



**UNIVERSIDAD CATÓLICA DE CUENCA**

*Comunidad Educativa al Servicio del Pueblo*

**UNIDAD ACADÉMICA DE INGENIERÍA**

**INDUSTRIA Y CONSTRUCCIÓN**

**CARRERA DE INGENIERÍA ELECTRÓNICA**

**BLOQUEO DEL SISTEMA DE ENCENDIDO DE UN  
VEHÍCULO ELÉCTRICO, MEDIANTE LA  
IMPLEMENTACIÓN DE UN ALCOHOLÍMETRO.**

**TRABAJO DE TITULACIÓN O PROYECTO DE INTEGRACIÓN  
CURRICULAR PREVIO A LA OBTENCIÓN DEL TÍTULO DE**

**INGENIERO ELECTRÓNICO**

**AUTOR: JOHN MARCELO GONZÁLEZ GONZÁLEZ**

**DIRECTOR: ING. JUAN CARLOS ORTEGA CASTRO**

**AZOGUES - ECUADOR**

**2020**

*Yo me gradué en los  
50 años de La Cato!*

©Copyright John González G  
All Rights reserved

# Bloqueo del sistema de encendido de un vehículo eléctrico, mediante la implementación de un alcoholímetro.

John Marcelo González González<sup>I</sup>  
[jmgonzalezg76@est.ucauce.edu.ec](mailto:jmgonzalezg76@est.ucauce.edu.ec)  
<https://orcid.org/0000-0001-9232-8043>

Juan Carlos Ortega Castro<sup>II</sup>  
[jcortegac@ucacue.edu.ec](mailto:jcortegac@ucacue.edu.ec)  
<https://orcid.org/0000-0001-6496-4325>

**Correspondencia:** [jmgonzalezg76@est.ucacue.edu.ec](mailto:jmgonzalezg76@est.ucacue.edu.ec)

- <sup>I</sup> Egresado de la carrera de Ingeniería Electrónica, Universidad Católica de Cuenca, Cuenca, Ecuador.
- <sup>II</sup> Docente de la carrera de Ingeniería Electrónica, Unidad Académica de Ingeniería, Industria y Construcción, Universidad Católica de Cuenca, Cuenca, Ecuador.
- 

## Resumen

Este artículo presenta un sistema de bloqueo, en el vehículo eléctrico de la Universidad Católica de Cuenca, mediante la implementación de un alcoholímetro, que contribuirá a la disminución de accidentes de tránsito, a través de la limitante generada a la conducción del vehículo por parte de personas en estado de embriaguez. El prototipo desarrollado tributa al proyecto de investigación Smart UniverCity Cato 2.0, además, es inclusivo por tener la capacidad de expandirse en trabajos futuros que mejoren la automatización del vehículo. En su desarrollo, se muestran: un circuito que, al reaccionar a la variable medida por un alcoholímetro, bloquea la energía suministrada hacia el switch de la llave de arranque; y, se explican, a detalle, los procesos a seguir para su correcto funcionamiento. Además, el sistema permite entregar al conductor la información necesaria para conocer el motivo del bloqueo o desbloqueo del vehículo, por medio de un monitor montado en el vehículo y los cambios de las variables en la plataforma ThingSpeak, culminado con la metodología para el desarrollo y programación de los dispositivos de hardware y software utilizados, así como también, las principales conclusiones que se obtuvieron en su ejecución.

**Palabras clave:** Alcoholímetro, vehículo eléctrico, sistema de bloqueo, microcontrolador, programación.

## **Abstract**

This article presents a locking system in the electric vehicle of the Catholic University of Cuenca through the implementation of a breathalyzer, which will contribute to the reduction of traffic accidents, through the limitation generated to the driving of the vehicle by people in a drunken state. The prototype developed contributes to the Smart UniverCity Cato 2.0 research project, additionally, it is inclusive because it has the ability to expand in future works that improve vehicle automation. In its development, a circuit that when reacting to the variable measured by a breathalyzer, blocks the energy supplied to the ignition key switch is shown, as well as are explained in detail the processes for its correct operation. In addition, the system allows the driver to be provided with the necessary information to know the reason for the locking or unlocking of the vehicle, by means of a monitor mounted on it, and the changes of the variables in the ThingSpeak platform, culminating with the methodology for the development and programming of the hardware and software devices used, as well as the main conclusions that were obtained in its execution.

**Keywords:** Breathalyzer, Electric Vehicle, Blocking System, Microcontroller, Programming.

## **CENTRO DE IDIOMAS**

Cuenca, 08 de septiembre de 2020

**EL CENTRO DE IDIOMAS DE LA UNIVERSIDAD CATÓLICA DE CUENCA, CERTIFICA QUE EL DOCUMENTO QUE ANTECEDE FUE TRADUCIDO POR PERSONAL DEL CENTRO PARA LO CUAL DOY FE Y SUSCRIBO**



DR. WLADIMIR QUINCHE  
ORELLANA  
Documento certificado  
digitalmente por Emergencia  
Sanitaria en Ecuador por  
COVID-19  
Centro de Idiomas Matriz: Cuenca  
- Ecuador  
2020-09-09 08:53+19:00

**Dr. Wladimir Quinche Orellana Msc.  
SECRETARIO CENTRO DE IDIOMAS**

## **Introducción**

El problema del consumo excesivo del alcohol se presenta en la sociedad de la ciudad de Cuenca y sus alrededores. Este problema involucra factores que llegan a afectar el bienestar personal y familiar del consumidor. Sin embargo, el alcance del problema crece cuando la persona que consume alcohol decide conducir bajo los efectos del mismo, estos generan accidentes de tránsito y siniestros viales. La Agencia Nacional de Tránsito determinó en el año 2014, a la conducción en estado de embriaguez como la principal causa de siniestralidad vial en el país. Este hecho presenta una tendencia ascendente en el Ecuador debido a los índices de motorización y aumento de la población (Gómez García et al., 2018).

El Código Orgánico Integral Penal COIP, sanciona a los infractores de tránsito con pena privativa de libertad, multa económica y rebaja de puntos en la licencia de conducir. Las multas aplicadas empiezan cuando el conductor del vehículo iguala o supera el valor de 0.3 gramos de alcohol por litro de sangre (Asamblea Nacional Constituyente et al., 2014). Estos controles no son efectivos en su totalidad, pues estos se los aplican una vez que el vehículo ya estaba en circulación, también, pueden ser evitados al circular por una ubicación diferente a donde se los aplica. La efectividad del control aumenta cuando el bloqueo se lo aplica antes de poner en marcha al vehículo

La revisión de publicaciones similares, (Casanova Vásquez, 2014) y (Cujano Reyes, 2011), en otras provincias, indican resultados positivos en la implementación de proyectos de la misma categoría. En estas se destaca la utilización de microcontroladores, sensores y actuadores como una opción para limitar la conducción en estado de embriaguez.

