

MÓDULO 4:

PROYECTO 9 "Rueda de Colores Motorizada"

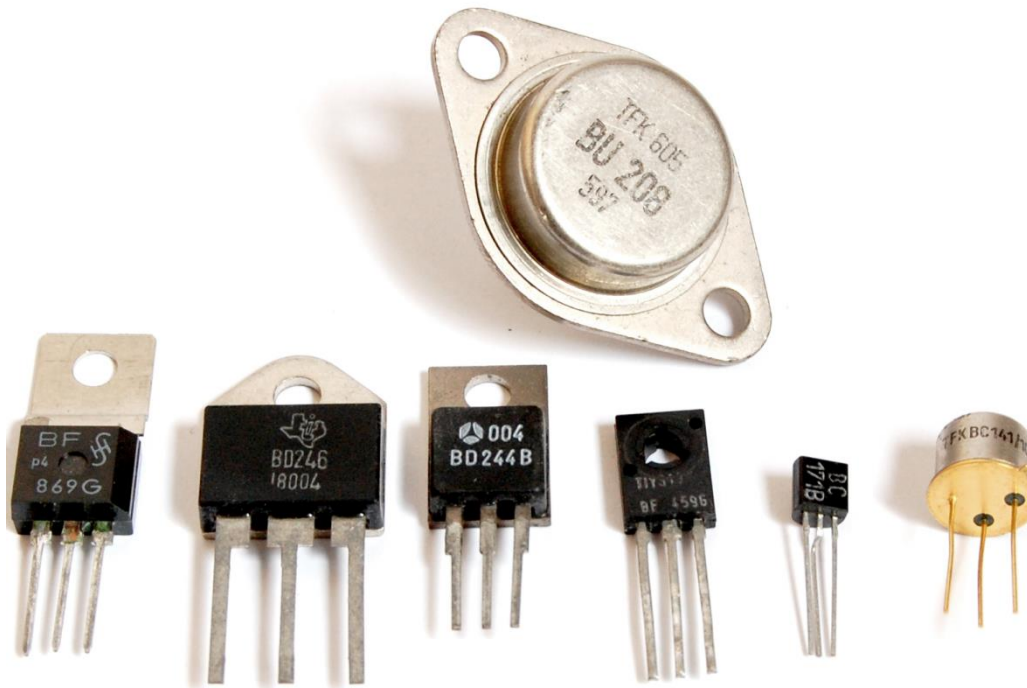
CURSO PROGRAMACIÓN DE PLACAS ROBÓTICAS



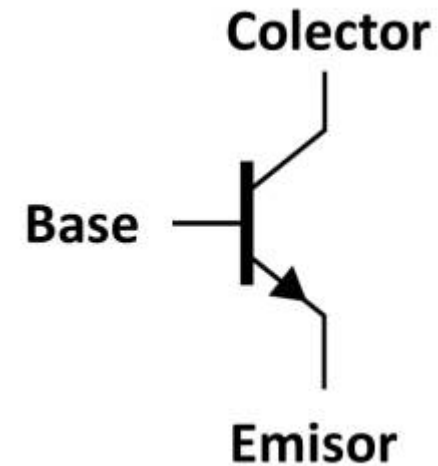
Introducción

- Controlar motores con Arduino es más complicado que controlar leds:
 - En primer lugar, un motor requiere más corriente de la que los pins de salida de Arduino pueden proporcionar.
 - En segundo lugar, un motor puede generar su propia corriente mediante un proceso llamado inducción, el cual, puede dañar nuestro circuito si no lo hemos tenido en cuenta.
- Nuestro Arduino tan solo puede suministrar 40 miliamperios (mA) de sus pines digitales, muchísimo menos de lo que los motores necesitan.
- Los **transistores** son componentes que nos permiten controlar elementos de alto voltaje e intensidad desde las salidas de baja corriente de nuestro Arduino.
- Cuando suministramos tensión a uno de los pins del transistor, llamado **base** (gate en inglés), se cierra el circuito entre los otros dos pins, llamados **emisor** y **colector** (en inglés, drain y source respectivamente).
- De esta manera, podemos activar o desactivar un motor de tensión y corriente elevadas con nuestro Arduino.

Introducción



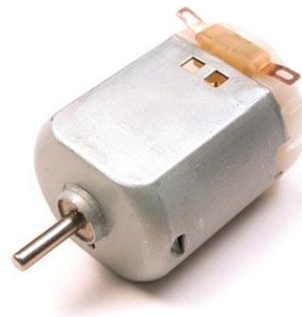
Diferentes tipos de transistores



Símbolo eléctrico del transistor

Introducción

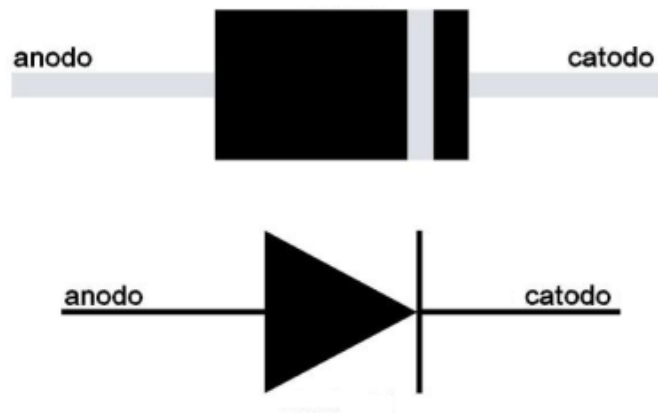
- Los **motores** son un tipo de dispositivo de inducción:
 - La inducción es un proceso por el cual, una corriente eléctrica variable que pasa por un cable, puede generar un campo magnético también variable alrededor de dicho cable.



Ejemplo de motor de corriente continua

Introducción

- Al contrario también funciona: un motor puede generar electricidad cuando el eje está girando. Esta tensión inversa, a veces denominada **tensión de retorno**, puede dañar nuestro transistor. Por esta razón, pondremos un diodo en paralelo al motor, y que de esta manera dicha tensión de retorno pase por el diodo y no por nuestro transistor. El diodo solo permite el paso de la corriente en un sentido, protegiendo de esta manera al resto del circuito



Montando el circuito

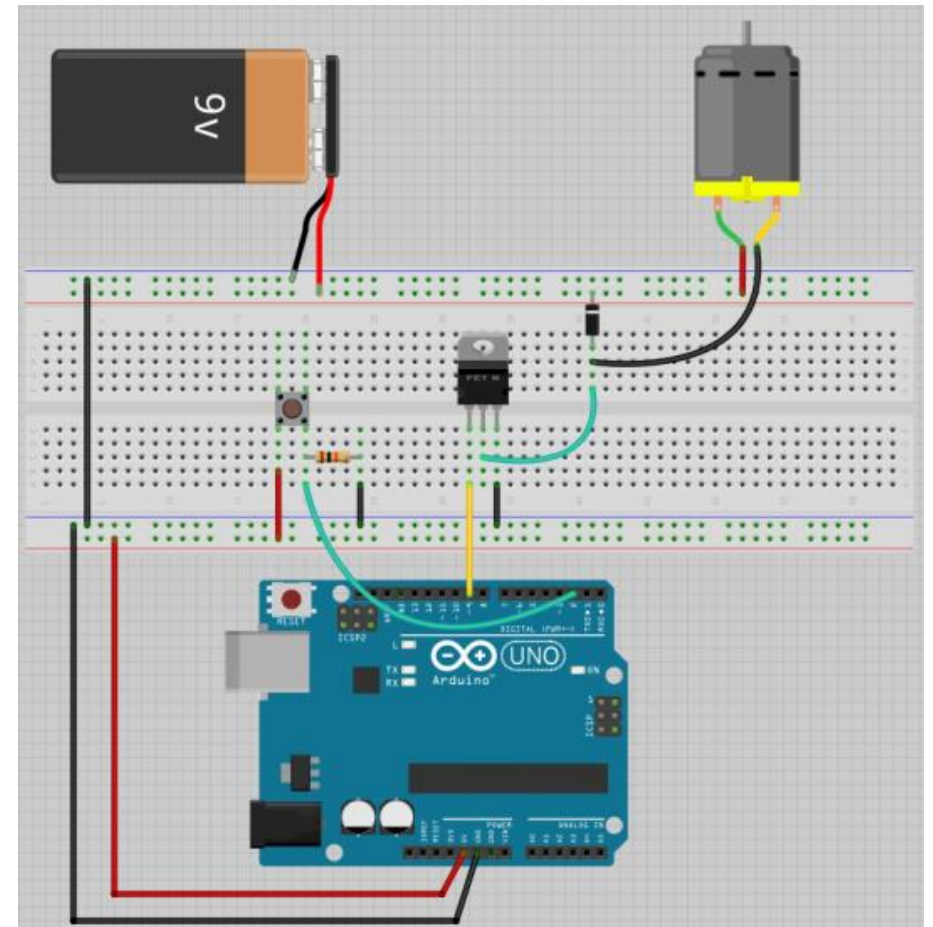
1. Tal y como hemos hecho en los proyectos anteriores, conectamos la alimentación y la tierra en nuestra protoboard.
2. Añadimos un pulsador, conectando la patilla de entrada a la alimentación y la de salida al pin digital 2 del Arduino. Añadiremos también la resistencia de $10K\Omega$ entre la patilla de salida del pulsador y tierra.
3. Cuando utilicemos circuitos con diferentes voltajes, debemos conectar sus tierras de forma conjunta, para de esta manera conseguir una tierra común. Seguidamente conectamos la batería de 9V a la protoboard. Unimos mediante un cable, la tierra de la batería con la del Arduino en la protoboard. Finalmente, conectamos la otra patilla del motor a la alimentación de 9V.

