LATIN-AMERICAN JOURNAL OF COMPUTING (LAJC), Vol VIII, Issue 1, January 2021

Vehículo Eléctrico con Algoritmo de Control de Velocidad y Freno Regenerativo y Diseño de una Aplicación Web Móvil Basada en IoT

Electric Vehicle with Speed Control Algorithm and Regenerative Braking and the Design of a Mobile Web App based on IoT

ARTICLE HISTORY

Received 13 October 2020
Accepted 02 November 2020

Alex Pulamarin
Department of Electrical and Electronic
National Polytechnic University
Quito, Ecuador
alexdiaz11_corp@hotmail.com

A. Pulamarin, "Electric Vehicle with Speed Control Algorithm and Regenerative Braking and the Design of a Mobile Web App based on IoT", Latin-American Journal of Computing (LAJC), vol. 8, no. 1, 2021.

Vehículo Eléctrico con Algoritmo de Control de Velocidad y Freno Regenerativo y Diseño de una Aplicación Web Móvil Basada en IoT

Electric Vehicle with Speed Control Algorithm and Regenerative Braking and the Design of a Mobile Web App based on IoT

Alex Pulamarin
Department of Electrical
and Electronic
National Polytechnic
University
Quito, Ecuador
alexdiaz11_corp@hotmail.com

Resumen— Actualmente, la mayoría de los vehículos eléctricos para la movilidad personal no mantienen una velocidad constante y no frenan automáticamente cuando el sistema lo requiere. Además, esta energía de frenado no se utiliza para alimentar el propio sistema. Para superar estos problemas, este trabajo presenta la implementación de algoritmos de control y una aplicación web para un vehículo eléctrico de movilidad personal. Se implementa un algoritmo de control en cascada donde el lazo interno es el par y el lazo externo es la velocidad. Además, se diseñó una aplicación web móvil basada en el internet de las cosas (IoT) para mostrar en tiempo real, información importante sobre el estado del sistema. Finalmente, se realizaron simulaciones y pruebas reales que indicaron una respuesta rápida de par y velocidad cuando el sistema está sujeto a diferentes escenarios de movilidad.

Palabras clave— *controlador PI, motor BLDC, algoritmo de velocidad, control adaptativo de referencia de modelo (MRAC), frenado regenerativo, base de fuego, interfaz de programación de aplicaciones (API).*

Abstract— Currently, most electric vehicles for personal mobility do not keep a constant speed and they do not brake automatically when the system requires. Moreover, this braking energy is not used to power the system itself. To overcome these problems, this work presents the implementation of control algorithms and a web app for an electric vehicle for personal mobility. A cascade control algorithm is implemented where the internal loop is torque and the external loop is speed. Additionally, a mobile web app based on IoT was designed to display in real time some important information about the system status. Eventually, simulations and real test were performed indicating fast torque and speed response when the system is subjected to different mobility scenarios.

Keywords— *PI controller, BLDC motor, speed algorithm, model reference adaptive control (MRAC), regenerative braking, firebase, application programming interface (API).*

I. INTRODUCTION

Electric vehicles are having great impact on society because they are more efficient than fossil fuel vehicles. Furthermore, electric mobility is sustainable, innovative and it generates no emission of greenhouse gases which directly benefits the environment [16].

This project focuses on an electric vehicle for personal mobility that will help to solve environmental problems and decrease traffic rush in big cities, as well as reducing destination time and road space. Moreover, this project presents solutions to improve the use of battery time and brakes without any mechanical devices. These advantages save cost and physical space. Additionally, a mobile web app based on IoT has been designed to indicate important variable information about the vehicle status [1].

To keep a constant speed and to brake automatically in electric vehicles this work focuses on the control algorithm design and the design of a web app based on IoT technologies. Eventually, tests are carried out, and results are analysed indicating energy efficiency and performance.

II. BACKGROUND AND RELATED WORK

Each year there are innovative steps in the electric design of vehicles, where new technologies and technics in software and hardware are included in their manufacturing [11][12]. This research will be presenting important features in the software design of an electric vehicle for personal mobility.

