Proyecto 2

Seguidor de Línea

Bohórquez P. Duvier, Vargas F. Darmael, Loaiza C. Braulio

***Resumen*— En el presente documento se presentan todos los procesos realizados para dar cumplimiento a los requerimientos de este proyecto, en términos de montaje y descripción de los componentes utilizados, junto con el análisis teórico que describe el algoritmo de control aplicado en este caso, el cual consiste, básicamente, en la aplicación del PID sobre los motores que proporcionan movimiento al robot seguidor de línea, con lo cual se garantiza que el sistema no tienda a inestabilizarse.**

***Índice de términos***— **Arduino, QTR-8RC, Driver de Motor, Control PID, Seguidor de Línea**

# Introducción

E

s importante entender el funcionamiento de los procesadores para, así mismo, tener entendimiento acerca de las utilidades que los mismos pueden tener, al distinguirse como un elemento muy importante en la electrónica actual, puesto que estos se pueden encontrar en la gran mayoría de dispositivos de esta época, que basan su funcionamiento en procesos y operaciones que se controlan digitalmente. Para el caso del controlador Arduino, el cual se puede programar de muchas maneras según la necesidad a suplir, permite una interacción más cercana con el procesador contenido allí.

Para este proyecto se podrá visualizar una de las aplicaciones más comunes que posee este dispositivo, en donde se puede realizar dicha interacción con el procesador, al programar el Arduino para que este se encargue de realizar el control consecutivo de la velocidad y sentido de un robot que se debe mover en la dirección indicada por una línea negra situada en el suelo. A lo largo de este informe se analizaran estas ideas.

# Objetivos

***Objetivo General***

* Construir un robot seguidor de linea eficiente que pueda circular por un circuito sin salirse del trayecto

