

Laboratorio remoto en cinemática: Una alternativa para la enseñanza de competencias científicas en la era digital

Remote laboratory in kinematics: An Alternative for Teaching Scientific Skills in the Digital Age

Lucas Gelatti¹, Francisco Esteban¹, Federico Rosales¹ y Sergio Ribotta¹

¹ Facultad de Ingeniería y Ciencias Agropecuarias
Universidad Nacional de San Luis

lucasgelatti97@gmail.com, festeban@unsl.edu.ar, fgrosale@gmail.com, sergioribotta@gmail.com

Científica N°1

Año: 2024

pp. 5 – 15

Historial del Artículo

Recibido:

12/04/2024

Enviado revisión:

19/04/2024

Aceptado:

06/05/2024

Área temática:

Enseñanza de la
ingeniería

Resumen: Este trabajo aborda el tema de los laboratorios remotos como una alternativa para el proceso de enseñanza de competencias científicas y tecnológicas en un entorno educativo cada vez más digitalizado. La pandemia de COVID-19 ha impulsado la necesidad de adaptar los métodos educativos tradicionales a modelos virtuales, lo que ha planteado desafíos para la interacción y formación práctica de los estudiantes. Los laboratorios remotos se presentan como una opción para proporcionar experiencias de aprendizaje prácticas y reales a través de internet, permitiendo a los estudiantes acceder y manipular equipos de laboratorio de manera remota. Este enfoque innovador ha demostrado ser efectivo para mejorar la enseñanza en diversos niveles educativos, especialmente en disciplinas científicas y de ingeniería. En este trabajo, se describe el desarrollo de un laboratorio remoto en cinemática llevado a cabo por un equipo de trabajo colaborativo, integrando tecnologías de hardware y software para su funcionamiento. Se espera que este laboratorio remoto contribuya al fortalecimiento de la educación en ciencias e ingenierías en modalidades a distancia y mixtas.

Palabras Claves:

Laboratorio remoto,
cinemática, ingeniería.

Keywords:

remote laboratory,
kinematics,
engineering.

Abstract: This paper addresses the topic of remote laboratories as an alternative for the teaching process of scientific and technological skills in an increasingly digitized educational environment. The COVID-19 pandemic has driven the need to adapt traditional educational methods to virtual models, posing challenges for student interaction and practical training. Remote laboratories emerge as an option to provide practical and real learning experiences over the internet, allowing students to access and manipulate laboratory equipment remotely. This innovative approach has proven effective in enhancing teaching across various educational levels, especially in scientific and engineering disciplines. The paper describes the development of a remote laboratory in kinematics carried out by a collaborative team, integrating hardware and software technologies for its operation. It is anticipated that this remote laboratory will contribute to strengthening science and engineering education in distance and blended learning modalities.

INTRODUCCIÓN

En el contexto actual de la educación superior, la integración de tecnologías digitales en la enseñanza se ha vuelto una necesidad imperante, debido a factores como la digitalización y la pandemia de COVID-19. Investigaciones recientes destacan la importancia de estas tecnologías para mejorar la calidad de la educación y la experiencia de aprendizaje de los estudiantes [1] y [2]. Con el advenimiento de la educación a distancia y la creciente demanda por experiencias de aprendizaje prácticas y accesibles desde cualquier lugar, surge la imperiosa necesidad de desarrollar plataformas educativas innovadoras que permitan a los estudiantes interactuar de manera efectiva con conceptos complejos, como es el caso de la cinemática en la física.

En este sentido, el presente artículo presenta el diseño y desarrollo de una plataforma remota diseñada específicamente para la enseñanza de cinemática en el ámbito universitario. Particularmente, la plataforma diseñada consiste en un móvil (vehículo a escala) que se desplaza por un riel cuya inclinación puede modificarse de manera remota. De esta manera puede realizarse un ensayo de movimiento rectilíneo uniformemente acelerado experimental de forma remota, y obtener curvas de velocidad, distancia y aceleración con el objetivo de que los estudiantes puedan comparar sus resultados teóricos con los obtenidos de manera experimental.

Esta plataforma está concebida para brindar a los estudiantes una experiencia real y práctica en el estudio de los movimientos, trayectorias y velocidades de los objetos, sin limitaciones geográficas ni de dispositivos, permitiendo su acceso desde cualquier computadora o dispositivo móvil conectado a internet.

El diseño de esta plataforma considera aspectos clave como la interactividad, la visualización ensayos realistas y la retroalimentación inmediata, con el objetivo de mejorar la comprensión de los conceptos de cinemática y fomentar un aprendizaje activo y significativo entre los estudiantes [3], [4] y [5]

A través de este artículo, se detallarán los aspectos técnicos y metodológicos que sustentan el diseño de esta plataforma, así como los resultados preliminares obtenidos en su implementación y evaluación piloto en un entorno universitario [6]. Se espera que este trabajo contribuya al avance en la integración efectiva de tecnologías educativas en la enseñanza de la física y áreas afines, promoviendo una educación más accesible, dinámica y orientada al desarrollo de competencias prácticas en los estudiantes.