Current electric vehicles for personal mobility do not maintain a fixed speed while riding. Therefore, it reduces their performance in speed and torque, when they are expose too many disturbances due to the unexpected road characteristics, and load [13]. One of the advances implemented is that the system brakes electrically and automatically without any mechanical component and without any frictional waste of energy. Instead, this energy from braking is tapping to power the entire system and recharge the battery pack. Furthermore, there are benefits such as saving space due to the nonuse of mechanical brakes and gears required to stop the vehicle.

To complement the system, a mobile web app has been designed in which important variables can be visualized. This information is crucial to optimize the life spam from the system. As a result, the user can monitor speed, temperature, battery level and other parameters. Besides that, the system can be locked and unlocked from the web for more security. Personal electric vehicles do not have these features like in the case of scooters, unicycles, skateboards, among others [14][15]. This research will start first with an analysis of a permanent brushless DC motor or simply BLDC motor in order to design the corresponding algorithms that will be executed on a chip to control this machine properly. These algorithms are fundamental to process and control variables such as current and speed. On the other side, these signals of current and speed have noise, thus both need to be filtered by a Butterworth filter. This filter improves substantially the output signal reducing the margin error less than 2%. Before the speed is processed, it is important to determine first the high and low states from the speed sensor. This information is used to determine the signal period and thus the synchronous speed.

In regard to the control algorithms, proportional and integral algorithms (PI) are used for both torque and speed. The current controller algorithm is based on the magnitude optimum criterion where the entire closed transfer functions become approximately one [4]. Then, the speed controller utilizes the criteria of the cancellation of poles and zeros to determine its gains. When the vehicle is subjected to many disturbances, the system slows down in speed response. Therefore, a type of a model reference adaptive algorithm (MRAC) is implemented in the speed loop, which updates its gains every 10 rpm.

In the interest of having more comfort when riding the vehicle, a web application based on the internet of things (IoT) has been developed. This app displays useful information such as speed, distance travelled, electronic board temperature, and power level. These data are first transferred from the chip to the firebase and then to the web app in real time.

III. CHARACTERISTICS OF A BLDC MOTOR

The principal part that provides force to the electric vehicle is a BLDC motor as shown in the Fig. 1. The principal parts of this motor are the stator and the rotor. The stator contains the coils and the rotor the permanent magnets.

Electrical and mechanical parameters of the BLDC motor are illustrated in Table I [1], which will be used to later to determine the controller parameters.

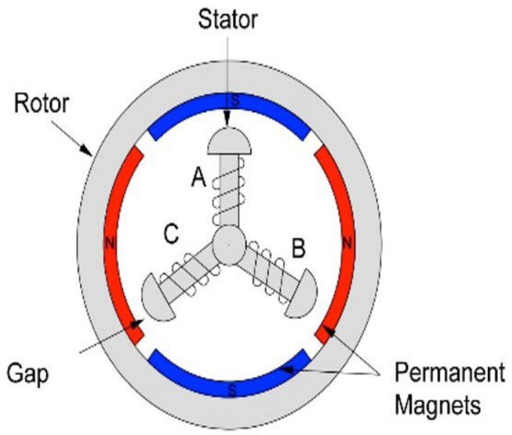

Fig. 1. Three-phase BLDC motor.

Table I. Electrical and mechanical parameters[1].

R	Phase AB resistance	0.5 ohms
L	Phase AB inductance	0.4 mH
τ_e	Electric time constant	0.0008 s
r	Wheel radius	0.21 m
T_{PWM}	PWM Time	$6.41 * 10^{-5} s$
τ_{mech}	Mechanical time constant	0.0678
B	Coefficient of viscosity	0.02336 Nms
J	Inertia	0.1375 Kgm^2
K_E	Torque Constant	1 Nm/A

IV. CURRENT AND SPEED MEASUREMENT

A.Current Measurement

Three Hall Effect sensors of 0-200A input and 0-50mA output are connected per phase as seen in Fig. 2. These current signals are conditioned by a differential amplifier circuit and then processed, filtered, and analysed by a microcontroller that gives feedback to the control current closed loop.

