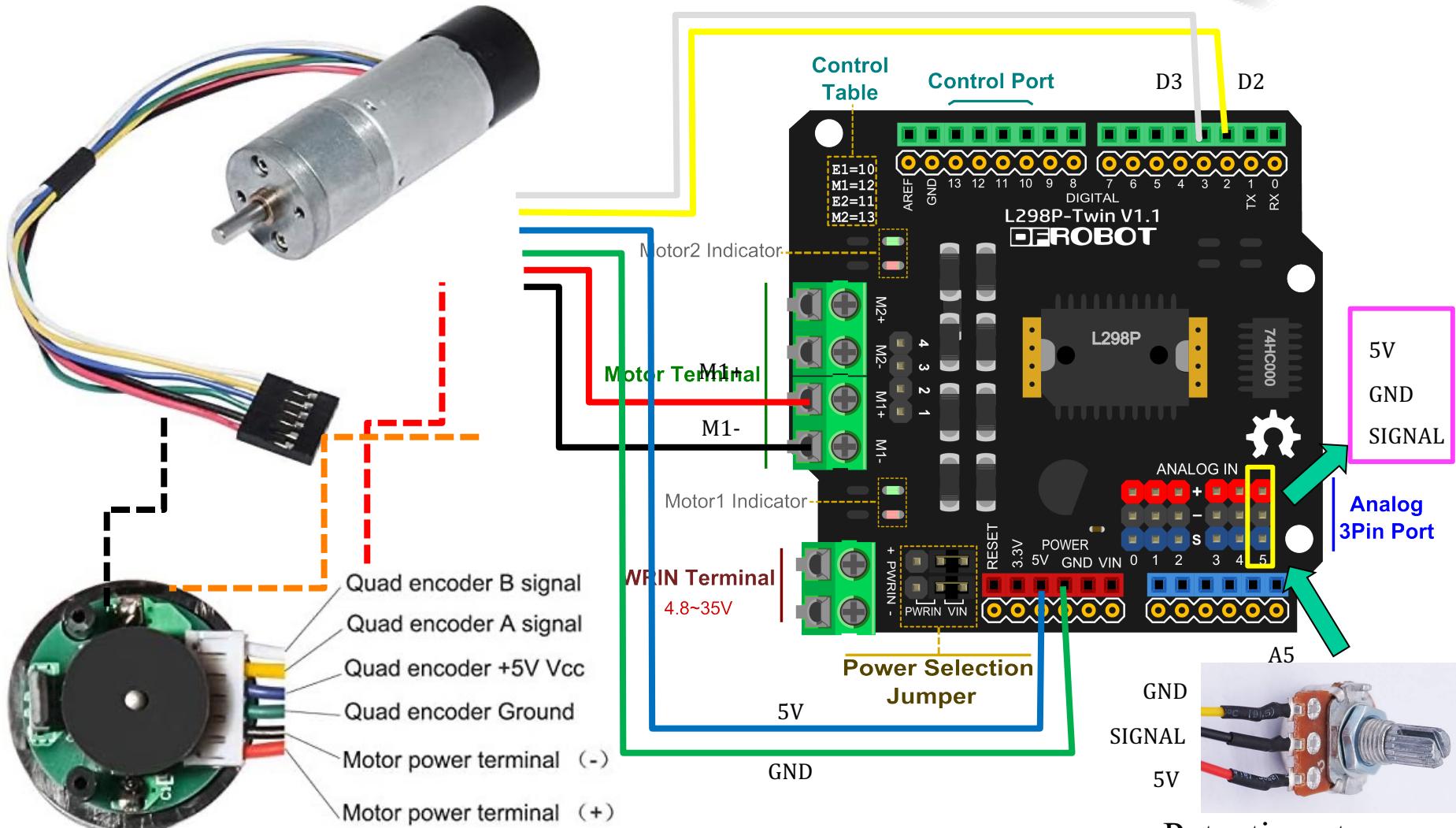


8. Wires are color-coded so make sure they match.

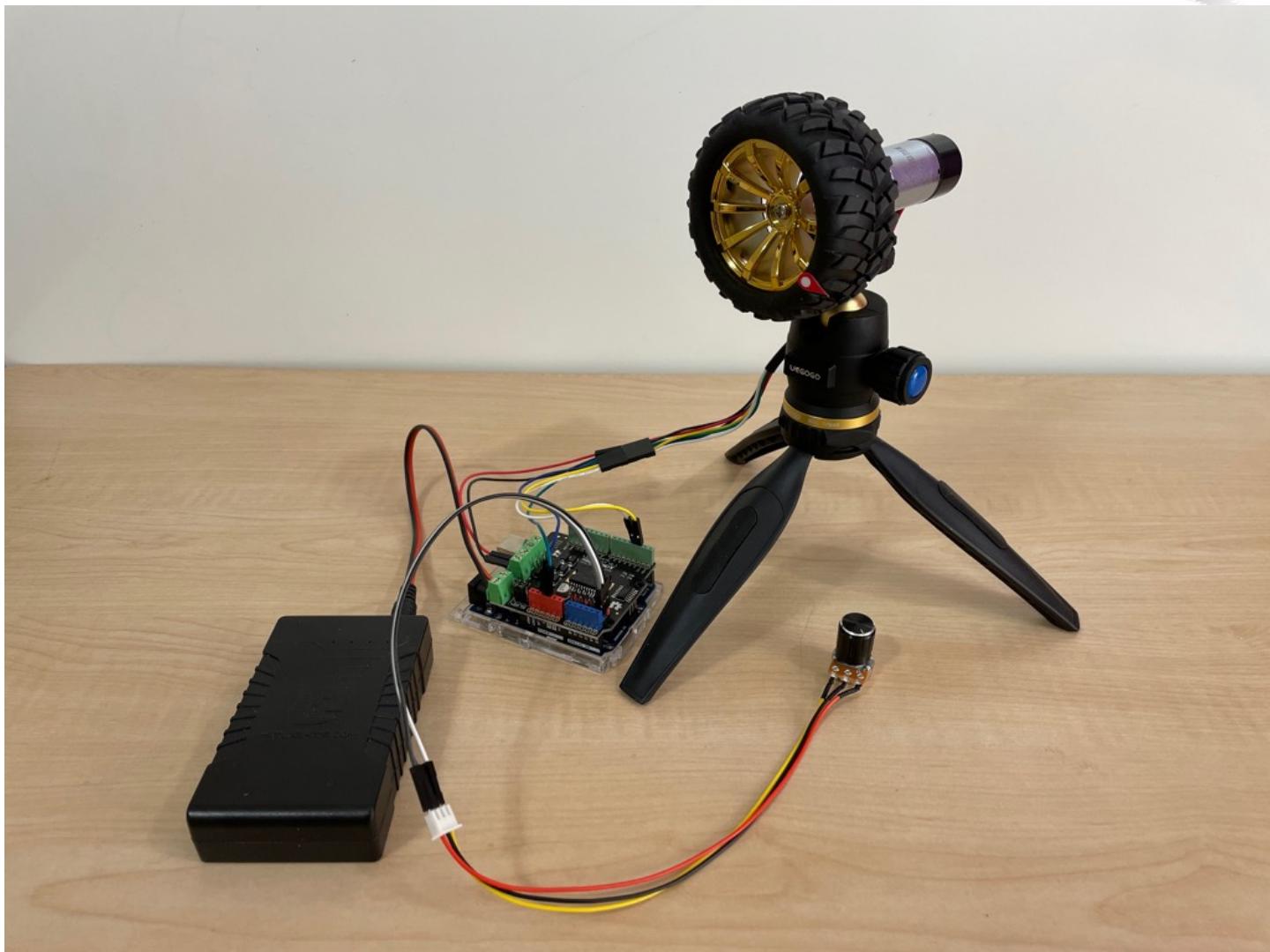


To motor shield

Motor Power and Encoder Connection

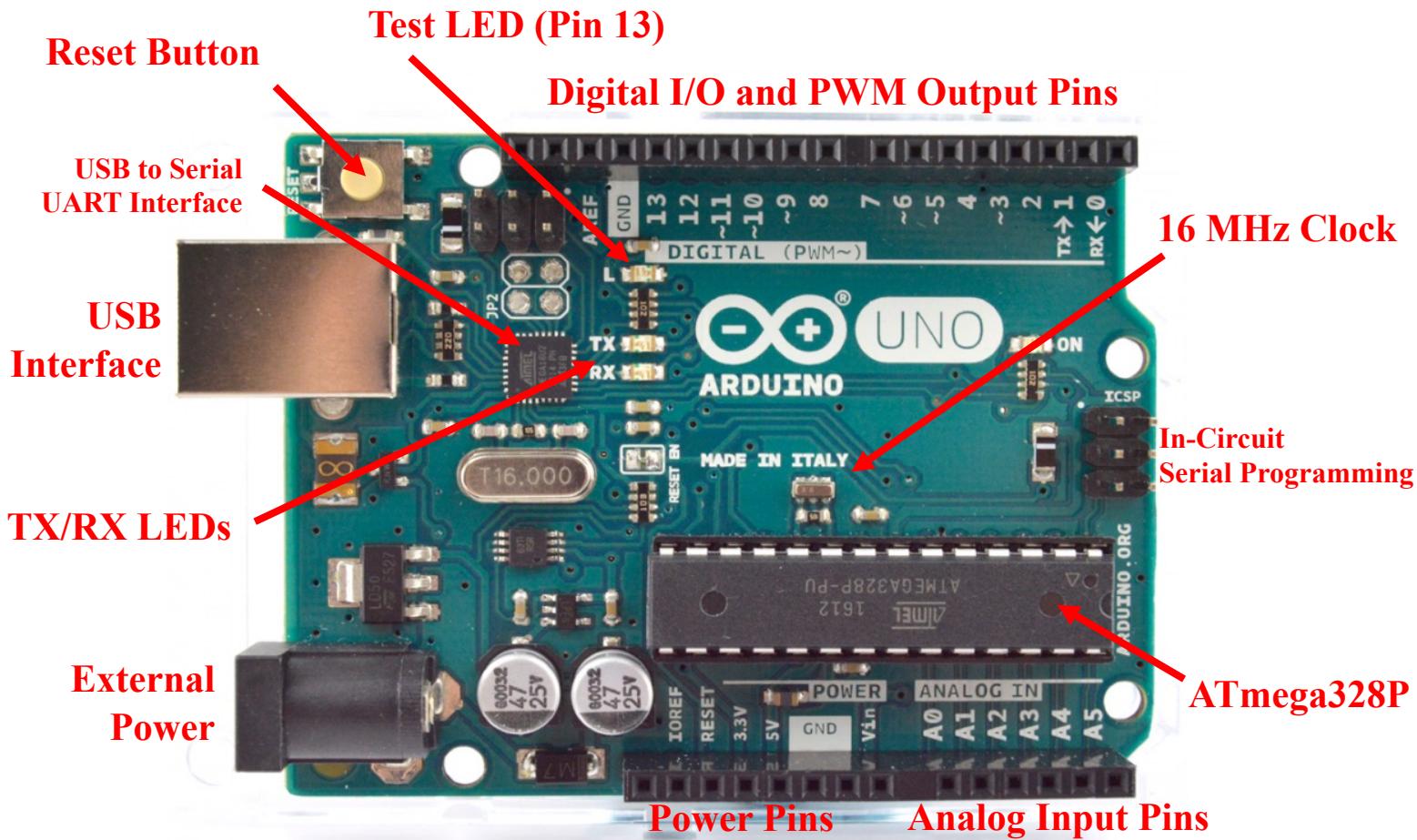


Fully Assembled DC Motor Setup



ARDUINO BACKGROUND

Arduino Uno Board



Setup Arduino for Programming



- Connect the Arduino to the computer via the USB cable.
- You must tell the software which of the many Arduino boards you are using. On the top menu bar, go to **Tools** → **Board** and select the appropriate board (e.g., “Arduino Uno”). Normally the type of board is automatically detected by the Arduino software.
- You must tell the software which “COM port” on the computer you are using to communicate with the Arduino. Go to **Tools** → **Serial Port** and make sure that the correct COM port is selected.