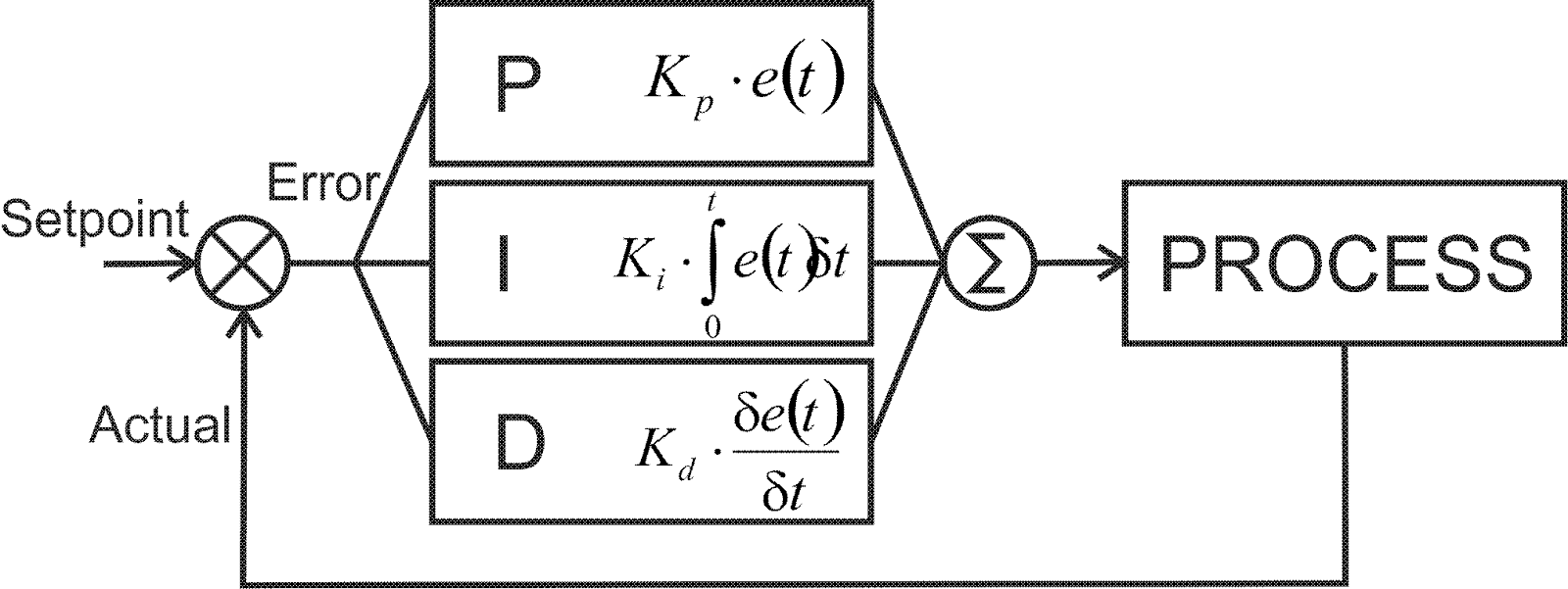
***Objetivos Específicos***

* Implementar un control PID efectivo.
* Usar componentes que se ajusten al bolsillo de los estudiantes.
* Realizar un montaje tal que no cause mucha reducción en la velocidad del carro.
* Hacer uso de una tarjeta de desarrollo y programar su microcontrolador de acuerdo a las diferentes necesidades
* Ganar la carrera contra otros seguidores de línea.

# marco teórico

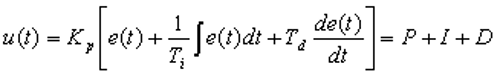
Para implementar el control PID se debe tener en cuenta que lo compone, para que sirve, como se puede optimizar y como funciona en conjunto con más partes.

El PID (control proporcional, integral y derivativo) es un mecanismo de control por realimentación que calcula la desviación o error entre un valor medido y el valor que se quiere obtener (set point, target position o punto de consigna), para aplicar una acción correctora que ajuste el proceso.



**Fig 1**. PID en lazo de control.

En el caso del robot velocista, el controlador PID, (que es una rutina basada matemáticamente), procesara los datos del sensor, y lo utiliza para controlar la dirección (velocidad de cada motor), para de esta forma mantenerlo en curso.

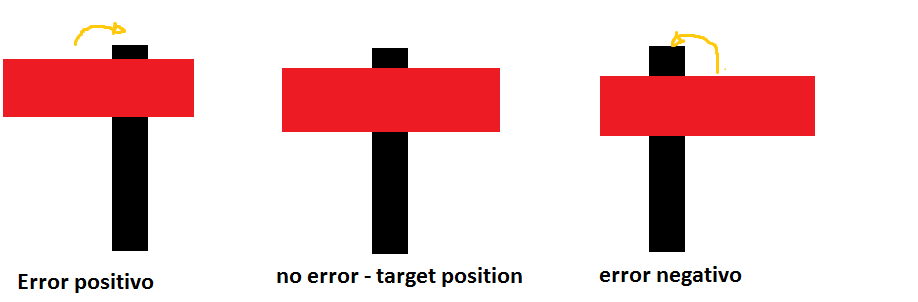


**Fig 2.** Ecuación del PID.

Valor de referencia o setpoint: Es aquel valor en donde los sensores se encuentran en la posición ideal, en este caso, los dos sensores en medio del arreglo se encuentran sobre la línea, para este valor de setpoint se tiene 3500

Posición: Es el valor leído y promediado por los sensores, determina como su nombre lo indica, la posición del carrito.

Error – Se denomina error a la diferencia entre la posición y el valor de setpoint, se busca que este valor sea cero usando el controlador, el objetivo es mantener este valor en cero.

**Fig 3**. Posiciones con respecto a la línea que toman los sensores infrarrojos y su respuesta ideal

Componente proporcional: Es la respuesta al error que se tiene que entregar de manera inmediata, es decir, el robot se encuentra en el centro de la línea, los motores , tendrán en respuesta una velocidad de igual valor, se aleja del centro, uno de los motores reducirá su velocidad y el otro aumentará.

Componente integrativa: La integral es la sumatoria de los errores acumulados, tiene como propósito el disminuir y eliminar el error en estado estacionario provocado por el modo proporcional, en otras palabras, si el robot se encuentra mucho tiempo alejado del centro (ocurre muchas veces cuando se encuentra en curvas), la acción integral se ira acumulando e irá disminuyendo el error hasta llegar al punto de referencia.

Componente derivativa: Es la derivada del error, su función es mantener el error al mínimo, corrigiéndolo proporcionalmente con la mismo velocidad que se produce, de esta manera evita que el error se incremente, en otra palabra, anticipara la acción evitando así las oscilaciones excesivas.

