

Plataforma experimental que emula la dinámica de orientación de nanosatélites Experimental platform that emulates the orientation dynamics of nanosatellites

Alan N. Castrellón-López ^a, Matias Ramirez-Rodriguez ^{a,*}, Emmanuel Gama-Moctezuma ^a, Kristian F. Maya-Gress ^a,
Raúl Villafuerte-Segura ^a

^aÁrea Académica de Computación y Electrónica, Universidad Autónoma del Estado de Hidalgo, 42184, Pachuca, Hidalgo, México.

Resumen

En el presente manuscrito se propone el diseño, la construcción y el control de una plataforma experimental que emula la dinámica del posicionamiento rotacional de un nanosatélite. Esta plataforma está inspirada en un *CubeSat*, satélite diseñado para orbitar alrededor de la Tierra, cuya estructura es escalable a un cubo de $0.1[m^3]$ de arista y masa inferior a $1.33[kg]$. Para la dinámica del posicionamiento rotacional, se propone un modelo matemático básico usando leyes de Newton y se presenta una identificación paramétrica. Estas dos metodologías permiten realizar un análisis matemático para sintonizar controladores empleando el criterio de Hurwitz. Sin embargo, también existen métodos experimentales como el segundo método de Ziegler-Nichols, el cual presentamos paralelamente con el propósito de mostrar una solución alterna de sintonización. Las tres metodologías anteriores garantizan la manipulación de la velocidad y el sentido de giro de una rueda de reacción, para orientar el posicionamiento de la plataforma experimental. Esto permite realizar el seguimiento de la fuente de luz más intensa, con el fin de cargar sus celdas solares, consigna ampliamente utilizada en el sector aeroespacial.

Palabras Clave: Nanosatélite, rueda de reacción, CubeSat, Ziegler-Nichols, Control Clásico.

Abstract

In this manuscript, we propose the design, construction and control of an experimental platform that emulates the rotational positioning dynamic of a nanosatellite. This platform is inspired by a *CubeSat*, a satellite designed to orbit the Earth, whose structure is scalable in cubes of edge $0.1[m^3]$ and mass less than $1.33[kg]$. In addition, a basic mathematical model using Newton's laws is proposed and a parametric identification of the above platform is presented (2nd order system). Although these two methodologies are essential to tune the controller by mathematical analysis, there are also experimental methods such as the second method of Ziegler-Nichols, which we present in parallel with the purpose of showing an alternative solution. The three methods described above allow the manipulation of the speed and direction of rotation of a reaction wheel to orient the positioning of the experimental platform. This allows the tracking of the most intense light source in order to charge its solar cells, a task widely used in the aerospace sector.

Keywords: Nanosatellite, reaction wheel, CubeSat, Ziegler-Nichols, Classic Control.

1. Introducción

En los últimos años se ha visto un crecimiento considerable en el desarrollo de sistemas de posicionamiento de vehículos espaciales, debido a proyectos como la estación espacial internacional (NASANET, 2024a), telescopios espaciales (NASANET, 2024b), sondas y satélites artificiales para la exploración de Marte (Geographic, 2022), así como una creciente demanda de satélites de alto rendimiento (High Throughput Satellites,

HTS) (infoespacio.com, 2023), entre otras tecnologías que lo requieren. Además, se estima que en los últimos cinco años se han pactado más de un millón de lanzamiento de satélites por parte de países de todo el mundo, 115 veces más que el número de satélites que operan actualmente alrededor de la Tierra (Confidencial, 2023). Y sólo 4 países, Estados Unidos, China, Reino Unido y Rusia concentran cerca del 85 % de los satélites en órbita. Según los datos de la Union of Concerned Scientists (UCS) hay 7 satélites de México en órbita, el más antiguo, el

* Autor para correspondencia: ra448324@uaeh.edu.mx

Correo electrónico: ca375730@uaeh.edu.mx (Alan Naresh Castrellón-López), ra448324@uaeh.edu.mx (Matias Ramirez-Rodriguez), ga446505@uaeh.edu.mx (Emmanuel Gama-Moctezuma), kristian_maya10493@uaeh.edu.mx (Kristian Freyri Maya-Gress), villafuerte@uaeh.edu.mx (Raúl Villafuerte-Segura).

Historial del manuscrito: recibido el 01/07/2024, última versión-revisada recibida el 03/09/2024, aceptado el 29/08/2024, publicado el 30/11/2024. DOI:<https://doi.org/10.29057/icbi.v12iEspecial4.13358>



Mexsat-3, lanzado en el 2012 (Fintualist, 2023). En este contexto, resulta atractivo impulsar en países como México el diseño y la construcción de plataformas experimentales que emulen las dinámicas de posicionamiento de sistemas reales, como las de los nanosatélites, ya que fomentan la creatividad, la innovación y la resolución de problemas complejos de la disciplina. Además, este tipo de plataformas pueden coadyuvar a la enseñanza-aprendizaje de la teoría de control en diversas consignas, como el seguimiento de fuentes de luz, empleando técnicas versátiles para manipular el posicionamiento del sistema en cuestión. Para tal efecto, es importante conocer los tipos de actuadores que se pueden emplear. (Rodríguez Vázquez, 2013) los clasifica en 2 tipos diferentes: externos, los cuales son capaces de controlar la acumulación de impulso; e internos, que son capaces de almacenar el momento angular, pero no pueden almacenar el impulso. Dentro de esta última categoría se encuentran las ruedas de reacción, comúnmente llamados "dispositivos de intercambio de momento". Estos dispositivos son capaces de añadir estabilidad a pesar de perturbaciones en algún sistema, y proveen un momento variable, permitiendo absorber torques cíclicos y para transferir el momento hacia el cuerpo de un satélite, para realizar maniobras de giro mediante el principio de la conservación del momento angular. Este principio brinda la oportunidad de desarrollar un controlador preciso y continuo para manipular la posición y orientación de gran variedad de vehículos espaciales (satélites, telescopios, naves espaciales, entre otros), lo cual es esencial en tareas como mantener la orientación de paneles solares apuntando hacia el sol, dirigir antenas de comunicación satelital hacia la Tierra o mantener la estabilidad de telescopios espaciales y otros instrumentos científicos. Es por ello que en el ámbito del control de posicionamiento de vehículos espaciales, investigaciones clave como la de (Froelich and Papapoff, 1959) han resaltado la importancia de las ruedas de reacción para la orientación precisa de estos sistemas. El estudio evalúa la eficiencia y fiabilidad del sistema, analizando la relación entre torque, velocidad y potencia, además proporciona recomendaciones de diseño para optimizar su rendimiento. Gracias a su enfoque integral, que incluye modelos de laboratorio y análisis de perturbaciones, este trabajo se ha convertido en un referente esencial para el diseño de sistemas de control en misiones espaciales. Posteriormente, en su estudio, (Cannon Jr, 1962) analiza el comportamiento dinámico de los sistemas de control de posicionamiento en vehículos espaciales, enfocándose en el uso de ruedas de reacción. Ya que se explora cómo estos sistemas pueden gestionar cambios de momento angular con eficiencia energética. Se destaca la importancia de una comprensión profunda de la dinámica del sistema y un diseño cuidadoso para lograr un control preciso de la orientación, analizando las respuestas del sistema a perturbaciones impulsivas y sinusoidales en entornos orbitales. Décadas después, (Sinclair et al., 2007) describen el uso de ruedas de reacción para orientar nanosatélites, destacando que, aunque la estabilización magnética ha sido tradicionalmente utilizada, no es adecuada para la mayoría de la instrumentación en nanosatélites. Los autores detallan el desarrollo de su rueda de reacción y subrayan su potencial para nuevas misiones, como observaciones astronómicas y maniobras de vuelo en formación, ampliando las capacidades de pequeños satélites en el espacio. Finalmente, en su artículo, (Snider, 2010) describe el diseño y

evaluación de un sistema de control de actitud (ACS) basado en ruedas de reacción para el simulador de satélite SimSat II de la Fuerza Aérea de EE. UU. El objetivo era lograr una precisión de posicionamiento de 0.01 grados y ejecutar maniobras de rotación de 10 y 30 grados rápidamente, resistiendo perturbaciones. Los resultados mostraron una precisión generalmente superior a la requerida, aunque no constante. Las maniobras descritas en el manuscrito, detallan que estas se realizaron en el tiempo especificado, pero con una precisión ligeramente inferior a la esperada. Puesto que se identificaron problemas con la ejecución de comandos de tiempo fijo y variaciones en el momento de inercia de los ejes. En cuanto a las aplicaciones prácticas en nanosatélites, (Krishna et al., 2018) analizan la dinámica y el control de un sistema de ruedas de reacción para nanosatélites, abordando el uso de motores BLDC y un controlador PID para el control en tres ejes. El estudio muestra que el sistema es viable en entornos espaciales, destacando la necesidad de magnetopares para manejar el momento acumulado. Un par de años más adelante, (Medina et al., 2021) destacan los CubeSats como una alternativa de bajo costo para la investigación espacial. Enfatizan la importancia del subsistema ADCS para una orientación precisa, y proponen un controlador de velocidad PID para ruedas de reacción en CubeSats de $1U = 1dm^3$. El estudio muestra que el PID mejora el tiempo de respuesta y la estabilidad, siendo eficaz para aplicaciones como la comunicación láser. En ese mismo año, (Afifa et al., 2021) presentan el modelado de un sistema de actitud para CubeSats usando una rueda de reacción. El estudio aborda el desarrollo de un sistema de control para ajustar y mantener la orientación en el espacio, proporcionando una base para futuras optimizaciones. Los resultados iniciales en simulaciones muestran que el sistema es prometedor para misiones espaciales, aunque se requiere más investigación para mejorar su eficiencia. Después, (Helmy et al., 2022) abordan la movilidad de CubeSats proponiendo dos controladores para el Sistema de Control de Actitud (ACS): un PID estándar y un PI-D modificado. En el cual utilizan una Optimización por Algoritmo Genético (GAO) para ajustar las ganancias de los controladores y mejorar el rendimiento en presencia de perturbaciones y ruido. Los resultados muestran que ambos enfoques son efectivos para mejorar la movilidad del CubeSat en condiciones adversas. En ese mismo año, (Alegre Bonet, 2022) presenta una tesis sobre el diseño mecánico de un sistema ADCS de tres grados de libertad para CubeSats de 1U, usando ruedas de reacción. El estudio incluye el análisis de tecnologías actuales, el diseño de componentes y dos configuraciones del sistema. Aunque el software de simulación no considera perturbaciones espaciales, verifica preliminarmente la capacidad de controlar los tres ejes principales del CubeSat. Los resultados indican que el diseño puede controlar los tres grados de libertad, pero la verificación es preliminar y no refleja el entorno espacial real. Como se ha observado, el posicionamiento de los satélites y la optimización del mismo son fundamentales en el sector aeroespacial, Además, el seguimiento solar y la orientación en aplicaciones espaciales son esenciales para garantizar la máxima eficiencia de los sistemas fotovoltaicos. Tal y como lo muestran, (Zlatanov and Weinrebe, 2014), los cuales desarrollaron un sistema de seguimiento solar con un microcontrolador para maximizar la producción de energía al alinear los paneles con el sol. El sistema usa software para ajustar la orientación de los paneles

