

CFV2: An Open-Source Robot Controller Board for Education and Research

Claudio Morales
Instituto Profesional Inacap
Santiago, Chile
claudio.morales55@inacapmail.cl

César Fuenzalida
Instituto Profesional Inacap
Rancagua, Chile
cesar.fuenzalida02@inacapmail.cl

Grinda Sierra
Instituto Profesional Inacap
Santiago, Chile
grinda.sierra@inacapmail.cl

Abstract—We introduce CFV2, a robot controller board made to make it easier to create mobile robots for research and education. This card was developed in accordance with the open-source philosophy in an effort to democratize robotics research and promote collaboration in technical and engineering education. Through pulse width modulation (PWM) control, independent quadrature encoder and current sensor reading, a gyroscope, an accelerometer, and signaling accessories—all coupled with a microcontroller that adds connectivity for wireless and serial communication—CFV2 enables the kinematic and dynamic control of robots with up to four motors. For a small fraction of the cost of comparable commercial systems, the CFV2 open source software enables its various stages to be configured as Robot Operating System (ROS2) nodes, simplifying their integration into high-performance robotic systems for teaching and research. This paper describes the features of CFV2 and provides an illustration of its application.

Keywords—robot control, engineering education, research and development

I. INTRODUCCIÓN

Integrar la robótica en la formación de técnicos e ingenieros sigue siendo un desafío para las instituciones educativas [1]–[3]. Si bien se ha alcanzado consenso sobre lo importante que es para los estudiantes utilizar tecnologías actuales para alcanzar las habilidades y conocimientos que demanda la Industria 4.0, existe aún una brecha entre las tecnologías que las instituciones educativas pueden poner a disposición de los estudiantes durante su período formativo y aquellas que se vienen incorporando progresivamente en los entornos reales de las empresas globales [4], [5].

La robótica ha resultado particularmente difícil de integrar en las instituciones de educación superior, debido a dos razones principales: 1) resulta imprescindible contar con robots físicos adecuados y bien configurados, y en un número y disponibilidad suficiente, y 2) se requiere un equipo de docentes bien calificados en el uso de estos equipos, a fin de diseñar e implementar experiencias educativas pertinentes [1], [2]. Ambos aspectos tensionan a las instituciones a invertir recursos cuantiosos en recursos educativos que, más aun, deben ser renovados frecuentemente debido al avance tecnológico.

Para enfocar correctamente este problema, se debe distinguir entre lo que se denomina *robótica educativa*—aquella

utilizada principalmente a nivel de educación primaria y secundaria para promover el desarrollo de habilidades de razonamiento y resolución de problemas en STEM (ciencia, tecnología, ingeniería y matemáticas)— y la robótica requerida para la formación de ingenieros y técnicos especialistas que se preparan para trabajar con sistemas robotizados reales en contextos productivos. La robótica educativa sí ha podido popularizarse bastante, gracias a la amplia disponibilidad de kits a bajo costo basados en plataformas de tipo Lego Mindstorm o Arduino, los que cuentan con abundante soporte y documentación en comunidades y foros de Internet. No ocurre lo mismo con los sistemas robóticos requeridos para la formación de ingenieros y técnicos que, en su futuro laboral, se desempeñarán en entornos productivos que emplean robots en sus tareas cotidianas.

Diversos sectores de la industria, el retail y la logística ya integran robots móviles de tipo AGV (Autonomous Guided Vehicles) en sus actividades. Entre las especificaciones mínimas de estos robots suelen incluirse cámaras RGB-D para medición de profundidad y sensores LiDAR (Light Detection and Ranging) integrados en sistemas de posicionamiento y navegación SLAM (Simultaneous Location And Mapping), lo cual requiere de computadoras con potencia de cálculo suficiente para ejecutar tareas de visión e inteligencia artificial, al mismo tiempo que intercambian gran cantidad de datos en tiempo real con otros múltiples sensores y actuadores, empleando protocolos de comunicaciones diversos. El más básico de este tipo de equipamiento de uso industrial cuesta alrededor de USD 15000, a lo cual se debe agregar el costo de las licencias de software para su programación y configuración.