La publicación (Casanova Vásquez, 2014) presenta la implementación de un alcoholímetro en una camioneta Mazda BT-50. Esta permite que los conductores se limiten a conducir cuando hayan ingerido bebidas alcohólicas. El análisis económico del proyecto resulta aceptable, y existe una posibilidad bastante aceptable para abrir este proyecto hacia un mercado.

La publicación (Cujano Reyes, 2011) demuestra que mediante el uso de: comandos AT, un microcontrolador y la red telefónica celular, se logra activar el bloqueo y desbloqueo de un automóvil. Este proyecto se sujeta a los instrumentos antes mencionados, utilizando la tecnología de envío de mensajes de texto SMS, consigue inmovilizar un automóvil mediante un teléfono celular.

La tecnología actual permite una lectura inmediata del nivel de etanol en el aire exhalado por la persona, siendo este dato utilizado según sea conveniente.

El objetivo de este proyecto consistió en eliminar el número de ocasiones en las cuales el vehículo eléctrico sea conducido por una persona que esté bajo los efectos del alcohol. Para esto se desarrolló un bloqueo aplicado en el vehículo, cuando este detecte que el aliento del conductor superó el valor de gramos de alcohol por litro de sangre permitido, el vehículo entrará a una fase de reposo, así este no puede ser puesto en circulación.

El presente proyecto se llevó a cabo mediante el desarrollo de las siguientes fases:

- Revisión de las leyes que implica el tránsito de vehículos con conductores en estado etílico.
- Diseño y construcción de un circuito electrónico que permita el bloqueo del encendido de un vehículo al detectar niveles pre establecidos de alcohol.
- Programación de microcontroladores y microprocesadores.
- Implementación del proyecto en el vehículo eléctrico de la Universidad Católica de Cuenca.
- Prueba y análisis de resultados obtenidos.

## Desarrollo

### Arquitectura del sistema.

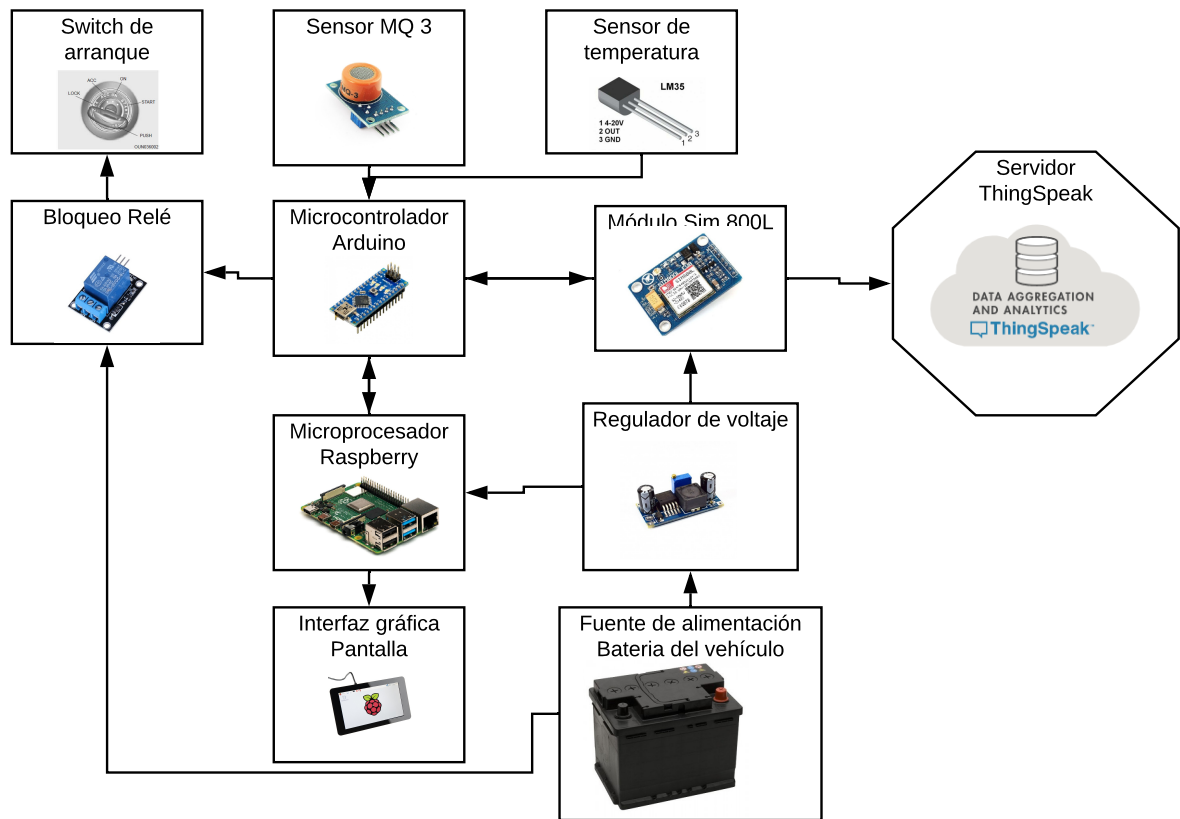


Fig. 1: Arquitectura del sistema.

Fuente: Autor

En la Fig. 1 se puede observar la arquitectura del sistema, en esta se integra la recepción de información analógica a través del sensor MQ 3, la cual se transporta hacia el microcontrolador Arduino, esta es evaluada, para transmitir los resultados hacia el Sim 800L y el Raspberry.

Las ordenes ejecutadas por el módulo y el microprocesador permiten el envío de los resultados hacia el servidor ThingSpeak y una interfaz gráfica mostrada en la pantalla del circuito.

La alimentación eléctrica es suplida por la batería del vehículo, la cual es convertida por un regulador de voltaje para que se pueda alimentar al Sim 800L y el Raspberry, este ultimo es el encargado de dar alimentación al Arduino, pantalla, sensor MQ 3 y relé de bloqueo.

Adicionalmente, al circuito principal se agrega la medición de sensores. La información generada por estos dispositivos es mostrada en la pantalla principal generada por Raspberry.

El sensor MQ 3 es un detector de gases como benceno, metano, hexano, etanol, etc. Sin embargo, registra su mayor sensibilidad al ser expuesto ante alcohol (Cytron Technologies, n.d.).

### **Alimentación del circuito.**

El circuito es alimentado por la corriente suministrada por la batería principal del vehículo, esta tiene un voltaje de 12 voltios, la cual es convertida a un valor de 5 voltios para no dañar componentes relacionados al circuito, esto se realiza por medio del integrado LM2596, este es un regulador de voltaje, que entrega un voltaje constante e inferior al proveído (Naylamp, 2018).

El regulador de voltaje entrega la corriente en el conector usb de Raspberry, este para generar el encendido del mismo, de la pantalla, de arduino y el sensor MQ 3, y en el puerto de entrada de voltaje del módulo Sim 800L.