MATERIALES Y MÉTODOS

El equipo de trabajo conformado por docentes y estudiantes de diversas carreras de ingeniería ha desarrollado un laboratorio remoto como parte de un proyecto de investigación. Para ello, se han utilizado tecnologías de la información y comunicación, hardware y software especializado para el control y manipulación de los equipos de laboratorio a distancia. El diseño y construcción del laboratorio remoto se ha llevado a cabo siguiendo criterios de accesibilidad, interactividad y seguridad para los usuarios.

Sistema mecánico implementado

En la siguiente subsección, se presenta el sistema mecánico implementado para el laboratorio remoto. El mismo posee partes fijas (base, riel, y sistema de frenado) que sirven de soporte para las partes móviles y permiten el frenado del móvil plástico.

A continuación, se mencionan los componentes principales del sistema implementado:

1. **Base:** La base del laboratorio se encuentra constituida por una tabla de madera en donde van montadas el soporte pivotante y el mecanismo de pivote.
2. **Riel y móvil:** Este experimento usa un riel de aluminio de 1,2m de largo y un móvil plástico de 4 ruedas. Estos elementos, al momento de la elaboración de este documento, son comercializados por la marca Pasco como parte de su kit "Dynamic Kart & Track System" [7].
3. **Soporte pivotante:** Esta pieza es el soporte principal sobre el cual rota el riel. Se encuentra compuesto por 8 piezas plásticas de PLA reciclable, fabricadas por método de deposición fundida (*Fused Deposition Modeling, FDM*), también llamado impresión 3D. Cada una de las piezas se encuentra unida mediante tornillos métricos M3 y M6, y la base cuenta con 4 orificios M8 que sirven para fijarla.

4. **Mecanismo de retención y frenado:** El mecanismo de retención es una pieza fabricada por FDM en Ácido poliláctico (*Polylactic Acid, PLA*), que se encuentra montada en el extremo izquierdo del riel. Este dispositivo cuenta con un electroimán comandado por un actuador a relé y además se encuentra protegido con un dispositivo de protección térmica para prevenir sobrecalentamientos. Su principal función es retener el móvil mediante la fuerza de atracción magnética producida por la interacción entre el núcleo del electroimán y los imanes de neodimio montados de fábrica en el móvil. En el extremo derecho del riel se encuentra montada una pieza plástica acolchada que sirve de tope para producir el frenado del móvil. Esta última viene incluida con el kit comercial mencionado anteriormente. Por último, se tiene montado en cada extremo del riel unas contenciones laterales para evitar el descarrilamiento del móvil. Estas contenciones también fueron fabricadas en PLA por el método FDM.
5. **Mecanismo de pivote:** El mecanismo de pivote que hace variar la inclinación de la rampa consta de una tuerca y un tornillo sin fin, el cual es conducido por un motor paso a paso NEMA17. Además, cuenta con distintos elementos adicionales de montaje.
6. **Soporte de sensores de barrera laser:** Estos soportes se encuentran montados a los lados del riel, fueron fabricados por FDM en plástico PLA reciclable. Dos tipos de soporte son utilizados, uno para el emisor y otro para el receptor. En total sobre el riel hay montados ocho soportes, cuatro de cada lado. De un lado los 4 emisores, y del otro, los 4 receptores.

En la Fig. 1. se muestra una ilustración de la plataforma diseñada, con sus respectivas partes detalladas.

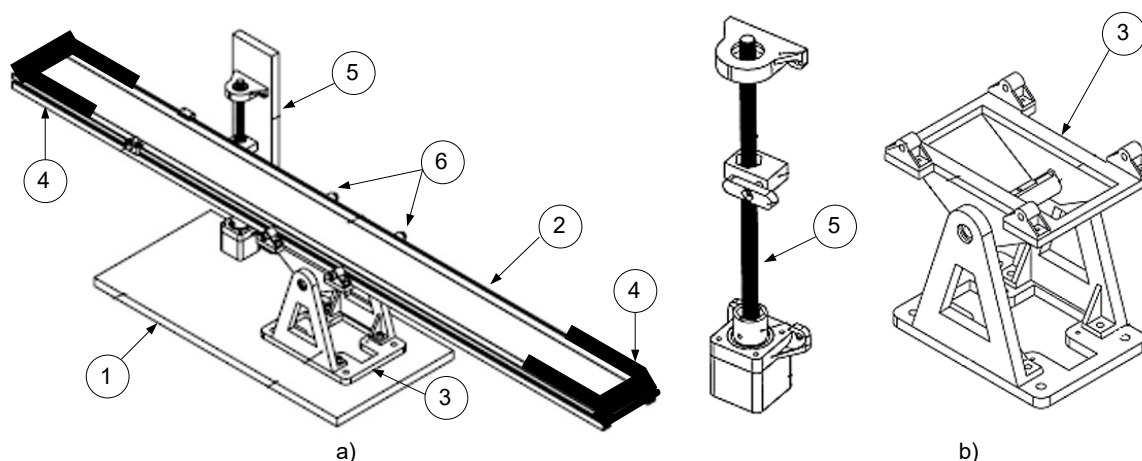


Figura 1. Ilustración de la plataforma implementada. a) Plataforma completa. b) Detalle del mecanismo de pivote (5) y soporte pivotante (3).

