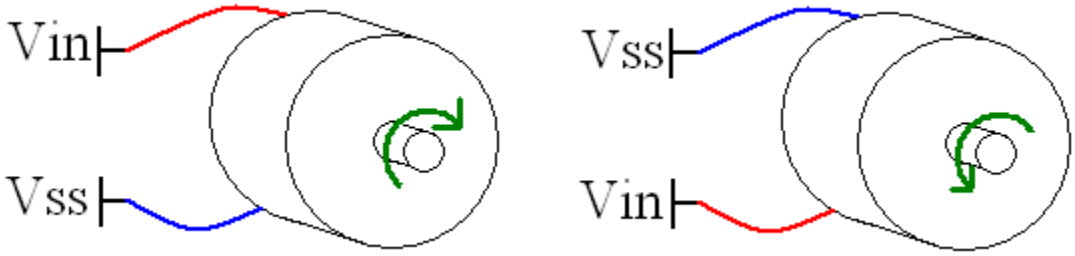
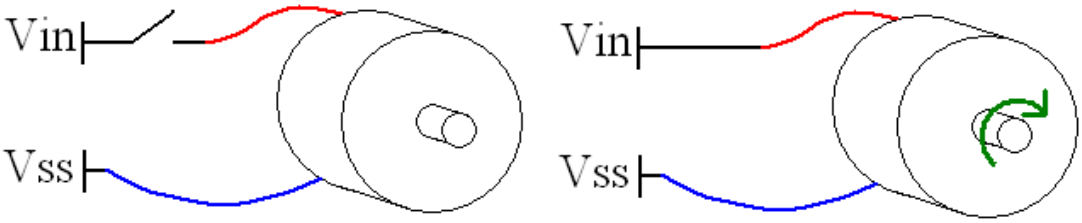
DC Motor Control

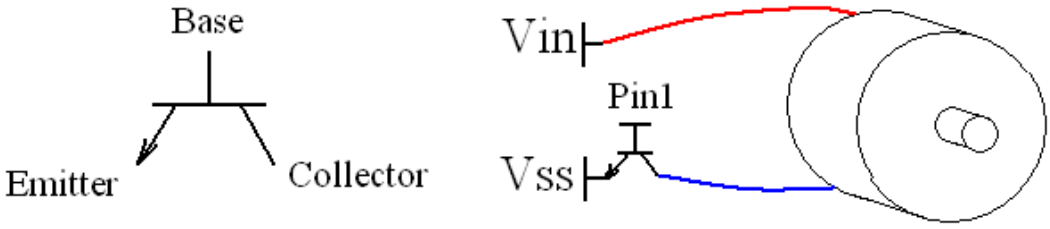
A very important part of robotics is actually having your robots move. This will give your robot a sense of being more that a pile of wires once it can physically interact with the environment. The main way to accomplish this is with motors. To use a motor you simply connect one terminal to voltage, and the other to ground. This will make the motor spin. If the terminals are revered, the motor will spin in the other direction.



The first way to control a motor is by a simple push button switch. By placing a switch in line with the motor, it works just like a light switch. By pushing the button, the circuit is closed which allows current to flow into the motor.