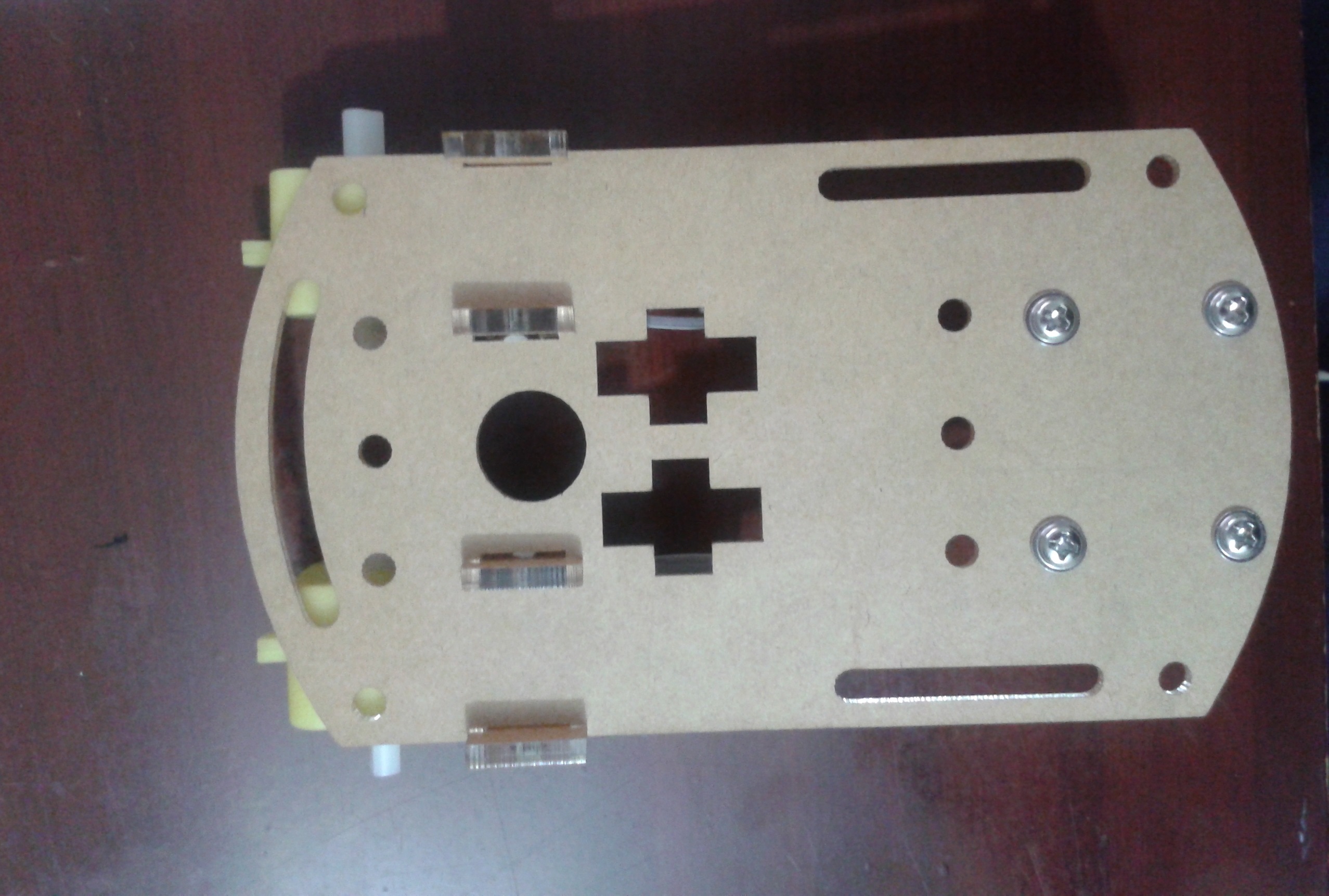
***Constantes***

Kp- Es un valor constante utilizado para aumentar o reducir el impacto de Proporcional. Si el valor es excesivo, el robot tendera responder inestablemente, oscilando excesivamente. Si el valor es muy pequeño, el robot responderá muy lentamente, tendiendo a salirse de las curvas

Ki- Es un valor constante utilizado para aumentar o reducir el impacto de la Integral, El valor excesivo de este provocara oscilaciones excesivas, Un valor demasiado bajo no causara impacto alguno.

