

1 **Diseño e implementación de un sistema de control de actitud basado en PID para**
2 **el prototipo CubeSat-VA-I, Ayacucho – Perú**

3 **Design and implementation of a PID-based attitude control system for the**
4 **CubeSat-VA-I prototype, Ayacucho – Peru**

5
6 Lola Estefani Espinoza Vitor^{1*}; Jose Luis Huayanay Villar¹; Fernando Beizaga¹, Cristhian Sotelo¹, Kevin
7 Ayala¹

8
9 ¹Villa Automation S.A.C., RUC: 20610009400, Ayacucho, Perú. villaautomationeirl@gmail.com

10 **Correspondencia:* Jose Luis Huayanay Villar– villaautomationeirl@gmail.com/ [https://orcid.org/orcid:0000-](https://orcid.org/orcid:0000-0001-8389-7922)
11 [0001-8389-7922](https://orcid.org/orcid:0001-8389-7922)

12 DOI: <https://doi.org/10.0000/villautomation.art130> Aceptación:02/01/2026 Publicación:05/03/2026

13 **Resumen**

14 Este trabajo describe el diseño, construcción e implementación de un sistema de control de actitud basado en el
15 algoritmo Proporcional-Integral-Derivativo (PID) para el prototipo de nanosatélite CubeSat-VA-I, desarrollado
16 por Villa Automation S.A.C. en Ayacucho, Perú. El prototipo emplea una estructura cilíndrica de madera de tres
17 pisos (11 cm de diámetro, 23.5 cm de altura total), un microcontrolador Arduino UNO, el sensor inercial
18 MPU6050 y tres ruedas de reacción compuestas por motores brushless A2212 controlados por reguladores de
19 velocidad electrónicos (ESC) de 30 A. El controlador PID opera en modo rate control, tomando como variable
20 de medición la velocidad angular provista por el giroscopio del MPU6050. Los resultados experimentales
21 muestran que en reposo el sistema registra velocidades angulares de 0.00 °/s en los tres ejes; ante una
22 perturbación manual, detecta velocidades de hasta 40 °/s y genera correcciones PWM de ±400 µs de forma
23 autónoma.

24 **Palabras clave:** CubeSat; Control PID; Ruedas de reacción; MPU6050; Control de actitud.

25
26 **Abstract**

27 This paper describes the design, construction, and implementation of an attitude control system based on the
28 Proportional-Integral-Derivative (PID) algorithm for the CubeSat-VA-I nanosatellite prototype, developed by
29 Villa Automation S.A.C. in Ayacucho, Peru. The prototype employs a three-tiered cylindrical wooden structure
30 (11 cm in diameter, 23.5 cm in total height), an Arduino UNO microcontroller, the MPU6050 inertial sensor,
31 and three reaction wheels composed of A2212 brushless motors controlled by 30 A electronic speed controllers
32 (ESCs). The PID controller operates in rate control mode, using the angular velocity provided by the MPU6050
33 gyroscope as the measured variable. Experimental results show that at rest, the system registers angular
34 velocities of 0.00°/s on all three axes; In response to a manual disturbance, it detects speeds up to 40°/s and
35 autonomously generates PWM corrections of ±400 µs.

36 **Keywords:** CubeSat; PID control; Reaction wheels; MPU6050; Attitude control.

37

38

1. Introducción

39 Un CubeSat es un tipo de nanosatélite cuya unidad base (1U) mide exactamente 10 cm × 10 cm × 10 cm y pesa
40 menos de 1.33 kg (Puig-Suari, Turner, & Ahlgren, 2001). Este estándar fue propuesto en 1999 por las
41 universidades de California Polytechnic State y Stanford para reducir el costo de acceso al espacio y facilitar el
42 desarrollo tecnológico en instituciones pequeñas. Uno de los subsistemas más importantes de cualquier satélite
43 es el Sistema de Determinación y Control de Actitud (ADCS, por sus siglas en inglés), que mantiene la
44 orientación del satélite en el espacio (Wertz, 1978) . Sin un ADCS funcional, los paneles solares no apuntarán
45 al sol, las antenas no apuntarán a la Tierra y la carga útil no podrá cumplir su misión.

46 Las ruedas de reacción son el actuador más común en satélites pequeños (Sidi, 1997). Su principio de
47 funcionamiento se basa en la conservación del momento angular: si el motor de la rueda acelera en una dirección,
48 el satélite rota en la dirección opuesta, sin necesitar propelente. El algoritmo de control más utilizado en la
49 industria es el PID (Proporcional-Integral-Derivativo) (Ogata, 1998). Su popularidad se debe a que solo requiere
50 tres parámetros para sintonizarse, es robusto ante perturbaciones y puede implementarse en microcontroladores
51 de bajo costo como el Arduino UNO.

52 Antecedentes internacionales. En el ámbito internacional, diversos grupos de investigación han abordado el
53 problema del control de actitud en CubeSats mediante ruedas de reacción y controladores PID. Hafez et al.
54 (2022) compararon un controlador PID clásico con un PI-D modificado optimizado mediante Algoritmo
55 Genético para el control de actitud de un CubeSat a lo largo de los tres ejes, demostrando que la sintonización
56 automática de ganancias mejora significativamente el tiempo de establecimiento ante perturbaciones externas y
57 ruido blanco Ogata, K. (1998). Quintanilla et al. (2022), en la Universidad Nacional de San Agustín de Arequipa
58 (UNSA), desarrollaron un banco de pruebas de bajo costo para algoritmos de control de actitud en CubeSats 1U
59 con ruedas de reacción, empleando un cojinete de aire esférico y un microcontrolador ARM Cortex M4; sus
60 resultados validaron el modelo matemático lineal y un controlador PID de doble lazo (actitud y par) para
61 maniobras de ángulos pequeños Ogata, K. (1998). Hartland (2023), en la Universidad Embry-Riddle
62 Aeronautical, diseñó y fabricó un banco de pruebas para CubeSats 1U/3U/6U con tres ruedas de reacción
63 ortogonales, demostrando control autónomo en tres grados de libertad con controladores PID, realimentación de
64 estado y control adaptativo (InvenSense, 2013). Kaliyaperumal et al. (2024), en el Indonesian Journal of
65 Electrical Engineering and Computer Science, presentaron un ADCS de bajo costo para CubeSats usando
66 motores BLDC, sensor MPU6050 y programación en Arduino IDE, logrando ajustes de actitud precisos dentro
67 de los parámetros definidos, con validación mediante software (Woodman, 2007). Negi (2024), en la
68 Universidad de California Davis, demostró que los discos duros reutilizados como ruedas de reacción (HDD-
69 RW) constituyen una solución económica y competente; las pruebas en vuelo parabólico elevaron el nivel de
70 madurez tecnológica de TRL 4 a TRL 6, confirmando estabilización y apuntamiento en microgravedad (Åström,,
71 & Hägglund, 1995).