y, aunque los resultados fueron satisfactorios, se identificaron limitaciones en el hardware que podrían mejorarse. La investigación concluye que el uso de microcontroladores es viable para aumentar la eficiencia de los sistemas fotovoltaicos. De igual importancia es el mencionar aquellos documentos clave, los cuáles ofrecen una visión amplia sobre los avances en tecnología satelital. En (Space, 1977) se presenta el manual del Sistema de Transporte Espacial (STS) de 1977, que brinda una visión general del STS, sus componentes y servicios, y es una referencia histórica para entender la evolución de los sistemas de transporte espacial. Otro documento relevante es el estándar para el desarrollo de nanosatélites elaborado por (University, 2015) en 1999, aplicable a instituciones educativas, empresas privadas y organizaciones gubernamentales. El laboratorio de CubeSat del Cal Poly ha lanzado 12 CubeSats y apoyado más de 175 misiones. Además, (Jamie et al., 2017) ofrecen una guía integral para desarrolladores de CubeSat, detallando conceptos, procesos y requisitos de certificación, incluyendo documentación, comunicación con el integrador de misión, y verificaciones de masa, batería, dimensiones y requisitos eléctricos, así como análisis de ventilación y pruebas de seguridad. Por último, (Liddle et al., 2020) analizan la evolución y el estado actual de los nanosatélites en misiones científicas, destacando cómo la innovación ha permitido desarrollar nanosatélites confiables y de alto rendimiento. El artículo también aborda desafíos técnicos, como el control térmico y la capacidad de apuntado, usando el satélite ASTERIA como ejemplo de cómo estos requisitos se pueden cumplir mediante un diseño adecuado y subsistemas comerciales. En este manuscrito se propone el diseño, la construcción y el control de una plataforma experimental para emular la dinámica de posicionamiento rotacional de un CubeSat. El diseño se realiza con SolidWorks, basado en la propuesta de (Pirateque et al., 2020), y se construye usando impresión 3D. Para el control rotacional, se utiliza una rueda de reacción acoplada a un motor de corriente directa (CD), controlado por un PID clásico basado en amplificadores operacionales. La señal de control se envía vía WiFi desde una consola de mando terrestre, utilizando dos tarjetas ESP32 para la comunicación y adquisición de datos. Un sensor con dos fotodiodos y un encapsulado impreso en 3D estima el ángulo de error entre la fuente de luz más intensa y la posición actual del nanosatélite. Para el seguimiento de la fuente de luz, se aplican tres técnicas para la sintonización del controlador PID: la primera se basa en un modelo matemático propuesto, usando Newton, que describe la dinámica de posicionamiento, la segunda en un método de identificación paramétrica de un sistema de segundo orden. En ambas técnicas se emplea el criterio de Hurwitz. En la tercera se aplica el método de Ziegler-Nichols para la sintonización sin modelo. Cabe mencionar que no es objetivo el comparar estas técnicas, sino ofrecer alternativas para la sintonización del controlador. Los resultados experimentales validan la propuesta, sugiriendo su potencial aplicabilidad en el sector aeroespacial. El resto del manuscrito se encuentra estructurado de la siguiente forma: el planteamiento del problema y las contribuciones se muestran en la Sección 2, mientras que los resultados preliminares se presentan en la Sección 3, referentes a criterios básicos, al modelo matemático, identificación de un sistema y método de Ziegler-Nichols. En la Sección 4 se presenta el diseño, la descripción y la puesta en marcha de la pla-

taforma experimental propuesta. En la Sección 5 se corroboran los resultados y la plataforma propuesta mediante la presentación de resultados experimentales. Finalmente, en la Sección 6 se presentan las conclusiones del presente.

2. Planteamiento del problema y puntos destacados

2.1. Planteamiento del problema

Como ya se mencionó anteriormente, en los próximos años se espera una gran demanda de sondas y satélites artificiales entorno a las telecomunicaciones y a la exploración del universo, esto conlleva a desarrollos tecnológicos y científicos en el sector aeroespacial, y en consecuencia, emergerán varios campos de oportunidad para diferentes áreas, sobre todo para aquellas interdisciplinarias, como la teoría de control. Desafortunadamente, estos desarrollos son propios de países de altos ingresos o “primermundistas”. Países como México no cuentan con la infraestructura y la interacción suficientes para atender problemas complejos de este sector. Además, típicamente entorno a la generación de recursos humanos no se cuentan con las plataformas experimentales suficientes para instruir a los estudiantes en estos quehaceres. Por lo que parece plausible, primeramente, proponer el desarrollo de mecanismos alternos que coadyuven a entender estas problemáticas, para posteriormente, incursionar y contribuir en este importante sector. Razon por la cual, el diseño, la construcción y el control de una plataforma experimental que emula la dinámica del posicionamiento rotacional de un nanosatélite, en tareas de seguimiento de una fuente de luz, sin la necesidad de emplear mecanismos de expulsión de masa (como los propulsores de cohetes), pueden apoyar al proceso de generación de recursos humanos, sobre todo de licenciatura; y contribuir al sector aeroespacial.

2.2. Puntos a destacar:

- Diseño y construcción de una plataforma experimental que emule la dinámica del posicionamiento de un nanosatélite, conocido como CubeSat.
- Control de velocidad y sentido de giro de una rueda inercial colocada en la parte superior del CubeSat, para manipular la orientación angular/rotacional de la plataforma, en tareas para el seguimiento de una fuente de luz.
- Contribuir a la generación de recursos humanos que puedan entender problemáticas del sector aeroespacial, mediante el desarrollo de plataformas experimentales que emulen las dinámicas de sistemas reales.

3. Resultados preliminares

Momento inercial. El momento inercial I es una magnitud que establece la resistencia que presenta un cuerpo al cambiar su velocidad angular ω . Con relación a un eje definido, el par (torque) $F(t)$ externo aplicado a un cuerpo rígido, se relaciona con la aceleración angular $\alpha(t)$ adquirida mediante

$$F(t) = I\alpha(t), \quad I = \int_m r^2 dm. \quad (1)$$

Rueda de reacción. La rueda de reacción o rueda inercial es un tipo de actuador que provee un par o torque resultante de la aceleración o desaceleración rotacional del volante de inercia.

La aceleración o desaceleración causa una reacción de alineación del torque generado con el eje del nanosatélite. El sentido de rotación y la velocidad angular $\omega_R(t)$ del volante de inercia pueden ser controlados por un motor eléctrico (CD, corriente directa) que permite ajustar el torque $F(t)$ mediante una señal de voltaje, es decir

$$F(t) = I_R \alpha_R(t), \quad (2)$$

donde $\alpha_R(t)$ es la aceleración angular e $I_R = \frac{1}{2}MR^2$ es el momento inercial de la rueda (disco), con masa M y radio R .

3.1. Modelo matemático

A continuación, se propone un modelo matemático simplificado que describe la dinámica de movimiento del nanosatélite en función del momento inercial generado por la rueda de reacción, para así manipular su posición angular y direccionar o alinear al CubeSat hacia la fuente de luz más intensa.

Considere el esquema de fuerzas dado en la Figura 1. Bajo la suposición de que la rueda inercial se encuentra sujeta firmemente al nanosatélite, entonces el momento inercial de la rueda de reacción induce una fuerza $F(t)$ que genera un desplazamiento en la posición angular del nanosatélites $\theta_S(t)$, en sentido contrario a las manecillas del reloj. Este movimiento es debido a la fuerza de rozamiento de la rueda inercial con el aire $F_r(t)$. Además, debemos considerar que la plataforma experimental cuenta con un sobrepaso y un subpaso, ocasionando oscilaciones al rededor del punto de convergencia, por lo cual se debe considerar una fuerza $F_w(t)$.

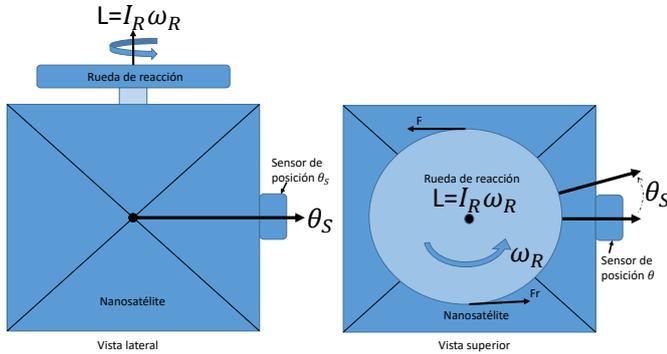


Figura 1: Esquema de nanosatélite con rueda de reacción

Así, usando la versión angular de la segunda ley de Newton, la cual relaciona las fuerzas netas $\sum_{i=1}^n F_i$ aplicadas sobre un objeto con el momento de inercia y la aceleración angular, tenemos que

$$I_S \ddot{\theta}_S(t) = F(t) - F_r(t) - F_w(t), \quad (3)$$

donde

- $I_S = \frac{1}{6}ml^2$ es el momento inercial del nanosatélite (cubo), con masa m y lado l .
- $\theta_S(t)$ es la posición angular del nanosatélite, respecto a una fuente de luz.
- $F(t) = I_R \alpha_R(t) = I_R(k_c V(t))$ es la fuerza inercial generada por la rueda de reacción. Aquí, k_c es una constante

positiva (empírica) con un valor de 0.6, $V(t)$ es el voltaje suministrado al motor de CD que regula la velocidad de la rueda de reacción.

- $F_r(t) = a_1 \dot{\theta}_S(t)$ es la fuerza de amortiguamiento causada por la fricción entre el nanosatélite y el aire. Aquí, a_1 es conocido como coeficiente de fricción.
- $F_w = a_2 \theta_S(t)$ es la fuerza de oscilación debida al sobrepaso y al subpaso del nanosatélite, respecto a la fuente de luz. Aquí, a_2 es el coeficiente de frecuencia natural.