Montando el circuito

4. A continuación, colocaremos el transistor en la protoboard. Hay que tener en cuenta que la cara metálica de dicho transistor no debe de quedar mirando hacia nosotros. El siguiente paso será conectar el pin digital 9 al pin izquierdo del transistor. A este pin se le denomina **base**. Un cambio de voltaje en la base provoca la conexión entre los otros dos pines del transistor. Seguidamente conectaremos uno de los terminales del motor al pin central del transistor. A dicho pin se le llama **colector**. Cuando Arduino activa el transistor suministrándole tensión a la base, el colector se conectará con la tercera patilla, llamada emisor. Finalmente conectamos el emisor a tierra.
5. Seguidamente, conectaremos la fuente de 9V al motor y a la protoboard. El último componente que nos queda es el diodo. Dicho elemento es un componente polarizado, esto quiere decir que solo permite el paso de la corriente en un sentido. Si nos fijamos en dicho diodo, veremos que en uno de sus extremos tiene pintada una línea blanca, dicha línea nos indica el polo negativo del diodo o cátodo. El otro extremo es el positivo o ánodo. Conectaremos el ánodo del diodo a la tierra del motor y el cátodo al positivo del mismo. De esta manera, nuestro diodo protegerá al circuito de la tensión de retorno producida por el motor, impidiendo que entre en el circuito. Recordemos que dicha tensión de retorno fluye en dirección opuesta a la de la tensión que le suministramos al motor.

Montando el circuito

Componentes	
Transistor MOSFET	1
Resistencia de 10K Ω	1
Diodo 1N4007	1
Motor de corriente continua.	1
Batería de 9V.	1
Conector que nos permita enlazar la batería a la protoboard.	1
Pulsador	1



Comentario del código

- El código es notablemente similar al utilizado por primera vez para encender un led. Lo primero de todo es establecer unas constantes para los pines del motor y del pulsador y una variable de nombre **switchState** para almacenar el valor del pulsador.
- Declararemos el pin del motor como salida (**OUTPUT**) y el del switch como entrada (**INPUT**).
- Seguidamente, en la función **loop()**, nos centraremos en comprobar el estado de **switchPin** mediante **digitalRead()**.
- Si el pulsador ha sido presionado, le mandaremos un valor **HIGH** a motorPin. Si no ha sido presionado, le mandaremos un valor **LOW**. Cuando el valor de motorPin es **HIGH**, el transistor se activará, cerrando el circuito del motor. Por contra, cuando es un valor **LOW**, el motor no girará.

Comentario del código

- El código es notablemente similar al utilizado por primera vez para encender un led. Lo primero de todo es establecer unas constantes para los pines del motor y del pulsador y una variable de nombre **switchState** para almacenar el valor del pulsador.
- Declararemos el pin del motor como salida (**OUTPUT**) y el del switch como entrada (**INPUT**).
- Seguidamente, en la función **loop()**, nos centraremos en comprobar el estado de **switchPin** mediante **digitalRead()**.
- Si el pulsador ha sido presionado, le mandaremos un valor **HIGH** a motorPin. Si no ha sido presionado, le mandaremos un valor **LOW**. Cuando el valor de motorPin es **HIGH**, el transistor se activará, cerrando el circuito del motor. Por contra, cuando es un valor **LOW**, el motor no girará.