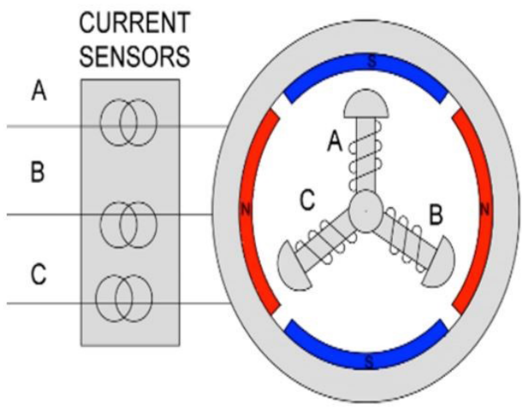

Fig. 2. Current sensors- connection per phase.

To have a better current signal a Butterworth filter is used, which eliminates noise produced by electromagnetic interference. In (1), u(s) is the filtered current signal and e(s) is the current signal nonfiltered.

$$H(s)=\frac{u(s)}{e(s)}=\frac{1}{(s+1)(s^2+s+a)} \qquad (1)$$

Equation (2) is in the frequency domain and it cannot be coded in a microcontroller therefore a differential equation from (6) is implemented as follows:

$$u(n)=Te[n-3]+u[n-1]-u[n-2] \qquad (2)$$
$$+u[n-3]-Tu[n-3]$$

B. Speed Measurement

To measure speed, hall effect sensors are embedded in the wheel and distributed in the coils 120 electrical degrees. These sensors can be high or low state (see Fig. 3).

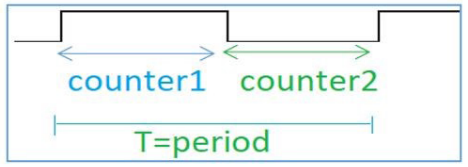

Fig. 3. Hall effect sensors high and low state signal.

In the microcontroller a timer is set to compute both high and low state to finally add up and obtain the signal period(T). Now, the synchronous speed can be easily calculated from (3) where p is the number of poles (p=56) and T is the period.

$$\Omega=\frac{120}{p*T} \qquad (3)$$

C. Temperature Measurement

The electronic board has three temperature sensors, which are located in each branch of the inverter. They alert the rider of overload and overcurrents, avoiding serious hardware damage. To calculate this temperature, it is known from the sensor datasheet that 1^o C/10mV, the supply voltage sensor is 3.3V and the microcontroller is 10bit-ADC(Analogue Digital Converter 1024) as indicated in (4).

$$Temperature=ADC*\frac{3.3}{1024}*100 \qquad (4)$$

V. CONTROLLERS ALGORITHM

To design the current and speed controllers a PI (proportional and integral) algorithm has been implemented. A PI algorithm has been chosen because the proportional response changes the output for a given change in the error and the integral response eliminates the steady state error that happens due to the proportional algorithm [10].

A.Current Controller Algorithm

Due to the electromagnetic torque is directly proportional to the torque constant and current, a current algorithm is implemented to control torque [2][3]. This algorithm is proportional and integral (PI) as indicated in (5), where k_{pi} is the proportional gain, and τ_i

$$PI(s)=k_{pi}\left(\frac{1+\tau_i s}{\tau_i s}\right) \qquad (5)$$

Fig. 4 illustrates a closed loop control. It is made up of a PI controller and the power electronics G_p and the BLDC motor transfer function G_M[5].

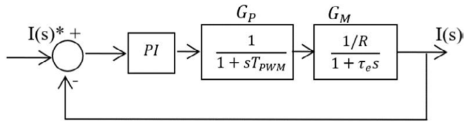

Fig. 4. Closed loop current.

Thus, the open-loop transfer function is:

$$G_{iaT}=$$
$$k_{pi}\frac{1+\tau_i s}{\tau_i s}\,\frac{1/R}{1+\tau_e s}\,\frac{1}{1+sT_{PWM}} \qquad (6)$$

Based on (6) and on the magnitude optimum criterion k_{pi} and τ_i are expressed as [4]:

$$k_{pi}=\frac{\tau_e R}{2T_{PWM}}=3.12 \qquad (7)$$

$$\tau_i=\tau_e=8*10^{-4} \qquad (8)$$

Fig. 5 is the closed loop current subroutine where errors are compute based on the current set point and the machine coil current then a control action is taken to generate a pulse width modulation or PWM signal.