Así, reescribiendo (3) se obtiene un modelo que describe la dinámica del posicionamiento del nanosatélite:

$$I_S \ddot{\theta}_S(t) + a_1 \dot{\theta}_S(t) + a_2 \theta_S(t) = I_R k_c V(t). \quad (4)$$

Lo anterior no es inesperado, ya que en (Åström and Hägglund, 2009) se menciona que la mayoría de los sistemas se pueden modelar por ecuaciones diferenciales de 1er o 2do orden.

3.2. Identificación paramétrica

A continuación, se presenta un método de identificación para obtener una aproximación del valor de los parámetros de la plataforma experimental. Bajo la suposición de que esta plataforma puede representarse mediante un sistema de segundo orden y la obtención de la respuesta escalón de la plataforma.

Considere la función de transferencia de una sistema de segundo orden con la forma

$$G(s) = \frac{Y(s)}{V(s)} = \frac{\omega^2}{s^2 + 2\delta\omega s + \omega^2}, \quad (5)$$

donde $\delta > 0$ y $\omega > 0$ son conocidas como amortiguamiento y frecuencia natural del sistema. Si $0 < \delta < 1$, entonces la respuesta temporal escalón con amplitud y_0 , es decir $V(s) = y_0/s$, está dado por

$$y(t) = y_0 - y_0 e^{-\delta\omega t} \left[\cos(\omega\sqrt{1-\delta^2}t) + \frac{\delta}{1-\delta^2} \sin(\omega\sqrt{1-\delta^2}t) \right]. \quad (6)$$

De la ecuación anterior, se sabe que la primera sobreelongación y_1 y la primera subelongación y_2 , están dadas por (Ogata, 2001):

$$y_1 = y_0 e^{-\frac{\delta\pi}{\sqrt{1-\delta^2}}} \quad \text{y} \quad y_2 = y_0 e^{-\frac{2\delta\pi}{\sqrt{1-\delta^2}}}.$$

De las ecuaciones anteriores, se puede deducir que

$$\delta_1 := \delta = \frac{|\ln(y_1/y_0)|}{\sqrt{\pi^2 + \ln^2(y_1/y_0)}} \quad \text{y} \quad \delta_2 := \delta = \frac{|\ln(y_2/y_0)|}{\sqrt{4\pi^2 + \ln^2(y_2/y_0)}}. \quad (7)$$

Así, se puede calcular un promedio como sigue:

$$\delta = \frac{\delta_1 + \delta_2}{2}. \quad (8)$$

Por otro lado, los dos primeros tiempo de subida t_{r_1} y t_{r_2} , se pueden obtener a partir de las siguientes ecuaciones (Ogata, 2001):

$$t_{r_1} = \frac{\pi - \beta}{\omega\sqrt{1-\delta^2}} \quad \text{y} \quad t_{r_2} = \frac{3\pi - \beta}{\omega\sqrt{1-\delta^2}}, \quad (9)$$

con $\beta = \sin^{-1}(\sqrt{1-\delta^2}) = \cos^{-1}(\delta)$. Si $T = t_{r_2} - t_{r_1}$, entonces

$$\omega = \frac{2\pi}{T\sqrt{1-\delta^2}}, \quad (10)$$

donde T es el periodo de la respuesta temporal transitoria.

3.3. Análisis para garantizar el seguimiento de fuentes de luz

En las subsecciones anteriores se propuso un modelo matemático de la forma (4) y se presentó una identificación para sistemas de la forma (5). En ambas técnicas se puede obtener la correspondiente ecuación característica del sistema en lazo cerrado con un controlador PID. Esta ecuación se puede emplear para garantizar seguimiento de una fuente de luz. A continuación se describe este proceso.

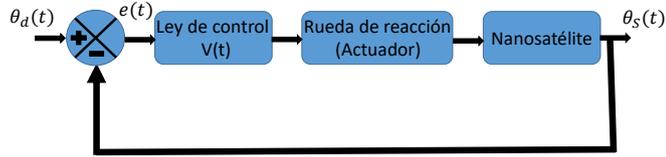


Figura 2: Diagrama a bloques del lazo cerrado.

Considere el diagrama a bloques en lazo cerrado dado en la Figura 2. Aquí, la variable $\theta_d(t)$ representa la posición de la fuente de luz más intensa (posición angular deseada), $V(t)$ es el voltaje suministrado al motor de CD (actuador) para variar la velocidad angular de la rueda inercial $\omega_R(t)$, mientras que $e(t) = \theta_d(t) - \theta_s(t)$ es el ángulo de error entre las posiciones angulares de la fuente de luz y el nanosatélite.

Observación 1. Claramente, para que el nanosatélite siga la fuente de luz más intensa localiza en $\theta_d(t)$, basta que el $e(t)$ converja a cero, cuando t tiende a infinito, es decir, que el sistema en lazo cerrado sea asintóticamente estable.

Teorema 2 (Criterio de Hurwitz, (Ogata, 2001)). *Un sistema en lazo cerrado es asintóticamente estable si y solo si todas las raíces de su correspondiente ecuación característica se encuentran del lado izquierdo del plano complejo \mathbb{C} .*

Así, para obtener la correspondiente ecuación característica es necesario calcular la función de transferencia $H(s)$ (función analítica, $s = \sigma + \omega i$, $i^2 = -1$, $\sigma, \omega \in \mathbb{R}$) del sistema en lazo cerrado:

$$H(s) = \frac{\mathcal{L}\{\theta_s(t)\}}{\mathcal{L}\{\theta_d(t)\}} = \frac{V(s)G(s)}{1 + V(s)G(s)}, \quad (11)$$

donde \mathcal{L} representa la transformada de Laplace y, $G(s)$ y $V(s)$ son las funciones de transferencia de la plataforma y del controlador, respectivamente. Aquí, la ecuación característica es

$$p(s) = 1 + V(s)G(s) = 0. \quad (12)$$

Si $G(s) = \frac{l_R k_c}{I_s s^2 + a_1 s + a_2}$ es la función de transferencia del modelo matemático (4) sugerido para representar el nanosatélite y $V(s) = k_p + k_d s + k_i/s$, $k_p, k_d, k_i \in \mathbb{R}$, es la función de transferencia del controlador PID. Entonces, la función de transferencia (11) se puede reescribir como

$$H(s) = \frac{l_R k_c [k_d s^2 + k_p s + k_i]}{I_s s^3 + [a_1 + l_R k_c k_d] s^2 + [a_2 + l_R k_c k_p] s + l_R k_c k_i}$$

Por lo que la correspondiente ecuación característica (12) es

$$p(s) = s^3 + \frac{[a_1 + l_R k_c k_d]}{I_s} s^2 + \frac{[a_2 + l_R k_c k_p]}{I_s} s + \frac{l_R k_c k_i}{I_s} = 0. \quad (13)$$

Por lo que, el problema se reduce a determinar las ganancias $k_p, k_d, k_i \in \mathbb{R}$ tales que (13) satisfaga el Teorema 2. Un método que se puede emplear es el conocido como asignación de polos. Otro método es el criterio de Routh-Hurwitz, ver (Ogata, 2001). Un razonamiento semejante se puede aplicar si ahora se considera a $G(s)$ como la función dada en (5). Parece que queda claro que esto se puede hacer si se cuenta con una función de transferencia $G(s)$, ya sea a partir de un modelo matemático o a partir del empleo de una identificación perimétrica. Sin embargo, ¿Qué pasa si no se cuenta con $G(s)$? En seguida se presenta una técnica para este fin.

3.4. Reglas de Ziegler-Nichols

Un método empírico para garantizar convergencia de $e(t)$ es dado por Ziegler-Nichols (Ogata, 2001), el cual se explica a continuación:

1. Experimentalmente, construya el lazo cerrado esquematizado en la Figura 2.
2. Para $k_d = k_i = 0$, incrementar k_p de 0 hasta un valor crítico k_{cr} , en donde la salida del sistema muestre oscilaciones sostenidas, ver Figura 3.

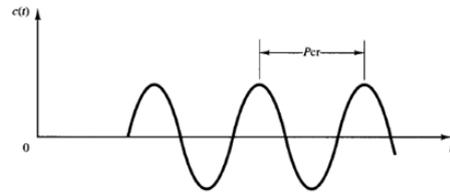


Figura 3: Oscilaciones sostenidas (Ogata, 2001).

3. Obtenga el período P_{cr} de la salida obtenida en el paso anterior.
4. Elija el tipo de ley de control a usar:

Tabla 1: Reglas de Ziegler-Nichols.

Control	k_p	k_i	k_d
P	$0.5k_{cr}$	∞	0
PI	$0.45k_{cr}$	$\frac{0.54k_{cr}}{P_{cr}}$	0
PID	$0.6k_{cr}$	$\frac{1.2k_{cr}}{P_{cr}}$	$0.075k_{cr}P_{cr}$

4. Diseño, descripción y puesta en marcha de plataforma

4.1. Diseño de la plataforma

La estructura de la plataforma propuesta se diseñó con base a la dada por (Rodríguez Pirateque et al., 2020), haciendo uso del software CAD, SolidWorks, como se muestra en la Figura 4. Esta consta de 4 paredes, 4 soportes con argolla, un soporte interno para los componentes electrónicos, una rueda unida a un cople para el eje de un motor DC, un soporte para sujetar dicho motor y una estructura que complemente el funcionamiento de un sensor de luz.

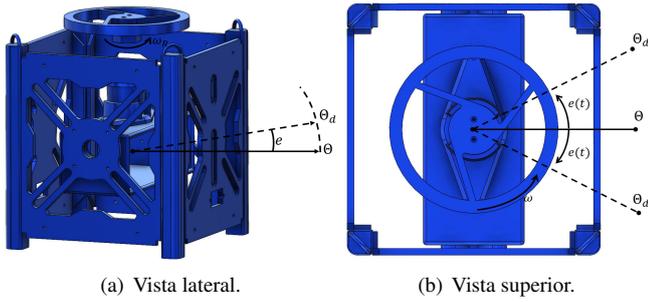


Figura 4: Diseños en SolidWorks de la plataforma.

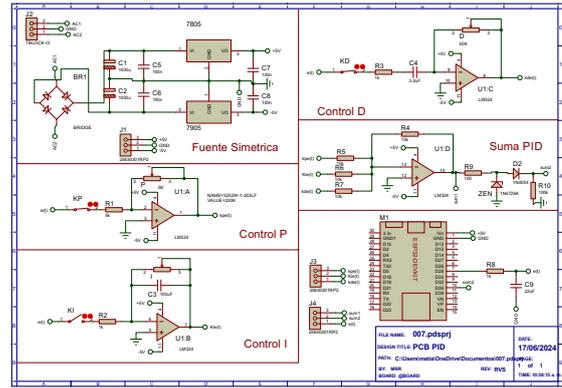


Figura 6: Esquemático del control PID que envía y recibe información por medio de la ESP32.