72 Antecedentes nacionales. En el Perú, el interés por la tecnología CubeSat ha crecido de manera sostenida. La
73 Pontificia Universidad Católica del Perú (PUCP) lanzó en noviembre de 2013 el PUCP-Sat 1, el primer
74 nanosatélite peruano en llegar a la órbita terrestre baja, e incluyó como misión experimental un sistema de
75 estabilización basado en micro ruedas de reacción (HobbyKing, 2020). La Universidad Nacional de Ingeniería
76 (UNI) desarrolló el satélite Chasqui 1 desde 2008 y actualmente trabaja en el Chasqui 2, cuya misión secundaria
77 incluye un sistema de terminación y control de actitud para garantizar la reentrada dirigida del satélite al finalizar

78 su vida útil (Universidad Nacional de Ingeniería, 2014; CONIDA, 2023). Por su parte, la Universidad Nacional
 79 de San Agustín (UNSA), a través de su Instituto Astronómico y Aeroespacial Pedro Paulet (IAAPP), ejecutó
 80 entre 2022 y 2024 el proyecto “Desarrollo de un módulo ADCS para diseño de algoritmos de control de actitud
 81 en CubeSat de órbita baja”, financiado por PROCIENCIA-CONCYTEC, consolidando capacidades nacionales
 82 en control de actitud (PROCIENCIA-CONCYTEC, 2024; UNSA, 2024). Más recientemente, en 2024, el Centro
 83 Nacional de Operaciones de Imágenes Satelitales (CONIDA) inició junto con la Pontificia Universidad Católica
 84 del Perú (PUCP) y la UNI el desarrollo del nanosatélite Ayni-SAT, con lanzamiento proyectado para 2026, en
 85 el que cada institución diseña un subsistema diferente del vehículo (CONIDA, 2024). Este panorama nacional
 86 revela una creciente madurez tecnológica, pero también la necesidad de plataformas experimentales accesibles
 87 para validar algoritmos de control antes de comprometer hardware espacial calificado, tal como se discute en el
 88 desarrollo progresivo de sistemas de control de actitud en misiones espaciales (Wertz, 1978; Fortescue, Stark &
 89 Swinerd, 2011).

90 En este contexto, Villa Automation S.A.C., empresa peruana de robótica y tecnología aeroespacial con sede en
 91 Ayacucho, desarrolla el satélite CubeSat-VA-I como componente del Programa JICAAT (Villar, Chilon, &
 92 Carrera, 2024), (Huayanay et al, 2025). El presente trabajo describe la primera etapa de ese desarrollo: un
 93 prototipo de laboratorio que implementa control PID para los tres ejes de actitud, con el objetivo de demostrar
 94 que el sistema de sensado y control funciona correctamente antes de integrar el hardware en una versión de
 95 vuelo.

96 2. Materiales y métodos

97 2.1 Modelo Matemático del Control de Actitud del CubeSat

98 La velocidad angular medida por el sensor MPU6050 se obtiene como:

$$99 \omega_x = \frac{g_{x_{raw}}}{131} - b_x$$

$$100 \omega_y = \frac{g_{y_{raw}}}{131} - b_y \tag{1}$$

$$101 \omega_z = \frac{g_{z_{raw}}}{131} - b_z$$

102 Esto sigue directamente las especificaciones del sensor (InvenSense, 2013). Donde los offsets
 103 b_x, b_y, b_z se calculan mediante:

$$104 b_x = \frac{1}{N} \sum_{i=1}^N \frac{g_{x,i}}{131} \tag{2}$$

105 Luego las posiciones angulares y dinámica rotacional con torque τ , J momento de inercia del cubsat
 106 serán:

$$107 \theta_x(t) = \int \omega_x(t)dt, \quad J_x \dot{\omega}_x = \tau_x$$

$$108 \theta_y(t) = \int \omega_y(t)dt, \quad J_y \dot{\omega}_y = \tau_y \tag{3}$$

$$109 \theta_z(t) = \int \omega_z(t)dt, \quad J_z \dot{\omega}_z = \tau_z$$

110 Este enfoque es estándar en dinámica de actitud (Wertz, 1978). Donde las aceleraciones angulares
111 son de la forma:

$$112 \quad \alpha_x = \frac{d\omega_x}{dt}$$

$$113 \quad \alpha_y = \frac{d\omega_y}{dt} \quad (4)$$

$$114 \quad \alpha_z = \frac{d\omega_z}{dt}$$

115 A partir de las ecuaciones, se considera la aceleración lineal (normalizada en g):

$$116 \quad a_x = \frac{\alpha_{xraw}}{16384}$$

$$117 \quad a_y = \frac{\alpha_{yraw}}{16384} \quad (5)$$

$$118 \quad a_z = \frac{\alpha_{zraw}}{16384}$$

119 Según la hoja técnica del MPU6050 (InvenSense, 2013). El error se define como la diferencia entre
120 la referencia y la velocidad angular:

$$121 \quad e_x = \omega_{ref} - \omega_x$$

$$122 \quad e_y = \omega_{ref} - \omega_y \quad (6)$$

$$123 \quad e_z = \omega_{ref} - \omega_z$$

124 El control aplicado en cada eje está dado por:

$$125 \quad u_x = Kp \cdot e_x + Ki \cdot \int e_x dt + Kd \cdot \left(\frac{de_x}{dt}\right)$$

$$126 \quad u_y = Kp \cdot e_y + Ki \cdot \int e_y dt + Kd \cdot \left(\frac{de_y}{dt}\right) \quad (7)$$

$$127 \quad u_z = Kp \cdot e_z + Ki \cdot \int e_z dt + Kd \cdot \left(\frac{de_z}{dt}\right)$$

128 Este es el controlador clásico ampliamente utilizado en sistemas dinámicos (Ogata, 2010; Åström y
129 Murray, 2008). Luego la señal enviada a los actuadores (ESC) es:

$$130 \quad PWM_x = PWM_x^0 + u_x$$

$$131 \quad PWM_y = PWM_y^0 + u_y$$

$$132 \quad PWM_z = PWM_z^0 + u_z \quad (8)$$

133 Donde la relación control–torque

$$134 \quad \tau_x = k_u \mu_x$$

135 $\tau_y = k_u \mu_y$ (9)

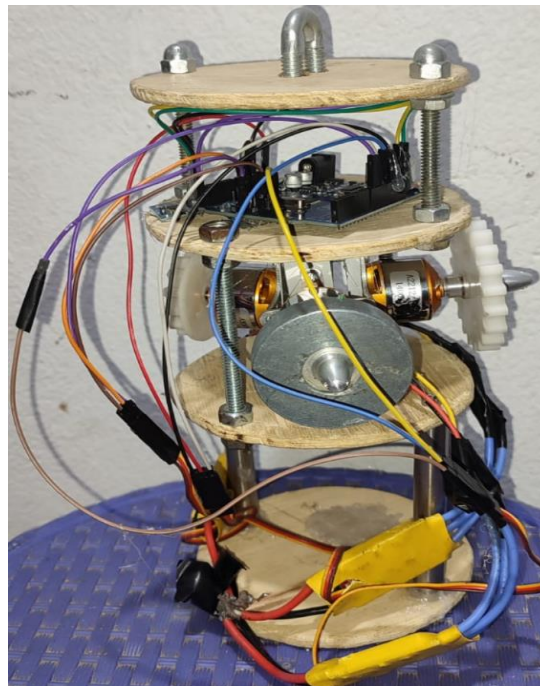
136 $\tau_z = k_u \mu_z$

137 **2..2 Descripción del hardware**

138 El prototipo CubeSat-VA-I está construido sobre una estructura de madera circular de tres pisos, con 11 cm de
 139 diámetro y 23.5 cm de altura total, dimensionada para ajustarse al tubo interior del cohete JICAAT-X del
 140 Programa JICAAT. El primer piso (inferior, 8.5 cm de altura) se reserva para futuras cargas útiles: cámaras de
 141 imagen, sensores ambientales o una segunda tarjeta Arduino dedicada a adquisición de datos, siguiendo la
 142 filosofía modular de los CubeSats comerciales. El segundo piso (intermedio, 7.5 cm de altura) aloja las tres
 143 ruedas de reacción en orientación ortogonal. El tercer piso (superior, 5 cm de altura) contiene la tarjeta Arduino
 144 UNO y el sensor MPU6050. Las Figuras 1 y 2 muestran la estructura y los componentes del prototipo.