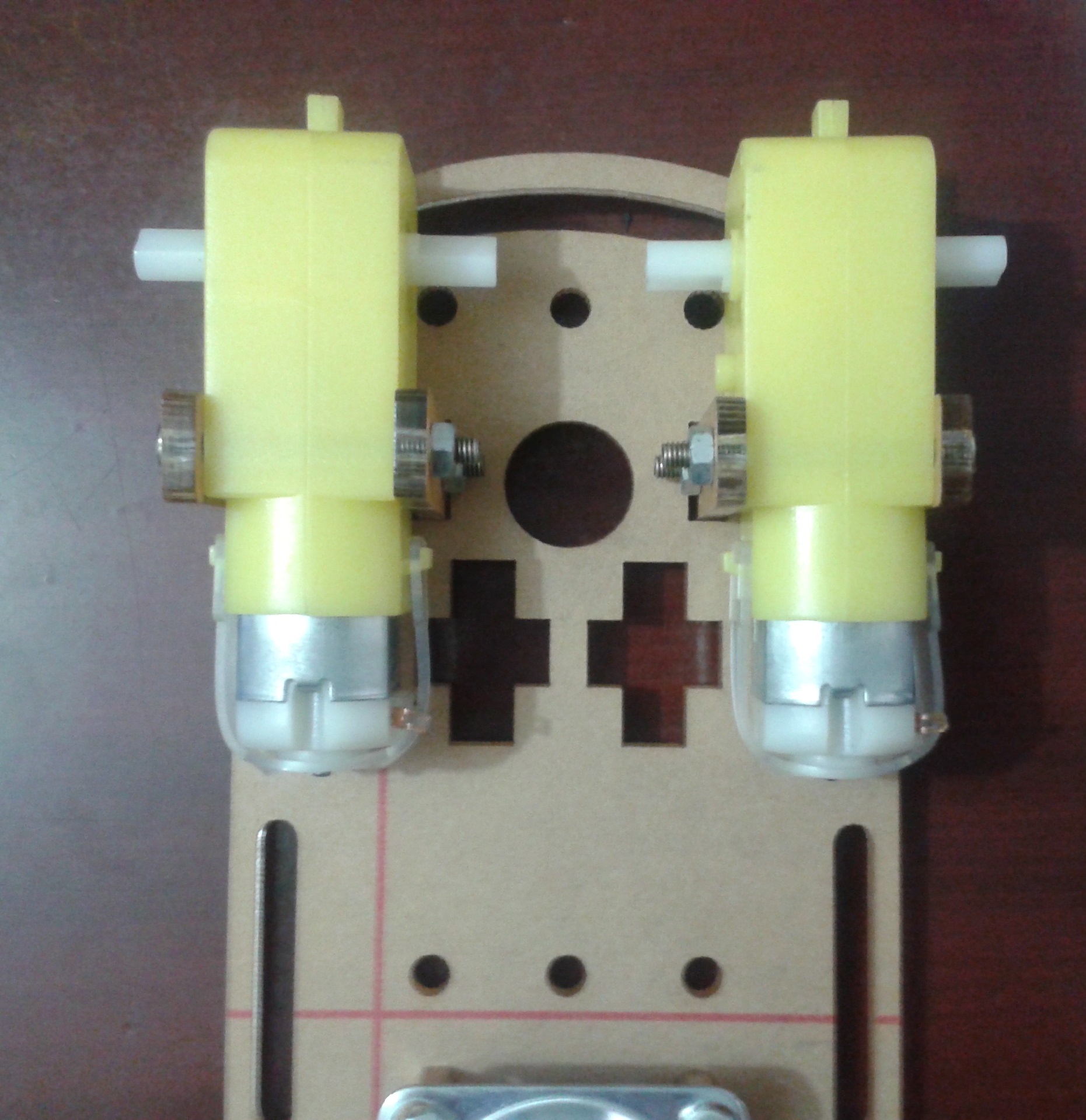
Kd- Es un valor constante utilizado para aumentar o reducir el impacto de la Derivada. Un valor excesivo provocara una sobre amortiguación, provocando inestabilidad.

# Procedimiento



**Fig. 4** Placa base donde se adecuaran los elementos que componen al carrito

En primer lugar, en cuanto a los componentes a utilizar, se hizo uso de la placa base mostrada, en la figura 4, la cual consiste en un chasis en acrílico, el cual se encuentra cubierto por una capa protectora superior e inferior con un adhesivo en color cartón que puede retirarse si se desea. Además este posee diversas perforaciones, las cuales son útiles para realizar una instalación de componentes, de manera más organizada, en términos del cableado que permite la interconexión de cada dispositivo a instalar en el mismo.