Los conocimientos y habilidades requeridos para trabajar con esta tecnología resultan de gran importancia para los futuros profesionales y técnicos, pero integrar sistemas robotizados de estándar industrial como recursos didácticos en los procesos formativos resulta prohibitivo para las instituciones. Esto ha motivado la búsqueda de alternativas que permitan contar con recursos didácticos asimilables a los robots y sistemas informáticos reales utilizados en producción, que sean flexibles para permitir la integración de tecnologías diversas y a un costo manejable para los presupuestos de las instituciones educativas [4], [7–11].

A fin de reducir los costos asociados a licencias de software, los robots utilizados en educación e investigación a

Este trabajo ha sido financiado por el Centro de Entrenamiento en Automatización y Robótica de Instituto Profesional Inacap.

nivel de ingeniería generalmente se basan en la plataforma de software libre ROS o ROS2 (Robot Operating System) [6]. Entre robots comerciales diseñados bajo este enfoque están Turtlebot 4 de Clearpath Robotics, un robot diferencial con un costo en el rango de USD 1200 a USD 1900, y Rosbot Pro de Roboworks, un robot holonómico con ruedas omnidireccionales, que se comercializa en alrededor de USD 4000.

Para reducir más el costo del equipamiento, prestigiosas instituciones educativas en distintas partes del mundo han desarrollado diseños propios, basados en componentes de más fácil acceso. Entre los ejemplos más ampliamente citados se encuentran MIT RACECAR (<https://racecar.mit.edu>), desarrollado por el Massachusset Institute of Technology a un costo estimado de USD 2800, y MuSHR (<https://mushr.io>), desarrollado por University of Washington, a un costo estimado de USD 900. Ambos corresponden a robots móviles diseñados para educación e investigación y cuentan con información pública detallada y software abierto [12].

La opción de las instituciones educativas por desarrollar diseños propios está motivada también por la posibilidad de construir sistemas abiertos que faciliten la actualización e innovación. Esto porque la evidencia empírica muestra que los usuarios finales –profesores y estudiantes– aun cuando tengan la posibilidad de contar con un robot comercial ya terminado, de todas formas buscan realizar modificaciones en el hardware o el software para ajustar el sistema a sus necesidades. Por este motivo, resulta deseable contar con sistemas modulares y flexibles, que puedan ser empleados en numerosas prácticas y actividades educativas diversas. Desde el punto de vista pedagógico, este tipo de sistemas facilita su uso en estrategias de aprendizaje basadas en proyectos, preferidas para desarrollar habilidades de creatividad y resolución de problemas. Desde el punto de vista de la gestión, es una forma de maximizar el uso de un recurso sobre el cual se ha hecho una inversión cuantiosa en equipamiento y capacitación, y retrasar su obsolescencia.

Estos diseños abiertos han venido mostrando ser la mejor alternativa para reducir costos en la implementación de sistemas de robótica móvil, al mismo tiempo que su propio desarrollo capacita al personal docente y promueve la creación de una comunidad educativa que colabora en la introducción de estas nuevas tecnologías.

En Santiago de Chile, el Centro de Entrenamiento en Automatización y Robótica de Instituto Profesional Inacap ha adoptado este enfoque y ha desarrollado una tarjeta de control para robots móviles bajo la filosofía *open-source*, denominada CFV2. Esta tarjeta de control ha sido diseñada específicamente para facilitar la construcción de robots destinados a educación e investigación, integrando características que permiten construir diferentes tipos de robots con componentes de bajo costo, pero también ser integrada a sistemas avanzados de robótica desarrollados sobre la plataforma ROS2.

En las siguientes secciones se presentan los requerimientos de diseño que motivaron el desarrollo de CFV2, la arquitectura de hardware y software implementadas, las posibilidades que entrega para el control cinemático y dinámico de robots móviles y un ejemplo de su aplicación en el desarrollo de un robot autónomo con fines educativos.

II. REQUERIMIENTOS DE DISEÑO

La tarjeta de control CFV2 tiene como propósito disponer, en una única placa de circuito impreso, de toda la electrónica necesaria para implementar robots móviles de distinto tipo y características, con propósitos de educación e investigación.