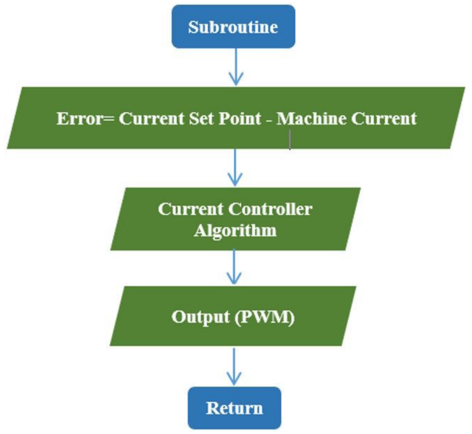

Fig. 5. Closed loop current subroutine.

B. Speed Controller Algorithm

The closed loop speed is shown in Fig. 6. It contains a PI algorithm, K_T is the torque constant and the mechanical transfer function $G\Omega$.

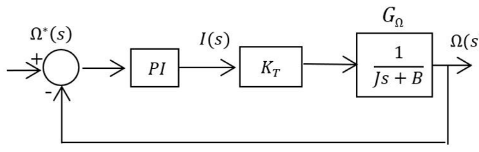

Fig. 6. Closed-loop speed.

The PI control algorithm is expressed as:

$$PI=k_{p\Omega}\frac{1+\tau_\Omega s}{\tau_\Omega s} \qquad (9)$$

The open-loop transfer function from Fig. 6 is:

$$G_{\Omega a}(s)=k_{p\Omega}\frac{1+\tau_\Omega s}{\tau_\Omega s}*\frac{1/B}{1+\frac{J}{B}s} \qquad (10)$$

Constants $k_p\Omega$ and $\tau\Omega$ are determined by the cancellation of poles and zeros and using Table I as follows:

$$\tau_\Omega=\frac{J}{B}=5.88 \qquad (11)$$

$$k_{p\Omega}=\tau_\Omega B=0.137 \qquad (12)$$

Fig. 7 is the closed loop speed subroutine where errors are calculated based on the speed set point and the machine mechanical speed. Then a control action is run, which generates an output equal to the current set point.

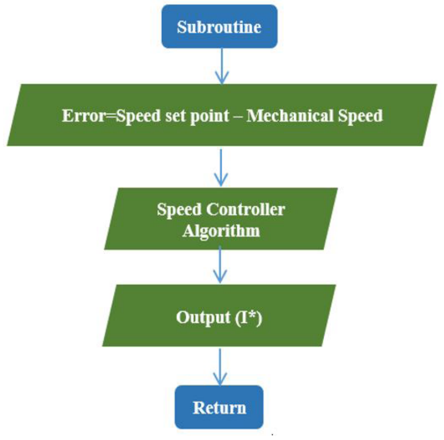

Fig. 7. Closed loop speed subroutine.

These constants are used to simulate the speed response in closed loop. In Fig. 8, the overshoot is 22% and the settling time is 0.422.

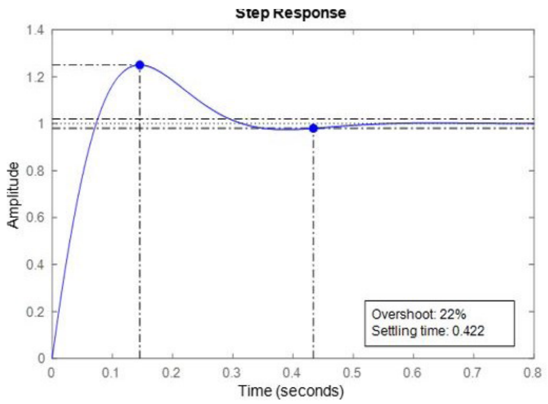

Fig. 8. Closed loop speed response simulation.

C. Adaptive Speed Control Algorithm

To improve the speed response a type of a model reference adaptive algorithm (MRAC) has been design (See Fig.9) [6].