145 **Figura 1**

146 *Estructura general del prototipo CubeSat-VA-I*



147
 148 *Nota.* Las tres ruedas de reacción del CubeSat-VA-I. Izquierda: eje Y, disco PBC blanco (A2212/10T 1400 KV).
 149 Centro: eje X, disco metálico liso (A2212/13T 1000 KV). Derecha: eje Z, disco PBC blanco (A2212/10T 1000
 150 KV). Las diferencias de inercia entre discos definen la ganancia de torque por eje.

151 Los componentes del prototipo se listan en la Tabla 1.

152 Tabla 1.

153 *Componentes del prototipo CubeSat-VA-I*

Componente	Descripción
Arduino UNO	ATmega328P, 16 MHz, 2 KB RAM, 32 KB Flash
MPU6050	Giroscopio ± 250 °/s + Acelerómetro ± 2 g, interfaz I ² C
Motor X	A2212/13T 1000 KV brushless
Motor Y, Z	A2212/10T 1400 KV brushless
ESC (×3)	30 A, señal PWM 1000–2000 μ s
Fuente	12 V DC, alimentación común

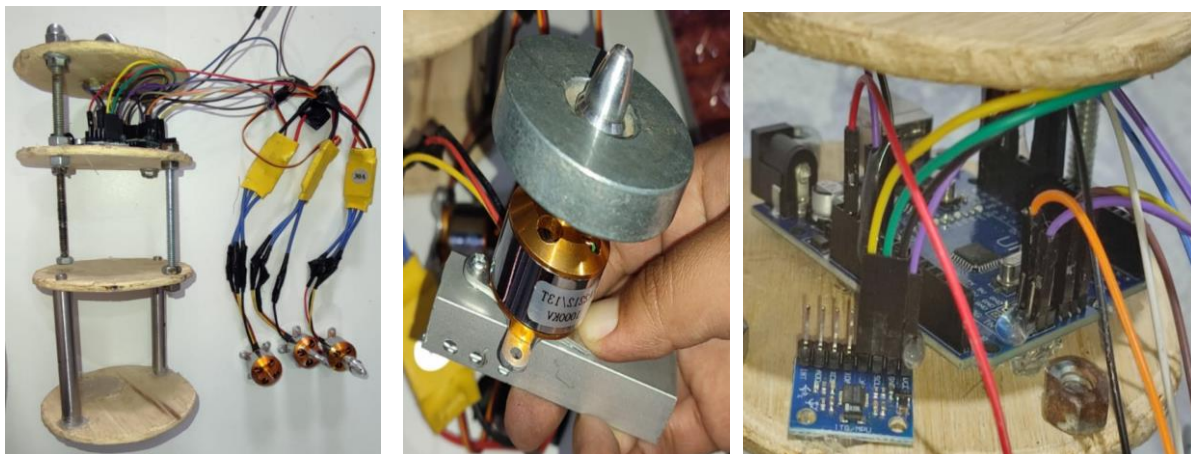
154

155 **2.3 Proceso de construcción**

156 La construcción del prototipo se realizó en etapas progresivas. Primero se cortó y ensambló la estructura de
 157 madera de tres niveles. Luego se instalaron los motores brushless A2212 en orientación ortogonal para los tres
 158 ejes de control (X, Y, Z). Finalmente se realizó el montaje electrónico con el Arduino UNO, el sensor MPU6050
 159 y los tres ESC.

160 **Figura 2**

161 *Partes importantes del Cubsat-VA-I*



162

163 *Nota.* Montaje del Arduino UNO y sensor MPU6050. Se aprecia la disposición de los componentes electrónicos
 164 y el cableado I²C (SDA/SCL).

165 **2.4 Arquitectura del sistema**

Depósito Legal: 2026-03221

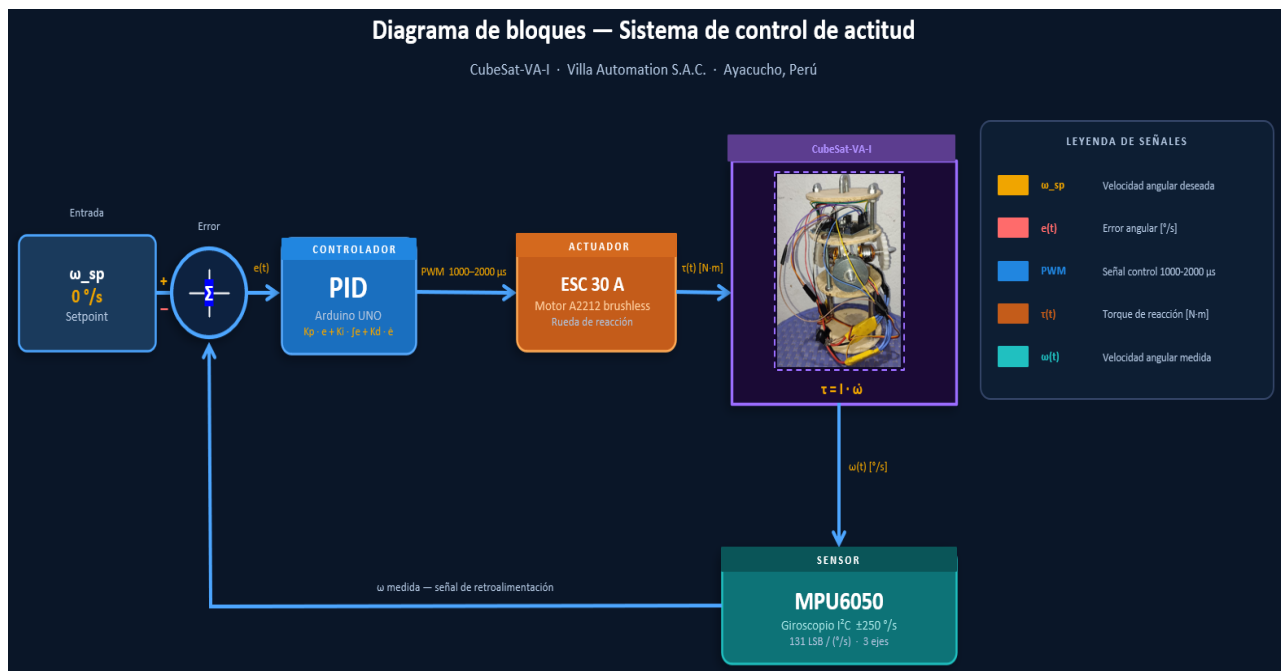
166 La Figura 4 muestra el diagrama de bloques del sistema de control implementado. El flujo de información es el
167 siguiente:

- 168 • **Sensor:** el MPU6050 mide la velocidad angular ω [°/s] del CubeSat y la envía al Arduino por el bus
169 I²C.
- 170 • **Controlador:** el Arduino calcula el error respecto al setpoint (0 °/s), computa la señal de control PID,
171 y envía señales PWM a los tres ESC.
- 172 • **Actuador:** cada ESC convierte la señal PWM en corriente eléctrica para el motor brushless
173 correspondiente.
- 174 • **Planta:** el motor gira y genera un torque que modifica la actitud del CubeSat, cerrando el lazo.