Los requerimientos técnicos que fueron tenidos en cuenta para la selección de los componentes y el diseño de la tarjeta de control incluían:

- Controlar el sentido de giro y la velocidad de hasta cuatro motores de corriente continua en forma independiente.
- La posibilidad de utilizar motores de 6 V, 12 V ó 24 V y manejar corrientes de hasta 2 A en forma continua.
- Registrar la posición y velocidad de rotación de cada motor utilizando *encoders* en cuadratura.
- Registrar el consumo instantáneo de corriente de cada motor en forma independiente.
- Contar con elementos de protección frente a sobrecargas o cortocircuitos en los motores.
- Contar con información de velocidad, aceleración y orientación del robot mediante IMU (Inertial Measurement Unit).
- Disponer de comunicaciones Bluetooth, Wi-Fi y una UART (Universal Asynchronous Receiver/Transmitter) cableada que permita implementar protocolos de comunicaciones de bus de campo, tales como CAN-bus.
- Disponer de accesorios de señalización luminosa y sonora que permitan conocer el estado del controlador.
- Controlar todo desde un microcontrolador que resulte simple de programar para quien esté familiarizado con los dispositivos Arduino.

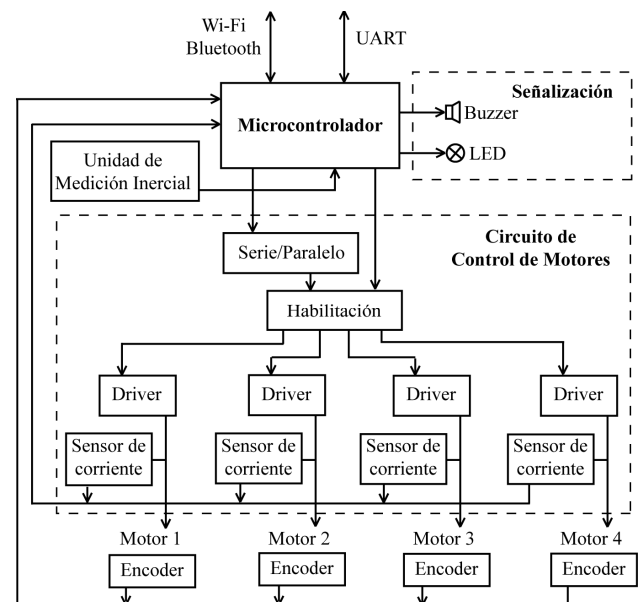


Fig. 1. Componentes de la tarjeta de control de robots CFV2.

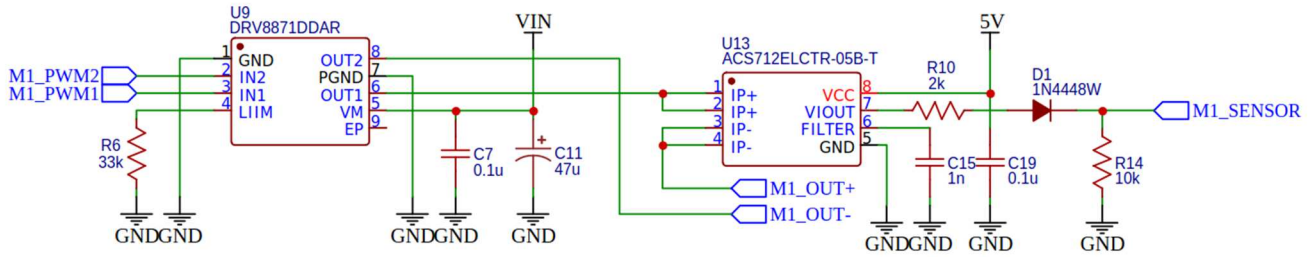


Fig. 2. Detalle de la etapa de control de motores para el motor M1: conexiones del driver DVR8871 y sensor de corriente ACS712.

III. ARQUITECTURA DE HARDWARE

El diagrama de para la tarjeta de control de robots CFV2 se muestra en la Fig. 1. Las distintas etapas, sus componentes y características principales se describen a continuación. El diseño electrónico ha sido realizado bajo la filosofía de *open-source hardware*. Tanto los detalles del circuito esquemático como el diseño de la placa de circuito impreso (PCB) se encuentran disponibles en Open Source Hardware Lab (<https://oshwlab.com/roboticainacap/control-robot-movil/>), y pueden ser utilizados y/o modificados libremente bajo licencia GPL 3.0.