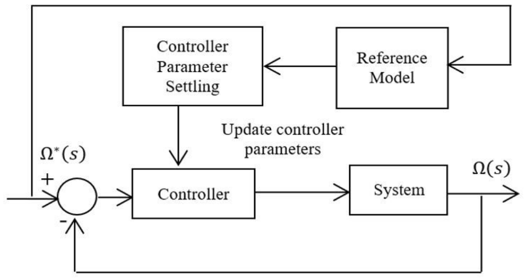

Fig. 9. Type of MRAC.

To implement this controller, speed is divided into intervals of 10 rpm and then speed constants change during this interval as indicated in Table II. k_pΩ and k_iΩ are determined using (10) and the root locus when K constant moves to the left [See Fig. 10(a)] [5].

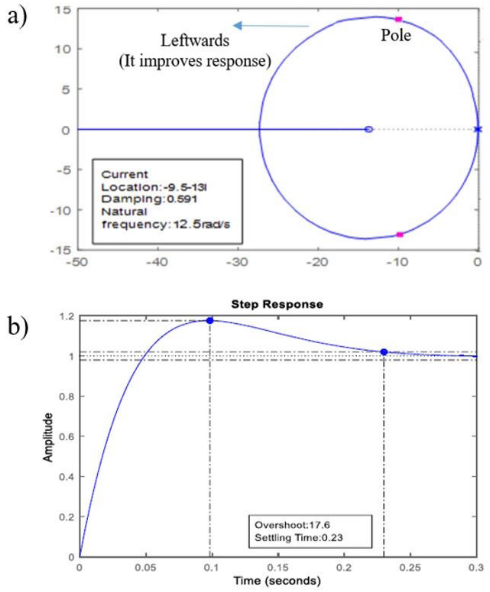

Fig. 10. a) Root Locus with K=1 b) Speed Response MRAC simulations.

In Fig. 10(b), there are improvements in speed response because the overshoot and the settling time go down to 17% and 0.23. Thus, speed response becomes faster.

Table II $k_p\Omega$ And $k_i\Omega$Speed Constants

Speed (rpm)	$k_{p\Omega}$	τ_{Ω}
0 a 10	0.137	5.88
10 a 20	0.28368	4.95
20 a 30	0.32256	4.23
30 a 40	0.40188	3.49
40 a 50	0.41472	3.30
50 a 60	0.4728	2.96
60 a 70	0.50688	2.69
70 a 80	0.56736	2.47
80 a 90	0.576	2.37
90 a 100	0.61464	2.28
100 a 110	0.64512	2.11
110 a 120	0.68556	2.04
120 a 130	0.6912	1.980
130 a 140	0.75648	1.85
140 a 150	0.8064	1.69

D. Difference Equations Algorithm

To implement the PI control algorithm into the microcontroller, it is necessary to transform from the frequency domain to Z domain and then into a difference equation. To discretize the controller, forward Euler method is used as shown in (13), where T is the sample time [7].

$$S \rightarrow \frac{z-1}{T} \qquad (13)$$

Thus, PI(z)=u(z)/e(z) where u(z) is the output and e(z) is the input or error:

$$u(z) = z^{-1}u(z) + k_{pi}e(z)$$
$$+(\tau^{-1}T - kp_i)z^{-1}e(z)s \qquad (14)$$

The inverse Z-transform is applied to (14) obtaining (15), which is programmed in the microcontroller.

$$u(n) = u(n-1) + kp_i\, e(n)$$
$$+(\tau^{-1}\,T - kp_i\,)e(n-1) \qquad (15)$$

VI. MOBILE WEB APP BASED ON IOT

The web app developed is based on the internet of things or IoT. It evolves multiple technologies such as, real-time analytics, sensors, wireless networks and embedded systems.

The web app components are classified in three main stages, microcontroller, database, and web development as shown in Fig. 11.

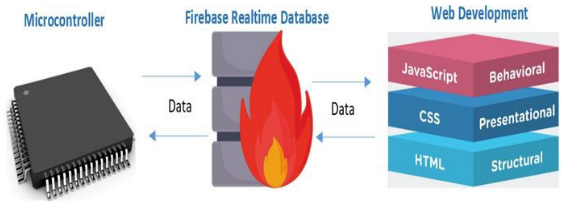

Fig. 11. Web app components.

