

## ***“Base móvil para robot”***

**Instrumentación Digital**

**Docente: Ramírez Barrueta Marcos**

*Flores Hernández Ylse*

*Padilla Torres Guadalupe Antonio*

*Rivas Bautista César René*

*Romero Martínez Jorge Alexis*

*Ingeniería en Sistemas Computacionales*

*9° cuatrimestre*

**7 de agosto de 2018.**

## Introducción

El siguiente proyecto ha sido desarrollado con la intención de mostrar lo aprendido en clase respecto de la Instrumentación Digital. Éste proyecto es una Base Móvil la cual servirá como inicio de un conjunto de proyectos que se unirán para formar un robot que contiene funciones diversas.

La Base Móvil tiene como función avanzar y retroceder con pulsos enviados desde una placa Arduino UNO y código en C. Esto se explicará en el desarrollo del proyecto, junto con los componentes utilizados para armar la base móvil.

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

Por otro lado, Arduino nos proporciona un software consistente en un entorno de desarrollo (IDE) que implementa el lenguaje de programación de arduino y el bootloader ejecutado en la placa. La principal característica del software de programación y del lenguaje de programación es su sencillez y facilidad de uso.

¿Para qué sirve Arduino? Arduino se puede utilizar para desarrollar elementos autónomos, conectándose a dispositivos e interactuar tanto con el hardware como con el software. Nos sirve tanto para controlar un elemento, pongamos por ejemplo un motor que nos suba o baje una persiana basada en la luz existente es una habitación, gracias a un sensor de luz conectado al Arduino, o bien para leer la información de una fuente, como puede ser un teclado, y convertir la información en una acción como puede ser encender una luz y pasar por un display lo tecleado.

Hay otro factor importante en el éxito de Arduino, es la comunidad que apoya todo este desarrollo, comparte conocimiento, elabora librerías para facilitar el uso de Arduino y publica sus proyectos para que puedan ser replicados, mejorados o ser base para otro proyecto relacionado.

El HW de Arduino es básicamente una placa con un microcontrolador. Un microcontrolador (abreviado  $\mu$ C, UC o MCU) es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida.

Características de un Microcontrolador:

- Velocidad del reloj u oscilador
- Tamaño de palabra
- Memoria: SRAM, Flash, EEPROM, ROM.
- I/O Digitales
- Entradas Analógicas
- Salidas analógicas (PWM)
- DAC (Digital to Analog Converter)
- ADC (Analog to Digital Converter)
- Buses
- UART
- Otras comunicaciones.

Arduino dispone de una amplia variedad de placas y shields para usar dependiendo de nuestras necesidades.

## Desarrollo del Proyecto

La base móvil consta de una placa de acrílico de 30x30cms que estará montada con 4 ruedas sobre motores que la harán girar. Los materiales utilizados en éste proyecto son los siguientes:

- Placa de acrílico de 30x30cms
- Protoboard
- 4 motores de 5V
- 4 ruedas
- Puente H
- Placa Arduino UNO
- Cable USB
- Batería externa
- Pila de 9V
- Pines para conexión
- Cinchos

Procedimiento de armado:

1. La placa de acrílico sirve como base del proyecto móvil, es por eso que se perfora con un taladro para atravesarle cinchos los cuales les darán el soporte a las piezas.
2. Se montan los motores en cada una de las bases con las ruedas.
3. Se montan las ruedas en cada una de las esquinas de la base de acrílico.
4. Se coloca la protoboard en la parte superior de la base.
5. Se montan la placa Arduino UNO y la pila de 9V a un lado de la protoboard.
6. Se hacen las conexiones necesarias con los pines respectivos de salida, voltaje y GND.

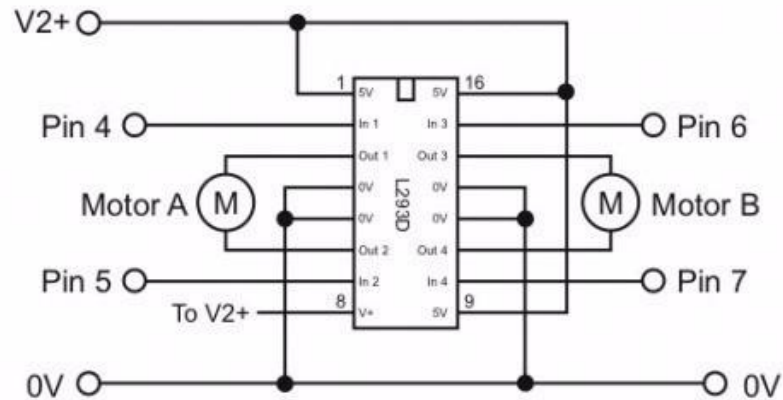
7. El puente H es conectado en la protoboard para darle el movimiento en ambos sentidos a cada una de las llantas de la base móvil.
8. Se realiza un código en C y se carga a la placa Arduino UNO
9. Se conectan los respectivos voltajes y se deja funcionar el código cargado al Arduino UNO.

