

Didactic-embedded platform for system identification applied to a BLDC motor

F.J. Villalobos-Piña* R. Alvarez-Salas**
J.A. Reyes-Malanche*** C.H. Saucedo-Zarate*
M.S. Esparza-González* L.A. Gonzalez-Murillo**

* *TecNM/Instituto Tecnológico de Aguascalientes, Aguascalientes, Ags., 20256 México (e-mail: fvillalobos@mail.ita.mx, carloshum777@hotmail.com).*

** *Facultad de Ingeniería, Universidad Autónoma de San Luis Potosí, San Luis Potosí, S.L.P., 78290 México (e-mail: ralvarez@uaslp.mx, luis.murillo@uaslp.mx).*

*** *Universidad Tecnológica de Aguascalientes, Aguascalientes, Ags., 20196 México (e-mail: josue.malanche@utags.edu.mx).*

Resumen: This paper presents a didactic-embedded platform for system identification applied to a BLDC motor like those found in drones. The main component of the embedded system is a Raspberry Pi 4 board with a slave board based on Microchip microcontroller PIC18F45K50. A detailed case study shows the platform's applicability to identify the Brushless Direct Current (BLDC) motor's parameters using the process reaction curve method.

Keywords: System identification, Embedded system, BLDC Motor

1. INTRODUCTION

Los cursos de control automático son una parte importante de la mayoría de los programas de ingeniería, una rama destacada dentro de esta disciplina es la identificación de sistemas. Sin embargo, para realizar el proceso de identificación completo se requiere disponer de plataformas experimentales en las cuales los estudiantes pueden determinar modelos matemáticos de diversos procesos para implementar alguna estrategia de control. En el mercado existen sistemas muy completos para tales fines, sin embargo son costosos. Una alternativa a esta problemática la constituyen las plataformas basadas en sistemas de desarrollo con procesadores digitales de bajo y programadas con software libre.

Existen muchas alternativas de plataformas digitales para el diseño del sistema de adquisición de datos y control, por ejemplo, en [Sobota et al. (2013)] se presenta una plataforma de bajo costo que permite la implementación de un algoritmo de control en tiempo real. El sistema consiste en un tarjeta Arduino en conjunto con una tarjeta Raspberry Pi que corre con el sistema de control REX.

Por otro lado, la estimación de parámetros eléctricos y mecánicos en los motores eléctricos es fundamental para la implementación de la mayoría de los controladores. En

[Arifin et al. (2021)] se hace una revisión de técnicas de estimación de parámetros basadas en el algoritmos de mínimos cuadrados y variantes, así como en técnicas de inteligencia artificial para las tipos más comunes de máquinas eléctricas rotatorias. En particular, se han reportado múltiples trabajos sobre el modelado, identificación de parámetros y control para el motor BLDC. Un método de determinación de parámetros de un motor BLDC usando pruebas electromecánicas y un algoritmo de mínimos cuadrados recursivos se describe en [Jimenez-Gonzalez et al. (2021)]. Una variante del algoritmo de estimación utilizando mínimos cuadrados ponderados se presenta en [Majdoubi et al. (2021)]. En [Danusuya et al. (2017)], se propone un esquema de indentificación de parámetros mediante el uso de técnicas de optimización y aprendizaje profundo con redes neuronales artificiales. [Kumpanya et al. (2015)] propone un método de identificación mediante optimización heurística. En [Anshory et al. (2021)], se utiliza lógica difusa para controlar la velocidad de un motor BLDC. Se han reportado también trabajos que describen el comportamiento del motor BLDC en términos de señales de vibración utilizando técnicas de elemento finito con modelos 3-D [Cho et al. (2018)].

El presente trabajo describe el desarrollo de una herramienta de bajo costo para la identificación de sistemas, la parte principal del sistema consiste en una tarjeta Raspberry Pi 4.0 model B con una segunda tarjeta esclava basada en el microcontrolador Microchip PIC18F45K50, la cual realiza las tarea de adquisición de datos para llevar

* This work was partially supported by AME-FI-UASLP, UASLP-CA-246 Sistemas Eléctromecánicos y Energía Sustentable, and TecNM/Instituto Tecnológico de Aguascalientes.