4.2. Descripción general de la plataforma propuesta

A continuación, se da una breve descripción de la plataforma experimental propuesta, ver Figura 7.

También, se diseñó un par de placas PCB, una para la instrumentación requerida por el nanosatélite CubeSat y otra para la instrumentación del controlador PID. Ambas placa fueron diseñadas para tener comunicación vía WiFi, usando dos tarjetas ESP32. En la Figura 5 se muestra el esquema de PCB en el CubeSat, tiene dimensiones de $104 \times 50 \times 0.8 [mm^3]$, es de dos capas, con FR4 como material de sustrato y se utilizó cobre de $1 [oz/ft^2]$ de grosor. Se empleó la tecnología THT (Through-Hole Technology) para facilitar el cambio de componentes, los cuales son: Microcontrolador ESP32 DevKit V1, CI L239D (Puente H), Sensor MPU6050, Sensor de Luz formado a partir de dos LDR, Header M-M, LM7805 regulador de voltaje 5[V], Switch on/off, Tblock i2, y resistencias de $10 [k\Omega]$. La PCB para el control PID, se diseñó usando amplificadores operacionales, ver Figura 6. Las dimensiones de la placa son de $129 \times 53 \times 0.8 [mm^3]$. Los componentes empleados son: Microcontrolador ESP32 DevKit V1, CI LM324, LM7805 regulador de voltaje 5[V], LM7905 regulador de voltaje -5[V], 4 Switch on/off, Tblock i2, resistencias de $1 [k\Omega]$, $10 [k\Omega]$, potenciómetros de $2 [k\Omega]$, $10 [k\Omega]$ y $20 [k\Omega]$, capacitores de $3.3 [\mu F]$, $150 [nF]$, $1000 [\mu F]$, así como diodos rectificadores 1n4004 y un diodo zener de $3.3 [V]$.

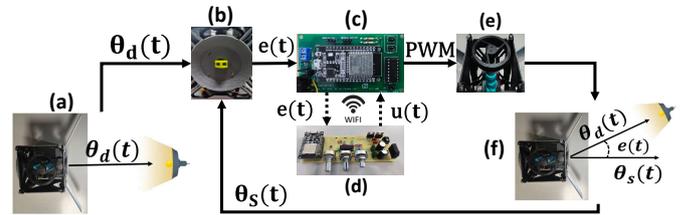


Figura 7: Diagrama a bloques del funcionamiento del prototipo. a) Posición del satélite deseado, b) Sensor de luz, c) PCB del CubeSat, d) PCB del control PID, e) Actuator, f) Posición de salida del CubeSat.

El sistema de control de orientación del CubeSat se compone de los siguientes elementos clave:

1. Rueda de Reacción: Es el actuador principal que ajusta la orientación del nanosatélite.
2. Sensores de Luz (LDR): Dos LDRs separados por una pequeña pared que proyecta una sombra, permitiendo medir el error de orientación con respecto a la fuente de luz.
3. Controlador PID: Ajusta la velocidad de la rueda de reacción para minimizar el error de orientación.
4. Estructura del CubeSat: impreso con material PET-G de diámetro $1.7 [mm]$.

El ciclo de trabajo comienza con la detección de una señal de referencia $\theta_d(t)$ provocada por una fuente luz en el entorno (Figura 7 (a)). Esto se logra al comparar las señales obtenidas de las fotorresistencias en el sensor de luz para estimar el error en el posicionamiento $e(t) = \theta_d(t) - \theta_s(t)$.

Este error es recibido por la primera tarjeta de adquisición ESP32 (placa 1) que se encuentra dentro de la estructura del CubeSat (Figura 7 (c)), después se envía a través de WiFi a una segunda tarjeta de adquisición ESP32 (placa 2) (Figura 7 (d)). En este punto la señal de error es procesada por el control PID analógico y el resultado enviado de regreso la ESP32 en el CubeSat (placa 1). La señal de control recibida por la ESP32 (placa 1) se convierte a una señal PWM adecuada para hacer funcionar

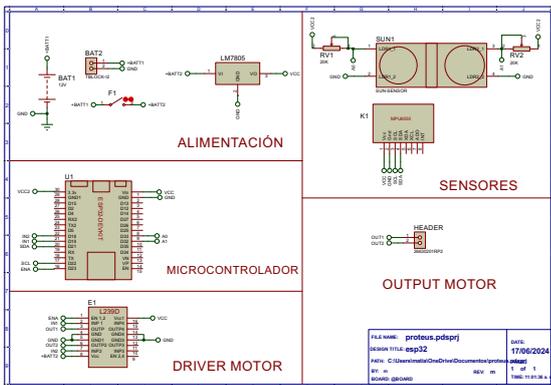


Figura 5: PCB nanosatélites.

el motor de la rueda inercial (Figura 7 (e)). La manipulación de la velocidad del motor de la rueda de reacción ocasionará una orientación o posicionamiento en el nanosatélite (Figura 7 (f)). La posición del nanosatélite se retroalimenta en tiempo real y el proceso termina cuando el $e(t)$ es cercano a cero.

4.3. Puesta en marcha del sistema

4.3.1. Sensor de Luz

El sensor para la detección de una fuente de luz se compone de una pared impresa en 3D y dos fotorresistencias R_{LDR1} y R_{LDR2} , ambas dispuestas en divisores de voltaje como se muestra en la Figura 8 (a). Así el voltaje de cada fotorresistencia está dado por las siguientes expresiones:

$$V_i = V \frac{R_i}{R_{LDRi} + R_i}, \quad i = 1, 2, \quad (14)$$

donde V_i es el voltaje en la fotorresistencia, V es el voltaje de entrada en el circuito, R_{LDRi} es el valor de la fotorresistencia y R_i es la resistencia del divisor de voltaje.

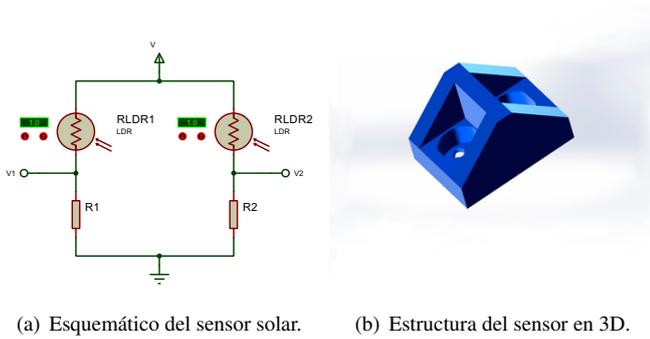


Figura 8: Diseño del sensor.

La estructura del sensor (Figura 8 (b)) está diseñada para que la longitud de la pared entre las dos fotorresistencias sea igual al diámetro de estas, esto con el propósito de proyectar una sombra sobre cada fotorresistencia a partir de una fuente de luz. Así, las resistencias de las mismas están dadas por una relación entre la longitud de la sombra proyectada y el ángulo de incidencia α , de la luz sobre el sensor. Note que cuando la luz incide completamente sobre una fotorresistencia, entonces esta proporciona su valor mínimo, $R_{\min 1}$ para la fotorresistencia izquierda y $R_{\min 2}$ para la fotorresistencia derecha. Asimismo, cuando se proyecta una sombra máxima (la sombra cubre completamente el sensor), estas adquieren un valor máximo ($R_{\max 1}$ para la izquierda y $R_{\max 2}$ para la derecha). Además se asume que:

- Solo una fuente de luz incide sobre el sensor con un ángulo α .
- El valor de las fotorresistencias es proporcional a la sombra proyectada.

Por lo cual, se propone la siguiente relación entre ángulo de

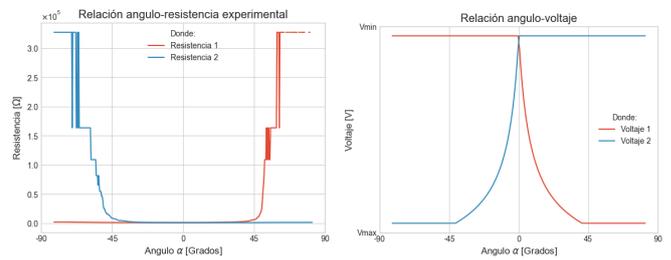
incidencia α y el valor de la fotorresistencia:

$$R_{LDR1}(\alpha) = \begin{cases} \alpha < 0 & \text{si } R_{\min 1} \\ 0 \leq \alpha \leq 45 & \text{si } R_{\min 1} + (R_{\max 1} - R_{\min 1}) \tan(\alpha) \\ \alpha > 45 & \text{si } R_{\max 1} \end{cases}, \quad (15)$$

Note que para la resistencia izquierda, la relación se aplica de la forma antes mencionada, para la resistencia derecha se toma el ángulo con signo contrario como argumento de la función, esto debido a que, es geoméricamente simétrico.

Siendo que el análisis de la segunda fotorresistencia es el mismo, pero geoméricamente invertido, si $R_{\min 1} = R_{\min 2}$ y $R_{\max 1} = R_{\max 2}$, ambas resistencias tiene la siguiente relación:

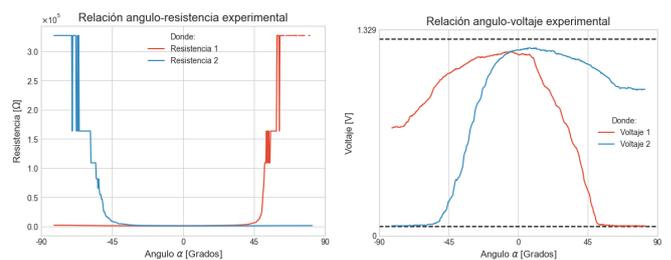
$$R_{LDR1}(\theta) = R_{LDR2}(-\theta). \quad (16)$$



(a) Gráfica del comportamiento de las fotorresistencias según la incidencia de la luz sobre el sensor. (b) Gráfica del comportamiento del voltaje sobre las fotorresistencias según la incidencia de la luz sobre el sensor.

Figura 9: Simulación de (15) realizada en Python.