### **Programación por bloques en Arduino.**

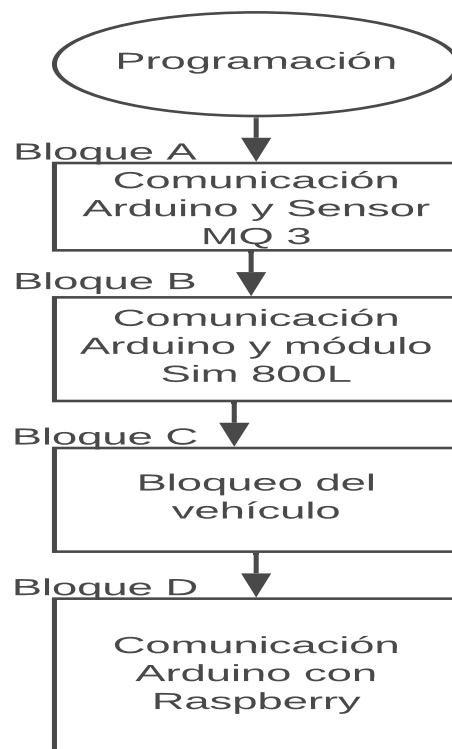


Fig. 2: Bloques de programación en Arduino

Fuente: Autor



### Bloque A, comunicación Arduino y sensor MQ 3.

La programación para lograr la comunicación entre Arduino y el sensor MQ 3 empieza por declarar la variable, en la cual se muestra el valor obtenido por dicho sensor. Luego, el dato que se receipta, es pasado por una operación aritmética para sacar un valor en voltaje correspondiente a la cantidad de mg/l de un alcoholímetro comercial.

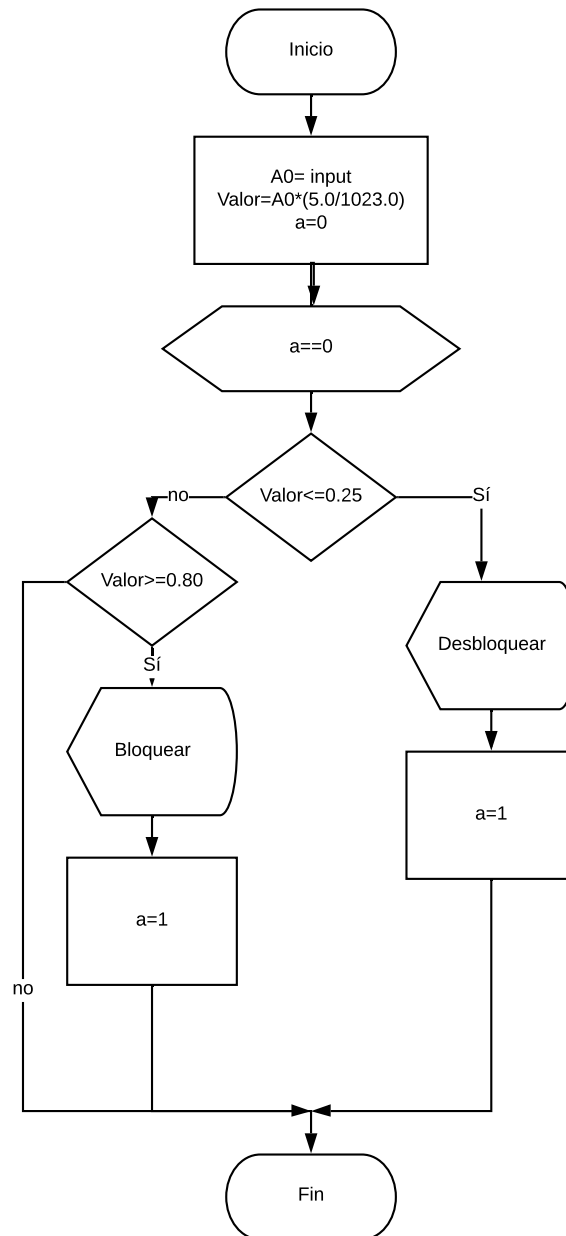


Fig. 3: Diagrama de flujo de datos, referente a la programación de Arduino con el sensor MQ-3

Fuente: Autor

La programación de este apartado corresponde al bloque A del código. Arduino, después de recibir la información del sensor MQ-3, la compara con los requerimientos y arroja 3 resultados. Los diferentes resultados indican lo siguiente:

Resultado 1: cuando el valor obtenido del sensor corresponda un voltaje menor de 0.8 V y mayor a 0.25, el bloque D de la programación pasa a ejecutar una función, en la cual, el sistema se mantendrá en una etapa de espera, hasta ser bloqueado o desbloqueado por algún usuario. Este resultado se lo obtiene, por lo general, cuando el sensor esta en una etapa de reposo, es decir, ningún usuario está depositando su aliento sobre el sensor.

Resultado 2: cuando el valor obtenido por el sensor corresponda a un voltaje mayor a 0.8, el bloque B, C y D de la programación ejecutarán funciones que correspondan al bloqueo del sistema. Estos valores del sensor se los obtiene cuando el usuario realice la medición, y este contiene una cantidad de alcohol por litro de sangre mayor a 0.3 mg/l, esta representada en voltaje correspondería a 0.8 V.

Resultado 3: cuando el valor obtenido por el sensor corresponda a un voltaje menor a 0.3, el bloque D de la programación adopta la ejecución de una función correspondiente al desbloqueo del sistema. Este resultado se lo obtiene cuando al comprobar la cantidad de alcohol por litro de sangre en el aliento depositado por el usuario, este no alcanza el valor de 0.3 mg/l.

La calibración del sensor MQ 3 se realiza por medio del ajuste de un potenciómetro integrado en el mismo, este se posicionó en el ángulo requerido para que los valores indicados en la siguiente tabla estén a conveniencia del circuito. Las mediciones tomadas en comparación con el Medidor de alcohol digital, que posee un rango de detección de : 0.000-0.199% BAC(gramos de alcohol por cada 100 ml de sangre, o 210 litros de aliento.(Terrones, 2019)) y 0.00-0.99 mg/L(miligramos por litro. (Terrones, 2019)) y una certificación: RoHS "Restriction of Hazardous Substances" "Restricción de sustancias peligrosas" (*Useful and Professional Manufacture Digital Alcohol Testers Breathalyzer Alcohol Breath Testers - China Alcohol Testers, Breathalyzer | Made-in-China.Com*, n.d.), son las siguientes:

Valor medidor de alcohol digital	Valor sensor MQ 3
Valores menores a 0.25 mg/l	Valores de voltaje entre 0.10 y 0.30
Valores mayores a 0.30 mg/l	Valores de voltaje entre 0.80 y 1.10
Valores entre 0.25 mg/l y 0.29 mg/l	Valores de voltaje entre 0.31 y 0.79

Tabla 1: Comparación de mediciones tomadas del Medidor de alcohol digital y el sensor MQ 3.

