

# INFORME PROYECTO COMPTADORAS Y SISTEMAS DE CONTROL

Bourlot Ezequiel, Federico Domoñi, Lorenzo Fernández, Joaquín Vignetta (E.E.S.T.R.Q N°7,  
[ezequielbourlot@impatrq.com](mailto:ezequielbourlot@impatrq.com), [joaquinvignetta@impatrq.com](mailto:joaquinvignetta@impatrq.com),  
[maximolorenzofernandez@impatrq.com](mailto:maximolorenzofernandez@impatrq.com), [federicorodriguezdomoni@gmail.com](mailto:federicorodriguezdomoni@gmail.com) )

## I. INTRODUCCIÓN

El proyecto consiste en un scooter eléctrico que cuenta con velocidad crucero. Este logra esto mediante el uso de una pi pico que va conectada al motor driver para regular la potencia de este. La velocidad se regula mediante un potenciómetro en el tablero del manillar, y la velocidad crucero se logra mediante un botón en el tablero que mantiene una velocidad constante. La velocidad se censa mediante tres imanes colocados en las ruedas, cada vez que detecta tres imanes lo detecta como una vuelta de rueda. La velocidad se regula con un potenciómetro, este se mide y se saca de la PWM, este se encuentra en el tablero del manillar.

## II. ABSTRACT

La placa se encarga de mandar una señal PWM para regular la potencia del motor. La PWM se obtiene mediante una raspberry controlada por un potenciómetro o sensor el cual censa 3 imanes conectado a las ruedas para hacer un cálculo de velocidad

## III. DESCRIPCIÓN FUNCIONAL

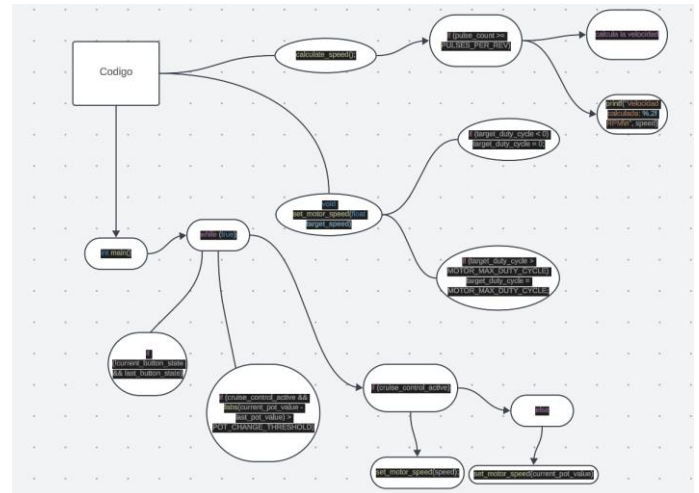
Este scooter funciona aprovechando la raspberry usando el GPIO 15 el cual se puede habilitar como PWM, esta señal se manda hacia una placa la cual cuenta con un transistor PNP y resistencias para evitar que se pueda quemar la raspberry.

La placa sería un driver para permitir conectar la raspberry, la cual estamos usando con 3V3, con un motor cuyo funcionamiento necesitaría de 12V.

Para regular la velocidad tiene dos modos el manual el cual hace un calculo basado en el valor que le manda un potenciómetro, el PWM va subiendo de a poco para evitar una acelerada o frenada de golpe.

La otra modalidad sería la velocidad crucero la cual se activaría al tocar un botón y se desactivaría de la misma manera o al girar el potenciómetro. Esta consiste en censar 3 imanes y hacer un calculo que consiste en que al detectar 3 imanes cuente una vuelta de rueda y dependiendo de las vueltas de rueda que cuenta calcula las revoluciones por minuto. Basado en estas revoluciones calcula la señal PWM que debería mandar para que se mantenga a la velocidad a la que está yendo.

## IV. DIAGRAMA DE CODIGO



(En este diagrama se encuentran las partes principales del código)

## V. DESCRIPCION DEL CIRCUITO

El circuito es un sistema de control para regular la potencia de un motor, tiene el motor, la pico que genera una señal PWM para regular el motor y un MOSFET con drivers para poder controlar con la pico. La PWM se genera basando en mediciones que haga la pico de un potenciómetro o de un sensor de velocidad, se selecciona cual con un botón.

## VI. ALCANCE LOGRADO

El alcance de este proyecto era intentar que un scooter eléctrico pueda andar devuelta y hacerlo funcionar con velocidad crucero mediante la raspberry pi pico, con la mayoría de los materiales y herramientas obtenidos en el taller de aviónica, incluyendo cosas tan importantes como la computadora, placas u otras cosas

## VII. CONCLUSIÓN

A lo largo de este proyecto hemos podido aprender y aplicar nuestro conocimiento visto en la materia y el de otras como la conexión de las baterías en serie, con la

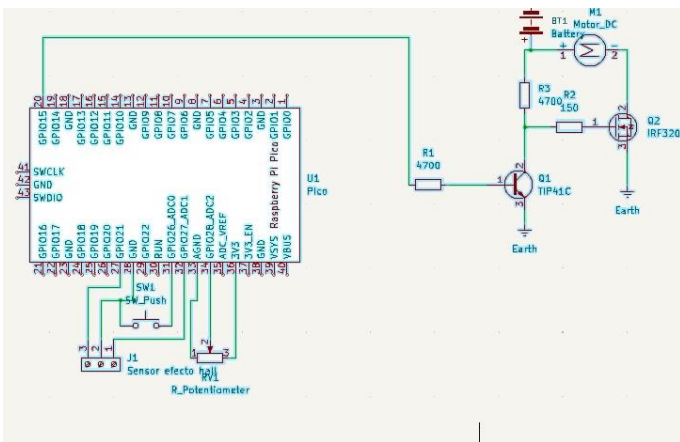
utilización del motor aplicamos prácticas de electrónica de potencia, como la utilización de la PWM para regular la potencia del motor usando una raspberry y el uso de imanes para medir velocidad (revoluciones por minuto).

También queremos destacar el diseño de la PSB y creación de la plaqueta, el uso de transistores para poder conectar la pi pico a la plaqueta o también el uso de soldadora y moledora para poder armar el scooter.

Y por último también hablar de que este proyecto nos ayudo a ejercitar la paciencia ya que hubo mucha prueba y error.

## VIII. ANEXOS

### ESQUEMATICO:



### PCB:

