

Name/NetID:

Teammate:

# Module: Driving the Vehicle using the Arduino

---

## Module Outline

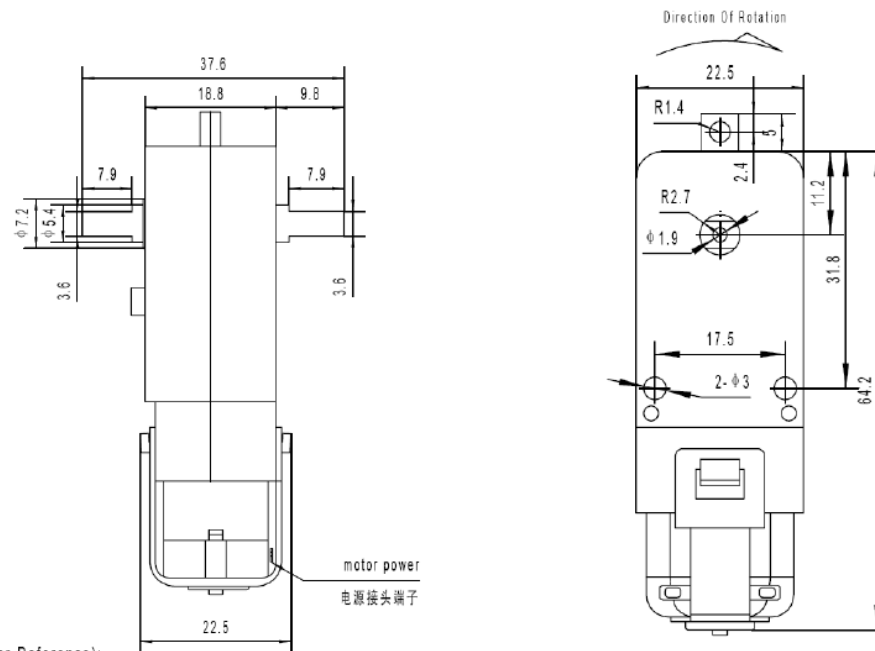
The vehicle connected to a battery with some resistors inserted to keep the velocity slow enough that the car does not crash is the first stage in building a fully autonomous car. As the design stands the vehicle can move in only one way, depending on how much resistance is inserted in series with the motor. To change how it moves you must manually change the arrangement and values of the resistors. Not very autonomous. The next step is to include a control circuit that can turn the motors on and off – for this task the Arduino is perfect. The digital I/O pins will be used to turn the motors on and off and control the speed of the motors as well. Only one problem needs to be overcome – an interface is needed between the relatively power hungry motors and the very small current sourcing capacity of most digital circuitry, the Arduino included.

# Procedures

The motors, as you will find out in a later lab, demand quite a lot of current. The current rating for the pins on the Arduino board – 40mA – suggests that the board cannot handle the current needed by the motor without damage. The datasheet for the motors on the Magician Chassis, shown in the figure below, indicates that when the motor is running freely at a voltage  $\sim 5V$  it will draw  $\sim 190mA$  and at most 250mA which is much greater than 40mA. You could try to limit the speed of the motors with resistors but the behavior will be unpredictable and there is a much better way.

This scenario is very common – how to drive a device or circuit that have high power needs but the control circuitry is not capable of providing that power. One solution is to use a transistor circuit as an interface between the two.

## DG01D-A130GEARMOTOR



GEAR BOX Specification(For Reference):

Suggested Voltage: 4.5V DC

- 1、 No Load Speed:  $90 \pm 10$ rpm
- 2、 No Load Current: 190mA (250mA MAX)
- 3、 Allowable max. Torque: 800gf.cm min

## Simple Test of the Transistor

The following instructions, along with the figures on the following page, are an intermediate stage in building the full motor drive circuitry. By building this circuit first you will be sure that the connections to the transistor are correct before connecting it to the Arduino.

- ✓ Following the directions included with the kit attach the top of the vehicle's chassis (if you have not done so already). For this lab you will be using both the Arduino board and the protoboard. There is a flat black holder that the SIK guide suggests that you attach the Arduino on one side and the protoboard on the other. The Arduino is screwed on and the protoboard has a sticky bottom revealed by peeling off the protective sheet. A less permanent option is to use the Velcro available in the lab. You can attach them any way you like.
- ✓ Insert the transistors available in the lab into the protoboard as shown in the figure below.
- ✓ Connect the transistor pins labeled **E** to ground labeled – on the protoboard as shown in the figure below.
- ✓ Connect the motor terminals of each motor with the positive terminal of the motors to the + terminal of the power strip and the negative terminal to the transistor labeled **C**.
- ✓ Put the vehicle on a stand – the small wooden blocks or your partner holding it will work fine.
- ✓ Connect the power supply to the power strips on the protoboard just as you did last week setting the voltage to 5V keeping it in the OUTPUT OFF mode until you are finished wiring the transistors.

NOTE: the instructions for constructing and attaching the protoboard, Arduino and the Magiacian's Chassis are suggestions so if you want to not attach the top, or not use the black holder feel free to improvise.