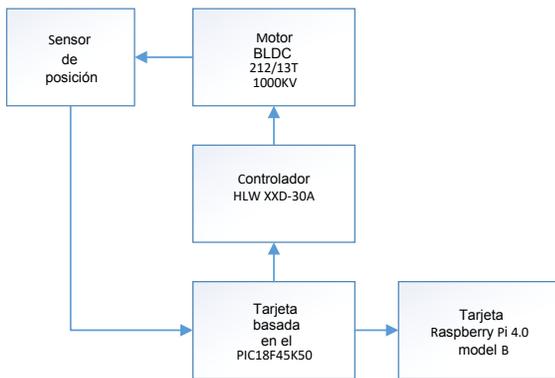


Figura 1. Diagrama de bloques de la plataforma.

acabo el proceso de identificación de parámetros de una planta. La tarjeta Raspberry Pi se utiliza para procesar los datos adquiridos, guardar la información en un medio de almacenamiento secundario y visualizar los resultados. La visualización se realiza mediante una pantalla HDMI. El software se desarrolló en Python para la tarjeta Raspberry Pi y en C para la tarjeta del PIC18F45K50. Como ejemplo de aplicación, se presenta un caso de estudio que consiste en determinar experimentalmente la curva de reacción de un motor BLDC comúnmente empleado en aeronaves no tripuladas pequeñas (drones).

En la sección 2, se describe la plataforma experimental. El software desarrollado se detalla en la sección 3. Las características de la interfaz gráfica del sistema desarrollado se muestran en la sección 4. La sección 5 presenta un ejemplo de aplicación de la plataforma para identificar la curva de reacción de un motor BLDC y, finalmente, las conclusiones se discuten en la sección 6.

2. DESCRIPCIÓN DE LA PLATAFORMA EXPERIMENTAL

En la figura 1 se presenta un diagrama de bloques del sistema implementado. Los elementos de la plataforma experimental se describen a continuación:

- Tarjeta Raspberry Pi 4.0 model B con una pantalla HDMI marca Sunfounder, un teclado y un mouse (figuras 2 y 3).
- Tarjeta esclava basada en el microcontrolador Microchip PIC18F45K50, de diseño propio, con conexión hacia la Tarjeta Raspberry Pi (figura 4).
- Motor trifásico BLDC marca QWinOut de 12V modelo A2212/13T 1000KV con sensor de posición (figura 5).
- Controlador marca Create idea modelo HLW XXD-30A, el cual es usado en sistemas de aeromodelismo y control de drones (figura 6).



Figura 2. Sistema embebido.



Figura 3. Tarjeta Raspberry Pi 4.0 model B.

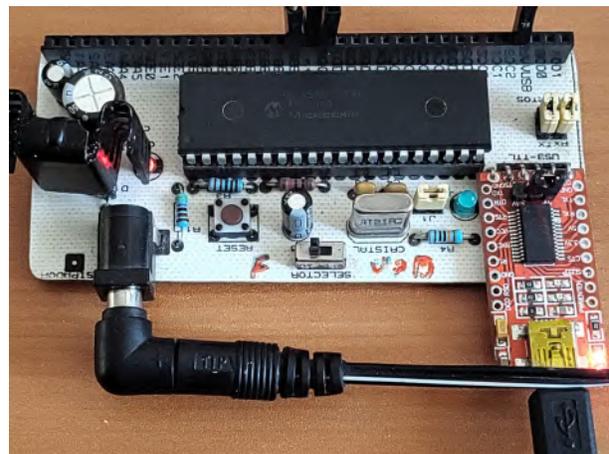


Figura 4. Tarjeta esclava.

2.1 Tarjeta Raspberry Pi 4.0 model B

El sistema de adquisición de datos consiste en una tarjeta Raspberry Pi modelo 4.0 model B y una capacidad de 8 Gb de memoria RAM, la cual se comunica con la tarjeta esclava mediante un puerto serial asíncrono (UART, por sus siglas en inglés). La tarjeta Raspberry Pi cuenta con bus de salida en la cual se encuentran conectadas las terminales del UART en GPIO14 (señal de transmisión TX) y en GPIO15 (señal de recepción RX). La computadora opera con niveles lógicos de voltaje de 3,3 V, pero es tolerante a los niveles de TTL que maneja el microcontrolador PIC.

