

# Diseño de un Carrito Seguidor de Línea de Código y Hardware Abierto.

line 1: 1<sup>st</sup> Cesar Antonio Pérez Rauda  
 line 2: *Escuela de Eléctrica y Electrónica*  
 line 3: *Itca-FEPADE*  
 line 4: Mejicanos, San Salvador  
 line 5: cesar.perez16@itca.edu.sv

line 1: 1<sup>st</sup> Julio Alexander Díaz Reyes  
 line 2: *Escuela de Eléctrica y Electrónica*  
 line 3: *Itca-FEPADE*  
 line 4: Izalco, Sonsonate  
 line 5: julio.diaz16@itca.edu.sv

line 1: 1<sup>st</sup> Jaime Mauricio Pérez Funes  
 line 2: *Escuela de Eléctrica y Electrónica*  
 line 3: *Itca-FEPADE*  
 line 4: Zaragoza, La Libertad  
 line 5: jaime.perez16@itca.edu.sv

line 1: 1<sup>st</sup> Raul Ernesto Aguilar Alvarez.  
 line 2: *Escuela de Eléctrica y Electrónica*  
 line 3: *Itca-FEPADE*  
 line 4: San Martín, San Salvador  
 line 5: raul.aguilar16@itca.edu.sv

**Abstract:** For our final project of micro controllers matter is the construction of a line follower, putting into practice each of the subjects seen in class, for the elaboration of this project we need two DC motors of twelve volts, two sensors Cny70, the construction of each of the plates used, these being those of the sensors and the one of the driver to control the motors. We also use what is a Mega Arduino which serves as a micro controller to automate our car through the code.

**Keywords:** Arduino, Motors, Micro controllers, Driver, Sensors, Code.

## I. INTRODUCCION.

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.

En la actualidad tiene un gran uso en distintos campos académicos a fines al estudio de la electrónica, debido que permite diseñar un sinnúmero de proyectos y programas prácticos, para el funcionamiento de los dispositivos con Arduino.

Además, en nuestro proyecto, nos permite aplicar la programación que se ha ido aprendiendo durante el inicio del módulo de microcontroladores y poder aplicar los conocimientos obtenidos teóricamente en clase.

Se tiene como objetivo en nuestro proyecto un carro seguidor de línea, el cual, se guiará por medio de unos sensores siguiendo un trazo en una pista. Los sensores funcionaran mediante un código hecho en Arduino, este enviara una señal ya sea al motor izquierdo o al derecho de lo que tendrá que hacer, en caso de que el sensor en el motor izquierdo se active entonces el motor derecho se activara y en caso de que el sensor derecho se active el motor derecho lo hará.

## II. METODOS DE DISEÑO.

Desde el inicio que se planteó el proyecto lo primordial era, definir el hardware que se usaría para llevarlo a la realidad. Y además saber la selección de los insumos electrónicos y el diseño de la circuitería que se aplicaría, con la ayuda de un software especializado (Eagle cad) para la creación de los circuitos que conformaran nuestro driver controlador de motores y los módulos de sensores infrarrojos.

### A. Motores.

La decisión de utilizar motores DC con reductores la hemos tomado ya que es un potente motor que proporciona 152 revoluciones por minuto con una potencia de 16,7 kg.cm que lo hace ideal para las aplicaciones de robótica. Poseen las características más adecuadas para nuestro proyecto, las cuales son:

- Voltaje de funcionamiento de 3V -12V DC.
- El peso del producto es de 0.030 kg, lo cual nos beneficia a la hora de colocarlo en el carro ya que este motor no pesa mucho.
- La corriente de carga es de unos 70 mA, 250 mA.
- El tamaño es de (L x A x H) 7.5 x 4 x 2.2 cm. [1]

### *B. Driver para los motores.*

El driver que hemos utilizado es el L293d debido a que es más accesible económicamente hablando y fácil de encontrar en las tiendas de electrónica, incluyen cuatro circuitos para manejar cargas de potencia media, en especial pequeños motores y cargas inductivas, con la capacidad de controlar corrientes hasta 600 mA en cada circuito y una tensión entre 4,5V a 36V.

Los circuitos individuales se pueden usar de manera independiente para controlar cargas de todo tipo y, en el caso de ser motores, manejar un único sentido de giro. Pero además, cualquiera de estos cuatro circuitos sirve para configurar la mitad de un puente H.

El integrado permite formar, entonces, dos puentes H completos, con los que se puede realizar el manejo de dos motores. En este caso el manejo será bidireccional, con frenado rápido y con posibilidad de implementar fácilmente el control de velocidad. [2]

### *C. Sensores.*

Los sensores CNY70 son sensores que incluyen un led infrarrojo y un fototransistor. Este se utiliza junto con el Arduino como medidores de distancia o detector de líneas. El circuito integrado tiene cuatro pines, el pin (1) que es el ánodo, el pin (2) que es el cátodo, el pin (3) que es el colector y el pin (4) que es el emisor. Siendo el pin (1) y (2) son para para el led emisor y el pin (3) y (4) son para el fototransistor.