A. Circuito de Control de Motores

El control de cada motor se realiza mediante un driver DRV8871, de Texas Instruments, el cual consiste en un circuito integrado de puente H que permite manejar cargas de hasta 3.5 A en forma continua a un voltaje de hasta 50 V. Este circuito integrado cuenta además con limitador de corriente, el cual que ha sido ajustado a 2.1 A, con lo cual se garantiza que el chip funcionará en una zona muy cómoda en cuanto a disipación de calor durante su uso continuo, incluso considerando su uso con motores alimentados a 24 V. El control de velocidad se realiza mediante PWM, a frecuencia de hasta 5 kHz.

A las conexiones que alimentan cada motor se ha agregado un sensor de corriente de efecto Hall ACS712, de Allegro Microsystems. Este circuito integrado permite sensar la corriente de cada motor mediante una señal proporcional de voltaje, la cual se envía directamente a una entrada analógica de conversión análogo/digital (ADC) del microcontrolador.

Las conexiones del driver DRV8871 y el sensor de corriente ACS712 se muestran en la Fig.2. La tarjeta CFV2 cuenta con cuatro circuitos independientes de este tipo, los cuales se controlan en forma conjunta mediante un circuito lógico de selección. Este conjunto de componentes conforma el Circuito de Control de Motores. El circuito lógico de selección habilita y define el sentido de giro de todos los motores simultáneamente, utilizando un registro de desplazamiento serie a paralelo. El circuito de habilitación permite utilizar una única señal PWM proveniente del microcontrolador para controlar simultáneamente la velocidad de los 4 motores. Mediante ambos circuitos es posible utilizar solo 4 pines GPIO (General Purpose Input/Output) del microcontrolador, en lugar de los 12 pines GPIO que serían requeridos para controlar individualmente de cada uno de los motores.

B. Unidad de Medición Inercial

Para la medición de los parámetros inerciales del robot se ha seleccionado el circuito integrado MPU-6050 de InvenSense Inc. Este circuito integrado contiene una unidad de giroscopio y acelerómetro de 3 ejes que entrega datos directamente al microcontrolador mediante comunicación I²C.

Las escalas de medición para los parámetros de giroscopio y acelerómetro son completamente programables mediante software, a fin de ajustarse a las necesidades específicas de cada robot que pueda implementarse utilizando la tarjeta CFV2. En el diseño de la placa de circuito impreso, se ha considerado la ubicación del chip MPU-6050 exactamente en el centro de la tarjeta y se ha indicado la orientación de los ejes de movimiento, a fin de facilitar los ajustes de lectura y configuración.

C. Microcontrolador

Para el control central se ha seleccionado el dispositivo SoM (System on Module) ESP-WROOM-32 de Espressif Systems, un componente de bajo costo y de prestaciones muy adecuadas para los requerimientos de esta tarjeta controladora.

Además de contar con un microcontrolador ESP32 de 32 bits a 240 MHz, con 4MB para memoria de programa y 512 MB de SRAM, este componente cuenta con comunicaciones Wi-Fi y Bluetooth integradas. Para la programación del ESP32 es posible utilizar C++ en entornos tan populares como Arduino IDE o PlatformIO, ambos con amplio soporte y documentación en la web, lo cual permite reducir notablemente la curva de aprendizaje. La programación también puede realizarse mediante MicroPython, LUA o Javascript, entre otros lenguajes de programación. En cuanto a la conexión física para programar el microcontrolador, en la PCB se ha incluido un circuito integrado CH340E, que permite cargar el código de programa mediante una conexión USB y establecer también una conexión serial con un computador. Un par de pulsadores han sido agregados a la tarjeta para poner al microcontrolador en modo de programación y reiniciar.

D. Fuente de Poder

La tarjeta CFV2 ha sido diseñada para ser alimentada mediante una única batería externa, la cual entrega energía simultáneamente a los motores y a los circuitos internos. El voltaje de las baterías puede seleccionarse libremente para ajustarse a los requerimientos de los motores, entre 7.2 y 28 V.