ANOTHER NOTE: The figures will probably not print out very well since everything is so red – you can get these from the website in Module 4.



Notes:

---

**Question 1:** Connect the pin labeled **B** on both transistors through a  $330\Omega$  resistor to the positive terminal on the power strip. (refer to the physical diagram for help) Turn the power supply on making sure it is set to 5V. Is the motor turning?

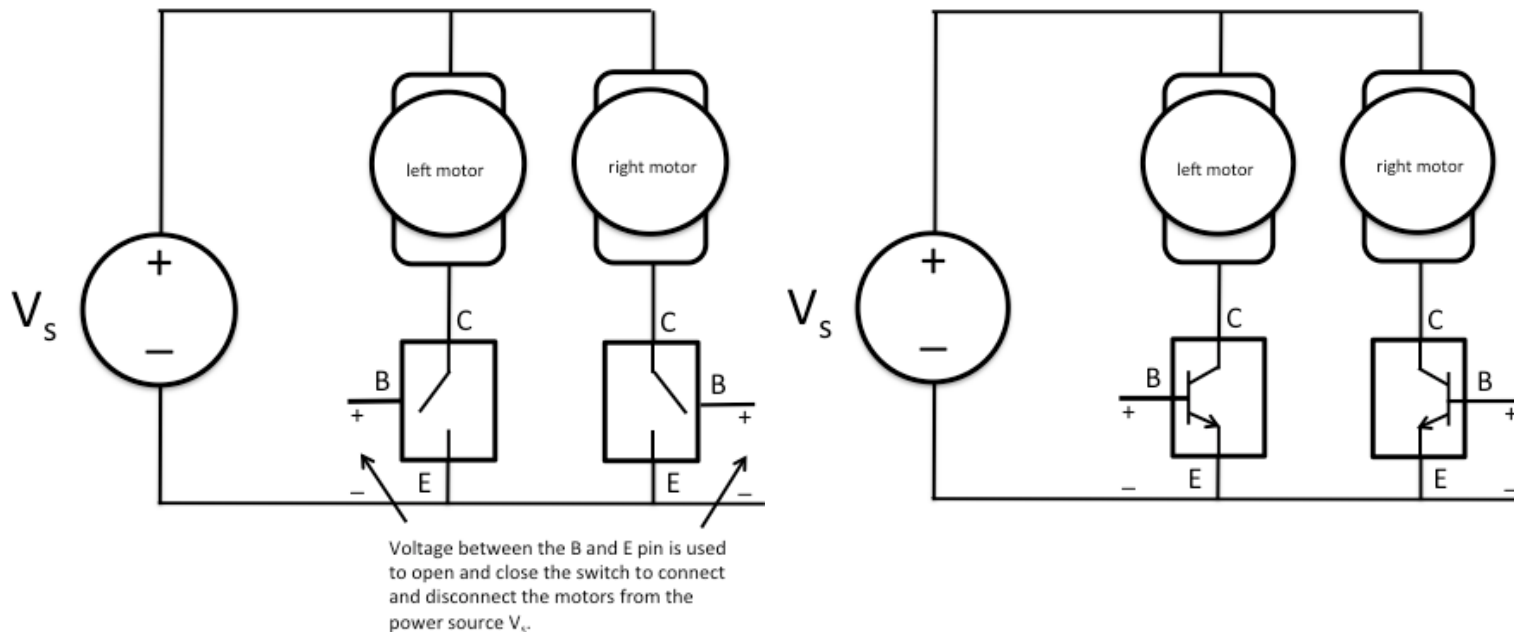
**Question 2:** If the answer to Question 1 was *no* then there is a wiring problem. Fix it – Is the motor turning?

Repeat until the answer is YES.

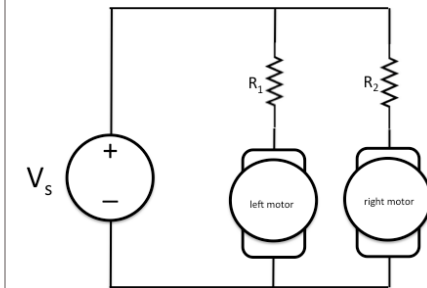
## What is a Transistor

In Module 2 you put together the chassis and connected the motors to first the power supply and then a battery pack to get the vehicle body to start moving on a table. To slow down the speed of the vehicle you inserted resistors in series with the power supply. Because the motors demand  $\sim 150\text{mA}$  of current it was difficult to find the right combination. Too little resistance and the resistor burns – too much and the motors will not be able to overcome the internal friction and will not turn.

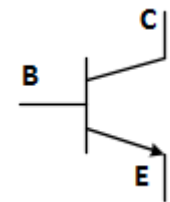
A special device is needed between the motor power source and the motors that acts as an electronic switch. The transistor is that special device – you will study it in class and characterize it in a later lab. For our purposes we are using the transistor as an electronic switch. There are many different types of transistors. The ones you will be using are Bipolar Junction Transistors (BJT) – a 2N5192 if you are looking for a datasheet. The pins of a BJT are named the Collector (**C**), the Emitter (**E**), and the Base (**B**). The desired current flows along a path between the collector and emitter. The signal that controls this current is applied to the base (see figure below).