Arduino Programming Environment



- Open source.
- Simplified C++ like development environment that is easy to program and to upload the code.
- Several examples are included that demonstrate various I/O capabilities.
- Built-in libraries that simplify data I/O tasks.
- Large user community.

The screenshot shows the Arduino IDE interface. The title bar reads "Arduino - 0015". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar contains icons for Open, Save, Print, and other functions. The central area is labeled "Menu Buttons" in red text, containing the code for the Blink sketch. A large green watermark reading "Sketch!" is overlaid on the code area. The status bar at the bottom displays "Done uploading." and "Binary sketch size: 976 bytes (of a 30720 byte maximum)".

```
/*
 * Blink
 *
 * The basic Arduino example. Turns on an LED on for one second,
 * then off for one second, and so on... We use pin 13 because,
 * depending on your Arduino board, it has either a built-in LED
 * or a built-in resistor so that you need only an LED.
 *
 * http://www.arduino.cc/en/Tutorial/Blink
 */

int ledPin = 13; // LED connected to digital pin 13

void setup() // run once, when the sketch starts
{
    pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop() // run over and over again
{
    digitalWrite(ledPin, HIGH); // turns the LED on
    delay(1000);
    digitalWrite(ledPin, LOW); // turns the LED off
    delay(1000);
}
```

Done uploading.
Binary sketch size: 976 bytes (of a 30720 byte maximum)

2

Sketch!

Code Area

Status Area

Arduino Code Structure



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 0022". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with various icons. The code editor contains the "Blink" sketch. The code is as follows:

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/

void setup() {
    // initialize the digital pin as an output.
    // Pin 13 has an LED connected on most Arduino boards:
    pinMode(13, OUTPUT);
}

void loop() {
    digitalWrite(13, HIGH);      // set the LED on
    delay(1000);                // wait for a second
    digitalWrite(13, LOW);       // set the LED off
    delay(1000);                // wait for a second
}
```

Comments

Initializes variables

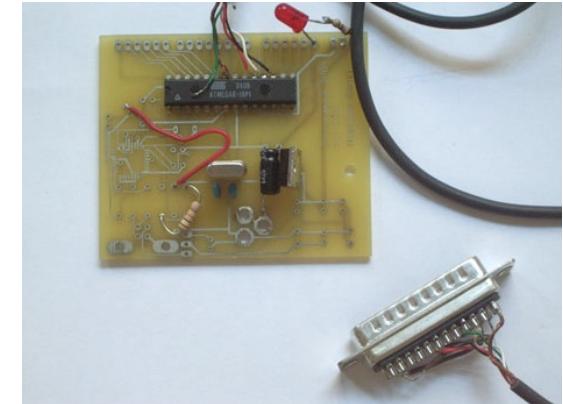
Executes commands and functions sequentially in the loop

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

History of Arduino

<http://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino>

- Named after a local bar (or a medieval king) in northern Italy...
- First release in 2005.
- One of the most popular microcontroller platforms around the world.
- The software development environment is based on a designer-friendly programming language called Processing developed by Casey Reas and Benjamin Fry at the MIT Media Lab in 2001.



Why Arduino

- Popular
- Open source
- Low cost
- Large user community
- Easy to use development environment
- Based on ATTEL's Atmega microcontroller chips
(<http://www.atmel.com/products/microcontrollers/avr/megaavr.aspx>)
- Newer boards have the ARM Cortex-M3 chips



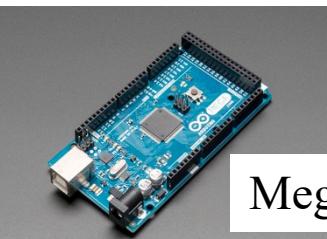
\$4



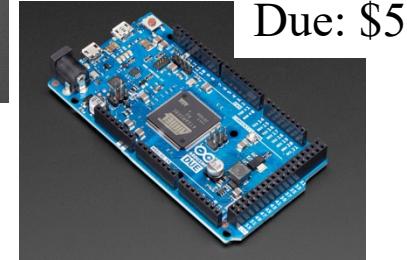
Nano: \$20



Uno: \$25



Mega: \$45



Due: \$50



Yun: \$57



101: \$30

<http://arduino.cc/en/Main/Products>