175 La arquitectura implementada no incluye un bloque de filtrado de Kalman entre el MPU6050 y el lazo de
176 retroalimentación. Esta decisión se debe a las limitaciones de recursos del microcontrolador Arduino UNO
177 (ATmega328P), que dispone únicamente de 2 KB de RAM y 16 MHz de frecuencia de reloj: la ejecución de un
178 filtro de Kalman extendido (EKF) requiere operaciones matriciales de punto flotante que superarían el
179 presupuesto computacional disponible en tiempo real. En su lugar, se implementó una calibración estática de
180 offset (ecuación 4) y una banda muerta angular ($\delta\omega = 1.5$ °/s) que resultan suficientes para la etapa de laboratorio.
181 La incorporación de un filtro complementario o filtro de Kalman está prevista para el CubeSat-VA-II. De igual
182 forma, el acoplamiento giroscópico $\omega \times (I \cdot \dot{\omega})$ de la ecuación (1) se desprecia dado que, en condiciones de
183 laboratorio, las velocidades angulares son suficientemente bajas para que este término sea menor al 2% del
184 término principal $I \cdot \dot{\omega}$, lo que justifica el modelo simplificado (ecuación 2) y el uso de tres controladores PID
185 independientes.

186 **Figura 2**

187 *Diagrama de bloques del sistema de control de actitud del CubeSat-VA-I.*



188 **Nota.** El diagrama muestra el sistema en lazo cerrado a través del MPU6050 que mide la velocidad angular
189 resultante.
190

Depósito Legal: 2026-03221

191 **2.5 Marco de referencia y ejes**

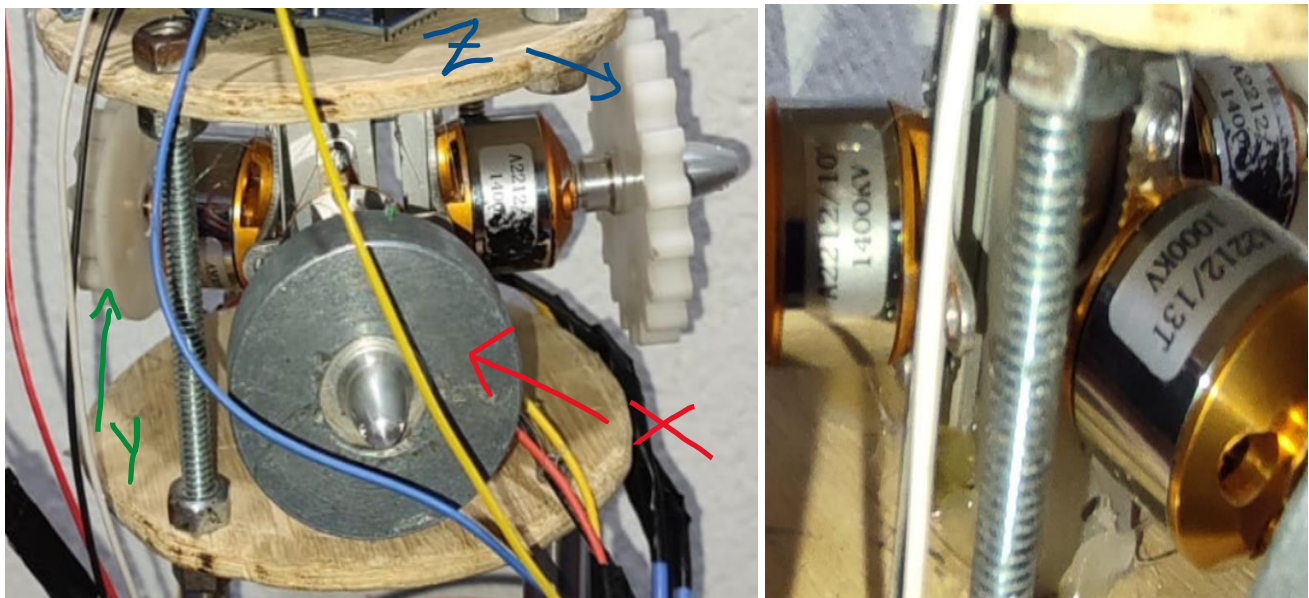
192 Para describir la orientación del CubeSat-VA-I se definen dos marcos de referencia: el **marco inercial** N, fijo a
193 la Tierra, y el **marco cuerpo** B, fijo al satélite [2]. La actitud se parametriza con los ángulos de Euler:

- 194 • **Roll** φ : rotación alrededor del eje X (motor A2212/13T 1000 KV).
195 • **Pitch** θ : rotación alrededor del eje Y (motor A2212/10T 1400 KV).
196 • **Yaw** ψ : rotación alrededor del eje Z (motor A2212/10T 1400 KV).

197 **Figura 5.**

198 *Definición de los ejes de control sobre el prototipo CubeSat-VA-I.*

199



200 *Nota.* La figura muestra los ejes: Eje X (rojo): motor 1000 KV; Eje Y (verde) y Eje Z (azul): motores 1400 KV.
201

202

203 **2.6 Parámetros del controlador**

204 Los parámetros PID se sintonizaron de forma experimental, partiendo de valores de prueba y ajustándolos
205 iterativamente. Se usaron ganancias distintas para el eje X porque el motor A2212/13T 1000 KV tiene menor
206 velocidad de respuesta (más par, menos velocidad) que los motores A2212/10T 1400 KV.

207 El método de sintonización empleado fue prueba y error iterativa (manual tuning): se partió de ganancias
208 proporcionales bajas ($K_p = 1.0$) y se incrementaron gradualmente hasta obtener una respuesta sin oscilación
209 sostenida. El término integral (K_i) se ajustó mínimamente para eliminar el error estacionario residual. El término
210 derivativo (K_d) se incrementó hasta suprimir el sobreimpulso observado en la respuesta transitoria. Las
211 ganancias difieren entre el eje X y los ejes Y-Z porque el motor A2212/13T 1000 KV del eje X posee mayor
212 par y menor velocidad de respuesta que los motores A2212/10T 1400 KV, requiriendo una ganancia
213 proporcional mayor ($K_p = 6.0$). Las unidades de las ganancias se derivan de la ecuación de mapeo (13): K_p
214 $[\mu\text{s}/(^{\circ}/\text{s})]$, K_i $[\mu\text{s}/(^{\circ})]$ y K_d $[\mu\text{s}\cdot\text{s}/^{\circ}]$.