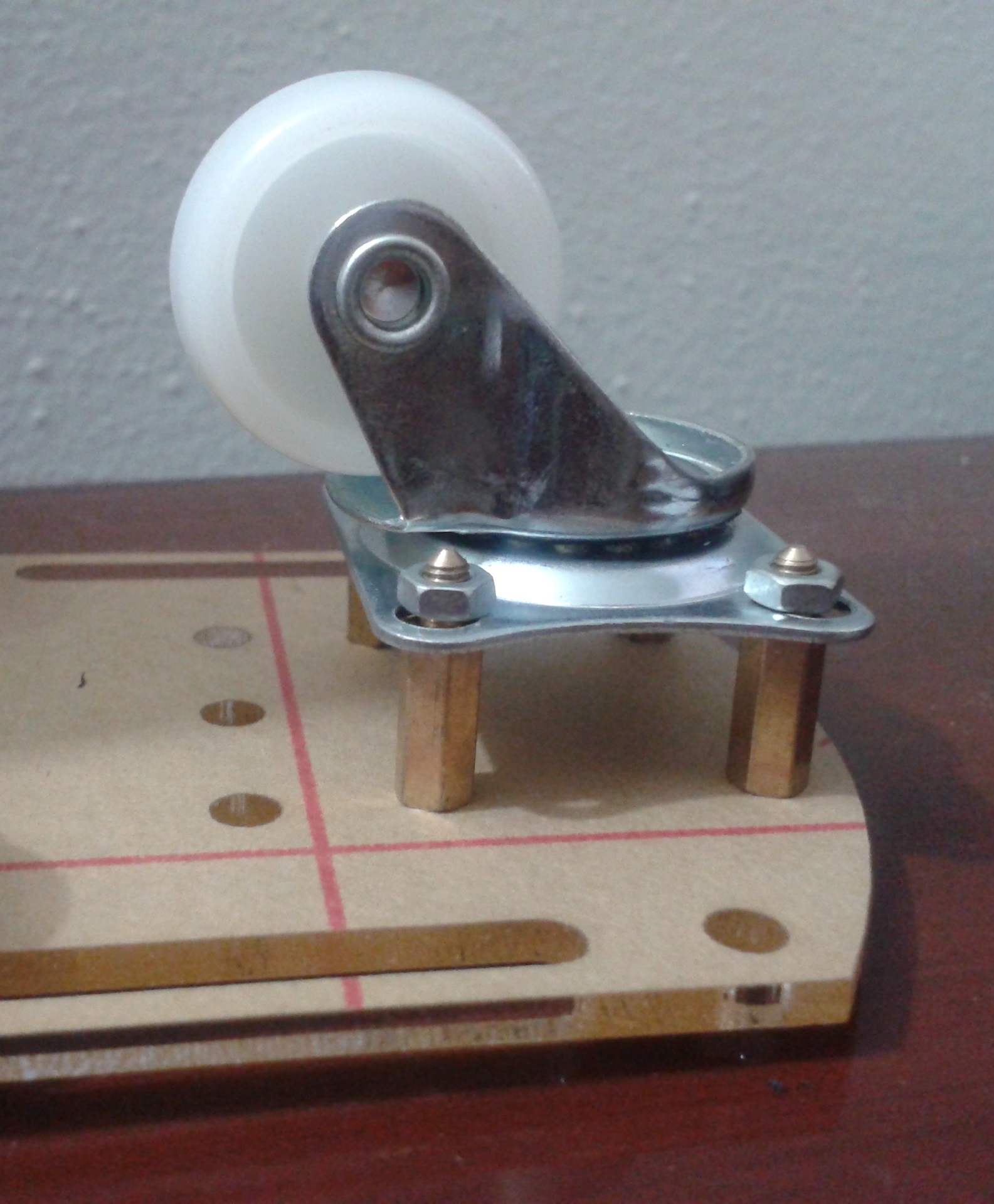
**Fig. 5** Motorreductores con caja reductora

A dicha placa se le añaden dos motorreductores, como se muestra en la figura 5, los cuales cumplen con la función de ejecutar el proceso de movimiento que mostrara el carrito, cuando este se encuentre en funcionamiento. Es importante destacar el hecho de que estos elementos posen un voltaje de alimentación situado en un rango de 3 a 6 voltios, además de contar con una relación de caja de cambios de 1:48, lo que quiere decir que cada uno de los motores poseen una rotación de 48 vueltas, por lo que el tacómetro permite capturar 192 pulsos (48x4) y 8 rpm del motor, así pues, el módulo de velocidad permite capturar 15.306000 pulsos/rpm.