De la misma manera se puede obtener la relación entre el voltaje y el ángulo de incidencia α con la fórmula del divisor de voltaje (Figura 9 (b)). En la gráfica se pueden observar como valores límite V_{\max} y V_{\min} , estos se entienden como el voltaje máximo y mínimo que puede suministrar una fotorresistencia.



(a) Gráfica del comportamiento del valor de las fotorresistencias según la incidencia de la luz sobre el sensor obtenidos de manera experimental. (b) Gráfica del comportamiento del voltaje sobre las fotorresistencias según la incidencia de la luz sobre el sensor obtenidos de manera experimental.

Figura 10: Implementación real del sensor descrito en (15).

Se realizaron pruebas experimentales para obtener la relación dada en (15), esto se puede observar en la Figura 10(a). De la misma manera se puede observar una similitud entre la relación ángulo-resistencia entre la Figura 10(b) y la Figura 9(b). Los resultados muestran que el rango de operación del sensor es ligeramente superior a 45 grados, esto puede deberse a pasar

por alto ciertos efectos en el comportamiento de la luz, y las fotorresistencias en el análisis de la relación entre el ángulo de incidencia y la resistencia.

4.3.2. Controlador PID analógico

De acuerdo con (Pertence, 1994), el amplificador operacional u OPAM (Operational Amplifier) es un dispositivo electrónico multietapa de alta ganancia que tiene dos entradas y una salida. Como se menciona en el libro, es difícil enumerar la totalidad de las aplicaciones de este componente electrónico. Sin embargo, para los efectos de este artículo, la aplicación del amplificador operacional se destaca por su facilidad para realizar operaciones matemáticas. En el contexto de nuestro trabajo sobre el control, es necesario utilizar un control analógico. Como menciona (Pertence, 1994) sobre los controladores PID analógicos, "su función básica es evaluar los errores y desviaciones de las variables controladas y enviar señales eléctricas a los dispositivos directamente relacionados con ellas, de forma que actúen corrigiendo los errores o desviaciones encontrados". De este modo, es importante mencionar las configuraciones del OPAM que se han utilizado para la realización de este artículo, así como sus respectivas funciones de transferencia.

A continuación se da una descripción de las configuraciones empleadas con OPAMs para la emulación de las acciones de un controlador PID, ver Figura 11. Esta figura fue realizada en PROTEUS 8.13.

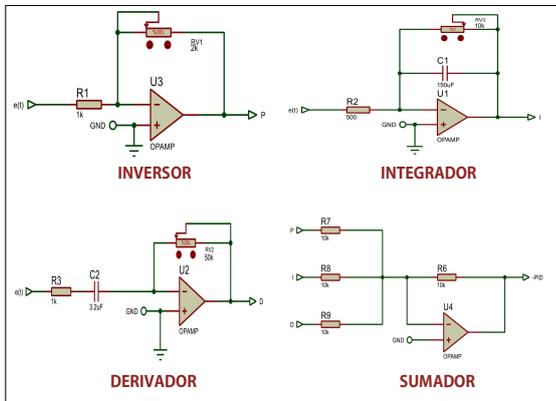


Figura 11: Esquemático del controlador PID.

Amplificador Inversor: La ecuación que describe la dinámica del voltaje en este amplificador es:

$$V_{out}(t) = -\frac{R_f}{R_1} \cdot V_i(t), \quad (17)$$

donde \$R_f\$ es una resistencia variable y \$R_1\$ es una resistencia fija. En la Figura 11, se muestra la simulación del esquema anterior utilizando Proteus, con \$R_f = 2[k\Omega]\$ y \$R_1 = 1[k\Omega]\$, así \$k_p \in [0, 2]\$. Note que si \$k_p = -\frac{R_f}{R_1}\$, \$u(t) = V_{out}(t)\$ y \$e(t) = V_i(t)\$, entonces tenemos una acción proporcional:

$$u(t) = k_p \cdot e(t), \quad (18)$$

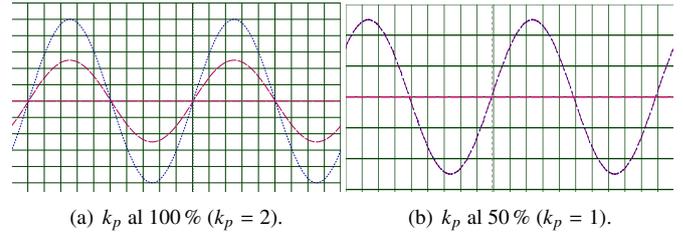


Figura 12: Simulación de acción P.

Amplificador Integrador: La ecuación que me describe la dinámica del voltaje en este amplificador es:

$$V_{out}(t) = -\frac{1}{R_1 C} \int_0^t V_i(\theta) d\theta, \quad (19)$$

donde \$R_1\$ es una resistencia fija y \$C\$ capacitancia fija. Así, si \$k_i = \frac{1}{RC}\$, \$u(t) = V_{out}(t)\$ y \$e(t) = V_i(t)\$.

$$u(t) = k_i \int_0^t e(\theta) d\theta. \quad (20)$$

Sin embargo, para este artículo se empleo una configuración de uso práctico como lo menciona (Pertence, 1994), en donde la relación de ganancia del OPAM Integrador está dada por la siguiente ecuación:

$$A_{vf} = \frac{\frac{R_f}{R_1}}{\sqrt{1 + (2\pi f R_f C)^2}}, \quad (21)$$

donde \$R_1\$ es una resistencia de entrada fija, \$R_f\$ es una resistencia variable, \$C\$ es una capacitancia y \$f\$ es una frecuencia. De este modo \$k_i\$ puede ser tomada como la ecuación (21). En la Figura 13 se muestra la simulación utilizando Proteus. Aquí, \$R_f = 10[k\Omega]\$, \$R_1 = 500[\Omega]\$, \$C = 150[\mu f]\$ y nuestra frecuencia de \$1[Hz]\$, podemos decir que \$k_i \in [0, 2]\$.

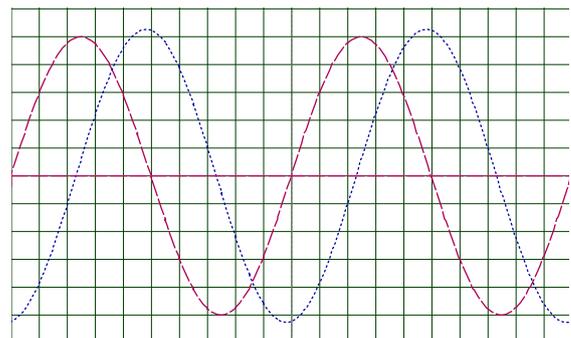


Figura 13: \$k_i\$ al 100% (\$k_i = 2\$).

Amplificador Derivador: La ecuación que me describe la dinámica del voltaje en este amplificador es:

$$V_{out}(t) = -R_1 C \dot{v}_i(t), \quad (22)$$

donde \$R_1\$ es una resistencia fija y \$C\$ es una capacitancia fija. Así, si \$k_d = -R_1 C\$, \$u(t) = V_{out}(t)\$ y \$\dot{e}(t) = \dot{v}_i(t)\$.

$$u(t) = k_d \dot{e}(t). \quad (23)$$

De igual manera como lo menciona (Pertence, 1994) existe una configuración de uso práctico para la configuración del OPAM

derivador donde la relación de ganancia está dada por la siguiente ecuación:

$$A_{vf} = \frac{R_f}{R_1} \frac{1}{\sqrt{1 + \left(\frac{1}{2\pi f C R_1}\right)^2}}, \quad (24)$$

donde R_1 es una resistencia de entrada fija, R_f es una resistencia variable, C es una capacitancia y f es la frecuencia. En la Figura 14 se muestra la simulación del esquemático de la Figura 11 utilizando Proteus. Aquí, $R_f = 50[k\Omega]$, $R_1 = 1[k\Omega]$, $C = 3.3[\mu f]$ y nuestra frecuencia es de $1[Hz]$. Por lo que, podemos decir que $k_d \in [0, 1]$.

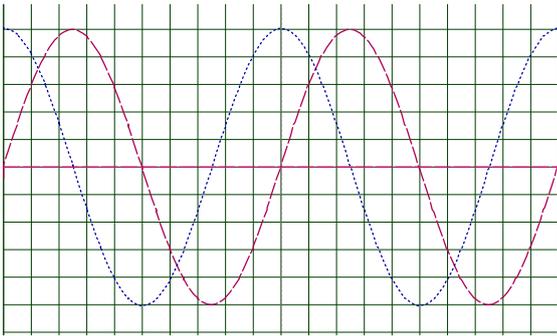


Figura 14: k_d al 100% ($k_d = 1$).

Amplificador Sumador: Por último, el Amplificador Sumador nos adiciona las acciones anteriores y es debido a esto que la siguiente ecuación describe la dinámica del voltaje :

$$V_{out}(t) = -\frac{R_f}{R} (v_1(t) + v_2(t) + v_3(t)). \quad (25)$$

Así, si $R_f = R$, $v_1(t)$ es igual a la ecuación (17), $v_2(t)$ es igual a la ecuación (19) y $v_3(t)$ es igual a la ecuación (22), podemos reescribir la ecuación (25), quedando de la siguiente forma:

$$V_{out}(t) = \frac{R_f}{R_1} \cdot V_i(t) + \frac{1}{R_1 C} \int_0^t V_i d(t) + R_1 C \cdot \dot{v}_i(t). \quad (26)$$

La cual tiene la estructura de un clásico control PID:

$$u(t) = k_p \cdot e(t) + k_i \cdot \int_0^t e(\theta) d\theta + k_d \cdot \dot{e}(t).$$

En la Figura 15 se muestra la PCB destinada para el control PID usando la descripción anterior y la cual emulará una estación de control terrestre. Cabe mencionar que esta placa también cuenta con una tarjeta ESP32 para conexión inalámbrica vía WiFi, con la PCB del CubeSat. Además de la comunicación, también las ESP32 operan como tarjetas de adquisición de datos.



Figura 15: PCB del PID.

4.3.3. Rueda de reacción y actuador

Como se menciona en la Subsección 4.1, el actuador utilizado es un motor DC de 12V a 5000RPM. A su vez, la rueda de reacción (Figura 16) posee un diámetro $D = 0.10[m]$ y un masa $M = 0.03193[kg]$.

La ecuación que describe la velocidad del motor, se basa en la siguiente relación entre el voltaje del motor y el error del sensor:

$$v_m(v_s) = e(v_s) * v_{mMax},$$

donde $v_m(v_s)$ es el voltaje aplicado al motor de la rueda de reacción, v_{mMax} es el voltaje máximo del motor, $e(v_s) = v_1 - v_2$ es el error dependiente del voltaje del sensor.