Cabe recalcar que, los valores de voltaje obtenidos del sensor MQ 3 que estén entre 0.31 y 0.79, los mismos que corresponden a 0.25 mg/l y 0.30 mg/l en el medidor de alcohol digital, no ejecutan ninguna función en el sistema, pero, al tomar una prueba luego de haber pasado un tiempo de 5 minutos, el valor de la medición bajara o subirá, dependiendo de si el conductor ingirió más bebidas alcohólicas. Esta zona de valores que no ejecutan funciones en el circuito es necesaria para descartar valores falsos provocados por el aire del ambiente en que se encuentra.

### **Bloque B, comunicación Arduino y módulo Sim 8001.**

La ejecución del bloque B de la programación se relaciona directamente con el resultado 2 del bloque A. Este apartado, luego de recibir la orden del bloque anterior, ejecuta una función, con el nombre "envioAT", que ordena el envío y recepción de información por medio de los puertos tx y rx de Arduino y del Sim 8001. La comunicación es bidireccional, ya que el microcontrolador envía los comandos, y el módulo devuelve las respuestas a los mismos.

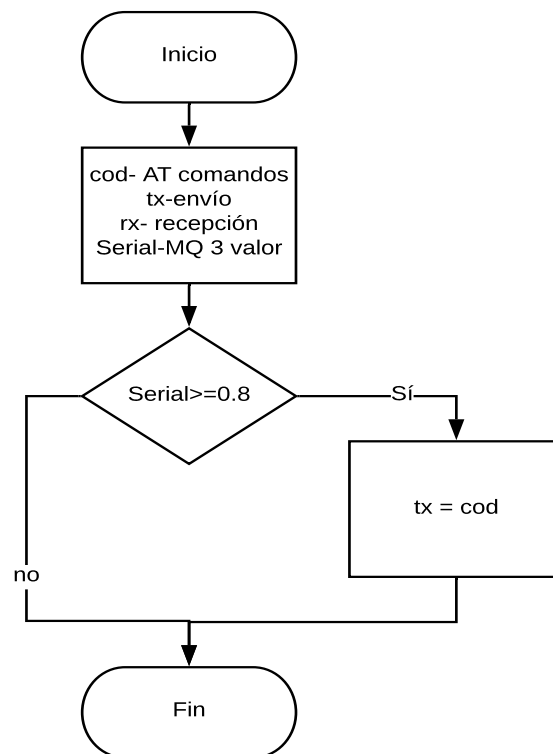


Fig. 4: Diagrama de flujo de datos, referente a la programación de Arduino con el módulo Sim 8001.

Fuente: Autor

### **Bloque C, bloqueo del vehículo.**

El desarrollo del bloque C de la programación recepta la orden emitida por el valor leído en el resultado 2 y 3 del bloque A, luego, una función ejecuta la interacción de un puerto digital del microcontrolador, el cual está conectado a un relé que corta, o deja pasar la alimentación suministrada por la batería del vehículo hacia los instrumentos del tablero del vehículo.

Al encender el circuito, dicho relé se mantiene en el estado abierto, es decir, el vehículo no podrá ser encendido sin antes recibir la sentencia ejecutada por el resultado 3 del bloque A. También, en el caso que la sentencia que se recepte sea la del resultado 2 del bloque de programación A, se mantendrá el relé en el estado abierto, así los instrumentos del tablero, necesarios para el encendido del vehículo, no tendrán alimentación eléctrica.

### **Bloque D, comunicación Arduino con Raspberry.**

La comunicación entre Arduino y Raspberry se realiza por medio de la clase serial, ubicada dentro de la librería "Serial" del lenguaje de programación Python (Antonio & Guerrero, 2017). Esta librería permite mantener información en los puertos seriales del microcontrolador y microprocesador, pudiendo ser ocupada en el momento que se requiera. La utilización de la librería se divide en dos partes:

#### *Librería Serial en Arduino:*

La utilidad de esta librería permite, luego de recibir las ordenes ejecutadas en los resultados 2 y 3 del bloque A de la programación, enviar hacia el puerto serial dos textos, útiles para el desarrollo del proceso en Raspberry.

La primera orden consta de enviar un texto que contiene información necesaria para el desbloqueo de la pantalla, la misma que esta conectada por medio del puerto HDMI con el microcontrolador.

La segunda orden ejecuta el envío de un texto que contiene información que, al ser verificada por el microprocesador, desbloqueará la pantalla implementada en el vehículo.

#### *Librería Serial en Raspberry:*

Esta librería se utiliza para la lectura del puerto serial, el cual contiene la información enviada por el microcontrolador. Antes de leer el puerto serial de Raspberry se mantendrá una pantalla de bloqueo en el vehículo.

Al leer la primera orden, enviada desde Arduino, se oculta la pantalla de bloqueo y por siguiente se muestra una pantalla temporal, durante 5 segundos, la cual indica que el vehículo ah sido desbloqueado. Finalmente, se observa una pantalla que indica que el vehículo está encendido y puede circular.

Al leer la segunda orden, se deshabilita la visibilidad de la pantalla de bloqueo y se muestra otra que indica que se ah superado el nivel de alcohol aceptado para desbloquear el vehículo, por lo tanto, el vehículo está bloqueado hasta que se apague el vehículo y sea puesto en funcionamiento posteriormente. También, se informa que la cantidad medida por el sensor fue enviada al servidor ThingSpeak.

Adicionalmente, en Raspberry se utiliza la librería threading”, la cual suma la función de realizar cualquier otro proceso en el microprocesador, y esta se desarrolle al mismo tiempo que los comandos escenciales, alojados en la función principal de la programación. Con esta librería se logra la solicitud, por parte de Raspberry hacia Arduino, de información adicional, esta es mostrada en la pantalla principal de la interfaz gráfica.

### Monitorización de los datos en el servidor.

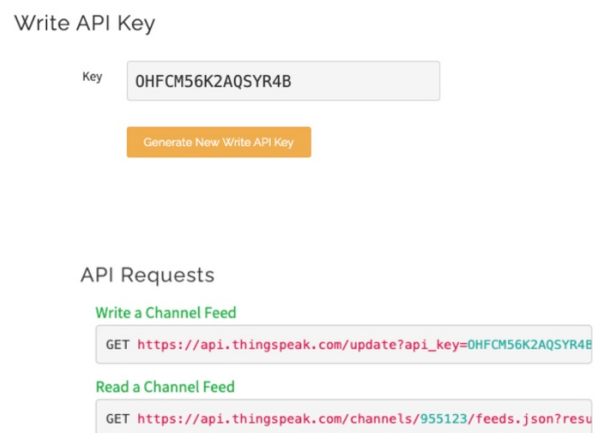


Fig. 5: Información proporcionada por el servidor ThingSpeak.

Fuente: Autor

El envío de los datos, hacia el servidor, se lo desarrolla por parte del Arduino, el cual manda comandos AT hacia el módulo Sim 8001 para que este los ejecute. Entre estos se encuentra el comando "AT+CIPSEND", el cual ejecuta el envío de un dato medido por el sensor MQ 3, en este caso alojado en una variable, hacia ThingSpeak. El comando mencionado anteriormente

necesita la ruta hacia la pagina y la clave API, proporcionada por el servidor, esta indicada en la Fig. 5, para ejecutarse de manera correcta.