Descripción del sistema eléctrico/electrónico implementado

En la siguiente subsección se especifica cómo está constituido el sistema eléctrico/electrónico que permite alimentar a los sistemas de accionamiento y sensado de variables a medir. Este se divide en tres partes que se comunican entre sí: base, móvil y servidor.

El subsistema eléctrico/electrónico de la base está formado por las siguientes partes:

- **Microcontrolador.** El microcontrolador utilizado es una placa de desarrollo NODEMCU que está basado en el micro ESP32 [8]. Sus principales funciones son:
 1. Conexión a la red wifi del laboratorio.
 2. Comunicación con Raspberry Pi 4 a través del protocolo MQTT.
 3. Controlar el motor NEMA17 a través del driver para modificar la inclinación del riel.
 4. Calibrar y leer los datos de los sensores.
 5. Calcular el ángulo de inclinación del riel mediante la integración de la velocidad angular proveniente del acelerómetro.

Fuente de alimentación principal: Se conecta a la red eléctrica de 220V (corriente alterna), y entrega una tensión de corriente continua de 12V.

- **Fuentes reductoras (step-down):** Se utilizan dos fuentes reductoras que convierten la tensión de la fuente de alimentación en un valor menor especificado por un potenciómetro. Una de estas fuentes se ajustó a una tensión de salida de 5V que sirve para alimentar el microcontrolador y los emisores de luz láser. La otra se ajustó a una tensión de 8.4V y se utiliza para cargar la batería del móvil.
- **Motor:** El motor utilizado es un paso a paso NEMA17 [9] que se utiliza para variar la inclinación de la rampa. Es controlado por un driver DRV8825 [10] que se encarga de generar las señales adecuadas para el correcto funcionamiento del motor. Este driver a su vez es controlado por el microcontrolador.
- **Sensores:** Los sensores con los que cuenta la base son:
 - Un acelerómetro MPU6050 [11]: mide la aceleración rotacional de la rampa y en base a este dato se calcula el ángulo.
 - Un final de carrera: se emplea para indicar que el riel se encuentra en una inclinación conocida que se utiliza de referencia para ajustar la posición angular del riel al valor deseado por el usuario.
 - Sensores de barrera laser: en el riel se encuentran cuatro de estos sensores que se utilizan para determinar el tiempo desde que se liberó el móvil hasta que se corta la barrera laser. Cada uno se encuentra constituido por un receptor y un emisor. Cada receptor que se utiliza cuenta con una fotorresistencia LDR-GL5516 [12], en la que cuando impacta la luz del emisor sobre la misma, su valor óhmico disminuye y es detectado por una GPIO (Entrada/Salida de propósito general, por sus siglas en inglés) del microcontrolador. Para cada emisor se utiliza un módulo comercial KY-008 que cuenta con un diodo emisor de luz láser que ilumina al receptor.
- **Otros componentes:** Otros componentes adicionales con los que cuenta el sistema implementado son:
 - Electroimán: se utiliza para sujetar el móvil en la posición inicial hasta que es energizado.
 - Módulo Relé: permite adaptar la tensión de salida del microcontrolador a la tensión de alimentación del electroimán.
 - Interruptor térmico de protección: Se utiliza para proteger el electroimán. El mismo interrumpe el circuito cuando la temperatura alcanzada por el electroimán supera el límite establecido.

El subsistema eléctrico/electrónico del servidor está formado por las siguientes partes:

- **Raspberry:** Este dispositivo es una computadora embebida en una sola placa [13]. Se alimenta a través de un cargador USB-C de 3,1A que va conectado directamente a la red domiciliar de 220V de corriente alterna. Sus funciones principales son:
 1. Proporcionar una interfaz web para la interacción del usuario con el laboratorio.
 2. Transmitir video en vivo del experimento.
 3. Montar un servidor de tipo Transporte de telemetría de colas de mensajes (*Message Queuing Telemetry Transport*, **MQTT**) para el envío de datos y comandos con el microcontrolador.
 4. Generar una red WIFI cerrada en donde se conectan los microcontroladores.

En particular, el modelo utilizado presenta las siguientes especificaciones:

- **Modelo:** Raspberry Pi 4 Computer Model B
- **Memoria RAM:** 8GB
- **Cámara web:** La plataforma cuenta con una cámara web HD modelo Xiaovv FullHD 1080p que graba el video en vivo del ensayo. La misma, se conecta a través de un cable USB a la Raspberry Pi

En la Fig. 2 se muestra un diagrama en bloques que representa la alimentación eléctrica de todos los módulos antes especificados.