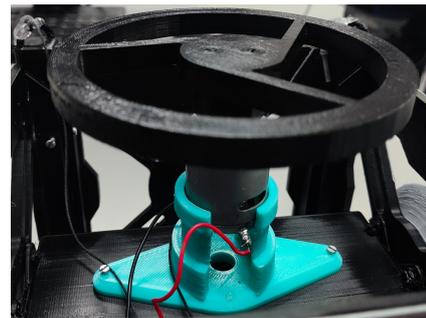


Figura 16: Rueda de reacción sobre el eje del motor (actuador).

4.3.4. Costos de la plataforma

Con un costo total de \$2220 pesos, siendo la batería el componente más caro, casi 1/3 del costo. Este proyecto proporciona una solución asequible y eficaz para aquellos interesados en el desarrollo de tecnología espacial, permitiendo una inmersión directa en el campo sin la barrera de un presupuesto elevado.

El desglose detallado de los costos se presenta en la Tabla 2:

Tabla 2: Costos de plataforma

Objeto	Descripción	Costo
1	Estructura del CubeSat de 15cmx15cm	\$ 300.00
2	La PCB fue diseñada en Proteus y mandada a hacer en JLCPCB	\$ 456.00
3	Motor de 12V	\$ 70.00
5	Estructura Metálica	\$ 70.00
6	Batería Lipo de 11.1v, 1300mA	\$ 725.00
7	Puente H	\$ 55.00
8	Regulador Lm7805	\$ 15.00
9	Sensor de Luz	\$ 20.00
10	Switch	\$ 10.00
11	Convertidor DC-DC	\$ 80.00
12	2 ESP32	\$ 300.00
13	Hilo	\$ 4.00
14	Cadena	\$ 20.00
15	Header M-H	\$ 10.00
16	Header M-M	\$ 10.00
17	Capacitores y resistencias	\$ 50.00
18	Quad opamp LM324	\$ 25.00
	Total	\$ 2220.00

5. Resultados experimentales

En seguida, se emplean el modelo matemático, el método de identificación paramétrica y el método de Ziegler-Nichols, dados en la sección anterior, para sintonizar las ganancias del controlador PID y obtener el desempeño de la respuesta en lazo cerrado. Los resultados presentados ilustran una posible aplicación para el posicionamiento angular del nanosatélite, en tareas de seguimiento de luz, dando indicios de ser una opción eficiente para este sector.

5.1. Implementación del modelo matemático

Aquí, se emplea el modelo matemático propuesto en la Subsección 3.1 y se emplea asignación de polos para cumplir con el criterio de Hurwitz para sintonizar el controlador PID y asegurar el seguimiento de la fuente de luz.

Considere el modelo matemático dado en (4). Sabiendo que la masa de la rueda de reacción es $M_R = 0.03193[kg]$ y su radio es $R_R = 0.05[m]$, se puede calcular el momento de inercia de la rueda:

$$I_R = \frac{1}{2} (0.03193 [kg]) (0.05 [m])^2 = 3.99125 \times 10^{-5} [kg \cdot m^2].$$

Asimismo, se puede calcular el momento de inercia del CubeSat, cuya masa es $M_S = 0.814[kg]$ y longitud de arista $l = 0.15[m]$:

$$I_S = \frac{1}{6} (0.814[kg]) (0.15[m])^2 = 0.003053[kg \cdot m^2].$$

Además, note que $a_1 = 2I_S\delta\omega = 2(0.003053)(0.1)(1.2) = 0.00073$, $a_2 = I_S\omega^2 = (0.003053)(1.2)^2 = 0.0044$ y $I_R k_c = (3.99125 \times 10^{-5})(0.6) = 2.4 \times 10^{-5}$. Así, empleando los datos anteriores, la ecuación (4) queda como

$$\ddot{\theta}_S(t) + 0.2391 \dot{\theta}_S(t) + 1.44 \theta_S(t) = 0.0078 V(t),$$

Finalmente, usando (13), los parámetros dados anteriormente y el método de asignación de polos se puede sintonizar el controlador PID. Sin embargo, para proponer un polinomio deseado $p_d(s)$ se debe tomar en cuenta que debido al diseño de control PID analógico, los valores de las ganancias k_p , k_d y k_i deben estar entre $[0, 1]$. En este sentido y sin pérdida de generalidad, se propone $p_d(s) = (s^2 + 0.2414s + 1.4471)(s + 0.0016)$. Así tenemos que

$$k_p = 0.8, k_d = 0.5 \text{ y } k_i = 0.3. \quad (27)$$

Además, solo se tiene la intención de ejemplificar la sintonización de controlador PID empleando un modelo matemático.

En la Figura 17, se muestra la respuesta temporal de la plataforma experimental en lazo cerrado, con el control PID analógico usando las ganancias dadas en (27).

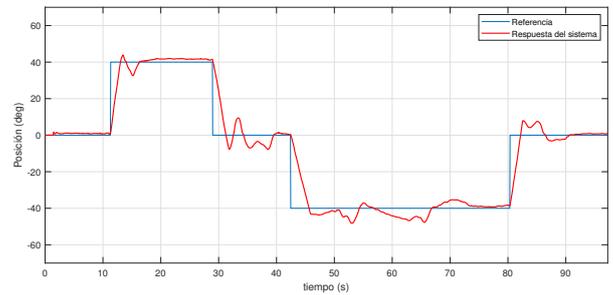


Figura 17: Respuesta real de la plataforma experimental en lazo cerrado con un PID usando las ganancias dadas en (27).

5.2. Implementación de la identificación paramétrica

Aquí, se implementa una técnica de identificación para la obtención de los valores de los parámetros de la plataforma, bajo la suposición de que esta puede representarse como un sistema de segundo orden de la forma (5). Además, se emplea asignación de polos para cumplir con el criterio de Hurwitz.

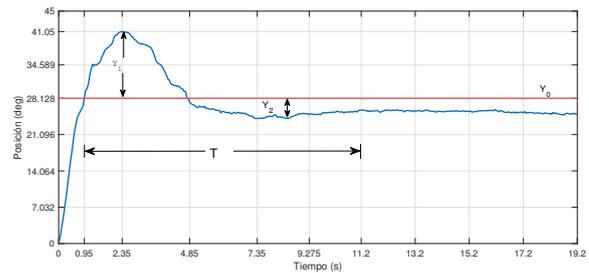


Figura 18: Sistema real con respuesta al impulso.

Considere la respuesta escalón de la plataforma obtenida experimentalmente y dada en la Figura 18. Como se mencionó en la Subsección 3.2, a partir de esta figura se puede obtener los valores de δ y ω . Aquí, $Y_0 = 28.128$, $Y_1 = 12.913$ y $Y_2 = 3.933$, así empleando (7) se tiene que

$$\delta_1 = \frac{\left| \ln \left(\frac{12.913}{28.128} \right) \right|}{\sqrt{\pi^2 + \ln^2 \left(\frac{12.913}{28.128} \right)}} = 0.2405382,$$

$$\delta_2 = \frac{\left| \ln \left(\frac{3.933}{28.128} \right) \right|}{\sqrt{4\pi^2 + \ln^2 \left(\frac{3.933}{28.128} \right)}} = 0.2988,$$

y usando (8) tenemos que

$$\delta = \frac{0.2405382 + 0.2988}{2} = 0.269674. \quad (28)$$

Ahora, dado que $T = 10.25[s]$, ver Figura (18), y usando el valor de δ obtenido en (28) tenemos que

$$\omega = \frac{2\pi}{(10.25[s]) \sqrt{1 - (0.269674)^2}} = 0.636577858[Hz]. \quad (29)$$

En la Figura 19 se muestra la respuesta temporal escalón correspondiente a la función de transferencia dada en (5), utilizando los valores de δ y ω obtenidos en (28) y (29), respectivamente. Aquí, el polinomio característico correspondiente es

$$p(s) = s^2 + 2(0.2697)(0.6366)s + (0.6366)^2. \quad (30)$$

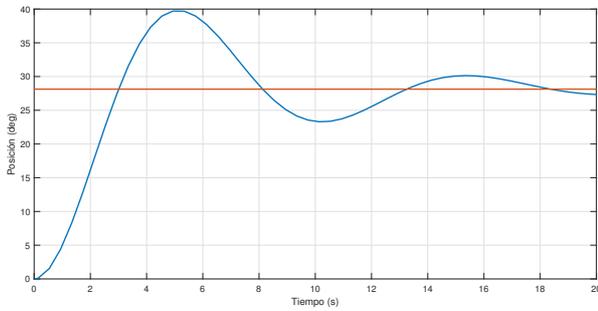


Figura 19: Simulación de la respuesta escalón.

En la Figura 20 se muestra la respuesta temporal de la plataforma experimental en lazo cerrado con un control PID analógico descrito en (26). Aquí, para la sintonización de las ganancias se empleó colocación de polos y la función de transferencia dada en (5) cuyos parámetros se obtuvieron empleando el método de identificación descrito anteriormente.

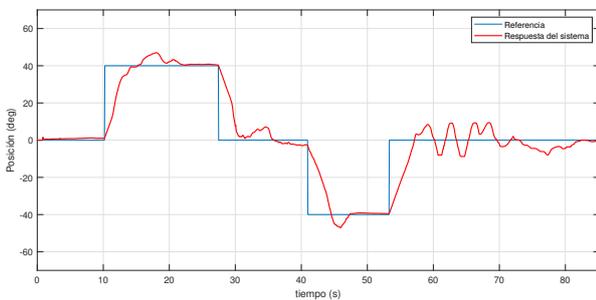


Figura 20: Respuesta temporal del sistema $\theta_S(t)$ en lazo cerrado con un PID usando el método de identificación paramétrica y asignación de polos.

5.3. Sintonización empleando de Ziegler-Nichols

Para aplicar el método de Ziegler-Nichols, se busca obtener oscilaciones sostenidas con un periodo P_{cr} haciendo uso de

una ganancia crítica K_{cr} . Para ello, las ganancias del controlador se configuran como $k_i = 0$ y $k_d = 0$, mientras que para la ganancia proporcional k_{cr} se realiza un barrido hasta lograr que el sistema oscile de manera periódica. De manera experimental se encontró que la ganancia crítica que satisface la condición de oscilación del sistema tomó un valor de $k_{cr} = 1.388$. Para la obtención del periodo crítico, se graficó la respuesta del sistema real, como lo muestra la Figura 21. A partir de esta, se puede obtener una medición del periodo crítico cuyo valor se estimó en $P_{cr} = 4.4s$.