### Interfaz gráfica.

Para la creación de la interfaz gráfica se utiliza Python, este es un lenguaje de programación utilizado para lectura y escritura de variables, la disponibilidad de bibliotecas agilitan el desarrollo de proyectos (Van Rossum, 2009). En este caso, la interfaz gráfica se desarrolla con un binding de la biblioteca grafica Tcl/Tk para Python llamada Tkinter, esta es una interfaz estándar de Python, viene pre instalada y se utiliza para desarrollar interfaces gráficas mediante programación de scrips. Esta desarrolla la interfaz mediante programación y configuración orientada a objetos. La programación está desarrollada por hilos, así, distintos procesos pueden ejecutarse al mismo tiempo sin pausas por los diferentes tiempos aplicados.

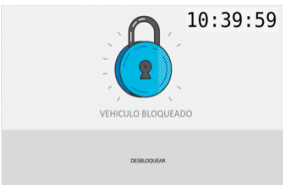


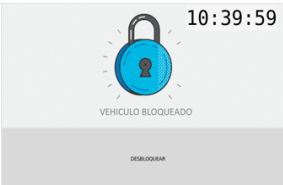

Medidor de alcohol digital.	Sensor MQ 3.	Texto enviado por el Arduino hacia Raspberry.	Orden ejecutada por Raspberry.	Primera pantalla mostrada.	Pantalla final mostrada.
Valor $\leq 0.25$ mg/l	Valor $\geq 0.1$ Valor $\leq 0.3$	Desbloqueo	Desbloqueo		
Valor $\geq 0.25$ Valor $\leq 0.29$	Valor $\geq 0.31$ Valor $\leq 0.79$		Espera		
Valor $\geq 0.30$	Valor $\geq 0.80$ Valor $\leq 1.10$	Bloqueo	Bloqueo		

Tabla 2: Secuencias de pantallas de la interfaz, según la variable leída por el sensor MQ 3.

Fuente: Autor

La Tabla 2, explica el funcionamiento de la interfaz de acuerdo a la información obtenida por el sensor MQ 3, esta información esta relacionada con el medidor de alcohol digital, este adquirido comercialmente.

La interfaz, generada de acuerdo a la información enviada por Arduino, se divide en tres etapas que son: espera, desbloqueo y bloqueo.

#### *Etapas de Espera.*

La etapa de espera es la etapa inicial donde la interfaz permanece en una prórroga hasta que el valor, enviado por Arduino, cambie este estado. Los procesos que se ejecutan en esta etapa reciben los datos del microcontrolador, estos datos indican si bloquear el vehículo cuando la medición del sensor MQ 3 supere el valor de 0.8, o desbloquear el vehículo si la lectura es menor a 0.30 V.

Para cambiar los estados de la interfaz se debe pulsar el botón desbloquear y luego proceder a exhalar el aliento en el sensor de alcohol.



Fig. 6: Primera pantalla (Orden Desbloqueo, Espera, Bloqueo).

Fuente: Autor

#### *Etapas de Bloqueo.*

La etapa de bloqueo surge luego de soplar sobre el sensor de alcohol y marcar una cantidad mayor a la establecida, los procesos que se incluyen este apartado son los del cambio de pantalla, desde la pantalla inicial, indicada en la Fig. 6, hacia la pantalla final, indicada a continuación.





Fig. 7: Pantalla final (Orden Bloqueo).

Fuente: Autor

#### *Etapa de Desbloqueo.*

La etapa de desbloqueo surge cuando la medición del sensor sea menor a 0.3, los procesos que intervienen en esta etapa son el cambio de pantalla, desde la primera pantalla, indicada en la Fig. 6, hacia la pantalla de desbloqueo, mostrada a continuación.



Fig. 8: Pantalla final (Orden Desbloqueo).

Fuente: Autor



En la Fig. 8, se puede visualizar una interfaz amigable para el usuario, en la cual se identifica dos botones, uno con la figura de un termómetro y otro con la figura de una batería, estas, al presionarlos, activan un cuadro de texto, ubicado en la parte superior de los botones, en estos se muestra información proporcionada por sensores ajenos al circuito, en estos se visualiza la temperatura en la que se encuentra la cabina y la cantidad de voltaje proporcionada por la batería del vehículo. También, se visualiza la hora en la que se encuentra, información de paginas web de la universidad Católica de Cuenca y su logo.

### **Protecciones de los circuitos.**

Para el primer encapsulado se tomo en cuenta las medidas de los componentes, y la ventilación que necesita cada uno de estos, dichos componentes son: el Raspberry, pantalla y cables.

Para la fabricación del segundo encapsulado, también, se tomo en cuenta la medida, y la ventilación necesaria para los dispositivos que este va a abarcar, estos son: Arduino, módulo Sim 8001 y el sensor MQ 3.

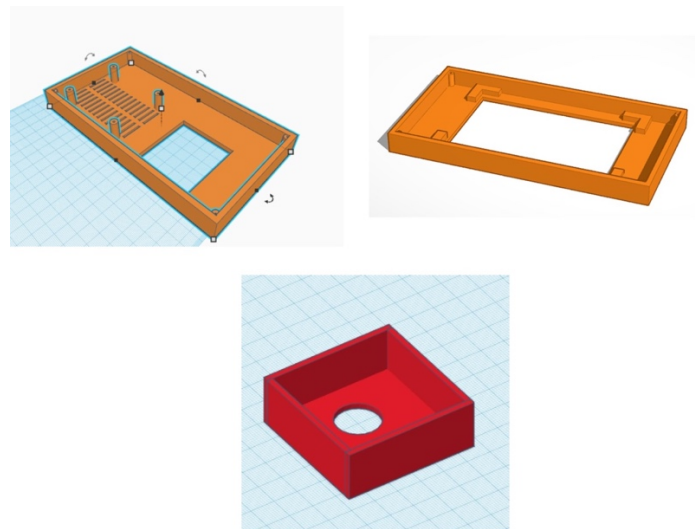


Fig. 9: Encapsulados: Primero (color tomate) y Segundo (color rojo).

Fuente: Autor

## Circuito implementado.



Fig. 10: Circuito en etapa de desbloqueo.

Fuente: Autor



Fig. 11: Circuito desbloqueado, vehículo encendido.