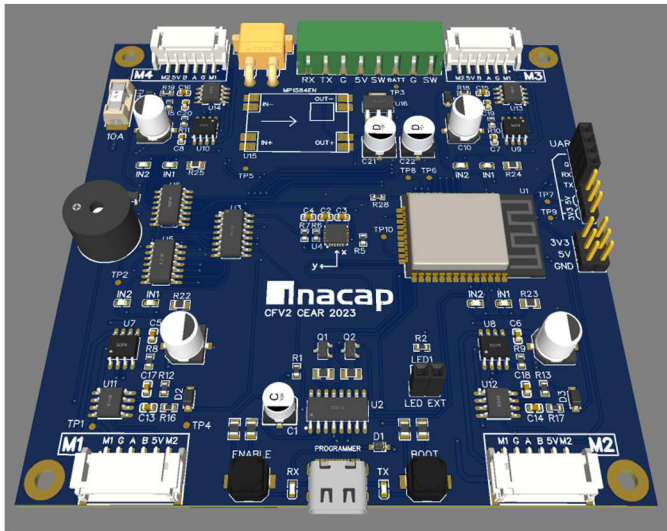


Fig. 3. Vista de la placa de circuito impreso de la tarjeta de control CFV2.

La alimentación a los driver de los motores se hace llegar directamente desde las baterías, con una protección general mediante fusible de 10 A. Para alimentar las etapas electrónicas se cuenta con dos etapas de conversión Step-Down, a fin de conseguir voltajes estabilizados y filtrados a 5V y 3.3V, empleando un módulo MP1584 de Monolithic Power Systems y un circuito integrado AMS1177-3.3 de Advanced Monolithic Systems. Terminales de alimentación para elementos externos a 5 V y 3.3 V han sido incluidos en la placa de circuito impreso, a fin de que puedan ser usados para energizar otros componentes.

E. Elementos Auxiliares

Varios elementos auxiliares han sido incluidos, con el propósito de facilitar la conexión de elementos externos e informar a los usuarios acerca del estado de la tarjeta.

Además de un indicador visual mediante LED (Light Emitting Diode) que indica que la tarjeta está energizada, en el driver que controla cada uno de los motores se han agregado LEDs que encienden junto con la señal PWM y, adicionalmente, indican el sentido de giro.

Se ha añadido un indicador sonoro tipo *buzzer* y un LED de propósito general, los cuales se controlan independientemente desde el microcontrolador. Ambos dispositivos están destinados a entregar mensajes luminosos y sonoros que pueden ser configurados a voluntad para informar sobre el estado del robot.

Se han agregado también terminales para la comunicación con elementos externos mediante una UART, con lo cual es posible insertar la tarjeta CFV2 en un bus de campo y permitir el envío y recepción de mensajes mediante protocolos de estándar abierto, tales como Modbus-RTU o CAN-bus. Mediante estos mismos terminales es posible comunicar el robot utilizando otros sistemas inalámbricos de más largo alcance, como puede ser ZigBee.

Todos los circuitos descritos, con los componentes auxiliares y conectores correspondientes, se han alojado en una placa de circuito impreso (PCB) de doble faz, con dimensiones

96 x 96 mm utilizando componentes de montaje superficial. Una vista del diseño de placa de circuito impreso resultante se muestra en la Fig. 3.

IV. ARQUITECTURA DE SOFTWARE

Todos los componentes de software para la tarjeta CFV2 han sido desarrollados y son mantenidos por el Centro de Entrenamiento en Automatización y Robótica (CEAR) de Inacap. El código fuente implementado en C++ y Python, la documentación, códigos de prueba y ejemplos han sido puestos a disposición de la comunidad bajo licencia GPL 3.0, en un repositorio de Github (<https://github.com/cear-inacap/>).

La arquitectura de software se conforma de cuatro componentes principales: (1) comunicaciones (comm); (2) módulo de medición de parámetros inerciales IMU (imu); (3) control de movimiento (motion_control); (4) información de estado (status_info), (5) control principal (main control).

El módulo de comunicaciones contiene las configuraciones y funciones necesarias para enviar y recibir datos entre CFV2 y dispositivos externos, por medio de enlaces inalámbricos Wi-Fi o Bluetooth o mediante comunicación serial cableada. En el módulo IMU se han implementado las funciones para la configuración y captura de datos desde la unidad de medición inercial MPU-6050. El control de movimiento, por su parte, se compone de un conjunto de funciones que se encargan del control cinemático y dinámico de los motores en respuesta a las instrucciones de velocidad de movimiento lineal y rotacional que son enviadas al robot. Las funciones para información de estado se encargan de establecer las señales luminosas y auditivas que informan al usuario sobre estados del robot y alertas. Por último, el control principal se encarga de levantar los servicios y coordinar la ejecución de los diferentes componentes de software.