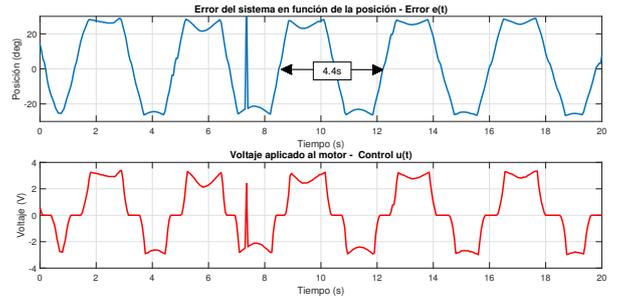


Figura 21: Respuesta del sistema con $K_{cr} = 1.388$.

Debido al requerimiento de realizar un controlador analógico para el sistema, se optó por implementar un controlador PID haciendo uso las reglas de Ziegler-Nichols mencionadas en la sección 3.4, y estimar los valores de los componentes electrónicos para asegurar las respectivas ganancias de k_p , k_i y k_d . Las reglas de Ziegler-Nichols establecen ganancias aproximadas para asegurar que la sintonización del controlador proporcione la estabilidad al sistema. De este modo, la estimación de las ganancias se realiza en función de los parámetros provistos por este método como se muestra en la Tabla 1. Entonces la ganancias resultantes son las siguientes:

$$k_p = 0.6(1.388) = 0.8328,$$

$$k_i = \frac{1.2(1.388)}{4} = 0.3785$$

$$k_d = 0.075(1.388)(4.4) = 0.4580.$$

Para la construcción de la configuración proporcional con un amplificador operacional se utiliza una resistencia de entrada R_1 con un valor de $1[k\Omega]$, por lo que el análisis de la resistencia R_f para obtener la ganancia k_p deseada se observa como

$$R_f = (0.8328)(1[k\Omega]) = 832.8[\Omega].$$

Experimentalmente se encontró que el sistema respondía mejor cuando la resistencia R_f de esta configuración es igual a $990[\Omega]$, lo que implica que la ganancia proporcional toma un valor de $k_p = 0.99 \approx 1$, lo cual no es de extrañarse, ya que como lo expresa (Ogata, 2001), este método es un punto de partida para la sintonización fina del controlador. La construcción de la configuración integral con un amplificador operacional utilizó una resistencia de entrada $R_1 = 510[\Omega]$, una capacitancia $C = 150[\mu F]$ y se estableció una frecuencia de trabajo de $1[Hz]$. El cálculo de la resistencia R_f para asegurar el valor de la ganancia k_i se puede obtener mediante el despeje de la ecuación

(21), tomando como resultado el valor positivo.

$$R_f = \frac{(0.3785)(510\Omega)}{\sqrt{1 - (2\pi \cdot 1[Hz] \cdot 150\mu F \cdot 0.3785 \cdot 510\Omega)^2}} \approx 195[\Omega].$$

La configuración derivativa construida con un amplificador operacional utiliza una resistencia de entrada $R_1 = 1[k\Omega]$, una capacitancia $C = 3.3[\mu F]$ y una frecuencia de trabajo de $1[Hz]$, igual a la del control integral. El cálculo de la resistencia R_f se obtiene mediante el despeje de la Ecuación (24)

$$R_f = (0.45804)(1k\Omega) \sqrt{1 + \left(\frac{1}{2\pi \cdot 1Hz \cdot 3.3\mu F \cdot 1k\Omega}\right)^2} \approx 22[k\Omega].$$

Las resistencias variables del circuito mostrado en la Figura 15 se configuraron según los cálculos realizados anteriormente.

En la Figura 22 se muestra la respuesta del sistema $\theta_S(t)$ y el voltaje de control $u(t)$ suministrado al motor (actuador). Aquí, la consigna es solo de estabilización, es decir, dejando una fuente de luz fija.

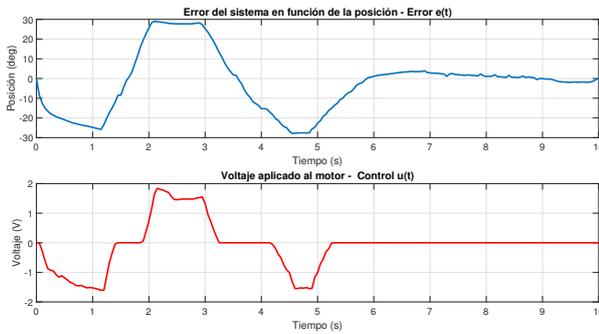


Figura 22: Respuesta del sistema a una fuente de luz fija con un control PID estimado con el segundo método de Ziegler-Nichols.

En la Figura 23 se muestra la respuesta del sistema $\theta_S(t)$ ahora cambiando la posición de la luz, variando entre -40° a 40° .

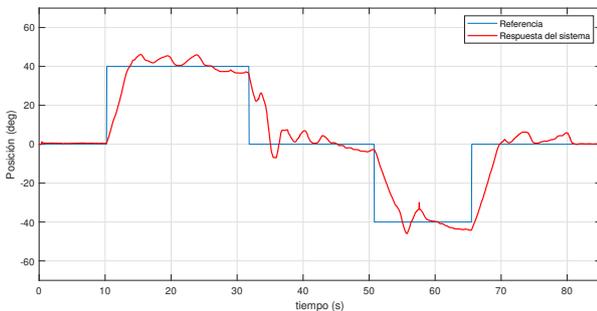


Figura 23: Posicionamiento real del nanosatélite usando Ziegler-Nichols.

6. Conclusiones

En el presente manuscrito se propone el diseño, la construcción y el control de una plataforma experimental inspirada en

un *CubeSat*. El diseño se realizó con base a la propuesta dada en (Rodríguez Pirateque et al., 2020) usando un software de diseño CAD, SolidWorks. Para la puesta en marcha se considero la construcción de dos PCB, con comunicación WiFi: una para la electrónica de nanosatélite y otra para la electrónica del PID analógico. Así como la construcción de un sensor para estimar el error $e(t)$, entre la posición del nanosatélite y la fuente de luz. Y el empleo de una rueda inercial para generar una fuerza externa para orientar el posicionamiento de dicha plataforma. También, se propuso un modelo matemático simple que describe la dinámica del posicionamiento del *CubeSat*. Para garantizar que el error del posicionamiento $e(t)$ converja a cero, se realiza una estimación paramétrica y se emplea el criterio de Ziegler-Nichols. Para corroborar el desempeño de la plataforma y los resultados obtenidos, en la Sección 5 se presentan los resultados experimentales. Estos resultados muestran que la plataforma experimental propuesta es una sugerente alternativa para emular la dinámica de orientación de un nanosatélite, la cual puede contribuir al entendimiento y enseñanza de este tipo de técnicas empleadas en el sector aeroespacial. Un vídeo del desempeño se puede observar en <https://youtu.be/zuoxnErKPhE?si=yIjk3WkkuEF6RbnZ>.

Referencias

- Afifa, L. N., Priyambodo, T. K., and Dharmawan, A. (2021). Preliminary modelling of a cubesat attitude control system using a reaction wheel. In *International Conference on Innovation in Science and Technology (ICIST 2020)*, pages 90–94. Atlantis Press.
- Alegre Bonet, M. (2022). Design of a 3dof attitude control for cubesats based on reaction wheels. Master’s thesis, Universitat Politècnica de Catalunya, Barcelona.
- Åström, K. and Hägglund, T. (2009). *Control PID avanzado*. isbn 9788483225110, Pearson, Madrid.
- Cannon Jr, R. H. (1962). Some basic response relations for reaction-wheel attitude control. *Ars Journal*, 32(1):61–74.
- Confidencial, E. (2023). Un millón de nuevos satélites esperan para llegar a la órbita de la tierra. Accessed on July 29, 2024.
- Fintualist (2023). ¿cuánto han aumentado los satélites artificiales en el espacio? Accessed on July 29, 2024.
- Froelich, R. and Papapoff, H. (1959). Reaction wheel attitude control for space vehicles. *IRE Transactions on Automatic Control*, 4(3):139–149.
- Geographic, N. (2022). La conquista de marte. Accessed on July 29, 2024.
- Helmy, M., Hafez, A. T., and Ashry, M. (2022). Cubesat reaction wheels attitude control via modified pi-d controller. In *2022 International Telecommunications Conference (ITC-Egypt)*, pages 1–6. IEEE.
- infoespacio.com (2023). Las constelaciones en órbitas baja y media impulsan el crecimiento de los satélites de alto rendimiento. Accessed on July 29, 2024.
- Jamie, C., Roland, C., Justin, F., and Alicia, J. (2017). Basic concepts and processes for first-time cubesat developers. *CubeSat101, CaliforDesign of a CubeSat Testbednia*.
- Krishna, N. S., Gosavi, S., Singh, S., Saxena, N., Kailaje, A., Datla, V., and Shah, P. (2018). Design and implementation of a reaction wheel system for cubesats. In *2018 IEEE Aerospace Conference*, pages 1–7. IEEE.
- Liddle, J. D., Holt, A. P., Jason, S. J., O’Donnell, K. A., and Stevens, E. J. (2020). Space science with cubesats and nanosatellites. *Nature Astronomy*, 4(11):1026–1030.
- Medina, I., Santiago, L., Hernández-Gómez, J., Castillo, R., and Couder-Castañeda, C. (2021). Speed pid controller simulation of a reaction wheel for cubesat orientation applications. In *Journal of Physics: Conference Series*, volume 1723, page 012013. IOP Publishing.
- NASANET (2024a). Estacion espacial internacional. Accessed on July 29, 2024.
- NASANET (2024b). Telescopios espaciales. Accessed on July 29, 2024.
- Ogata, K. (2001). *Modern Control Engineering*. Prentice Hall PTR, USA, 4th edition. ISBN:978-0-13-060907-6.