Fuente: Autor

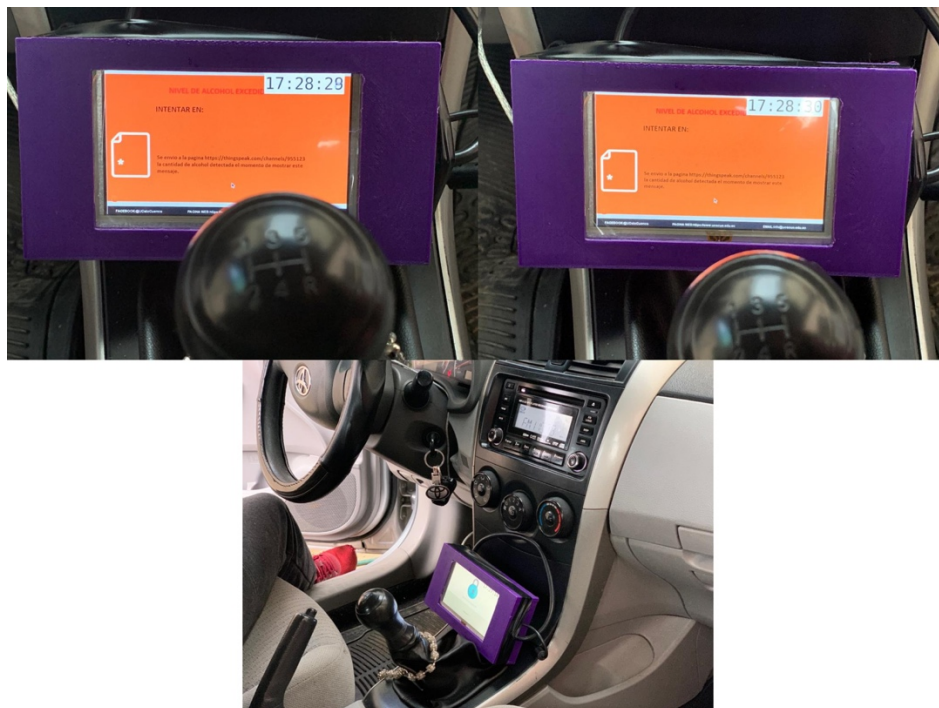


Fig. 12: Circuito en etapa de bloqueo.

Fuente: Autor



Fig. 13: Circuito del sensor MQ 3 implementado.

Fuente: Autor

## Esquema electrónico del sistema.

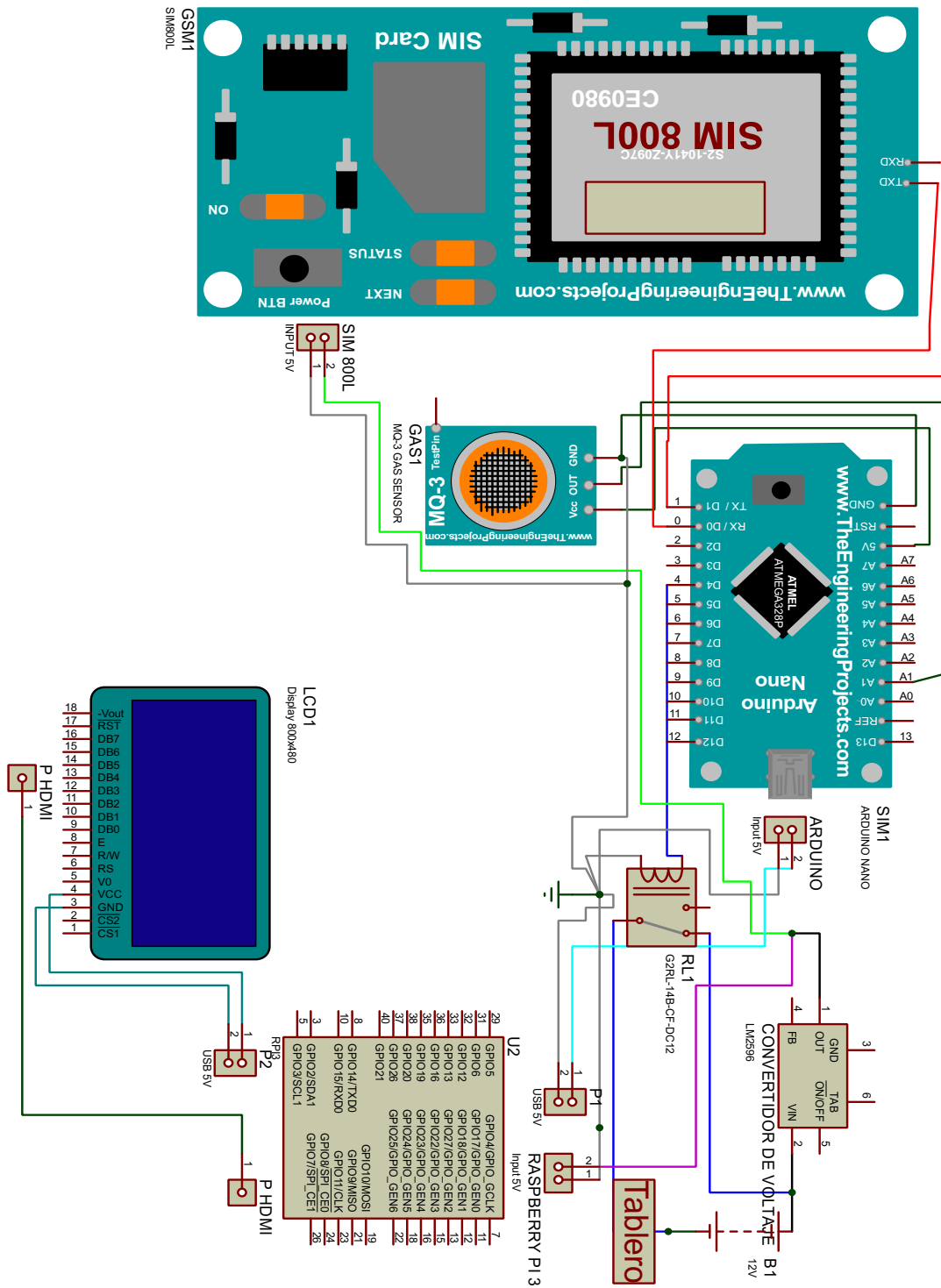


Fig. 14: Circuito del proyecto.

Fuente: Autor

# Resultados

## Sensor MQ-3.

Los resultados, al ser puesto en funcionamiento el sensor con Arduino, muestran una lectura lineal mientras no se detecte ninguna interacción con el sensor. La variable del microcontrolador obtiene valores menores a 0.3V cuando el aliento exhalado en el sensor no presenta residuos de alcohol. Esto indica, mediante la programación, el desbloqueo del circuito. Por otra parte, al interactuar el sensor con un aliento que contiene alcohol, se obtienen valores en las variables, mayores a 0.8V, estas establecidos en la programación, se indica que el circuito pasa a la etapa de bloqueo.