El control principal se ha desarrollado en dos versiones: una versión de pruebas que permite el control del robot utilizando simplemente instrucciones de texto codificado en ASCII y otra versión que inicializa independientemente cada componente de software como un nodo, servicio o acción de ROS2. Utilizando ROS2 se han construido también los componentes de software necesarios para interconectar la tarjeta CFV2 con cualquier otro tipo de sensores, actuadores y dispositivos de control, facilitando con ello la construcción de robots avanzados destinados a la educación e investigación en el campo de la robótica móvil.

V. CFV2 PARA EDUCACIÓN E INVESTIGACIÓN

La tarjeta controladora CFV2 puede ser adquirida utilizando diversos servicios de fabricación de PCB que se ofertan a través de Internet, a partir de los archivos de diseño que han sido puestos a disposición. El costo de fabricación de una tarjeta, con los componentes ya montados, es de alrededor de USD 95 más gastos de envío.

La funcionalidad de esta tarjeta ha sido validada con la implementación de un robot móvil holonómico para Instituto Profesional Inacap, el cual ha pasado a formar parte del equipamiento estándar para las asignaturas de Introducción a la Automatización y Robótica y Robótica Avanzada, en el plan de estudios de Ingeniería en Automatización y Robótica.



Fig. 4. Robot educativo que utiliza la tarjeta de control CFV2 en un chasis fabricado por HexMan Robotics.

Este robot móvil ha sido implementado utilizando el chasis y componentes del robot YORK, fabricado por HexMan Robotics. El sistema de tracción está compuesto por 4 motores de corriente continua de 24 V, con caja reductora a 90 rpm, que entregan un torque de 9 kg/cm² y cuentan con *encoder* en cuadratura acoplado al eje del motor. Las ruedas son de tipo *mecanum* de 96 mm de diámetro. La tarjeta CFV2 controla los movimientos del robot. La integración con componentes adicionales permite el desarrollo de algoritmos avanzados de navegación, los cuales se han implementado en un computador de placa reducida Nvidia Jetson Nano, el cual a su vez recibe datos desde un LiDAR 2D modelo D300-LD19 de LDRobot y una cámara de profundidad YDSENS RGB-D. La Fig. 4 muestra el aspecto físico de este robot, el cual incluye la tarjeta de control CFV2 en su interior.

La configuración básica de este robot, considerando únicamente el chasis, batería, ruedas, tarjeta de control y algunos accesorios, fue fabricado a un costo de alrededor de USD 850 más impuestos de importación. Esta configuración básica resulta suficiente para realizar prácticas de programación de robots teleoperados, para los cuales toda la programación puede hacerse directamente sobre el microcontrolador ESP32 que está integrado en la tarjeta de control CFV2.

En la configuración completa se incluye una computadora Nvidia Jetson Nano, un sensor LiDAR y una cámara RGB-D, que permiten implementar algoritmos avanzados de localización y navegación autónoma. El costo total de componentes para esta implementación resulta en alrededor de USD 2000 más costos de importación, y entrega prestaciones y funcionalidades equivalentes a plataformas móviles comerciales del rango de USD 4000 a USD 8000, tales como Rosbot Pro o Mecabot X de Roboworks.

VI. RESULTADOS

Dos características relevantes de la tarjeta CFV2 se presentan a continuación, mediante la presentación de resultados de su aplicación en el robot móvil de la Fig.4., éstas son: 1) la medición de corriente de los motores y 2) el registro de movimiento mediante encoders.

La medición de corriente en cada motor permite contar con retroalimentación para los algoritmos de control de torque, con lo cual se hace posible tener control sobre la velocidad del robot. Un algoritmo para la medición de corriente de los motores ha sido implementado y testeado, con el propósito de conocer el comportamiento de los motores desde el estado de reposo del robot hasta alcanzar velocidad máxima. Para esta prueba, se encendieron simultáneamente los cuatro motores a velocidad máxima, todos en el mismo sentido de giro, y se midió la corriente de uno de los motores, a una tasa de muestreo de 1 ms, por medio de los circuitos que conforman la tarjeta CFV2. En la Fig. 5 se muestra gráficamente el resultado de esta medición. La gráfica resultante refleja con buen detalle la corriente de partida del motor, mostrando un peak de corriente de aproximadamente 1.5 A al momento del encendido, con duración de aproximadamente 150 ms, luego de lo cual la corriente se estabiliza en aproximadamente 100 mA, que es la corriente nominal de los motores. Aun cuando los datos leídos directamente desde el circuito de medición de corriente tienen bastante ruido, en la gráfica se aprecia que un sencillo filtro digital de valor promedio con 10 muestras es suficiente para tener una representación adecuada de la corriente del motor.