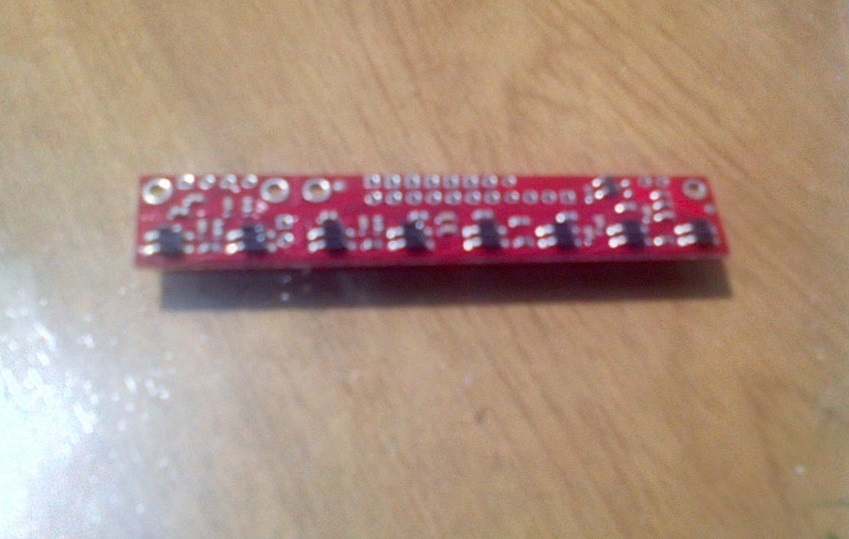
**Fig. 6** Ruedas

La figura 6 muestra el conjunto de ruedas que serán ubicadas en cada uno de los motores descritos anteriormente. En cuanto a sus dimensiones, estas poseen 6 cm de diámetro y 2.5 cm de ancho.