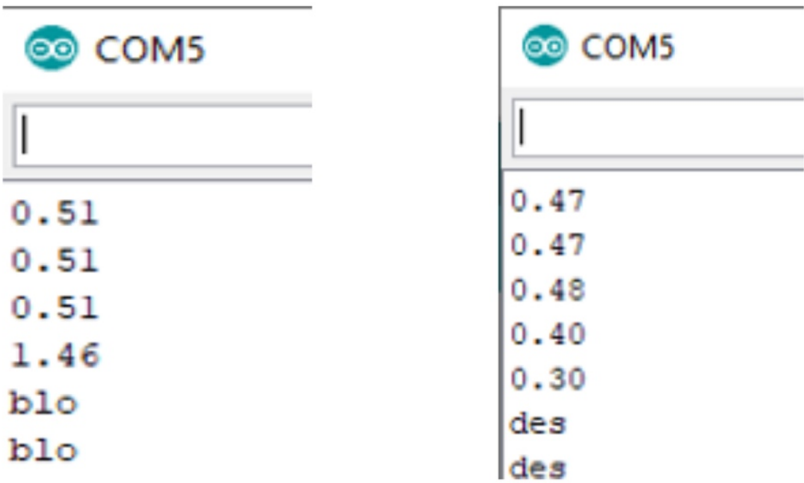


Fig. 15: Lecturas tomadas del puerto serial de Arduino.

Fuente: Autor

## Interacción circuito en el vehículo con ThingSpeak

Al bloquear el circuito, mediante la detección de alcohol en el sensor conectado a Arduino, se activan una serie de procesos, entre los cuales están: el envío hacia ThingSpeak de la variable correspondiente a la lectura del sensor MQ.



Fig. 16: Lectura de variable en ThingSpeak.

Fuente: Autor

*Tabla de resultados obtenidos.*

Para la toma de resultados de aplico el proyecto en 4 personas, estas con las siguientes características:

- Persona 1 Stalin, Hombre, 75 kg edad 22 años.
- Persona 2 Patricia, Mujer, 55 kg edad 20 años
- Persona 3 Víctor, Hombre, 80 kg edad 46 años
- Persona 4 Isabel, Mujer, 72 kg edad 43 años.

Al aplicar la medición se puso a prueba 2 tipos de licor en 2 cantidades diferentes. Por lo tanto, al clasificar estas características se tienen 4 bebidas "B".

B1.- 100 ml de cerveza con 4.2 grados de alcohol en su composición.

B2.- 400 ml de cerveza con 4.2 grados de alcohol en su composición.

B3.-50 ml de ron con 35 grados de alcohol en su composición.

B4.- 100 ml de ron con 35 grados de alcohol en su composición

Luego de exponer las variables implicadas en las mediciones, estos son los resultados.

	Tipo de bebida.	Tiempo luego de ingerir la bebida.	Valor en Medidor de alcohol digital (mg/l).	Valor en Sensor MQ 3 (V).	Imagen mostrada en Interfaz.	Envío de variable hacia ThngSpeak.	Bloqueo del vehículo.
Persona 1	Ninguna.	0 minutos	0	0.10	Fig. 8	No	No
Persona 2	Ninguna	0 minutos	0	0.8	Fig. 8	No	No
Persona 3	Ninguna	0 minutos	0	0,8	Fig. 8	No	No
Persona 4	Ninguna	0 minutos	0	0,9	Fig. 8	No	No
Persona 1	B1	5 minutos	0.17	0.15	Fig. 8	No	No
Persona 2	B1	5 minutos	0.32	0.80	Fig. 7	Si	Si
Persona 3	B1	5 minutos	0.20	0.22	Fig. 8	No	No
Persona 4	B1	5 minutos	0.35	0.82	Fig. 7	Si	Si
Persona 1	B2	60 minutos	0.45	0.90	Fig. 7	Si	Si
Persona 2	B2	60 minutos	0.67	0.98	Fig. 7	Si	Si
Persona 3	B2	60 minutos	0.56	0.73	Fig. 7	Si	Si
Persona 4	B2	60 minutos	0.70	1.02	Fig. 7	Si	Si
Persona 1	B3	5 minutos	0.13	0.14	Fig. 8	No	No
Persona 2	B3	5 minutos	0.20	0.17	Fig. 8	No	No
Persona 3	B3	5 minutos	0.14	0.16	Fig. 8	No	No
Persona 4	B3	5 minutos	0.19	0.21	Fig. 8	No	No
Persona 1	B4	10 minutos	0.30	0.81	Fig. 7	Si	Si
Persona 2	B4	10 minutos	0.42	0.87	Fig. 7	Si	Si
Persona 3	B4	10 minutos	0.28	0.56	Fig. 8	No	Si
Persona 4	B4	10 minutos	0.34	0.85	Fig. 7	Si	Si

Tabla 3: Resultados obtenidos en el proyecto.

Fuente: Autor

En la Tabla 3, se visualiza todos los resultados correspondientes al funcionamiento del proyecto. Entre estos se obtvo un envío satisfactorio de la información hacia el servidor ThngSpeak y el bloqueo del vehiculo, mediante la activación del relé que corta el suministro eléctrico del tablero del vehículo.

## **Conclusiones**

Las visualizaciones de las variables obtenidas se facilitan al observarlas mediante una interfaz gráfica, y, el procesamiento de las mismas, se manejan de mejor manera al ser enviadas hacia un ambiente de internet de las cosas (IoT), como lo es ThingSpeak.

Para el correcto encendido de Raspberry, Arduino y demás dispositivos del proyecto, se disminuye el voltaje de 12 a 5 voltios dc, así se evita el sobrecalentamientos y fallas en el circuito.

Se investigaron diferentes alternativas para el desarrollo de la aplicación, siendo Arduino la plataforma que posee mejores prestaciones, en virtud de que cuenta con características que ayudan al manejo de las variables, sensores y actuadores, actuando bidireccionalmente en la dotación de datos. Tambien, el sensor utilizado presta un funcionamiento muy optimo para este proyecto, este puede tiene la ventaja de modificar su sensibilidad, pudiendo asi calibrarlo para obtener resultados similares a un alcoholímetro comercial.

El circuito, en su apartado de bloqueo, corta la energia en el interruptor electrico montado en el tablero del mismo, esta decisión facilita el montaje de este proyecto en diferentes vehiculos, ya que la mayoría de vehiculos pueden bloquearse al cortarles el suministro electrico que acciona el arranque principal.

En el sistema operativo Android la aplicación propia de ThingSpeak permite: la visualización de datos, editar los gadgets disponibles y ejecutar alarmas en la pantalla principal del dispositivo. En su similar, IOS, no existe una aplicación oficial de ThingSpeak, por lo tanto, se utilizan aplicaciones de terceros en las cuales existen aplicaciones de pago y gratuitas, para fines de este trabajo se utilizó las gratuitas, en las cuales solo permite la visualización de valores sin dejar que se altere ninguna. En sistemas operativos como Windows o macOS las aplicaciones disponen de los mismos beneficios que en Android