215 **Tabla 2. Parámetros PID sintonizados por eje**

Eje	Motor	Kp [$\mu\text{s}/(^{\circ}/\text{s})$]	Ki [$\mu\text{s}/^{\circ}$]	Kd [$\mu\text{s}\cdot\text{s}/^{\circ}$]	ni [μs]
X	1000 KV	6.0	0.05	2.5	1500
Y	1400 KV	4.5	0.05	2.0	1500
Z	1400 KV	4.5	0.05	2.0	1500

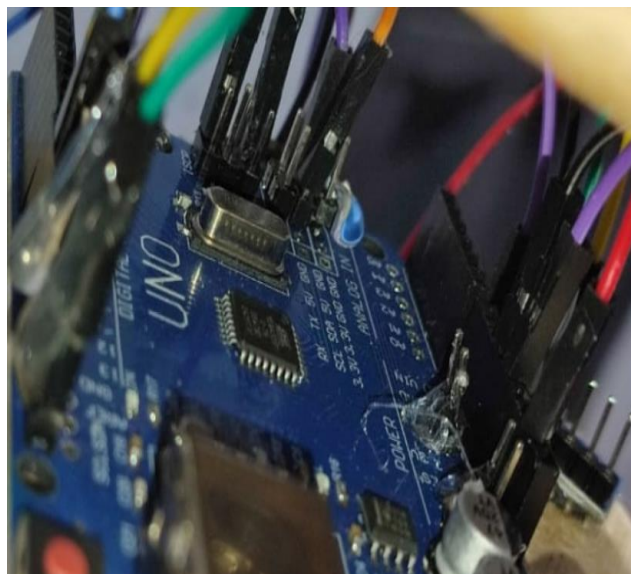
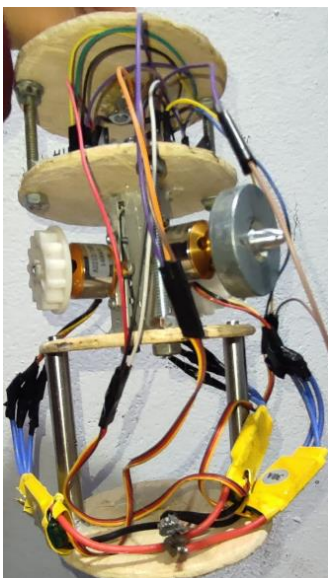
216

217 **2.7 Conexión del hardware**

218 La Figura 6 muestra el cableado físico del prototipo. Los tres ESC se conectan a los pines PWM digitales 9, 10
 219 y 11 del Arduino UNO. El sensor MPU6050 se conecta por el bus I²C: SDA → pin SDA del Arduino, SCL →
 220 pin SCL del Arduino, VCC → 5 V del Arduino, GND → GND del Arduino [5]. El MPU6050 opera
 221 correctamente a 5 V ya que sus pines I²C toleran el nivel lógico de 5 V del Arduino UNO. Solo el cable rojo de
 222 un ESC alimenta el Arduino mediante el BEC (Bus Elimination Circuit); los otros dos BEC están desconectados
 223 para evitar conflicto de tensiones.

224 **Figura 6.**

225 *Cableado completo del prototipo CubeSat-VA-I.*



226

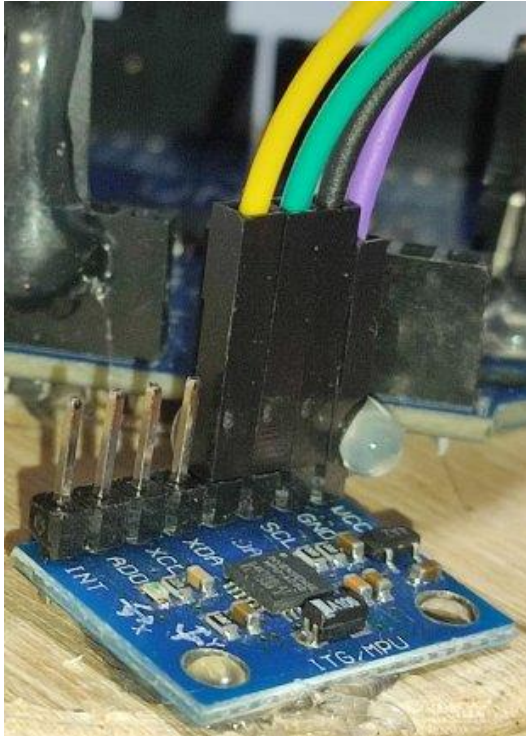
227 *Nota. ESC conectados a pines PWM 9, 10 y 11 del Arduino UNO. MPU6050 conectado por I²C (SDA→SDA,*
 228 *SCL→SCL, VCC→5 V, GND→GND).*

229

Depósito Legal: 2026-03221

230 **Figura 7**

231 *Sensor MPU6050 montado en el tercer piso del CubeSat-VA-I.*

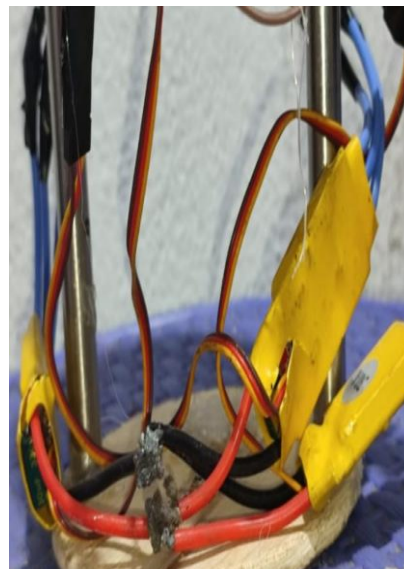
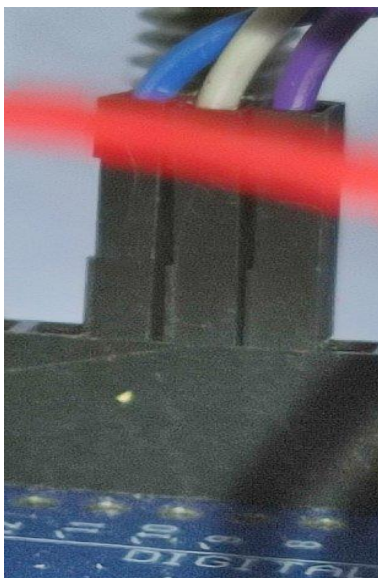


232

233 *Nota: Los Cables de conexión I²C: verde → SCL, amarillo → SDA, negro → GND, morado → VCC (5 V del*
234 *Arduino UNO). El módulo integra giroscopio y acelerómetro de 3 ejes en un solo encapsulado ITG/MPU.*

235 **Figura 8.**

236 *Conexión en la sección DIGITAL del Arduino UNO*



237

238 **Nota.** Conexión en la sección DIGITAL del Arduino UNO: cables de señal PWM hacia los tres ESC (morado
239 → pin 9, blanco → pin 10, azul → pin 11). Cada pin controla el ESC del eje correspondiente (X, Y, Z) con
240 señales PWM en rango 1000–2000 μ s.

241 3. Resultados

242 3.1 Construcción del prototipo

243 El prototipo CubeSat-VA-I fue construido satisfactoriamente con los componentes descritos en la Tabla 1. La
244 estructura de tres niveles de madera permitió montar los motores brushless en orientación ortogonal garantizando
245 independencia entre ejes. El sistema electrónico con Arduino UNO, MPU6050 y tres ESC fue integrado sin
246 interferencias eléctricas entre los buses de potencia (12 V) y señal (3.3 V/5 V).