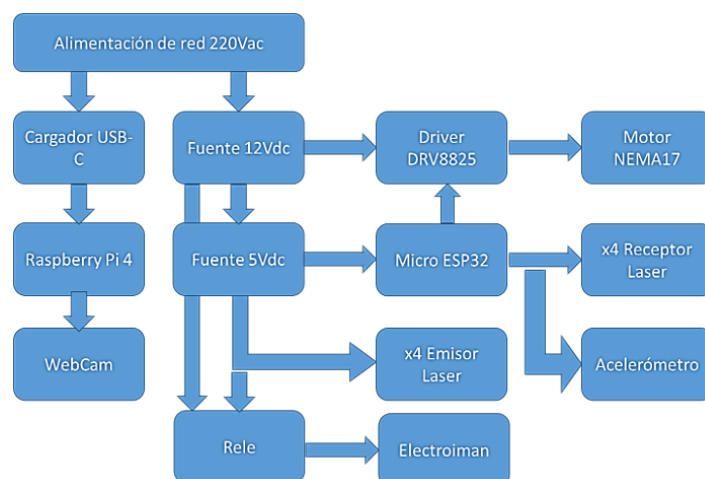


Figura 2. Diagrama de bloques del esquema de alimentación del servidor y la base.

Por último, el subsistema eléctrico/electrónico del móvil está formado por las siguientes partes:

- **Batería:** Está formada por dos pilas de ION DE LITIO, y un módulo de protección (*Battery Management System, BMS*). Dicha batería se encarga de alimentar el microcontrolador.
- **Microcontrolador.** El microcontrolador utilizado es una placa de desarrollo basada en ESP-32.

Sus principales funciones son:

1. Conectarse a la red wifi generada por la Raspberry
2. Adquirir los datos del acelerómetro
3. Enviarlos a la Raspberry a través del protocolo MQTT.

- **Sensor:** Es un acelerómetro MPU6050 encargado de adquirir datos de la aceleración del móvil de acuerdo al tiempo de muestreo elegido.
- **Interruptor principal:** Permite el apagado del móvil cuando no se está utilizando.
- **Conector:** es la ficha donde se enchufa el cargador para energizar las baterías.

A continuación, se desarrolla la etapa de alimentación utilizada para energizar el vehículo (móvil) empleado para realizar el ensayo. En la Fig. 3 se muestra un diagrama en bloques de la etapa de alimentación diseñada.

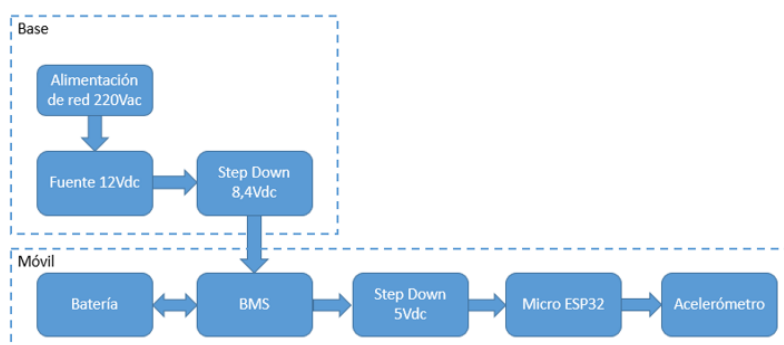


Figura 3. Diagrama de bloques del esquema de alimentación utilizado para el móvil.

En la Fig. 4 se muestra el diagrama esquemático que muestra el conexionado de alimentación y comunicación entre la fuente de alimentación, los sensores, el electroimán, el motor paso a paso y el módulo del microcontrolador.