Elegimos este sensor ya que es bastante sencillo de hacer su diagrama en Eagle cad, ya que este solo necesita resistencias y un transistor para su buen funcionamiento.

Este sensor infrarrojo es de un rango de corto alcance (menos de 5cm) que se utiliza para detectar los colores de los objetos. El uso usualmente es para la construcción de pequeños robots, en nuestro caso un robot seguidor de línea. [3]

### *D. Chasis del carro.*

El chasis del carro esta hecho en su totalidad de madera tipo plywood ya que es un material bastante liviano y estable, por lo tanto a la hora de su manejabilidad de conectar los componentes es mucho más fácil, la forma es rectangular con una medida de 18.5cm de largo y 9cm de ancho.

En la parte del frente estará ubicada una rueda giratoria o también conocida como rueda loca la cual le permite girar hacia donde se encuentre la línea, detrás de esta estarán ubicados los sensores, a una altura de 1cm, en la parte de arriba estarán las placas de los sensores, el driver puente h y el Arduino Mega, todo esto unido mediante cables pin header. Por la parte de atrás están ubicados los motores DC con las llantas.

Esto nos permite una mayor estabilidad a la hora de que el carrito se mueva por la pista, y que no se pierda mientras este sigue la línea.

### *E. Arduino Mega.*

El Arduino Mega 2560 es un tablero de microcontroladores basado en el ATmega2560. Tiene 54 pines digitales de entrada / salida (de los cuales 15 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio. Contiene todo lo necesario para soportar el microcontrolador; simplemente conéctelo a una computadora con un cable USB o con un adaptador de CA a CC o batería para comenzar. Por esta razón hemos tomado a bien el utilizar el Arduino Mega para tener un microcontrolador con la suficiente memoria para nuestro código. [4]

### *F. Código.*

El código de este proyecto esta hecho en Arduino ya que hemos optado por utilizar este microcontrolador, en el código mostramos en cada una de las líneas lo que estas hacen, por ejemplo las primeras líneas se muestran que el pin 10 y el pin 11 de nuestro Arduino recibirán los datos desde los sensores, los datos serían inicialmente en "1" significando que visualiza el color blanco, al pasar el sensor por la línea negra se muestra como dato el número "0".

Luego se declaran los pines 3,4 y 6,7 estos para la entrada de los motores desde el driver, así puedan hacer lo que el sensor mande de señal.

Cabe destacar que cuando ambos sensores se encuentren en el color blanco o sea en "1" las llantas del robot seguidor de línea irán hacia atrás y cuando ambos sensores se encuentren en negro o sea "0" ambos motores irán hacia delante. Si un sensor esta en "1" y el otro esta en "0" el sensor que muestre el "1" hará que el motor se detenga para que a la hora de dar un giro este solo este funcionando un motor independientemente del lado que sea.

## **II. RECOMENDACIONES.**

- La recomendación más básica seria, empezar por la mecánica tomando en cuenta por supuesto, la potencia de los motores, que deben tener la suficiente fuerza para soportar al chasis, cabe destacar que todo debe estar alineado, de la forma más precisa posible, ya que, de lo contrario, el control tendrá que lidiar con esos defectos en la alineación.
- Algo que es muy importante, es el peso, ya que mientras menos peso exista, menor será la inercia que vencer, tanto en el arranque como en las curvas. con menos peso también, los motores tendrán que hacer menos esfuerzo y al hacer menos esfuerzo, necesitaran menos corriente, al necesitar menos corriente, se podrán ocupar

- baterías más pequeñas y al usar baterías más pequeñas, se podrá tener menos peso.
- Parte del diseño de la mecánica debe ser el tipo de pilas a ocupar, para que estas, no queden mal puestas o desequilibren el centro de gravedad del móvil, así que se les debe considerar desde el principio.

### III. CONCLUSIONES.

Se diseñó e implementó, un robot seguidor de línea, el funcionamiento del robot fue adecuado a la meta propuesta. Se demostró que a pesar de que se usaron componentes básicos de electrónica son suficientes para realizar un proyecto interesante y vistoso. El robot cumple las expectativas esperadas además este sirve como base para adentrarse en el campo de la electrónica y robótica.

Se noto que la ubicación de los sensores encargados de mantener el carro sobre la línea es prácticamente fundamental ya que, si no se encuentran donde debería estar el robot no hace lo deseado así la parte electrónica y de programación estén en perfecto estado.

### IV. REFERENCIAS

- [1] «Super Robotica,» 18 julio 2013. [En línea]. Available: <http://www.superrobotica.com/s330140.htm>.
- [2] E. J. Carletti, «Robots,» 2017. [En línea]. Available: [http://robots-argentina.com.ar/MotorCC\\_L293D.htm](http://robots-argentina.com.ar/MotorCC_L293D.htm).
- [3] «Tecnosefarad,» 03 marzo 2014. [En línea]. Available: <http://www.tecnosefarad.com/2014/03/sensor-de-infrarrojos-cny70-como-entrada-digital/>.
- [4] Arduino, «Ardunio,» [En línea]. Available: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>.