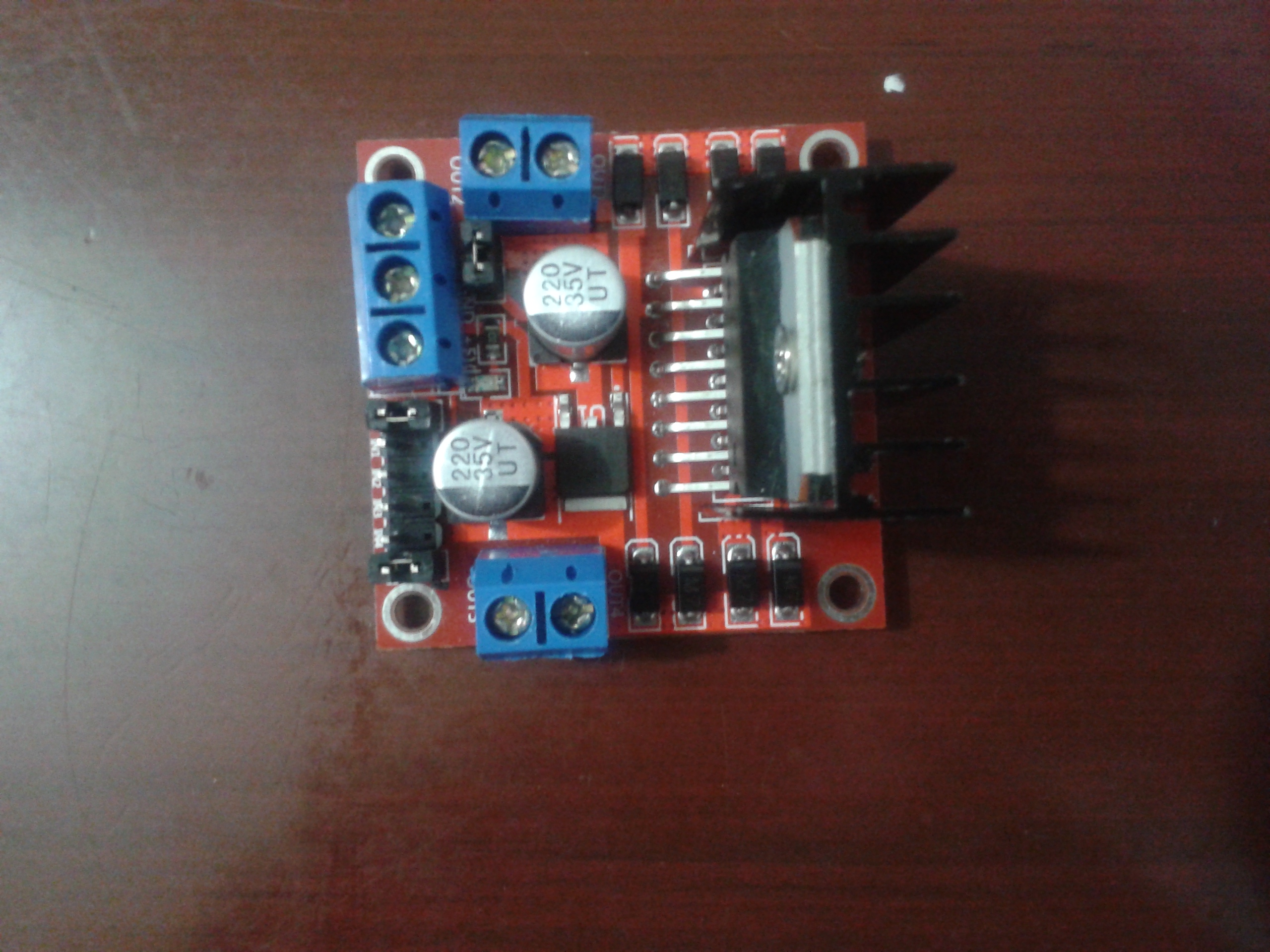
**Fig. 7** Rueda delantera

La rueda que se observa, en la figura 7, se debe utilizar para que el carrito conserve un equilibrio permanente, al tener la capacidad de orientarse en cualquier dirección. En cuanto a las dimensiones se debe destacar que esta posee 2.2 cm de diámetro y 1.2 cm de ancho. Por otra parte, observando esta imagen, se puede apreciar que esta se encuentra levantada por medio de un conjunto de tornillos y esto se hace con el fin de lograr un balance y/o nivelación del chasis, para que no ocurran problemas de estabilidad en su funcionamiento.



**Fig. 8** Sensor Qtr-8rc

En este proyecto es necesario hacer uso de un conjunto de sensores que permitirán la detección de la línea negra que seguirá el carrito. La figura 8 muestra dicho elemento, el cual es una placa que contiene un arreglo de 8 sensores montados a 9.5 mm uno del otro, por lo que es un gran detector para seguidores de línea. En este caso cada sensor proporciona una salida digital, la cual puede ser medida de forma separada. Las dimensiones de este objeto son de 2.5’’x 0.5’’x 0.125’’ sin pines y funciona a partir de un rango de voltajes situado entre 3.3 y 5 voltios. Además de esto, dicho dispositivo se debe alimentar con una corriente de 100 mA y una distancia optima de detección debe ser de 3 mm, aunque la máxima distancia recomendada de detección es de 9.5 mm.



**Fig. 9** Driver puente H L298n doble motor

La figura 9 muestra un driver puente H, el cual será utilizado en la implementación del seguidor de línea, puesto que este permite dar funcionamiento a los motores, a partir de la información que suministre el controlador arduino. Este elemento es, básicamente, un controlador de motores de alto voltaje, alta corriente y es diseñado para aceptar estándares de niveles lógicos TTL. Este dispositivo se encuentra en la capacidad de manejar cargas inductivas tales como relés, solenoides, motores DC y de paso. En cuanto a sus especificaciones técnicas, este dispositivo cuenta con un voltaje de control de 5 voltios, una corriente de control situada entre 0 y 36 mA, soporta temperaturas desde -20 hasta 135 grados centígrados, presenta un voltaje de salida desde 5 voltios hasta 35 voltios, soporta una corriente máxima de 2 A, una potencia máxima de 25 Watts, sus dimensiones externas son de 43x 43x 27 mm y pesa 30 gramos.