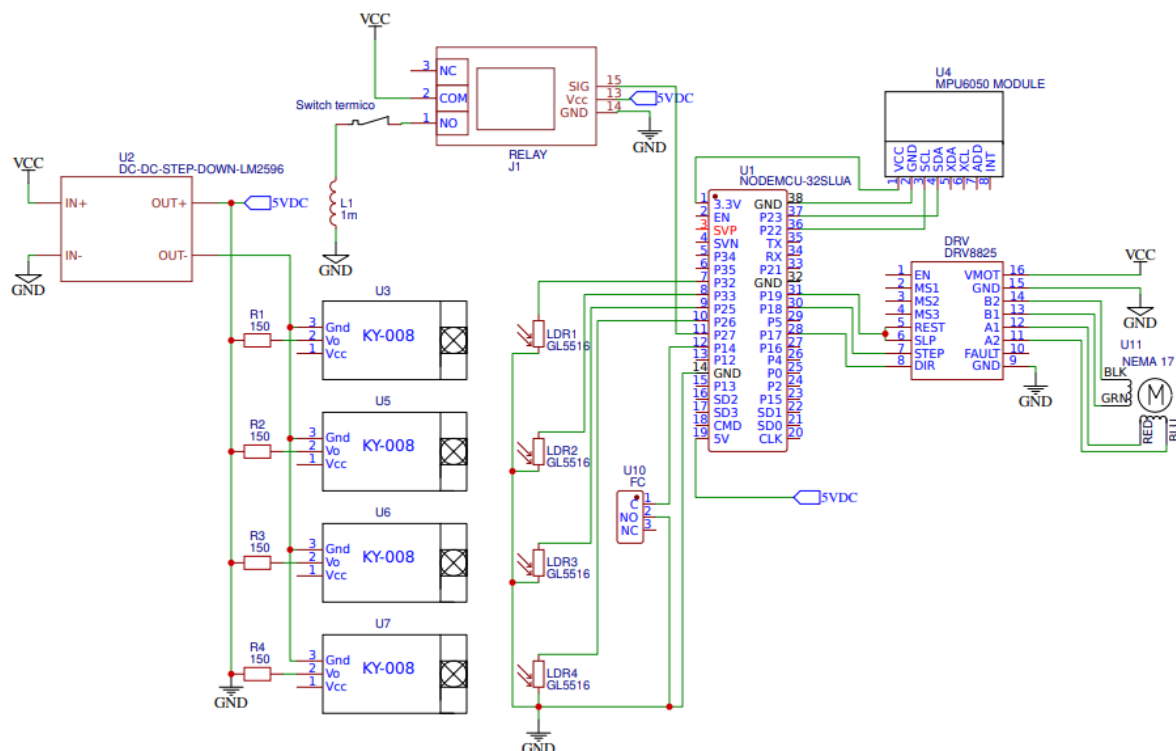


Figura 4. Diagrama esquemático del microcontrolador de la base.

En la Fig. 5, se muestra el diagrama esquemático que muestra el conexionado de alimentación y comunicación del móvil.

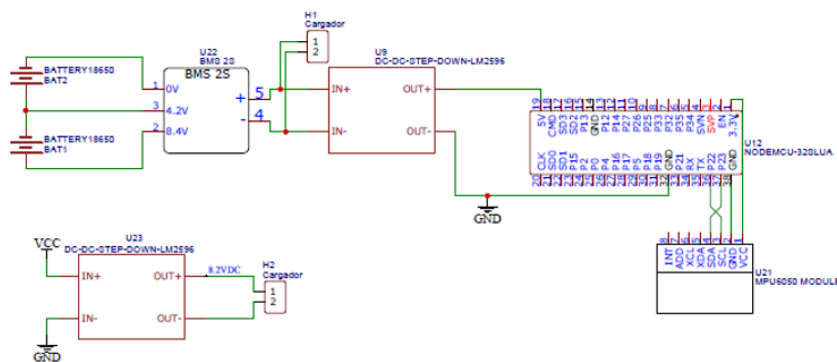


Figura 5. Diagrama esquemático de alimentación y comunicación del módulo móvil.

Finalmente, el diagrama de bloques de la Fig. 6 muestra las comunicaciones entre los diferentes módulos de la plataforma. El usuario se comunica por medio del Protocolo de Transferencia de Hipertexto (*Hypertext Transfer Protocol*, **HTTP**) con el microcontrolador (Raspberry Pi4), que luego utiliza Wifi para comunicarse con los dos microcontroladores (ESP32) (base y móvil). Estos últimos utilizan comunicación *Serial* para conectarse con los acelerómetros. Finalmente, los receptores laser se conectan a los puertos de entrada salida de propósito general (*General Purpose Port Input Output*, **GPIO**).

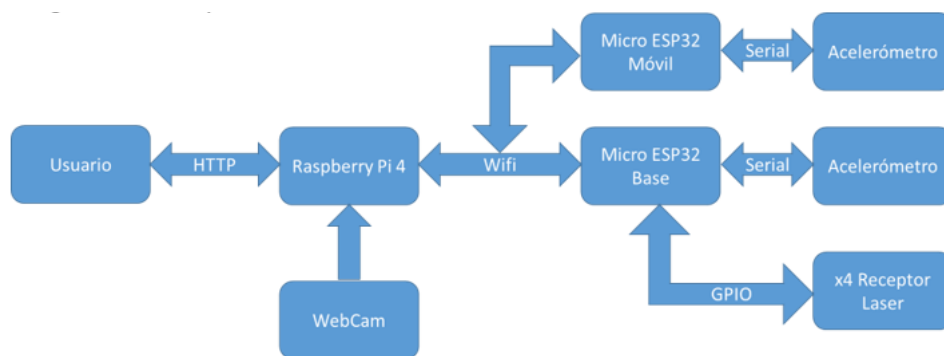


Figura 6. Diagrama en bloques del esquema de comunicación.

Software desarrollado

El firmware de la base cuenta de un programa principal y dos librerías de desarrollo propio. Este programa hace uso del sistema operativo freeRTOS [14] incorporado en el microcontrolador. Este permite la creación de tareas que se ejecutan de forma paralela para un mejor aprovechamiento de los dos núcleos de procesamiento con los que cuenta este dispositivo.