A. Microcontroller

The microcontroller board has got a Wi-Fi chip that allows connectivity to the internet and it is compatible with TCP / IP protocol. To set up the microcontroller, libraries are attached to receive and send data to the Firebase. In addition, an application programming interface or API key is necessary to exchange data. These data are speed, distance traveled, battery level and temperature.

B.Firebase Realtime Database

The firebase real time database is a cloud-hosted database. Data is stored and synchronized in real time to the microcontroller.

The firebase real time database allows secure access to the database directly from the microcontroller. Data is persisted locally, and when offline, real time events continue. When the microcontroller is connected, the real time database synchronizes the local data changes with the remote updates that occurred while the microcontroller was offline, solving any trouble automatically [8]

C. Web Design

The web app was design by using Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It displayed images and other objects such as interactive forms that may be embedded into the page. HTML is assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. CSS brings style to the sheet by giving color, layout, and fonts. Java script instead manipulates data from the page itself and from the firebase. Java establishes connectivity with and the firebase through an API key along with other parameters such as domain, uniform resource locator (URL) and an identifier (ID). To run API in Java, three main components are required Java compiler, Java virtual machine and Java API as illustrated in Fig. 12 [9].

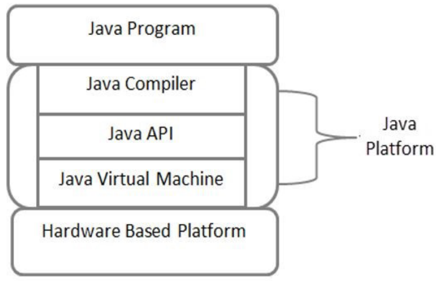

Fig. 12. Basic Java structure.

VII. RESULTS

The vehicle was tested under different conditions of speed, current and disturbances and which results are presented below.

A. Torque Controller Response

Tests were performed for a reference value of 15Nm torque, which results are presented in Fig. 13, presenting an overshoot of Mp<18% and an steady state of error<3%.

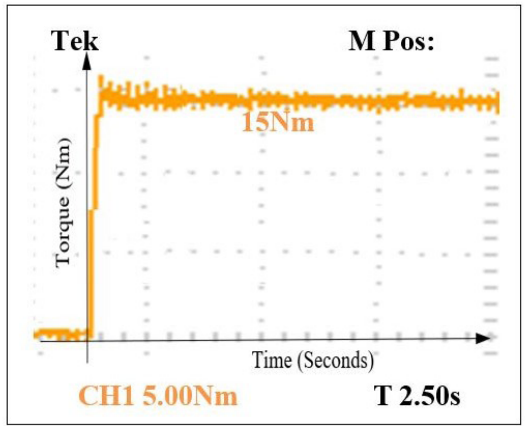

Fig. 13. Current controller response in closed loop for 15 Nm.

B. Speed controller response

Tests were performed for speed references of149 rpm and the results obtained are shown in Fig. 14 with an overshoot Mp<20% and an error<4%.

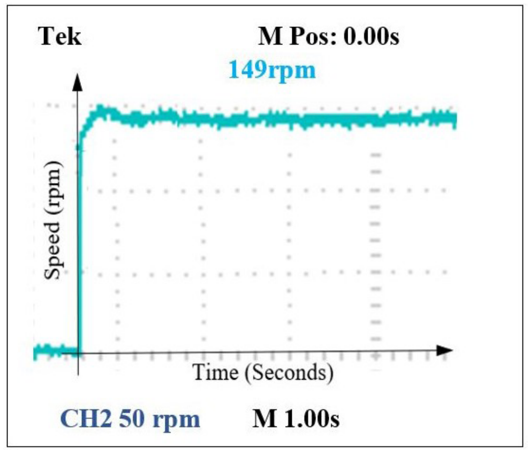

Fig. 14. Speed controller response in closed loop for149 rpm.