La conexión de los motores para el funcionamiento de la base móvil es muy sencilla. Para que ésta avance sólo se necesita de conectar los polos positivos y negativos de cada motor a los respectivos polos de la pila conectada a la protoboard; esto hará que las llantas den en un sentido, pero, ¿cómo hacer que los motores corran en ambos sentidos?

Es aquí donde entra la magia de la placa Arduino UNO y el código de programación. Ésta placa se conecta a la protoboard y con ayuda de pines y conexiones múltiples se le da salida a cada puerto gracias al activado del código en C. Aquí sólo se inicializan los puertos y se carga un código que nos sirva para mandar pulsos de energía con las órdenes de HIGH (abierto) y LOW (cerrado).

Si se requiriera el encendido de dos motores, basta con inicializar en enteros dos pines de salida y hacer que la energía sea constante con la palabra reservada HIGH. Pero, ¿Qué pasa si queremos que los motores giren en sentido contrario?

Para hacerlos funcionar en sentido horario, el código fue explicado anteriormente. Para la energía en sentido anti horario es necesario el uso de un circuito integrado llamado Puente H (L293D). Éste circuito funciona de la siguiente manera:

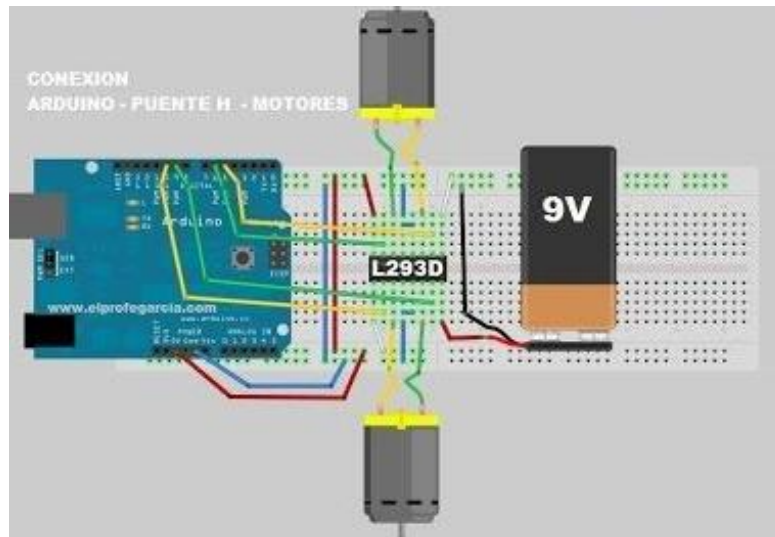


Es entonces que se comprende que un motor conectado a éste circuito puede ser manipulado para cambiar la corriente de sus polos. Para entender mejor, el funcionamiento del circuito, basta con explicar las salidas de cada una de las 16 patas:

1. Voltaje de Arduino (5V)
2. Pin de Controlador arduino
3. Motor positivo
4. GND
5. GND
6. Motor negativo
7. Pin de Controlador arduino
8. Voltaje Externo (9V)
9. Voltaje de Arduino (5V)
10. Pin de Controlador arduino
11. Motor positivo
12. GND
13. GND
14. Motor negativo
15. Pin de Controlador arduino
16. Voltaje de Arduino (5V)

## Diagrama

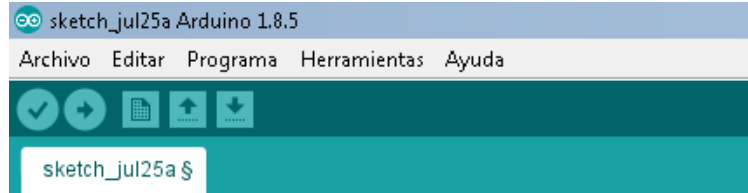
Las conexiones a los pines y las patas del arduino, la placa madre y el circuito integrado quedan de la siguiente manera:



## Código

El código cargado en el Arduino para el funcionamiento del circuito es el siguiente:

- Parte1: Declaración de Variables
- Parte 2: Inicialización de puertos
- Parte 3: Ejecución de un loop para el funcionamiento de las llantas



```
int izqA = 5;
int izqB = 6;
int derA = 9;
int derB = 10;
int vel = 255; // Velocidad de los motores (0-255)
```

```
void setup() {
  pinMode(derA, OUTPUT);
  pinMode(derB, OUTPUT);
  pinMode(izqA, OUTPUT);
  pinMode(izqB, OUTPUT);
}
```

```
void loop() {
  analogWrite(derB, 0); // Detiene los Motores
  analogWrite(izqB, 0);
  delay (500);
  analogWrite(derA, vel); // Frente 2 segundos
  analogWrite(izqA, vel);
  delay (2000);

  analogWrite(derA, vel); // Derecha 0,5 segundos
  analogWrite(izqA, 0);
  delay (500);

  analogWrite(derA, 0); // Izquierda 0,5 segundos
  analogWrite(izqA, vel);
  delay (500);

  analogWrite(derA, 0); // Detiene los Motores
  analogWrite(izqA, 0);
  delay (500);
  analogWrite(derB, vel); // Reversa 2 segundos
  analogWrite(izqB, vel);
  delay (2000);
}
```

---



## Ajustes

Al principio se tuvo problemas con el proyecto ya que no se encontraba la manera de hacer que las llantas o motores tuvieran el movimiento en ambos sentidos.

Cuando se estudió el sensor L293D o puente H, se determinó que se podían conectar 2 motores y se podía cambiar la polaridad de la energía para que estos tuvieran el movimiento requerido. Las conexiones encontradas en tutoriales y cursos de programación para Arduino eran demasiado complejas y no nos arrojaban ningún resultado.

Se buscó tener una respuesta al problema con un código sencillo que nos proporcionara estabilidad y fuera fácil de entender, ya que los códigos buscados en internet eran demasiado complejos y contenían librerías que no eran fáciles de entender.

Al entender perfectamente el funcionamiento del sensor L293D se resolvió el problema, pues pudimos ver que las 4 esquinas del puente H eran las que requerían voltaje, pero sólo una de ellas debía estar conectada al voltaje de la pila de 9V, y las demás eran controladas con la corriente distribuida por la placa Arduino.

Otro de los ajustes fue el de comprar una pila de 9V nueva, ya que la que se estaba usando en aquel momento era usada y no daba el movimiento de las ruedas las cuales son muy pesadas. Se creyó que las conexiones estaban mal, o que los pines no servían, inclusive que el código cargado desde la computadora, el cual no presentaba fallas, estaba mal. Al final pudimos observar que al utilizar otra batería el problema se resolvió.

Aun así, es un proyecto tan ambicioso que requiere de mucha energía, es por eso que se da el uso de la corriente externa y el de una pila de 9V.

## Conclusiones

El proyecto pudo funcionar gracias a la colaboración de todos los integrantes del proyecto, así como el ingenio que tuvo todo el equipo en el armado, la obtención de materiales y la creación de un código sencillo y práctico que todos pudieran comprender.

Se concluyó que una base de madera no era viable ni factible ya que pesaba mucho y las pilas conectadas no daban la potencia necesaria para el avance de las llantas.

También se concluyó que de utilizar una fuente de poder más alta podrían ocurrir daños en los sensores o incluso en la protoboard.

El código cargado en C hacia la placa Arduino UNO consta de variables enteras para los pines, inicializadores para cada puerto y un loop que funciona como un ciclo el cual se repite para dar la ejecución de las ruedas en ambos sentidos del reloj.

El sensor L293D o puente H es de mucha ayuda en este tipo de proyectos ya que permite cambiar la polaridad a diferentes dispositivos de entrada y salida. En nuestro caso, los motores que requerían avanzar y retroceder.

Las conexiones deben ser perfectas y correctas para que el sensor L293D pueda distribuir la energía y las órdenes necesarias a cada una de las partes de la base móvil.

## Bibliografía

- Torrente Artero, Óscar. (2013). Arduino. Curso práctico de formación. México D.F. Alfaomega Grupo Editor, S.A. de C.V.
- Reyes Cortés, F.; Cid Monjarra, J. Arduino. (2015). Aplicaciones en robótica, mecatrónica e ingenierías. México D.F.: MARCOMBO, S.A.
- Tojeiro Calaza, G. (2014). Taller de Arduino. Un enfoque práctico para principiantes. Barcelona. MARCOMBO, S.A.
- Pomares Baeza, Jorge. (2009). Control por computador. Manual de Arduino. Universidad de Alicante. GITE - IEA