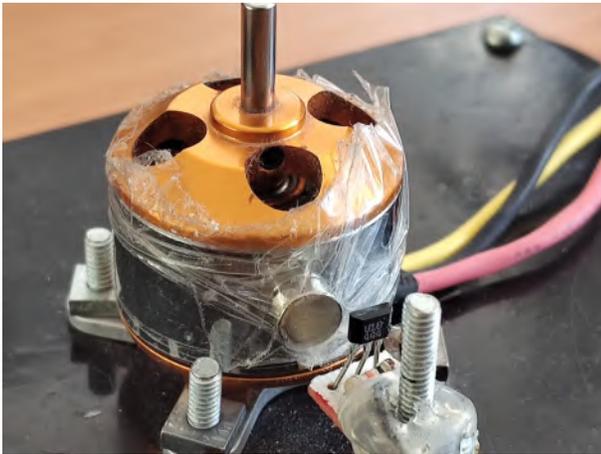


Figura 5. Motor BLDC con sensor de posición.

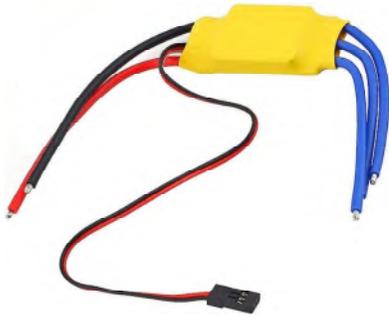


Figura 6. Controlador modelo HLW-30A.

La tarjeta Raspberry Pi funciona con el sistema operativo Raspbian, variante de la distribución Debian Linux.

2.2 Tarjeta esclava

La Tarjeta esclava se basa en el microcontrolador Microchip PIC18F45K50, el cual opera a una velocidad real de 12 MHz y se logran 12 MIPS. Se utiliza el módulo de captura con que cuenta del microcontrolador para estimar el periodo de la señal digital generada por el transistor de efecto Hall que se utiliza como sensor de posición acoplado al motor BLDC, con dicho periodo, es posible estimar la frecuencia de rotación mecánica del motor y, a su vez, calcular la velocidad angular en revoluciones por minuto (RPM). La tarjeta cuenta también con un convertidor USB-Serial basado en el circuito integrado FTDI 232.

2.3 Motor BLDC con sensor de posición

Para estimar la velocidad angular del motor BLDC, fue necesario instrumentar éste, ya que no contaba con ningún

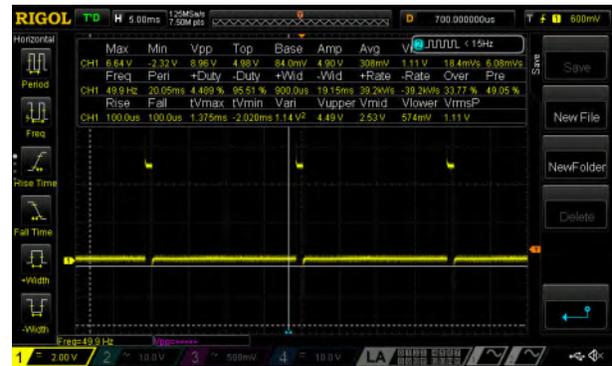


Figura 7. Señal PWM para el control del motor BLDC.

tipo de sensor para tal fin. Se pegó un imán de neodimio sobre un costado del motor y se montó a un lado de éste un transistor de efecto Hall modelo U18444, con una resistencia de carga tipo pull-up de 10kΩ, la cual permite generar una señal de lógica digital de niveles TTL.

2.4 Controlador y señal PWM

Para operar el motor BLDC con el controlador HLW-30A, éste requiere una señal que tenga un periodo de 20 msec, la duración en alto mínima es de 800 μseg y la dinámica para modificar la velocidad de 0% al 100% debe definirse mediante un cierto algoritmo. Esto se hace al aplicar la energía al controlador, éste genera una serie de tonos que son audibles durante los cuales el controlador espera el valor mínimo de PWM y, posteriormente, el valor máximo. Para el presente trabajo se definió una dinámica de 1,5 msec, es decir, el algoritmo de PWM genera una señal en alto con una duración entre 800 μseg y 2,3 msec sumando el tiempo mínimo en alto y el tiempo que define la velocidad desde su valor mínimo al máximo. De esta manera, el motor BLDC alcanzó una velocidad máxima de 23000 RPM.

En la figura 7 se ilustra una imagen capturada con un osciloscopio RIGOL modelo MSO1105 Z en la cual se aprecia la señal PWM generada por la tarjeta esclava con un temporizador operando a una frecuencia de salida de 50 Hz.