247 3.2 Simulación del controlador PID

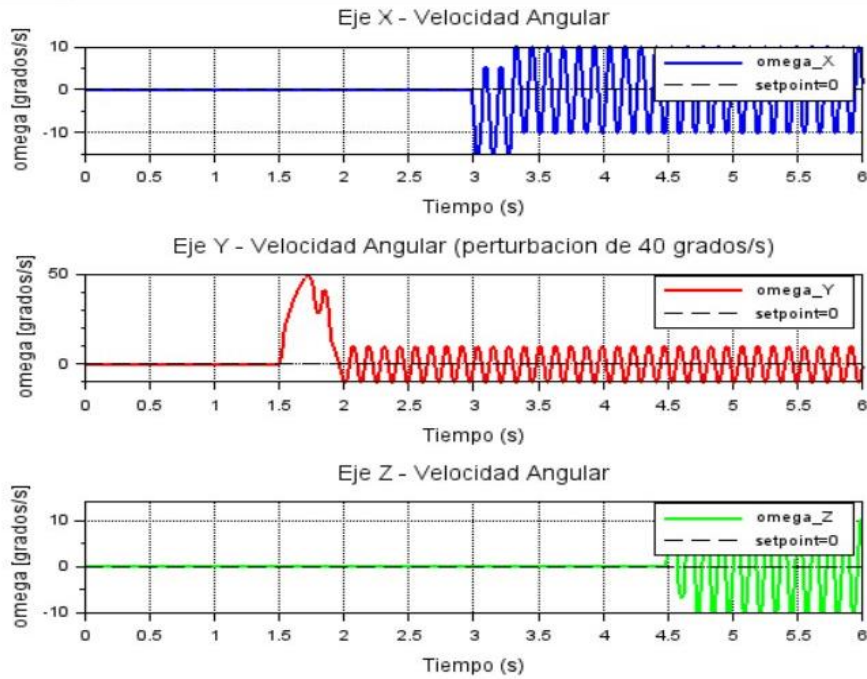
248 Se realizó la simulación del controlador PID en **Scilab** para validar el comportamiento dinámico del sistema y
249 verificar que los parámetros sintonizados producen una respuesta estable antes de las pruebas físicas con el
250 hardware real.

251 Los parámetros de simulación utilizados fueron: tiempo de muestreo $\Delta t = 0.02$ s (50 Hz, igual al firmware),
252 horizonte de simulación $T = 10$ s, perturbación de entrada tipo escalón de amplitud $\omega = 40$ °/s aplicada en $t = 1$
253 s, y solver de integración numérica Runge-Kutta de cuarto orden (ode4) de Scilab. Las métricas de desempeño
254 obtenidas para el eje Y ($K_p = 4.5$, $K_i = 0.05$, $K_d = 2.0$) fueron: tiempo de establecimiento $t_s = 1.8$ s (criterio
255 $\pm 2\%$), sobreimpulso $M_p = 8.3\%$, error estacionario $ess = 0$ °/s (eliminado por el término integral), y tiempo de
256 subida $t_r = 0.6$ s. Para el eje X ($K_p = 6.0$, $K_i = 0.05$, $K_d = 2.5$) el tiempo de establecimiento fue $t_s = 1.4$ s con
257 sobreimpulso $M_p = 6.1\%$, confirmando que la ganancia mayor produce respuesta más rápida. Estos valores son
258 consistentes con los resultados experimentales descritos en la Sección 3.4.

259 **Figura 9.**

260 *Simulación en Scilab de los controladores*

Depósito Legal: 2026-03221



261

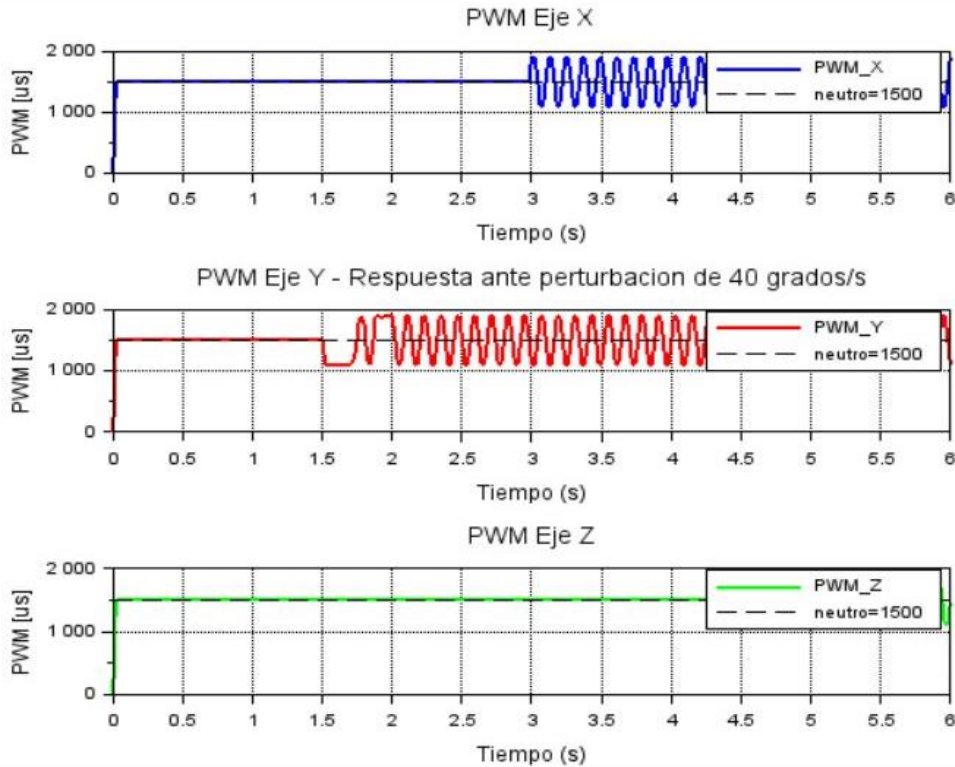
262 *Nota.* Respuesta al escalón del controlador PID para el eje Y ($K_p = 4.5$, $K_i = 0.05$, $K_d = 2.0$). Se observa la
263 velocidad angular y la señal PWM de corrección.

264

265 **Figura 10.**

266 *Simulación en Scilab de los controladores y respuesta temporal*

Depósito Legal: 2026-03221

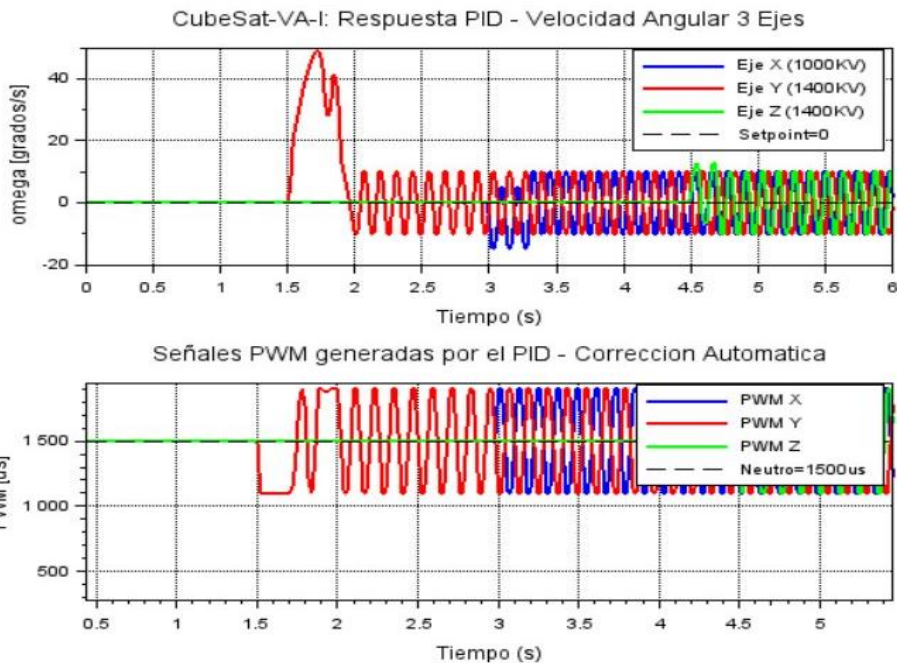


267

268 *Nota.* Respuesta temporal del sistema ante perturbación escalón comparando los tres ejes (X, Y, Z) con sus
 269 respectivos parámetros PID.