- Pertence, A. (1994). *Amplificadores operacionales y filtros activos: Teoría, proyectos y aplicaciones prácticas*. McGraw-Hill. Madrid. ISBN 84-7615-660-X.
- Pirateque, G. W. R., De La Peña, N. A., and García, E. D. C. (2020). Sustainable Design of a NanoSatellite Structure TypeCubeSat as a Modular Platform for Tests. *Revista Ciencia y Poder Aéreo/Ciencia y poder aereo*, 15(1):108–134.
- Rodríguez Pirateque, G. W., Arzola de la Peña, N., and Cortes García, E. D. (2020). Diseño sostenible de una estructura de nanosatélite tipo cubesat como plataforma modular para pruebas. *Ciencia y Poder Aéreo*, 15(1):108–134.
- Rodríguez Vázquez, A. L. (2013). Algoritmo de control magnético puro para satélites de órbita baja. *Dialnet*.
- Sinclair, D., Grant, C., and Zee, R. E. (2007). Enabling reaction wheel technology for high performance nanosatellite attitude control.
- Snider, R. E. (2010). *Attitude control of a satellite simulator using reaction wheels and a PID controller*. PhD thesis, Air Force Institute of Technology.
- Space, N. (1977). National aeronautics and space administration. Retrieved from *National Aeronautics and Space Administration: www.nasa.gov*.
- University, C. P. S. (2015). ¿qué es un cubesat? Accessed on May 27, 2024, California Polytechnic State niversity, San Luis Obispo, Building 192 (Engineering IV) Room 101, San Luis Obispo, California — 93407.
- Zlatanov, H. and Weinrebe, G. (2014). Csp and pv solar tracker optimization tool. *Energy Procedia*, 49:1603–1611.

Apéndice A. Código C++ empleado para la ESP32 del control PID

```

1 #include <esp_now.h>
2 #include <WiFi.h>
3
4 uint8_t broadcastAddress[] = {0xB0, 0xB2, 0x1C, 0
5     x97, 0x89, 0xA8};
6
7 int Pint;
8 int Iint;
9 int Dint;
10
11 int SignalPID;
12 float Kpf = 0;
13 float Kif = 0;
14 float Kdf = 0;
15
16 int incomingA;
17 int incomingB;
18 int incomingDeltaUnsigned;
19
20 String success;
21
22 struct GainsInt{
23     int Kpint;
24     int Kiint;
25     int Kdint;
26 };
27
28 typedef struct struct_message {
29     int A;
30     int B;
31     int Delta;
32 } struct_message;
33
34 typedef struct PID_message {
35     int Signal;
36 } PID_message;
37
38 struct_message incomingReadings;
39
40 PID_message PID;
41
42 class Gains {
43 public:
44     byte GP;
45     byte GI;

```

```

46     byte GD;
47     byte Gsig;
48
49     Gains(byte G_P, byte G_I, byte G_D, byte
50         G_signal);
51     float readGainP();
52     float readGainI();
53     float readGainD();
54     int readPIDSignal();
55 };
56
57 Gains::Gains(byte G_P, byte G_I, byte G_D, byte
58     G_signal) {
59     GP = G_P;
60     GI = G_I;
61     GD = G_D;
62     Gsig = G_signal;
63 }
64
65 int Gains::readPIDSignal(){
66     return analogRead(Gsig);
67 }
68
69 float Gains::readGainP(){
70     const int GainP = map(analogRead(GP), 0, 4095,
71         0, 1024);
72     return float (GainP/1024.0);
73 }
74
75 float Gains::readGainI(){
76     const int GainI = map(analogRead(GI), 0, 4095,
77         0, 1024);
78     return float (GainI/1024.0);
79 }
80
81 float Gains::readGainD(){
82     const int GainD = map(analogRead(GD), 0, 4095,
83         0, 1024);
84     return float (GainD/1024.0);
85 }
86
87 esp_now_peer_info_t peerInfo;
88
89 void OnDataSent(const uint8_t *mac_addr,
90     esp_now_send_status_t status) {}
91
92 void OnDataRecv(const uint8_t * mac, const
93     uint8_t *incomingData, int len) {
94     memcpy(&incomingReadings, incomingData, sizeof(
95         incomingReadings));
96     incomingA = incomingReadings.A;
97     incomingB = incomingReadings.B;
98     incomingDeltaUnsigned = incomingReadings.Delta;
99 }
100
101 int PIDScale = 0;
102
103 const byte PinKp = 39;
104 const byte PinKi = 34;
105 const byte PinKd = 35;
106 const byte SignalDAC = 25;
107 const byte SignalADC = 4;
108
109 Gains Gains(PinKp, PinKi, PinKd, SignalADC);
110
111 void setup() {
112     Serial.begin(115200);
113     analogReadResolution(10);
114     WiFi.mode(WIFI_STA);
115
116     if (esp_now_init() != ESP_OK) {
117         return;
118     }

```

```

112 esp_now_register_send_cb(OnDataSent);
113
114 memcpy(peerInfo.peer_addr, broadcastAddress, 6)
115 ;
116 peerInfo.channel = 0;
117 peerInfo.encrypt = false;
118
119 if (esp_now_add_peer(&peerInfo) != ESP_OK){
120     return;
121 }
122 esp_now_register_rcv_cb(OnDataRecv);
123 }
124
125 void loop() {
126     getReadings();
127     PID.Signal = SignalPID;
128     updateDisplay();
129
130     esp_err_t result = esp_now_send(
131         broadcastAddress, (uint8_t *) &PID, sizeof(
132         PID));
133     delay(50);
134 }
135 void getReadings(){
136     SignalPID = analogRead(PinKd);
137     incomingA = 10000 + incomingA;
138     incomingB = 10000 + incomingB;
139     PIDEscale = 10000 + SignalPID;
140 }
141
142 void updateDisplay(){
143     dacWrite(SignalDAC, incomingDeltaUnsigned);
144     Serial.print(incomingA);
145     Serial.print(" ");
146     Serial.print(incomingB);
147     Serial.print(" ");
148     Serial.println(PIDEscale);
149 }

```

Listing 1: Código de la tarjeta de adquisición de datos ESP32 para el control PID

Apéndice B. Código C++ empleado para la ESP32 del nanosatélite

```

1 #include <esp_now.h>
2 #include <WiFi.h>
3
4 struct SolarData {
5     int dataA;
6     int dataB;
7     int dataDelta;
8 };
9
10 class SolarSensor {
11 public:
12     byte sensorA;
13     byte sensorB;
14
15     SolarSensor(byte sensor_a, byte sensor_b);
16     float readDelta();
17     int readA();
18     int readB();
19     SolarData readAll();
20 };
21
22 SolarSensor::SolarSensor(byte sensor_a, byte
23     sensor_b) {
24     sensorA = sensor_a;

```

```

25 }
26
27 float SolarSensor::readDelta() {
28     const int senA = analogRead(sensorA);
29     const int senB = analogRead(sensorB);
30
31     return float(senA - senB) / float(max(senA,
32     senB));
33 }
34
35 int SolarSensor::readA() {
36     return analogRead(sensorA);
37 }
38
39 int SolarSensor::readB() {
40     return analogRead(sensorB);
41 }
42
43 SolarData SolarSensor::readAll() {
44     SolarData solardata;
45
46     solardata.dataA = analogRead(sensorA);
47     solardata.dataB = analogRead(sensorB);
48     solardata.dataDelta = solardata.dataA -
49     solardata.dataB;
50
51     return solardata;
52 }
53
54 class MotorController {
55 public:
56     byte pwm;
57     byte in1;
58     byte in2;
59
60     MotorController(byte PWM, byte In1, byte In2);
61     void setSpeed(int speed, int sign);
62 };
63
64 MotorController::MotorController(byte PWM, byte
65     In1, byte In2) {
66     pwm = PWM;
67     in1 = In1;
68     in2 = In2;
69
70     pinMode(in1, OUTPUT);
71     pinMode(in2, OUTPUT);
72 }
73
74 void MotorController::setSpeed(int speed, int
75     sign) {
76     if (sign < 0) {
77         digitalWrite(in1, HIGH);
78         digitalWrite(in2, LOW);
79         analogWrite(pwm, speed);
80     } else {
81         digitalWrite(in1, LOW);
82         digitalWrite(in2, HIGH);
83         analogWrite(pwm, speed);
84     }
85 }
86
87 struct struct_message {
88     int A;
89     int B;
90     int Delta;
91 } SatReadings;
92
93 struct incomingReadings {
94     int Signal;
95 } PID_message;
96
97 const int del = 50;

```

```

95
96 uint8_t broadcastAddress[] = { 0xC4, 0xDE, 0xE2,
    0x5B, 0x9E, 0x3C };
97
98 const byte solarPin1 = 32;
99 const byte solarPin2 = 33;
100 const byte bridge1 = 19;
101 const byte bridge2 = 18;
102 const byte bridgePWM = 23;
103
104 int UnsignedSpeed;
105
106 int LectureA;
107 int LectureB;
108 int LectureDeltaSigned;
109 int LectureDeltaUnsigned;
110
111 String success;
112
113 MotorController motorController(bridgePWM,
    bridge1, bridge2);
114 SolarSensor solarSensor(solarPin1, solarPin2);
115
116 esp_now_peer_info_t peerInfo;
117
118 void OnDataSent(const uint8_t *mac_addr,
    esp_now_send_status_t status) {}
119
120 void OnDataRecv(const uint8_t *mac, const uint8_t
    *incomingData, int len) {
121     memcpy(&PID_message, incomingData, sizeof(
        PID_message));
122     UnsignedSpeed = PID_message.Signal;
123 }
124
125 void setup() {
126     Serial.begin(115200);
127
128     WiFi.mode(WIFI_STA);
129
130     if (esp_now_init() != ESP_OK) {
131         return;
132     }
133
134     esp_now_register_send_cb(OnDataSent);
135
136     memcpy(peerInfo.peer_addr, broadcastAddress, 6)
    ;
137     peerInfo.channel = 0;
138     peerInfo.encrypt = false;
139
140     if (esp_now_add_peer(&peerInfo) != ESP_OK) {
141         return;
142     }
143     esp_now_register_recv_cb(OnDataRecv);
144 }
145
146 void loop() {
147     getSolarReadings();
148     SatReadings.A = LectureA;
149     SatReadings.B = LectureB;
150     SatReadings.Delta = LectureDeltaUnsigned;
151
152     esp_err_t result = esp_now_send(
        broadcastAddress, (uint8_t *)&SatReadings,
        sizeof(SatReadings));
153
154     motorController.setSpeed(UnsignedSpeed,
        LectureDeltaSigned);
155     delay(50);
156 }
157
158 void getSolarReadings() {
159     LectureA = solarSensor.readAll().dataA;
160     LectureB = solarSensor.readAll().dataB;
161     LectureDeltaUnsigned = abs(solarSensor.readAll
        ().dataDelta);
162     LectureDeltaSigned = solarSensor.readAll().
        dataDelta;
163
164     LectureDeltaUnsigned = map(LectureDeltaUnsigned
        , 0, 4095, 0, 255);
165 }

```

Listing 2: Código de la tarjeta de adquisición de datos ESP32 para el sistema del satélite