El programa principal se encarga de realizar las siguientes tareas, dadas en orden de ejecución:

1. Establecer una conexión mediante wifi con la Raspberry. Al momento de ejecutar esta tarea debe estar disponible la red, de lo contrario nunca se establecerá la conexión.
2. Utilizando la red anterior se establece una conexión con el servidor MQTT y se configuran la recepción de comandos y transmisión de datos. En el momento de ejecución de este código, el servicio de MQTT se debe encontrar disponible, de lo contrario no se establecerá la conexión.
3. Configurar todas las entradas y salidas del microcontrolador para poder establecer la comunicación con los sensores y el driver del motor.
4. Conectar con el acelerómetro que mide la inclinación de la rampa, configurarlo y calibrarlo. Para esto se utiliza la librería MPU6050_calibration.h de que fue elaborada para este proyecto. Es importante que el acelerómetro se encuentre conectado al momento de ejecución de esta etapa, ya que de lo contrario el programa no continuara y se deberá reiniciar el microcontrolador.
5. Crear una tarea que se encargue de recibir y decodificar cada comando.
6. Crear una tarea que lea la velocidad angular del acelerómetro y la integre para obtener la posición angular de la rampa. Estas dos tareas (5 y 6) se ejecutan constantemente y de forma paralela gracias a la incorporación del sistema operativo en tiempo real (*Real Time Operating System*, RTOS) del microcontrolador.
7. Cuando un comando es recibido, una función se encarga de decodificarlo y ejecutar la acción pertinente. Los distintos comandos posibles fueron enumerados en la sección del protocolo MQTT, a continuación, se explica el funcionamiento de cada uno.
 - **“com0” NOOP:** Esta función imprime en la terminal de depuración que se ha ejecutado esta función. La misma comprueba la conexión con el servidor MQTT.
 - **“com1” Inicio de Experimento:** Los pasos que se ejecutan en esta tarea son:
 1. Suelta el móvil, activando la salida correspondiente al relé que controla el electroimán y guarda el valor del contador del microcontrolador en la posición 0 del vector de tiempos.
 2. Espera 25 ms antes de añadir las llamadas a interrupciones de los sensores de barrera. Esto es necesario debido a que al conectar o desconectar el electroimán se producen picos de tensión transitorios que disparan los sensores.
 3. Añade las interrupciones de cada sensor. Cada vez que se dispara un sensor, se guarda el valor del contador en la posición siguiente del vector de tiempos. Estos valores están dados en microsegundos.
 4. Antes de continuar, espera a que los sensores hayan sido disparados o que hayan transcurrido 5 segundos desde el inicio del experimento.
 5. Desactiva la salida del electroimán.

6. Elimina la tarea.
- **“com2” Enviar Telemetría:**
 1. Calcula la diferencia entre cada componente del vector de tiempos y la guarda en un vector dt . Tiene la forma $dt_i = \frac{t_i - t_{i+1}}{1000}$ para $0 \leq i \leq 3$.
 2. Envía el vector “ dt ” a través del tema “/labRem/cinematica/baseTiempo”.
 3. Elimina la tarea.
- **“com3 ang” Inclinar rampa:**
 1. Lleva el riel hasta el origen, es decir, hasta el final de carrera.
 2. Inclina la rampa hasta que el error entre el ángulo ingresado y el medido sea menor a $0,25^\circ$.
 3. Elimina la tarea.
- **“com4” Reiniciar experimento:**
 1. Inclina el riel hasta el final de carrera, y luego lo lleva a 0° .
 2. Elimina la tarea.
- **“com5” Enviar estado:**
 1. Envía el valor de la variable estado a través del tema “/labRem/cinematica/baseInfo”
- **“com6” Hard reset:**
 1. Hace un reinicio completo del microcontrolador, es equivalente a desconectar la alimentación y volver a conectar.

Por otra parte, el firmware del móvil también cuenta de un programa principal y una librería de desarrollo propio.

El programa principal se encarga de realizar las siguientes tareas, dadas en orden de ejecución:

1. Establecer una conexión mediante Wifi con la Raspberry. Al momento de ejecutar esta tarea debe estar disponible la red, de lo contrario nunca se establecerá la conexión.
2. Utilizando la red anterior se establece una conexión con el servidor MQTT y se configura la recepción de comandos y transmisión de datos. En el momento de ejecución de este código, el servicio de MQTT se debe encontrar disponible, de lo contrario no se establecerá la conexión.
3. Configurar todas las entradas y salidas del microcontrolador para poder establecer la comunicación con el acelerómetro.
4. Conectar con el acelerómetro que mide la aceleración del móvil, configurarlo y calibrarlo. Para esto se utiliza la librería MPU6050_calibration.h de elaboración propia. Es importante que el acelerómetro se encuentre conectado al momento de ejecución de esta etapa, ya que de lo contrario el programa no continuara y se deberá reiniciar el micro.
5. Crear una tarea que se encargue de recibir y decodificar cada comando.
6. Cuando un comando es recibido, una función se encarga de decodificarlo y ejecutar la acción pertinente. Los distintos comandos posibles fueron enumerados en la sección del protocolo MQTT, a continuación, se explica el funcionamiento de cada uno.
 - **“com0” NOOP:** Esta función imprime en la terminal de depuración que se ha ejecutado esta función. Sirve para comprobar la conexión con el servidor MQTT.
 - **“com1” Inicio de Experimento:** Los pasos que se ejecutan en esta tarea son:
 1. Activa la interrupción del acelerómetro.
 2. Lee los datos de aceleración en x, en z y el tiempo cada 10ms.
 3. Una vez que llega a 100 puntos termina el sensado.
 4. Inicia una tarea para enviar los datos.
 5. Elimina la tarea.
 - **“com2” Enviar Telemetría:**
 1. Da formato a los datos para formar el mensaje: “ t_i, ax_i, az_i ” donde t es el tiempo en el que se tomó el dato de aceleración medido desde el inicio del experimento, ax es la aceleración en el eje paralelo al movimiento del auto y z es el eje normal al plano inclinado.
 2. Por cada punto envía un mensaje al tema “labRem/cinematica/movilTiempo”
 3. Elimina la tarea.
 - **“com4” Consultar estado:**