3. SOFTWARE

En esta sección se describen las principales funciones de los programas desarrollados para las tarjetas Raspberry Pi y PIC18F45K50.

3.1 Tarjeta Raspberry Pi

En la tarjeta Raspberry Pi se implementó básicamente un algoritmo que solicita al microcontrolador PIC18F45K50 un buffer de datos de 1 KB de información, ésta restricción se debe al tamaño de la memoria del microcontrolador. La velocidad de muestreo puede ser variable, pero

para fines de este trabajo se utilizó 1 KHz . Es decir se captura un periodo de un segundo de información de la velocidad en tiempo real.

Por otra parte, la tarjeta Raspberry Pi implementa un esquema de comunicación con el microcontrolador en el cual se intercambian tramas con el siguiente formato:

[COMANDO][LONGITUD][DATOS][CRC-16]

donde [COMANDO] es la acción a realizar, por ejemplo: `CMD='D'` (ASCII) (captura de datos), LONGITUD es el tamaño de la trama de datos que puede ser de hasta 1 KB , [DATOS] es el paquete de datos y [CRC-16] es una verificación para validar que el paquete intercambiado sea correcto con un algoritmo de redundancia cíclica (CRC, por sus siglas en inglés). En caso de que la verificación presente algún problema, el algoritmo implementado solicita nuevamente dicho comando y así se garantiza una comunicación segura. Una vez recibido el paquete de datos por parte de la tarjeta Raspberry Pi, se visualiza utilizando para ello la librería Tkinter de Python en una ventana.

Un algoritmo adicional permite solicitar el sistema operativo Raspbian el uso del puerto serial asíncrono, siempre y cuando se encuentre disponible por parte del BIOS de la tarjeta. Mediante una caja de dialogo es posible seleccionar entre diferentes posibles direcciones del puerto serial.

3.2 Tarjeta esclava

El microcontrolador PIC18F45K50 es básicamente un sistema de adquisición de datos, el cual cuenta con un algoritmo que hace uso de un temporizador que permite el muestreo y almacenamiento de datos en memoria RAM del procesador, la información capturada es el tiempo de disparo de una señal cuadrada producida al rotar el motor mediante el uso de un sensor de efecto Hall y un imán de neodimio. El hardware utilizado del microcontrolador para estimar la frecuencia de rotación es el módulo de captura. Dicho módulo produce una interrupción que se programa por flancos de subida y/o bajada o cierta cantidad de flancos de subida y/o bajada. Una vez que se genera dicha interrupción en conjunto con el temporizador se logra medir el tiempo entre flancos, como consecuencia se calcula la frecuencia y, finalmente, se obtiene la velocidad de rotación en RPM del motor.

4. INTERFAZ GRÁFICA

Se desarrolló un programa en Python con las librerías Tkinter y Matplotlib para la tarjeta Raspberry Pi con el fin de disponer de un entorno de desarrollo que facilite el análisis de los resultados.

4.1 Pantalla principal

El programa desarrollado de la interfaz gráfica permite efectuar la captura de la curva de reacción en tiempo

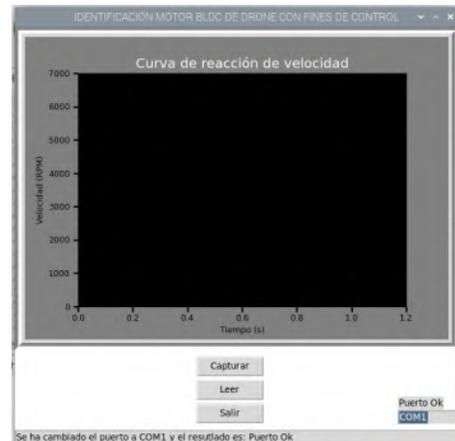


Figura 8. Pantalla principal.

real, el guardado de ésta en un medio de almacenamiento secundario, la lectura de datos que se tengan previamente guardados y su visualización, cambiar el puerto serial en caso de que la tarjeta tenga más de uno. Cabe mencionar que el programa diseñado funciona de igual forma en una computadora con Linux como en una computadora con Windows, esto se logra prácticamente sin cambios; excepto un ajuste menor en las configuraciones para el acceso al puerto serial en el caso de la tarjeta Raspberry Pi. En la figura 8, se ilustra la ventana que muestra el aspecto del programa visual diseñado utilizando la librería Tkinter.