Under disturbances, the controller also responds property and which its results are shown in Fig. 15 for a reference speed of 70rpm.

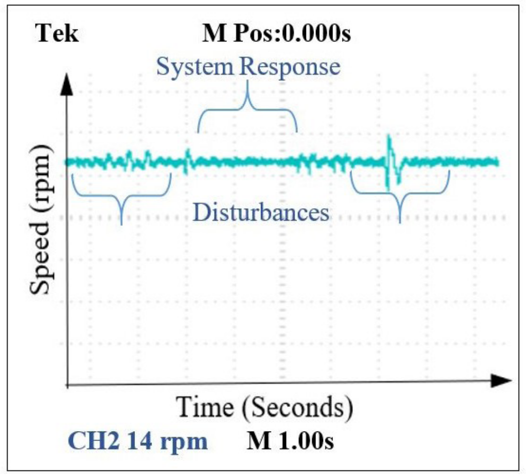

Fig. 15. Speed controller in closed loop for 70 rpm under disturbances.

C. System Response in Regenerative Braking Mode

The system is ridden in different scenarios where A is flat terrain and B, C are descending slopes as indicated in Fig. 16.

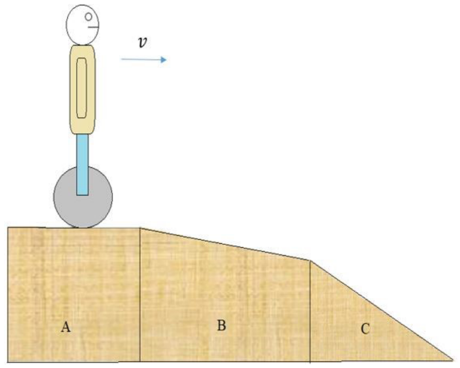

Fig. 16. Mobility scenarios.

In Fig. 17 can be observed that when the vehicle is ridden on flat terrain the torque positive , therefore there is energy consumption but when the vehicle is ridden on descending slopes the torque is negative therefore there is energy regeneration. This energy can be used to supply power to the electronic board and recharge the pack of batteries.

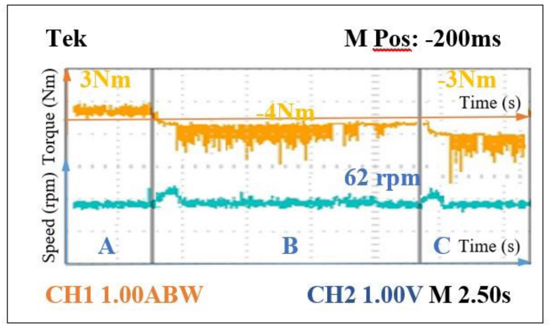

Fig. 17. Energy regeneration test to 62 rpm.

D. Mobile Web App

Fig. 18 is the web app design which indicates speed in Km/h, distance travelled , battery level, MOSFET temperature, the led can change from red to green indicating whether the vehicle saves energy or consumes energy and also allows us the option to lock the vehicle.

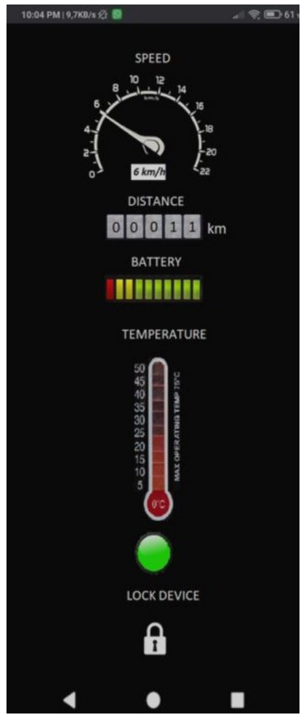

Fig. 18. Mobile web app.

VIII. CONCLUSION AND RECOMMENDATIONS

The software for an electric vehicle for personal mobility was designed and implemented with torque, speed, and energy regeneration.