La detección del movimiento de las ruedas mediante encoder, por otra parte, entrega información relevante para la odometría del robot. La tarjeta CFV2 permite la lectura simultánea de cuatro encoders en cuadratura, acoplados a los ejes de cada uno de los motores del robot. La lectura de los pulsos provenientes de estos encoders permite conocer la velocidad y sentido de giro de cada rueda. Mediante conversiones matemáticas, estos datos permiten estimar la posición del robot.

En la Fig. 6 se muestra el registro de pulsos acumulados de cada uno de los motores, para una rutina de demostración consistente en una secuencia cuatro movimientos. Los datos recibidos desde los encoders fueron procesados en el microcontrolador, sumando en un acumulador los pulsos que

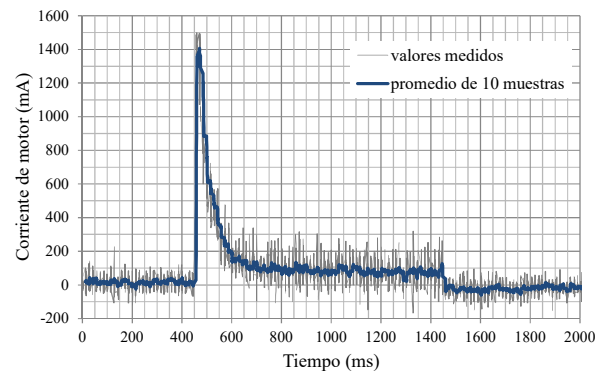


Fig. 5. Corriente de partida de uno de los motores del robot de pruebas, registrada por medio de los circuitos internos de la tarjeta CFV2. Se observa que un filtro de valor promedio con 10 muestras es suficiente para una correcta representación de la magnitud.

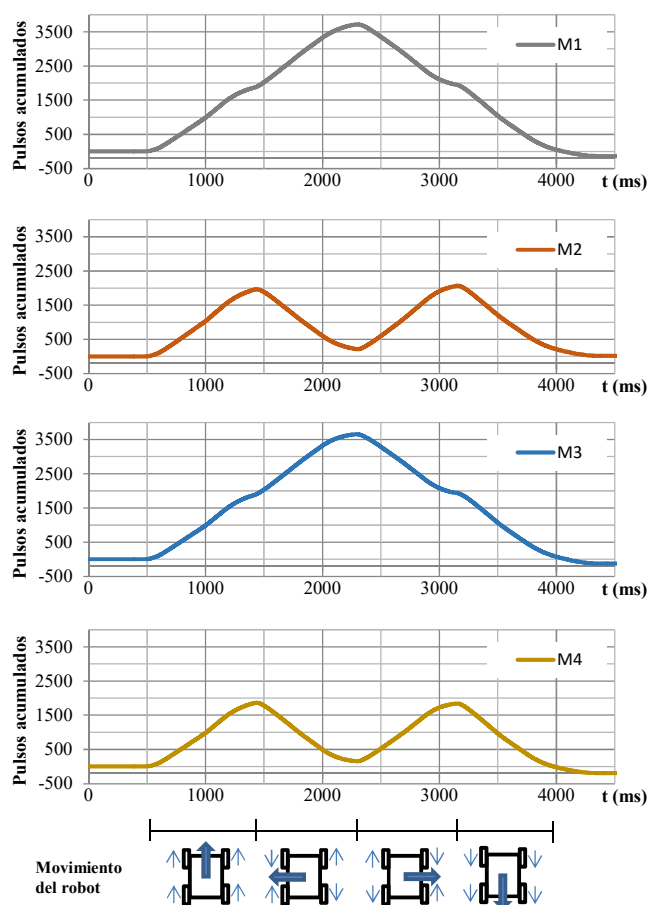


Fig. 6. Registro de los encoder acoplados a cada uno de los motores M1 a M4, para una rutina de movimientos de demostración.

representan movimiento en el sentido avance y restando los pulsos que representan movimiento en sentido opuesto. Para esta rutina de demostración, la posición final del robot es igual a la posición inicial. Los gráficos de la figura 6 muestran los pulsos acumulados para cada uno de los cuatro motores por separado. Se observa que cada motor retorna a la posición inicial luego de concluir la rutina, según lo esperado.