4.2 Captura de datos

Mediante un proceso de comunicación serial asíncrona $RS232$, se establece la comunicación con la tarjeta esclava a una velocidad de 19200 Bauds , 8 bits de datos, 1 bit de paro, sin paridad. Además, se muestrea la velocidad del motor BLDC con una frecuencia de muestreo de 1 KHz y se captura una cantidad de 900 muestras con un tamaño de palabra de 16 bits, como se ilustra en la figura 9.

4.3 Almacenamiento de datos

Una vez que se ha capturado de forma experimental los datos de la curva de reacción de velocidad, es posible almacenarla en un medio de almacenamiento secundario de la tarjeta para su posterior análisis. La información se almacena en formato tipo texto con una estructura matricial de 900 renglones y 3 columnas, donde la primera columna es el número consecutivo de dato, la segunda columna representa el instante de tiempo en que fue capturada la muestra y, finalmente, la tercera columna representa la velocidad en RPM en el instante de captura. En la figura 10, se muestra la ventana invocada mediante la librería Tkinter que permite hacer uso de la API del sistema operativo correspondiente al cuadro de dialogo para almacenar archivos.

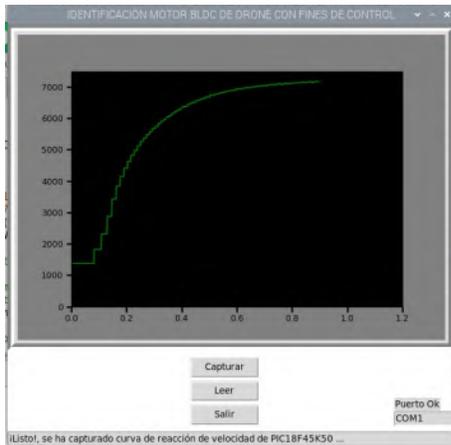


Figura 9. Captura de la curva de reacción de velocidad.



Figura 11. Lectura de archivos.

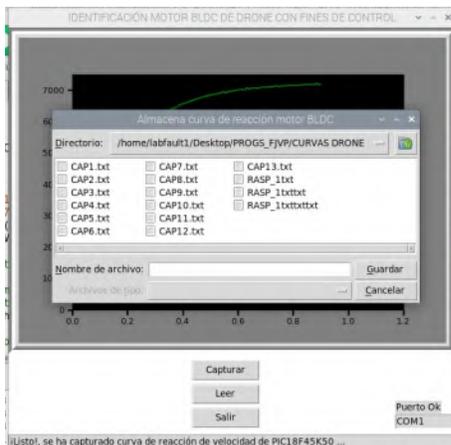


Figura 10. Almacenamiento de datos en archivo.

4.4 Lectura de archivos

En la figura 11, se muestra el cuadro de diálogo del sistema operativo que permite navegar y seleccionar algún archivo tipo texto para visualizar la curva de reacción de velocidad previamente almacenada.

5. IDENTIFICACIÓN DEL SISTEMA

Una vez que se cuenta con una base de datos de curvas de reacción de velocidad del motor BLDC, es posible efectuar un proceso de identificación del sistema. A manera de ejemplo, para fines del presente trabajo se procedió al diseño de un programa en MATLAB que permite aproximar la curva de velocidad a la de un sistema de primer orden dado por (1).

$$G(s) = \frac{K}{\tau s + 1} \quad (1)$$

En la figura 12, se presentan las curvas de 10 experimentos.

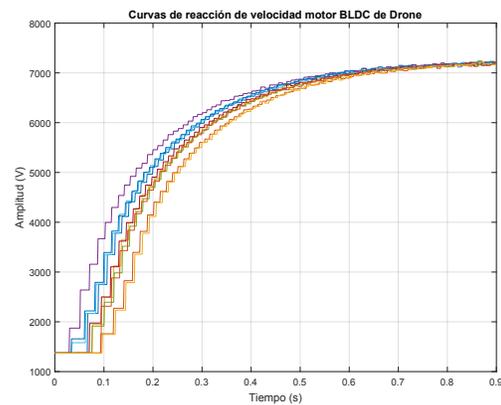


Figura 12. Curvas de reacción de velocidad experimentales.