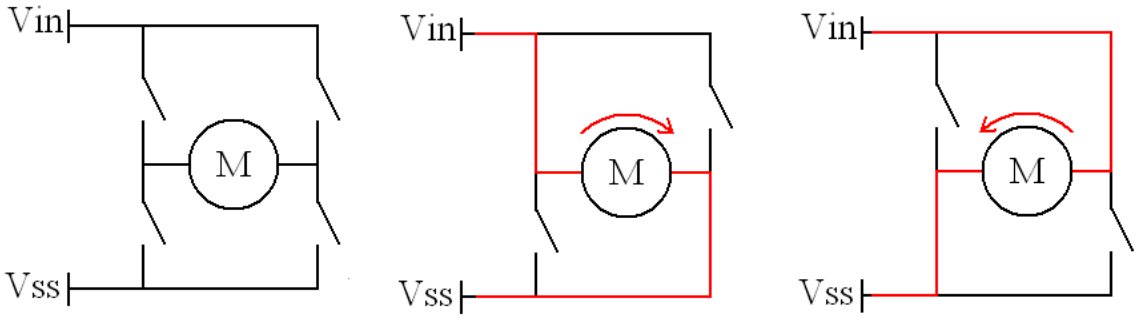
If we would like to control this motor using the BS1 instead of the button we can use an electronic device known as a transistor. A transistor is used just like a switch, but instead of physical input to close the circuit, we use a voltage to close the circuit. The basic schematic for a transistor is shown below. There are three legs on the transistor, Base, Collector, and Emitter. The Collector pin is connected to the part of the circuit you would like to control, the Emitter pin is connected to ground, and the Base pin is used as our switch. When 5V is applied to the Base, the circuit is closed, and when the Base is 0v, the circuit is open. So by connecting the base pin to pin 1 of our BS1, the emitter pin to ground, and the collector to one terminal of the motor, with the other terminal connected to power. We can now control our motor using our BS1. The following program will turn the motor on and off every half a second.



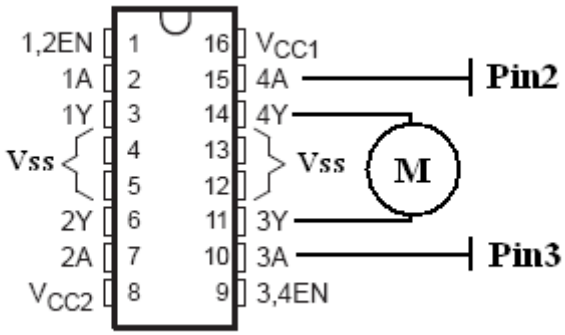
Initialize:
  LOW 1

Main:
  HIGH 1
  PAUSE 500
  LOW 1
  PAUSE 500
GOTO Main

If we would like to control the motor in both directions, we have to use what is called an H-Bridge. It is called an H-Bridge because the schematic looks like an H. See the schematics below. The switches shown below will be can simply be substituted with transistors to use the BS1 to control it. By closing the switches diagonally from each other we can control the motor in both directions.



Instead of recreating this circuit each time we would like to control a motor, we can use a pre-built circuit like the one shown below. This is a SN754410 chip. It can control up to 2 motors, using 2 input pins per motor. Pins 1, 9, & 16 are connected to 5v; pin 8 is connected to the voltage you would like to run the Motor. The 4, 5, 12, & 13 pins are connected to ground. The motors are connected to the Y pins, and the control pins from the BS1 are connected to the A pins. The 1 and 2 pins are tied together, and the 3 and 4 pins are ties together. So if we set 4A to ground, and 3A to 5v, power will run out of the 3Y pin and the 4Y pin will complete the circuit. By switch the 3A and 4A pins, the direction of the motor will change. The program below will show this.