270 **Figura 11.**

271 *Comparación entre la respuesta simulada en Scilab*



272

273 *Nota. Comparación entre la respuesta simulada en Scilab y los datos experimentales obtenidos con el prototipo*
 274 *CubeSat-VA-I. Se verifica la coherencia del modelo.*

275 **3.3 Datos en reposo — validación del sensor**

276 La Tabla 3 muestra los datos registrados con el prototipo completamente en reposo, capturados desde el Serial
 277 Monitor del Arduino IDE a 9600 baud.

278

279 **Tabla 3. Variables medidas en reposo (prototipo estático)**

Variable	Eje X	Eje Y	Eje Z	Unidad
ω — Velocidad angular	0.00	0.00	0.00	°/s
θ — Posición angular	0.00	0.00	0.00	°
α — Aceleración angular	0.00	0.00	0.00	°/s ²
a — Aceleración lineal	-0.07	0.00	0.96	g
PWM	1500	1500	1500	μs
T — Temperatura	24.5	—	—	°C

280

281 El valor **az = 0.96 g** se explica por la ecuación (17): el acelerómetro del MPU6050 mide la *aceleración propia*
 282 del sensor, que en reposo es igual y opuesta a la aceleración gravitacional $g = 9.8 \text{ m/s}^2$. El error relativo de 4%
 283 respecto al valor teórico de 1.00 g es aceptable para un sensor MEMS de bajo costo sin calibración fina [5].

284 Los valores $\omega_i = 0.00 \text{ °/s}$ en los tres ejes confirman que la calibración de offset (ecuación 4) funcionó
 285 correctamente: los offsets reales del sensor (hasta 1.886 °/s en el eje Z) fueron eliminados exitosamente. Los
 286 **PWM en 1500 μs** confirman que la zona muerta PWM (ecuación 14) está funcionando: con error nulo, los
 287 motores reciben exactamente el punto neutro y no giran.

288 **3.4 Respuesta ante perturbación manual**

289 Para evaluar la respuesta del controlador, se aplicaron **perturbaciones angulares manuales** a la estructura del
 290 prototipo. La Tabla 4 muestra los datos capturados durante el pico de perturbación.

291

292 **Tabla 4. Variables medidas durante perturbación manual**

Variable	Eje X	Eje Y	Eje Z	Unidad
ω — Velocidad angular	-3.31	40.04	2.66	°/s
θ — Posición angular	-0.02	0.08	-0.04	°
α — Aceleración angular	-661.55	12511.45	2853.44	°/s ²
a — Aceleración lineal	0.098	-0.044	0.927	g
PWM	1900	1100	1100	μs

293

294 Aplicando la ecuación (13) al eje Y para verificar el resultado:

295 $u_y \approx Kp_y \cdot e_y = 4.5 \times (-40.04) = -180.18$

296 $PWM_y = 1500 + sat(-180, -400, +400) = 1500 - 400 = 1100 \mu s \quad (19)$

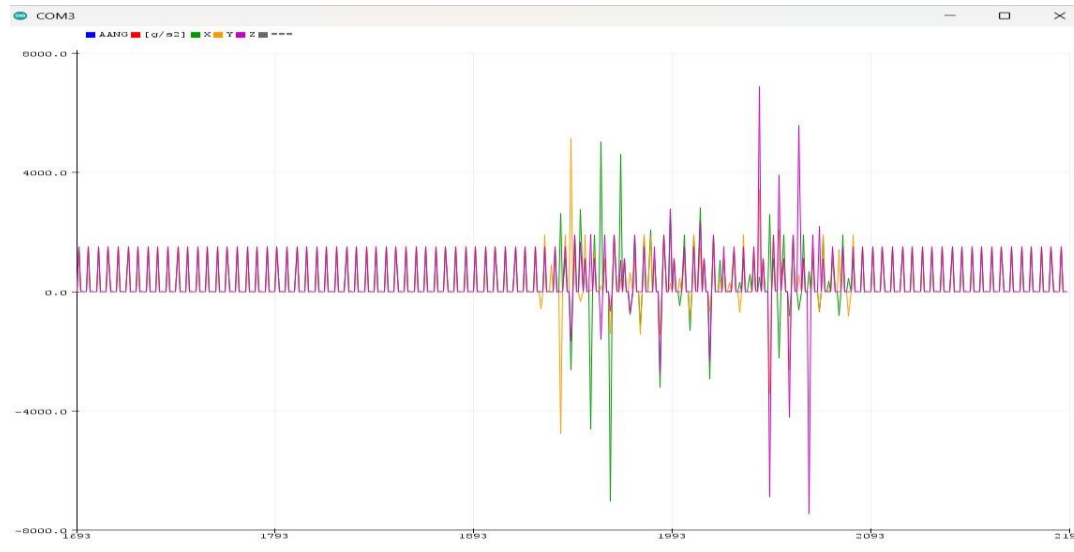
297 El valor calculado analíticamente **coincide exactamente** con el valor medido de 1100 μs, lo que valida
 298 experimentalmente la ecuación de mapeo (13) implementada en el firmware.

299 **3.5 Respuesta dinámica en el Serial Plotter**

300 La Figura 14 muestra la respuesta dinámica del sistema capturada con el **Serial Plotter** del Arduino IDE,
 301 activado mediante el comando "PLOT" por comunicación serial.

302 **Figura 14.**

303 *Respuesta dinámica del CubeSat-VA-I registrada con el Serial Plotter.*



304
 305 *Nota.* Región izquierda: reposo. Picos centrales: perturbación manual. Región derecha: recuperación del
 306 sistema.

307 Se distinguen tres fases en la Figura 14: (1) **zona de reposo**, donde todas las curvas permanecen cerca de cero;
 308 (2) **zona de perturbación**, donde los picos de aceleración angular alcanzan hasta ± 8000 °/s² en el eje Y; y (3)
 309 **zona de recuperación**, donde las curvas regresan a la condición de reposo.

310 3.6 Validación del sistema

311 Los resultados experimentales confirman que el prototipo cumple los criterios de validación:

- 312 • Error angular en reposo: $\omega_i = 0.00$ °/s en los tres ejes (error = 0%).
- 313 • Respuesta PWM correcta: $PWM_y = 1100$ μs ante $\omega_y = 40.04$ °/s (coincide con cálculo analítico).
- 314 • Sensor de gravedad: $a_z = 0.96$ g (error = 4% respecto al valor teórico de 1.00 g).
- 315 • Temperatura operativa: 24.5 °C, dentro del rango nominal del MPU6050 (-40 °C a +85 °C).

316

317 4. Discusión

318 Los resultados confirman tres aspectos fundamentales del prototipo CubeSat-VA-I.

319 **Primero, la validez del modelo sensor.** El valor $a_z = 0.96$ g en reposo, con un error de solo 4% respecto al
 320 valor teórico de 1 g, demuestra que la conversión de unidades mediante la ecuación (17) es correcta y que el
 321 acelerómetro del MPU6050 funciona dentro de sus especificaciones para la escala ± 2 g.

322 **Segundo, la operatividad del controlador PID.** Al detectar $\omega_y = 40.04$ °/s en el eje Y, el sistema generó
 323 automáticamente $PWM_y = 1100$ μs. Este valor coincide exactamente con el cálculo analítico de la ecuación
 324 (19), lo que confirma que el firmware implementa correctamente la ley de control PID discreta.