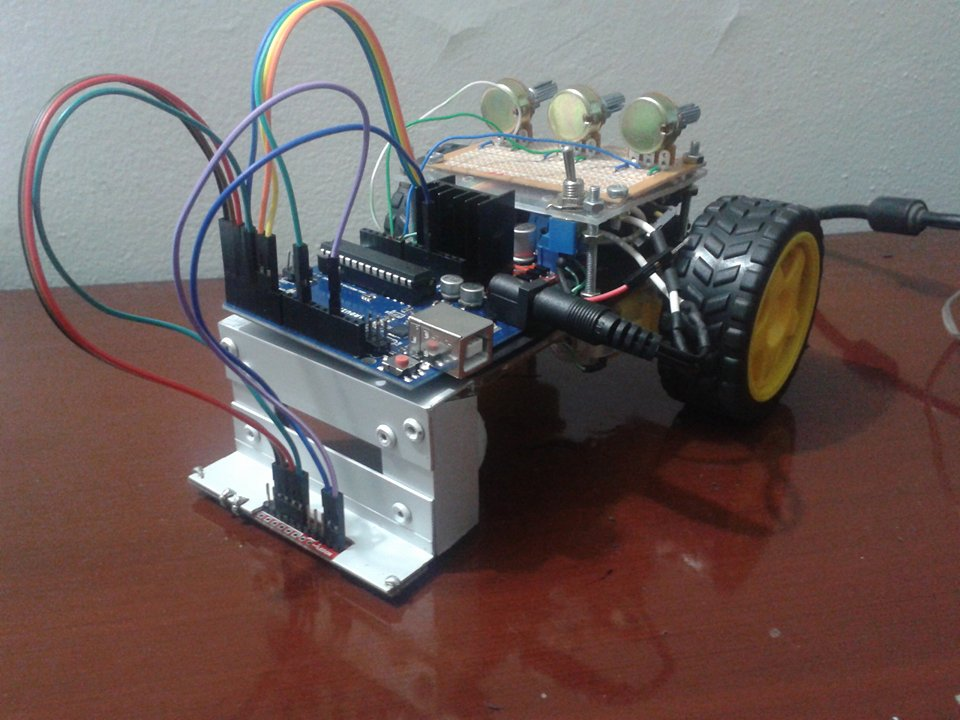


**Fig. 10** Tarjeta Arduino UNO R3 mod junto al montaje provisional

Se hizo uso de la tarjeta Arduino UNO ya que esta es más pequeña que el Arduino Mega 2650 y cumplen la misma función a costa de pines ADC, pero esto no es obstáculo ya que el sensor es digital y solo se necesitaron tres entradas análogas para las diferentes constantes.

Para las baterías se usaron pilas de 9v pero éstas se consumían en muy poco tiempo, ya que debían alimentar al Arduino y el puente H, la solución a esto fue dos baterías de celular recargables y conectadas en serie.

También se ubicó en la parte superior una baquela con los potenciómetros que calibrarían las constantes, estos alimentados con el 5v del Arduino.

**Fig 11 Montaje final.**

# Conclusiones

* Finalmente se logra el ensamblaje de todos los componentes que hacen parte del carrito seguidor de línea, como se muestra en la figura 11, con el cual ha sido posible realizar las pruebas correspondientes, con respecto al algoritmo implementado para aplicar el control PID, teniendo en cuenta todos los parámetros especificados previamente, con respecto al modo de funcionamiento que posee cada dispositivo, el diagrama de conexiones establecido según las condiciones y/o configuraciones establecidas en el código que se ha implementado, y la precisión que posee los sensores utilizados para la identificación de la posición en la que se encuentra la línea negra, con respecto al carrito seguidor de línea, para aplicar el control.
* En términos generales, con respecto al desarrollo del proyecto y la implementación del carrito seguidor de línea, se concluye el mismo con la adquisición de experiencia suficiente, en cuanto al funcionamiento del procesador contenido en la placa Arduino UNO utilizada, en este caso, puesto que se ha podido establecer una interacción continua con dicho dispositivo, al realizar un análisis a nivel algorítmico, observando cada sentencia aplicada y relacionando cada macro de la librería donde se encuentran almacenadas todas las funciones y macros que se utilizan para interactuar con el procesador, a partir de operaciones a nivel de bits.

Referencias

1. A. Vera, W. Jimenez & J. Cardona “Fundamentos y Aplicaciones de los Microcontroladores AVR” 1ra edición, Armenia, 2008