VII. CONCLUSIONES

Con el desarrollo de la tarjeta de control CFV2 en Instituto Profesional Inacap se pudieron reducir los costos de adquisición de equipamiento educacional hasta el nivel que ha sido posible contar con un robot móvil de altas prestaciones para cada una de las 15 sedes que imparten la carrera de Ingeniería en Automatización y Robótica. Las actividades didácticas que actualmente se preparan con el uso de la tarjeta CFV2 serán aplicadas en diversas asignaturas y momentos del plan de estudio, cubriendo desde los conceptos introductorios de automatización y robótica, electrónica, programación, redes de comunicaciones y sistemas de control, hasta alcanzar los

tópicos más avanzados de visión artificial, inteligencia artificial, optimización y técnicas de mapeo y localización.

El diseño de la tarjeta CFV2, como también los algoritmos de software, han sido puestos a disposición del público en general bajo licencia GPL 3.0. De esta forma, cualquier institución educativa –y también entusiastas de la robótica– pueden utilizar libremente este diseño en su forma actual o realizar modificaciones, además de colaborar con el desarrollo de mejoras y nuevas aplicaciones.

AGRADECIMIENTOS

El desarrollo de CFV2 ha sido financiado por el Centro de Entrenamiento en Automatización y Robótica de Instituto Profesional Inacap, Puente Alto, Santiago, Chile. Agradecemos la confianza y facilidades para llevar adelante este trabajo.

REFERENCIAS

- [1] M. Hernandez-de-Menendez, C. Escobar, R. Morales-Menendez, "Engineering education for Smart 4.0 technology: a review", *International Journal on Interactive Design and Manufacturing*, vol 14, pp 789-803, 2020.
- [2] M. Lanz, R. Pieters, R. Ghabcheloo, "Learning environment for robotics education and industry-academia collaboration", *Procedia Manufacturing*, 31, 79-84.
- [3] E. Altamirano, D. Prado, I. Diaz, C. Vásquez, "A Digital Twin Implementation for mobile and collaborative robot scenarios for teaching robotics based on Robot Operating System", *2022 IEEE Global Engineering Education Conference (EDUCON)*, Tunis, Tunisia, pp. 556-564, 2022.
- [4] S. Evripidou, K. Georgiou, et al., "Educational robotics: Platforms, competition and expected learning outcomes", *IEEE Access*, vol 8, pp 219534-219562, December 2020.
- [5] S. Elayyan, "The future of education according to the fourth industrial revolution", *Journal of Educational Technology and Online Learning*, vol 4 (1), pp 23-30.
- [6] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woododall, "Robot Operating System 2: Design, architecture, and uses in the wild," *Science Robotis*, vol. 7, May 2022.
- [7] R. Amsters and P. Slaets, "Turtlebot 3 as a robotics education platform" in "Robotics in education", *Advances in Intelligent Systems and Computing*, vol 1023, Springer, 2019.
- [8] F. Arvin, J. Espinosa, B. Bird, A. West, S. Watson and B. Lennox, "Mona: an affordable open-source mobile robot for education and research", *Journal of Intelligent & Robotic Systems*, vol 94, pp 761-775, 2019.
- [9] J. Cañas, E. Perdices, L. García-Pérez and J. Fernández-Conde, "A ROS-based open tool for intelligent robotics education", in *Advances in Artificial Intelligence Learning Technologies, Applied Sciences*, 10(21), 7419, 2020.
- [10] S. Tselegkaridis and T. Sapounidis, "Simulators in educational robotics: A review", *Education Science*, vol 11 (1), 2021.
- [11] C. Lytridis, C. Bazinas and G. Papakostas and V. Kaburlasos, "On measuring engagement level during child-robot interaction in education", in *Robotics in education, RiE 2019, Advances in Intelligent Systems ad Computing*, vol 1023.
- [12] S. Srinivasa, P. Lancaster, J. Michalove, M. Schmittle, C. Summers, M. Rockett, and F. Sadeghi, "MuSHR: A low-cost, open-source robotic racecar for education and research", arXiv preprint arXiv:1908.08031, 2019.