1. Envía el estado del microcontrolador en el tema "labRem/cinematica/movillInfo".
- **"com6" Hard reset:**
 1. Reiniciar completamente el microcontrolador
- **"com7" Deep sleep:** Configura el microcontrolador en modo de reposo/ ahorro de energía, desconectándolo de Wifi. Por lo que para salir de ese estado se debe reiniciar.

Por ultimo en la Raspberry se instaló un servidor web de pruebas provisorio, que sirve para realizar los ensayos y comprobar el funcionamiento del prototipo del laboratorio remoto. Además, se instaló un bróker MQTT que es el servicio de red que permite la comunicación bidireccional entre los microcontroladores y el servidor web.

El servidor de pruebas se realizó en lenguaje Python con el módulo Flask [15], el cual tiene las siguientes funciones:

1. Configurar un cliente y conectarse con el servidor MQTT.
2. Proveer una página web en la red local donde se encuentra conectada la Raspberry.
3. Leer la señal de video de la cámara web y transmitirlo a través de la página.
4. Permitir la interacción con el laboratorio remoto por parte del usuario a través de un navegador web.

El bróker MQTT utilizado es un software gratuito desarrollado por "Eclipse" llamado "Mosquitto Broker" [16]. Se encuentra configurado para funcionar solo en la red wifi generada por la Raspberry, por lo que así se evitan conexiones indeseadas desde la red local, de lo contrario, cualquier dispositivo podría enviar comandos a los microcontroladores puenteando la página web.

En la Fig. 7 se muestra una imagen de la plataforma de laboratorio desarrollada, ya montada en su lugar correspondiente para realizar los ensayos.

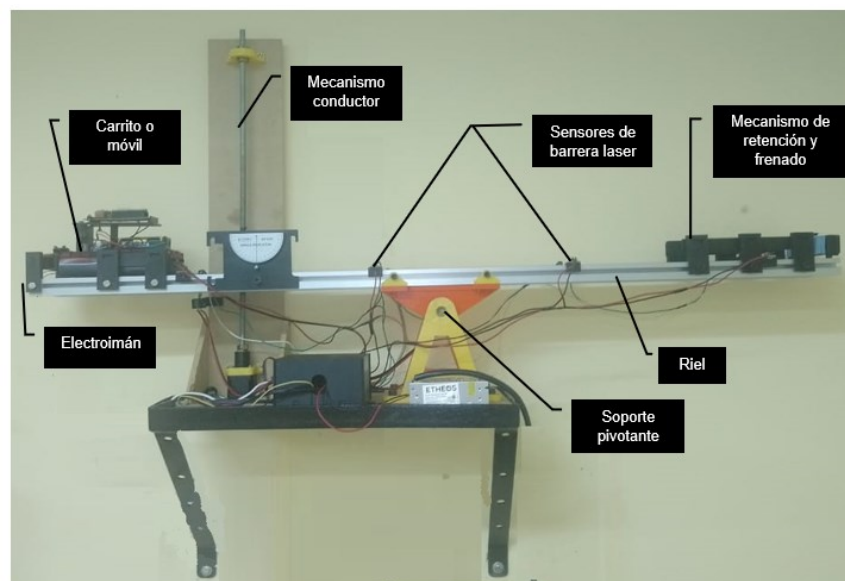


Figura 7. Imagen del prototipo implementado.

En la Fig. 8, se muestra una imagen de la interfaz gráfica de usuario implementada, la misma cuenta con la visualización en tiempo real de la imagen adquirida por la cámara web, que muestra el ensayo. Además, cuenta con un cuadro de diálogo donde se ingresa el ángulo deseado para realizar el experimento, y las funciones de soltar (inicio de ensayo), reinicio y reinicio forzado (en caso de que durante el ensayo se presente alguna falla).