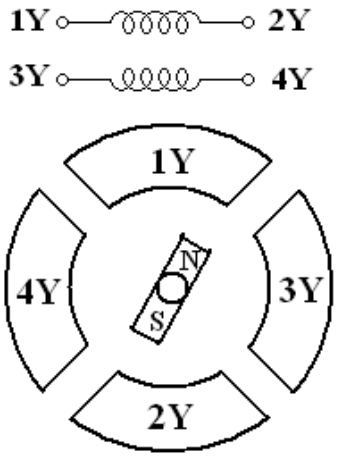


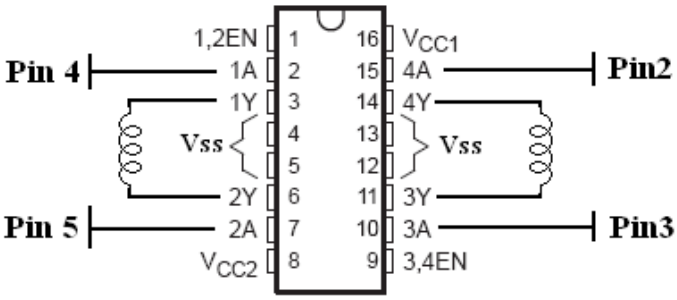
```
Initialize:
  LOW 2
  LOW 3

Main:
  HIGH 2
  LOW  3
  PAUSE 500

  LOW 2
  LOW 3
  PAUSE 500

  LOW  2
  HIGH 3
  PAUSE 500

  LOW 2
  LOW 3
  PAUSE 500
GOTO Main
```

Stepper Motor Control

When you would like some control over how much a motor turns, ones way is to use a stepper motor. Instead of applying voltage and the motor spinning until you disconnect the motor, each time you apply voltage to one of the wires, the motor only turns a small precise amount. The motor we will be using is a 400 steps per revolution motor, so each step we make, rotates the shaft 0.9 deg This motor has 4 wires in total, each of them is connected to the end of a coil. When wiring the circuit for this stepper motor you place the two wires from one of the coils just like the dc motor was set up. The other two wires will hook up like a second dc motor. This will allow you to plow current both directions through the coil. See the schematic below.

The theory behind stepper motors deals with magnetism. When you send current through a coil, the magnet on the motor shaft is attracted to the coil. So if we sequentially send current through the coils we can pull the shaft around. Since the coils alternate in position the coils have to be activated in a specific order to make it work correctly. Working through the diagram below, if we want to rotate the motor clock-wise we need to activate 1Y, and then activate 3Y, then 2Y, then 4Y. To go counter-clock-wise we activate 1Y, and then 4Y, then 2Y, then 3Y. During the steps in which you could is not activated, the pin must be set to ground. The speed of the motor is dependant on how long each coil is activated, the longer its activated, the slower it will go and vise-versa. If you try to go too fast, the motor will not respond properly. So the program below will show how to turn the motor for 200 steps in one direction, and then change direction for 200 steps. Each step is held for 15ms.





SYMBOL counter = **B0**

```
Main:
  FOR counter = 0 TO 50
    PULSOUT 2, 1500
    PULSOUT 5, 1500
    PULSOUT 3, 1500
    PULSOUT 4, 1500
  NEXT
  FOR counter = 0 TO 50
    PULSOUT 2, 1500
    PULSOUT 4, 1500
    PULSOUT 3, 1500
    PULSOUT 5, 1500
  NEXT
GOTO Main
```

Additional Programs

DC Motor with Transistor and Button

```
SYMBOL  x = B0

Initialize:
  LOW 1

Main:
  x = PIN0
  IF x = 0 THEN On
  LOW 1
 GOTO Main

On:
  HIGH 1
GOTO Main
```

DC Motor with H-Bridge and Buttons

```
SYMBOL  x = B0
SYMBOL  y = B1

Initialize:
  LOW 2
  LOW 3

Main:
  x = PIN0
  y = PIN7
  IF x = 0 AND y = 0 THEN Initialize
  IF x = 0 THEN Left
  IF y = 0 THEN Right
  LOW 2
  LOW 3
GOTO Main

Left:
  HIGH 2
GOTO Main

Right:
  HIGH 3
GOTO Main
```

Stepper Motor with H-Bridge and Buttons

```
SYMBOL  x = B0
SYMBOL  y = B1

Main:
  x = PIN0
  y = PIN7
  IF x = 0 AND y = 0 THEN Main
  IF x = 0 THEN Left
  IF y = 0 THEN Right

GOTO Main

Right:
  PULSOUT 2, 1500
  PULSOUT 5, 1500
  PULSOUT 3, 1500
  PULSOUT 4, 1500
GOTO Main

Left:
  PULSOUT 2, 1500
  PULSOUT 4, 1500
  PULSOUT 3, 1500
  PULSOUT 5, 1500
GOTO Main
```