## Referencias Bibliográficas

- Antonio, F., & Guerrero, G. (2017). *Vehículo autónomo para gestión de trayectorias basado en Arduino y Raspberry*. <https://riuma.uma.es/xmlui/handle/10630/12800>
- Asamblea Nacional Constituyente, Pucuna, S. Y., Geovanny, B., Cajo, H., & Fallis, A. . (2014). CÓDIGO ORGÁNICO INTEGRAL PENAL Año I-Nº 180. In *Código Orgánico Integral Penal*. (Vol. 53, Issue 9). <https://doi.org/10.1017/CBO9781107415324.004>
- Casanova Vásquez, M. P. (2014). *Diseño, construcción e instalación de un alcoholímetro electrónico con dispositivo de bloqueo de un vehículo*. <http://dspace.esPOCH.edu.ec/handle/123456789/3732>
- Cujano Reyes, J. P. (2011). Diseño e Implementación de un Sistema de Control Electrónico para Bloquear / Desbloquear Automóviles Mediante Mensajes MSM Encriptados. *Tesis*. <http://dspace.esPOCH.edu.ec/handle/123456789/612>
- Cytron Technologies. (n.d.). *MQ3 Alcohol Sensor Module*. Retrieved August 25, 2020, from <https://components101.com/sensors/mq-3-alcohol-gas-sensor>
- Gómez García, A. R., Ayala Heredia, M., & Campos Villalta, Y. Y. (2018). Caracterización de 1.967 casos de fallecimientos por accidentes de tránsito en Ecuador. *Espiraes Revista Multidisciplinaria de Investigación*, 2(21). <https://doi.org/10.31876/re.v2i21.342>
- Naylamp. (2018). *Convertidor Voltaje DC-DC Step-Down 3A LM2596 - Naylamp Mechatronics - Perú*. <https://naylampmechatronics.com/conversores-dc-dc/196-convertidor-voltaje-dc-dc-step-down-3a-lm2596.html>
- Terrones, J. (2019). “*Diseño e implementación de un sistema de monitoreo y control operativo de alcoholemia para la empresa Galvanoplastia Yhabal S.A.C.*”
- Useful and Professional Manufacture Digital Alcohol Testers Breathalyzer Alcohol Breath Testers - China Alcohol Testers, Breathalyzer | Made-in-China.com*. (n.d.). Retrieved July 8, 2020, from <https://m.made-in-china.com/product/Useful-and-Professional-Manufacture-Digital-Alcohol-Testers-Breathalyzer-Alcohol-Breath-Testers-881234935.html>
- Van Rossum, G. (2009). *Tutorial Python version 2.6.2*. <http://ralsina.me/static/tutorial-8.pdf>

# ARTÍCULO\_JHON\_GONZÁLEZ

*por Jhon González*

---

**Fecha de entrega:** 02-sep-2020 09:15p.m. (UTC-0500)

**Identificador de la entrega:** 1378617958

**Nombre del archivo:** Art\_culo\_Gonz\_lez.pdf (3.11M)

**Total de palabras:** 4790

**Total de caracteres:** 25609

# ARTÍCULO\_JHON\_GONZÁLEZ

## INFORME DE ORIGINALIDAD

5%

INDICE DE SIMILITUD

5%

FUENTES DE  
INTERNET

1%

PUBLICACIONES

2%

TRABAJOS DEL  
ESTUDIANTE

## FUENTES PRIMARIAS

1

[biblioteca.esPOCH.edu.ec](http://biblioteca.esPOCH.edu.ec)

Fuente de Internet

1%

2

[repositorio.uancv.edu.pe](http://repositorio.uancv.edu.pe)

Fuente de Internet

1%

3

[naylampmechatronics.com](http://naylampmechatronics.com)

Fuente de Internet

<1%

4

Submitted to Universidad Ricardo Palma

Trabajo del estudiante

<1%

5

[www.bcnet.upc.es](http://www.bcnet.upc.es)

Fuente de Internet

<1%

6

[pt.scribd.com](http://pt.scribd.com)

Fuente de Internet

<1%

7

Submitted to Universidad Cesar Vallejo

Trabajo del estudiante

<1%

8

[www.coursehero.com](http://www.coursehero.com)

Fuente de Internet

<1%

9

[openaccess.uoc.edu](http://openaccess.uoc.edu)

Fuente de Internet

<1%

10	<a href="http://www.revistaespirales.com">www.revistaespirales.com</a> Fuente de Internet	<1 %
11	<a href="http://dspace.ucuenca.edu.ec">dspace.ucuenca.edu.ec</a> Fuente de Internet	<1 %
12	<a href="http://docplayer.com.br">docplayer.com.br</a> Fuente de Internet	<1 %
13	<a href="http://dspace.esPOCH.edu.ec">dspace.esPOCH.edu.ec</a> Fuente de Internet	<1 %
14	<a href="http://busca.com">busca.com</a> Fuente de Internet	<1 %
15	<a href="http://www.mazdatres.com">www.mazdatres.com</a> Fuente de Internet	<1 %
16	<a href="http://pcarloscrespi.galeon.com">pcarloscrespi.galeon.com</a> Fuente de Internet	<1 %
17	"XXV IUFRO World Congress: Forest Research and Cooperation for Sustainable", XXV IUFRO World Congress: Forest Research and Cooperation for Sustainable, 2019 Publicación	<1 %

Excluir citas

Apagado

Excluir coincidencias

Apagado

Excluir bibliografía

Apagado

## UNIDAD ACADÉMICA DE INGENIERÍA, INDUSTRIA Y CONSTRUCCIÓN-CARRERA DE INGENIERÍA ELECTRÓNICA

Oficio Nro. UCACUE-SA-UAIIC-IE -2020-057-OFC  
Azogues, 07 de septiembre de 2020

Asunto: Solicitud de baja de Tutorías de Trabajos de Titulación

Señor Ingeniero  
Nelson Federico Córdova González, Msc  
**DECANO DE LA UNIDAD ACADÉMICA DE INGENIERÍA, INDUSTRIA Y  
CONSTRUCCIÓN**  
Presente.

De mi consideración:

Con un cordial y atento saludo, Por medio de la presente, me permito comunicar que el Ingeniero Juan Carlos Ortega Castro, ha solicitado la baja de las tutorías que se encontraban a su cargo, documento que adjunto para que sea conocido en Sesión de Consejo Directivo y se pueda tomar la resolución más acertada para que los estudiantes que estaban bajo su tutoría puedan continuar con sus trabajos de Investigación sin interrupciones.

Por la atención que sabrá dar a este requerimiento, le presento mi agradecimiento.

Atentamente,  
**DIOS, PATRIA, CULTURA Y DESARROLLO**  
**AÑO JUBILAR, QUINCUAGÉSIMO ANIVERSARIO FUNDACIONAL**

Ing. Jaime Rojas Coronel. Msc  
**DIRECTOR DE LA CARRERA DE INGENIERÍA ELECTRÓNICA**

Elaborado por:	Eco. Nancy Peralta I.	
Aprobado por:	Ing. Jaime Rojas C.	

## El Bibliotecario de la Sede Azogues

### CERTIFICA:

Que: **GONZALEZ GONZALEZ JOHN MARCELO**, con cédula de ciudadanía Nro. **0302138276**, de la Carrera de: **INGENIERÍA ELECTRÓNICA**

No adeuda libros, a esta fecha: **3 de septiembre del 2020**.



Byron Alonso Torres Romo  
**Bibliotecario**

Biblioteca Universitaria  
MONS. "FROILAN POZO QUEVEDO"