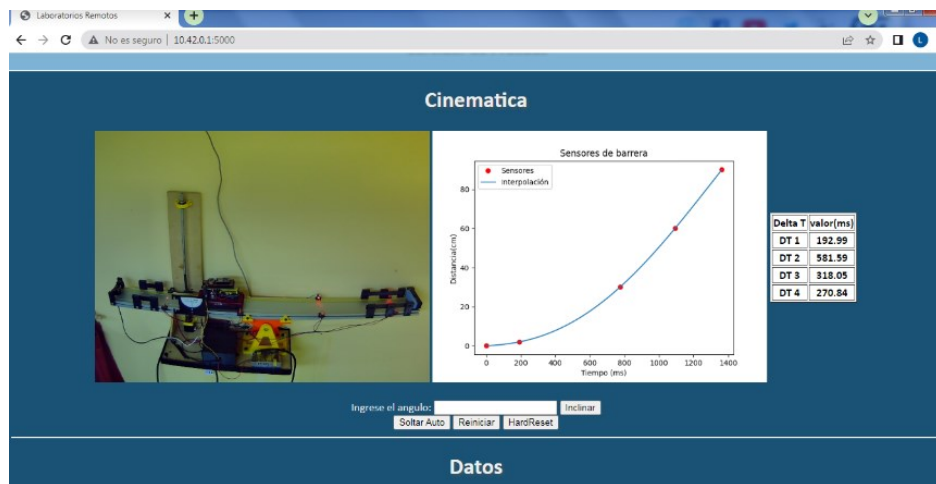


Figura 8. Imagen de la interfaz gráfica de usuario (vista de la cámara y medición de sensores).

En la Fig. 9 se muestra una segunda imagen de la interfaz gráfica de usuario, con las tres gráficas obtenidas luego de realizar un ensayo. Las curvas obtenidas luego de realizar el ensayo son:

- **Espacio recorrido:** se grafica el espacio recorrido en (m) en función del tiempo de duración del ensayo en (ms).
- **Velocidad:** en esta grafica se muestra la curva de velocidad (m/s) con respecto al tiempo en (ms).
- **Aceleración:** esta gráfica muestra la aceleración obtenida de forma teórica y experimentalmente, y se mide en (m/s^2).

Se debe aclarar que debido a que se trata de un ensayo en un plano inclinado, es decir movimiento rectilíneo uniformemente acelerado, el espacio recorrido se incrementa exponencialmente conforme aumenta el tiempo, la velocidad crece linealmente y la aceleración es constante (ver Fig. 9).



Figura 9. Imagen de la interfaz gráfica de usuario (vista de las curvas de espacio, velocidad y aceleración obtenidas).

CONCLUSIONES

Los laboratorios remotos se presentan como una solución efectiva para enfrentar los desafíos educativos actuales, especialmente en el ámbito científico-tecnológico. La posibilidad de acceder a experiencias prácticas y reales de manera remota a través de internet brinda oportunidades de formación más flexibles y accesibles para estudiantes de todo el mundo. La implementación exitosa del laboratorio remoto por parte del equipo de trabajo demuestra su viabilidad y utilidad para mejorar la enseñanza y el aprendizaje de competencias científicas. Se espera que esta innovadora herramienta contribuya significativamente al fortalecimiento de la

educación en ciencias e ingenierías en modalidades a distancia y mixtas, preparando a los estudiantes para los retos de la sociedad digital.

REFERENCIAS

- [1] Johnson, A. B. (2019). The impact of digital tools on student learning outcomes in higher education. *Educational Technology Research and Development*, 12(3), 75-89.
- [2] Chen, W. (2020). Integrating digital technologies into higher education: A review of current issues. *Journal of Educational Technology*, 15(2), 45-60.
- [3] Smith, D. E. (2020). Designing interactive simulations for physics education: Best practices and challenges. *Simulation & Gaming*, 25(3), 120-135.
- [4] Garcia, M. L. (2021). Enhancing physics learning through interactive digital platforms. *Journal of Physics Education*, 28(4), 102-115.
- [5] Martinez, S. P. (2022). Assessing student progress in virtual learning environments: Tools and strategies. *Assessment in Higher Education*, 18(2), 55-68.
- [6] Lopez, J. R. (2023). Implementing remote learning platforms in university physics courses: A case study. *Journal of Online Education*, 5(1), 30-42.
- [7] Pasco. "Dynamic Kart & Track System". <https://www.pasco.com/products/guides/dynamics-system-configuration>
- [8] Espressif. ESP32-Wroom datasheet. https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf
- [9] NEMA17 datasheet. <https://pages.pbclinear.com/rs/909-BFY-775/images/Data-Sheet-Stepper-Motor-Support.pdf>
- [10] Texas Instruments. DRV8825 datasheet. <https://www.ti.com/lit/ds/slvsa73f/slvsa73f.pdf?ts=1713962224464>
- [11] InvenSense. MPU6050 datasheet. <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>
- [12] GL5516 datasheet. <https://www.kth.se/social/files/54ef17dbf27654753f437c56/GL5537.pdf>
- [13] Raspberry Pi. Raspberry Pi 4 Model B página oficial. <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>
- [14] FreeRTOS. <https://www.freertos.org/index.html>
- [15] Flask documentación, <https://flask.palletsprojects.com/en/3.0.x/>
- [16] Eclipse. Mosquitto página oficial. <https://mosquitto.org/>