Notes:



Circuit you built in Module 2



Circuit symbol for an NPN BJT – you will learn more about these devices and what NPN means during lecture and in future labs.

NOTE: this next portion is *optional* but interesting and emphasizes that the transistor is a device that can control the flow of current through a power hungry device connected to beefy power source by using a signal that can be driven by a completely different power source (sharing the same common or ground connection).

- ✓ (optional) Hook up the power supply from your neighbors' bench to the other power strip on the protoboard and connect the grounds on both sides together.
- ✓ (optional) Now connect the pin labeled **B** on both of the transistors through the  $330\Omega$  resistor to the positive terminal of the other power strip.
- ✓ (optional) Set the voltage of your neighbors' power supply to 5V.

**Question 3:** (Optional) Vary your power supply and describe the behavior of the current from each supply. Your supply is driving the motor while your neighbors' is controlling the state of the transistor keeping it in the ON state.

## Using the Arduino to Drive the Motors

Now you have what you need to interface the vehicle's motors to the Arduino – that is the first step to making your design autonomous. Although you will only need to connect 4 wires between the protoboard and the Arduino this step can be a little confusing because there are several different ways to do this. Most of the confusion arises because there are several ways to configure the power to the motors and to the board. So let's do the simplest part first – attaching the signals that will control the motors.

- ✓ Disconnect the power supply from the board.
- ✓ Connect the wires that currently connect the base of the transistors to the positive terminal of the power supply through the  $330\Omega$  resistors to the digital I/O pins 9 and 11 on the Arduino. If you look at the board and the labeling of the digital I/O pins you will see that some but not all of them have a tilde  $\sim$  beside the number. These pins are the only pins that can output the time-varying signal programmed using the `analogWrite` function. Both pins 9 and 11 have this capability.

All that is left to do is to connect the protoboard to the power source used to run the motors. But you also need to power the board. From Module 3 you know that when you hook the Arduino up to the computer it draws power through the USB port. This is how you have to upload programs to the board and you can use the computer to power the external circuitry you build as you are debugging and working on a design. But for true autonomy the vehicle needs to be powered using a battery (and the USB current ratings are lower). The battery will be plugged into the black connector – called a barrel connector. The circuitry on the board can detect when the computer is connected to a power source and can distinguish between a battery pack and the USB connection. The board also has circuitry to give priority to the barrel connection even if both sources are connected. There is another way but it must be used with care and has no advantages for your design.

The Arduino can run on voltages ranging from 6-20V. Several types of batteries that are available in the lab – 9V batteries that can be attached with the snap on connector in your kit, a 7.2V rechargeable battery pack, and 4 AA batteries that fit in the cradle that came with your chassis kit. Your TAs will give you these options and explain the benefits of each choice. The method of connecting the protoboard that is given here will allow you to never have to rewire the motor drive circuitry for different any of the methods of powering the board.





Notes:

**Question 4:** Plug 3 different battery packs into the barrel connector – the cradle that holds 4 AA batteries, a 9V with the snap-on connector included in your kit, and the 7.2V rechargeable battery pack. Make a qualitative assessment of how the motor responds to each. Use the 7.2V battery as the comparison and fill in the table below. How fast the motors turned as compared to the 7.2V pack and compare the force needed to stop the motors turning by gently grabbing both wheels as compared to the 7.2V pack.

Battery voltage	Motor speed	Force
4 AA ~6V		
7.2V		
9V		

## Speed Control

Using the transistor you can play around with controlling the vehicle using the digital I/O pins on the Arduino. The most useful parameter to control is the speed of the motors. In Module 2 you played with slowing the vehicle down by adding resistors in series with motors and the supply. While using the Arduino this method is difficult, wastes power, and is very imprecise. Luckily, there is a very simple method. You have already used it to vary the intensity of an LED using the *analogWrite* function to turn the digital signal on and off very quickly. By changing the ratio of the amount of time that the signals is on compared to the amount of time it is off determines the speed of the motor. You will learn more about this method called Pulse Width Modulation (PWM) in a later lab.

- ✓ Connect the Arduino to the computer with the red USB cable.
- ✓ Upload a similar code from Module 2 to the board. Note the addition of the control of Pin 11.

```
void setup() {  
  // put your setup code here, to run once:  
  
  //declare both Pin 9 and Pin 11 to be outputs  
  
  pinMode(9,OUTPUT);  
  pinMode(11,OUTPUT);  
  
}  
  
void loop() {  
  
  // put your main code here, to run repeatedly:  
  
  analogWrite(9,200);  
  analogWrite(11,200);  
  
}
```

**Question 5:** Change the value passed to the *analogWrite* function between 0 and 255. Describe how the motor speed changes.

Notes:

---

## Open loop Driving

Now you are ready to go for a test drive. Combining only the simple code statements that you have already learned you can have the car perform pre-programmed movements. By turning the motors on and off in different ways you can get the car to go straight, make turns, or spin in place.

**Question 6:** Using any combination of the *digitalWrite*, *delay*, and *analogWrite* functions play with navigating your car. Include the code and path the vehicle took in several experiments.