Additionally, tests were performed under minimum, normal, and maximum operating conditions demonstrating a good system performance. The torque response has an overshoot less than 18% and a steady state error less than 3% while the speed response has an overshoot less than 20% and an error less than 4%, thus, meeting the design criteria. Under disturbances the system reacts fast and smoothly. On the other hand, the mobile web app becomes handy when riding because it presents valuable information of the vehicle status. Additionally, it helps to increase the lifetime of the vehicle when it warns high temperature values.

In regard to the regenerative braking, the energy produced can supply energy to the electronic board. Moreover, it can recharge the battery pack y some time intervals. In this stage, the time of the battery can last for 20 to 30 minutes and it can regenerate an average energy of 35 Wh. Furthermore, when the system is in regenerative braking mode, it brakes automatically, keeping a fixed speed and eliminating mechanical brakes and frictional waste, which are great economic and space saving benefits.

Other control algorithms can be implemented to improve speed response and energy efficiency. In regard to the web app parameters such as speed record, graphs and GPS can incorporated.

REFERENCES

[1] P. Alex, "Design and implementation of speed and torque control with regenerative braking for a platform prototype of an electric unicycle," February 2019.

[2] T. Miller, "Brushless permanent magnet and reluctance Motor Drives," Oxford University, USA, 1989.

[3] Mohan Ned, "Electric drives, an integrate approach," USA, MNPERE Minneapolis, 2001.

[4] G. Papadopoulos, "PID controller tuning using the magnitude optimun criterion," Springer, Suiza, 2015.

[5] K. Ogata, "Modern Control Engineering," PEARSON, 5th Ed., Madrid, 2010.

[6] Embetion, "Adaptive algorithms," [Online]. Available: https://www.embention.com [Accessed: Sep.11,2020].

[7] MIT, "Discrete Approximation of Continuous Time Systems," [Online]. Available: https://ocw.mit.edu/terms/ [Accessed: Sep.11,2020]

[8] Firebase, "Firebase Realtime Database," [Online]. Available: https://firebase.google.com/docs/database [Accessed: Nov.05, 2020]

[9] Firebase, "App development platform," [Online] Available: https://firebase.google.com/docs/database [Accessed: Nov.06, 2020]

[10] C. Kuo "Automatic control systems," PRENTICE HALL, 7th Ed, New York, 2010

[11] D. Stefano, C. Pablo, S. Aldo, G. Patrik, P. Pietro and V. Fabio " Torque-fill control and energy management for a four-wheel-drive electric vehicle layout With two-speed transmissions," " IEEE Transactions on Industry Applications, October 2016.

[12] M. Arash, S. Aldo, G. Patrick, F. Saber and S. Jasper "A fast and parametric torque distribution strategy for four-wheel-drive energy-efficient electric vehicles" IEEE Transactions on Industry Applications," IEEE Transactions on Industrial Electronics, July 2016.

[13] R. Desna, H. Eko, S. Randen and D. Dadet "PENS-Wheel (One-Wheeled Self Balancing Vehicle) Balancing Control using PID Controller," Conference International Electronics Symposium 2016.

[14] R. Jiageng Ruan and S. Qiang "A Novel Dual-Motor Two-Speed Direct Drive Battery Electric Vehicle Drivetrain," IEEE Access, April 2019.

[15] R. Bakhtiar, H.Eko, S. Raden and P. Dadet "PENS-Wheel (Self Balancing One-Wheel Vehicle) Mechanical Design and Sensor System," Conference International Electronics Symposium 2016.

[16] NewScientist, "Electric cars really are a greener option than fossil fuel vehicles," [Online]. Available: www.newscientist.com [Accessed: Nov. 11, 2020]

AUTHOR

Alex Pulamarin

Alex Pulamarin he was born in Cayambe-Ecuador. He carried out his studies at National Polytechnic University. His major is in Electronic and Control Engineering. He has participated in several research projects in electric vehicles in the department of power electronics at National Polytechnic University and at Massachusetts Institute of Technology MIT. Areas of interest: Electrical Machines, Power Electronics, Embedded Systems, Instrumentation, and Industrial Control.

A. Pulamarin, “Electric Vehicle with Speed Control Algorithm and Regenerative Braking and the Design of a Mobile Web App based on IoT”, Latin-American Journal of Computing (LAJC), vol. 8, no. 1, 2021.