El cálculo de la ganancia K de la función de transferencia se obtuvo considerando el voltaje de excitación máximo aplicado al sistema, $V_s = 12\text{ V}$, y la velocidad máxima de rotación angular alcanzada por el motor V_{max} de acuerdo con (2):

$$K = V_{max}/V_s \quad (2)$$

La ganancia fue $K = 600,2219$ y la constante de tiempo $\tau = 0,1880$, dando la siguiente función de transferencia identificada:

$$G(s) = \frac{600,2}{0,188s + 1} \quad (3)$$

En la figura 13, se presenta la comparativa entre la curva de reacción promedio experimental y la respuesta al escalón de amplitud V_s utilizando la función de transferencia identificada. Cabe mencionar que se trata de una aproximación básica que sirve como base para aplicar algoritmos más elaborados de identificación de sistemas.

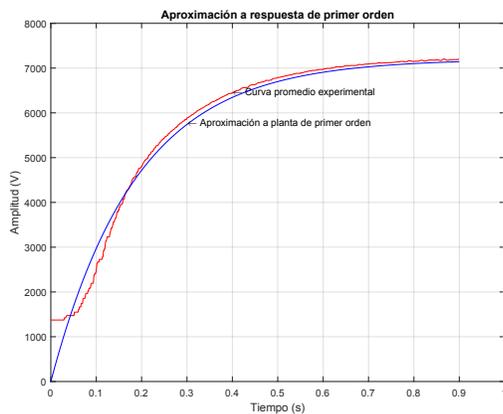


Figura 13. Comparativa de las respuestas experimental e identificada.

6. CONCLUSION

Se presentó el desarrollo de una plataforma embebida de bajo costo para realizar tareas de identificación de sistemas. Se utilizó como planta un motor BLDC de drone para ilustrar las funcionalidades del sistema. Sin embargo, la plataforma se puede emplear para determinar los parámetros de otros procesos, siempre y cuando éstos dispongan de la instrumentación adecuada para que las variables de interés puedan ser adquiridas por el sistema descrito.

REFERENCIAS

- Anshory, I., Hadidjaja, D., and Sulistiyowati, I. (2021). Measurement, modeling, and optimization speed control of bldc motor using fuzzy p so based algorithm. *Journal of Electrical Technology UMY (JET-UMY)*, 5(1), 1–9, <https://journal.umy.ac.id/index.php/jet/article/view/12113>, ISSN: 2550–1186.
- Arifin, B., Nugroho, A.A., Suprpto, B., Prasetyowati, S.A.D., and Nawawi, Z. (2021). Review of method for system identification on motors. *2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 1–6, <https://doi.org/10.23919/EECSI53397.2021.9624259>.
- Cho, S., Hwang, J., and Kim, C.W. (2018). A study on vibration characteristics of brushless dc motor by electromagnetic-structural coupled analysis using entire finite element model. *IEEE Transactions on Energy Conversion*, 33(4), 1–16, <https://doi.org/10.1109/TEC.2018.2833493>.
- Danusuya, P., Balamuruga, K., and Mahalakshmi, R. (2017). Parameter identification in bldc motor using optimization technique. *Journal of Applied Science and Engineering Methodologies*, 3(2), 465–470, <http://www.jasem.in/2017/32465470>, ISSN: 2395–5341.
- Jimenez-Gonzalez, J., Gonzalez-Montañez, F., Jimenez-Mondragon, V.M., Liceaga-Castro, J.U., Escarela-Perez, R., and Olivares-Galvan, J.C. (2021). Parameter identification of bldc motor using electromechanical tests and recursive least-squares algorithm: Experimental validation. *MDPI actuators*, 10(143), 1–16, <https://doi.org/10.3390/act10070143>.
- Kumpanya, D., Thaiparnat, S., and Puangdownreong, D. (2015). Parameter identification of bldc motor model via metaheuristic optimization techniques. *Elsevier Procedia Manufacturing*, 4(1), 322–327, <https://doi.org/10.1016/j.promfg.2015.11.047>.
- Majdoubi, R., Masmoudi, L., Bakhti, M., Elharif, A., and Jabri, B. (2021). Parameters estimation of bldc motor based on physical approach and weighted recursive least square algorithm. *International Journal of Electrical and Computer Engineering (IJECE)*, 11(1), 1–13, <https://doi.org/10.11591/ijece.v11i1.pp133-145>, ISSN: 2088–8708.
- Sobota, J., Pisl, R., Balda, P., and Schlegel, M. (2013). Raspberry pi and arduino boards in control education. *10th IFAC Symposium Advances in Control Education*, 1–6, <https://doi.org/10.3182/20130828-3-UK-2039.00024>.