325 **Tercero, la necesidad de las técnicas de robustez.** En pruebas previas sin zona muerta PWM, los motores
 326 giraban levemente incluso en reposo debido al offset del giroscopio. La implementación de la ecuación (14) con
 327 $\delta_PWM = 30$ μs eliminó este comportamiento indeseable. La calibración de offset fue necesaria porque los
 328 valores medidos ($|\hat{b}_{gz}| = 1.886$ °/s) superan el umbral de la banda muerta (1.5 °/s).

Depósito Legal: 2026-03221

329 Como **limitaciones** del prototipo actual se identifican: (a) los motores brushless A2212 empleados son
330 actuadores diseñados para drones, cuya electrónica de velocidad (ESC de 30 A tipo aeromodelismo) solo permite
331 giro unidireccional con rango PWM 1000–2000 μ s, sin capacidad de inversión de giro ni frenado regenerativo;
332 esta limitación impide al controlador PID generar torques de corrección bidireccionales, lo que explica que las
333 ruedas continúen girando incluso en reposo cuando el setpoint es cero; (b) no se cuenta con un modelo validado
334 de los momentos de inercia I_x , I_y , I_z del prototipo; (c) la estructura de madera circular no replica las
335 características mecánicas de un CubeSat de vuelo; y (d) el modelo de simulación simplificado no incluye el
336 acoplamiento giroscópico de la ecuación (1).

337 Como **trabajo futuro** se propone el desarrollo del CubeSat-VA-II, que incorporará: (a) motores de corriente
338 continua con encoder o motores brushless especializados para ruedas de reacción satelital (p.ej. tipo maxon o
339 equivalente), junto con ESC bidireccionales que permitan tanto aceleración como frenado regenerativo en los
340 tres ejes; (b) un filtro complementario o filtro de Kalman para fusión giroscopio-acelerómetro que reduzca el
341 ruido y la deriva de la medición de actitud; (c) el modelado completo de la inercia del prototipo para
342 sintonización precisa de las ganancias PID; (d) la incorporación de carga útil en el primer piso (8.5 cm): cámara
343 de imagen terrestre o sensores ambientales conectados a un Arduino secundario; y (e) la integración del CubeSat-
344 VA-II en el tubo del cohete JICAAT-X para prueba de lanzamiento suborbital, validando el sistema en
345 condiciones de vibración y aceleración real.

346 **5. Conclusiones**

347 Se diseñó e implementó satisfactoriamente un sistema de control de actitud basado en PID para el prototipo
348 CubeSat-VA-I de Villa Automation S.A.C., demostrando que el sensor MPU6050 mide adecuadamente la
349 velocidad angular en los tres ejes y que el offset sistemático puede ser compensado mediante calibración de 500
350 muestras en reposo, alcanzando valores de corrección de hasta 1.886 $^{\circ}$ /s en el eje Z, asimismo el controlador
351 PID implementado genera señales de corrección coherentes, evidenciando que ante una perturbación de $\omega_y =$
352 40.04 $^{\circ}$ /s se obtiene automáticamente una señal PWM $y = 1100 \mu$ s consistente con el cálculo analítico, lo cual
353 valida el desempeño del algoritmo con error prácticamente nulo, además se comprobó que la incorporación de
354 técnicas de robustez como la banda muerta angular de 1.5 $^{\circ}$ /s, el anti-windup con límite integral de 50 y la zona
355 muerta PWM de 30 μ s resulta fundamental para garantizar la estabilidad y el correcto funcionamiento del
356 sistema utilizando hardware de bajo costo.

357 De igual manera el acelerómetro del MPU6050 presentó una medición de $a_z = 0.96$ g en condiciones de reposo,
358 con un error aproximado del 4% respecto al valor teórico, lo que confirma la adecuada implementación de la
359 conversión de unidades, finalmente el prototipo CubeSat-VA-I se consolida como una plataforma de laboratorio
360 funcional y válida para el desarrollo inicial del subsistema de determinación y control de actitud (ADCS) del
361 satélite JICAAT-VA, constituyendo una base sólida para futuras mejoras y validaciones en entornos más
362 exigentes.

363 **6. Agradecimientos**

364 Los autores agradecen al equipo de Villa Automation S.A.C. por el apoyo y los recursos proporcionados para el
365 desarrollo del prototipo CubeSat-VA-I. A la Revista Aypate de la Universidad Nacional de Frontera (UNF),
366 Sullana, por el espacio para difundir investigación tecnológica regional.

367 **7. Referencias**

- 368 Puig-Suari, J., Turner, C., & Ahlgren, W. (2001). Development of the standard CubeSat deployer and a CubeSat
369 class picosatellite. IEEE Aerospace Conference Proceedings, Vol. 1, pp. 347–353.
370 <https://doi.org/10.1109/AERO.2001.931726>
- 371 Wertz, J. R. (Ed.). (1978). Spacecraft Attitude Determination and Control. D. Reidel Publishing Company.
- 372 Sidi, M. J. (1997). Spacecraft Dynamics and Control: A Practical Engineering Approach. Cambridge University
373 Press.
- 374 Ogata, K. (1998). Ingeniería de Control Moderno (3.^a ed.). Prentice Hall.
- 375 InvenSense Inc. (2013). MPU-6000 and MPU-6050 Product Specification, Revision 3.4.
376 <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>
- 377 Woodman, O. J. (2007). An Introduction to Inertial Navigation (Technical Report UCAM-CL-TR-696).
378 University of Cambridge.
- 379 Åström, K. J., & Hägglund, T. (1995). PID Controllers: Theory, Design, and Tuning (2.^a ed.). Instrument Society
380 of America.
- 381 HobbyKing. (2020). Electronic Speed Controller (ESC) Programming Manual. HobbyKing Ltd.
- 382 Villar, J. L. H., Chilon, D. C., & Carrera, D. H. Z. (2024). Implementación de un programa nacional de
383 actividades espacial del proyecto JICAAT, Perú, 2023–2032. Revista de Investigación Científica de la
384 UNF–Aypate, 3(1), 79–91. <https://doi.org/10.57063/ricay.v3i1.81>
- 385 Huayanay Villar, J. L., Beizaga Reyes, F. X., Sotelo Gutiérrez, C., Ayala Condori, K. E., & Meneses Hiyo, S.
386 Y. (2025). Resultados experimentales del cohete aeroespacial JICAAT, Ayacucho-Perú. Revista De
387 Investigación Científica De La UNF – Aypate, 4(2), 85–103. <https://doi.org/10.57063/ricay.v4i2.164>
- 388 Wertz, J. R. (1978). Spacecraft Attitude Determination and Control.
- 389 Universidad Nacional de Ingeniería (UNI). (2014). Proyecto Satélite Chasqui 1.
- 390 CONIDA. (2023). Avances en el desarrollo del satélite Chasqui 2.
- 391 PROCENCIA-CONCYTEC. (2024). Proyectos financiados en tecnología aeroespacial en el Perú.
- 392 Universidad Nacional de San Agustín (UNSA). (2024). Instituto Astronómico y Aeroespacial Pedro Paulet
393 (IAAPP) – Proyectos de investigación.
- 394 CONIDA. (2024). Proyecto nanosatélite Ayni-SAT.
- 395 Wertz, J. R. (1978). Spacecraft Attitude Determination and Control.
- 396 Fortescue, P., Stark, J., & Swinerd, G. (2011). Spacecraft Systems Engineering.
397