

A Multiprocessing-Based GPS Solar Tracking System with Dual-Axis Control for Remote Energy-Aware Stations

Mohammed Al shareef¹, M.M. El-Halwany^{2,3}, Islam Ismael⁴, Abdelhady Ghanem⁴, Ramadan Madi Ali Bakir⁵

¹ Mechanical Engineering Department, Lecturer in the Mechanical Engineering Department, Najran College of Technology, 61411, KSA.

² Department of Mathematics and Physics Engineering, Faculty of Engineering, Mansoura University, Elgomhouria St, El-Mansoura 35516, Egypt.

³ Faculty of Engineering, Mansoura National University, Gamsa, Egypt

⁴ Department of Electrical Engineering, Faculty of Engineering, Mansoura University, Elgomhouria St, El-Mansoura 35516, Egypt.

⁵ Semiology Department, National Research Institute of Astronomy and Geophysics (NRIAG), Cairo 4037101, Egypt. email: ahr_bakir@yahoo.com

HOW TO CITE:

Mohammed Al Shareef, M.M.
El-Halwany, Islam Ismael,
Abdelhady Ghanem, Ramadan
Madi Ali Bakir (2026). A
Multiprocessing-Based GPS Solar
Tracking System with Dual-Axis
Control for Remote Energy-Aware
Stations.
International Journal of Special
Education, 41(11s), 43-63

COPYRIGHT STATEMENT:

Copyright: © 2026 Authors.
Open access publication under the
terms and conditions
of the Creative Commons Attribution
(CC BY)
license (<http://creativecommons.org/licenses/by/4.0/>).

ABSTRACT:

This work presents the design, control architecture, and initial performance characterization of a fully autonomous, GPS-based dual-axis photovoltaic tracking system intended for off-grid and remote energy applications. Conventional fixed-tilt PV installations are passive and low-cost but cannot continuously align to the sun, which limits incident irradiance and daily energy yield, especially at low solar elevation. Commercial sun trackers address this limitation but typically rely on optical sensors, require continuous actuation, or assume permanent infrastructure and backend connectivity. The gap we address is the need for a self-contained, low-power, self-orienting tracker that can operate without external supervision, compute sun position from first principles, manage its own energy budget, and report performance data over telemetry. The proposed system combines a Raspberry Pi (high-level computation, telemetry, and supervisory power management) with an Arduino Nano (deterministic sensor acquisition and motor actuation). Using GPS latitude/longitude, day-of-year, and UTC time, the controller computes sun azimuth and elevation through astronomical equations, converts those angles into azimuth/elevation motor commands, and steers two geared DC drives accordingly. The system also measures PV voltage, current, and power in real time and logs these quantities alongside orientation commands. A three-day rooftop deployment (15–17 June 2025, Cairo latitude $\sim 30^\circ\text{N}$) was used to simulate energy capture for a 10 W panel under different tracking update scenarios. For realistic scenarios between 5 and 40 minutes, total daily harvested energy remained approximately 96.2–96.3 Wh, while an extremely slow, quasi-fixed behavior (60-minute cadence) produced about 95.75 Wh/day, indicating only ~ 0.5 – 0.7% loss under favorable summer conditions. These data confirm that high actuation frequency is not always required;

instead, a hybrid strategy with dense updates near sunrise/sunset and slower updates midday can preserve energy yield while reducing mechanical wear and control overhead. The study establishes the functional baseline embedded control, tracking logic, autonomous power management, and telemetry framing and sets the stage for future work on long-duration field validation, adaptive energy management policies, and multi-panel scaling.

Keywords: Dual-axis solar tracker; GPS-based sun tracking; Photovoltaic systems; Embedded control; Autonomous power management; Solar position algorithm; Real-time power monitoring; Telemetry.

1. Introduction

1.1. Motivation

The efficiency of traditional, fixed-position solar panels is inherently limited because they are only optimally aligned with the sun's rays for a brief period each day [1]. To maximize the energy capture from a solar panel, it must continuously adjust its orientation to follow this dynamic path [2]. This project is motivated by the need to develop a robust, intelligent, and cost-effective solar tracking system that overcomes the limitations of fixed mounts. The proposed solar tracker uses a Global Positioning System (GPS) module to precisely determine the geographical location and time. This data serves as the foundation for a highly accurate astronomical algorithm, which calculates the sun's exact position [3], [4]. This approach offers a significant advantage over passive systems that rely on light-dependent resistors (LDRs), which can be unreliable due to cloud cover, shadows, or other environmental factors. By employing a predictive, GPS-based method, our system can anticipate the sun's movement and proactively adjust the solar panel's position, ensuring continuous, optimal alignment and maximizing energy generation [5]. The system's intelligence is further enhanced by integrating an Arduino microcontroller to manage motor control and a Raspberry Pi to act as a data logger and Internet of Things (IoT) gateway [6].

1.2. Literature review

Solar tracking systems are broadly categorized into fixed, single-axis, and dual-axis mechanisms. Fixed systems are stationary and least expensive, but suffer from reduced energy capture due to their inability to adjust to the sun's changing position throughout the day. While these systems require minimal maintenance and have lower initial costs, their static nature significantly limits their energy harvesting potential compared to dynamic tracking alternatives. Research has demonstrated that single-axis trackers can significantly improve energy output compared to fixed systems. Sayeedurrahman et al. [7] proposed a basic LDR and stepper motor-based controller system for single-axis tracking. Despite its simplicity, this system significantly enhanced energy capture and was easy to deploy in rural regions, making it particularly suitable for off-grid applications where maintenance capabilities may be limited. Dual-axis trackers follow both azimuth and altitude angles of the sun, offering the highest energy gains among all tracking configurations. These sophisticated systems can achieve maximum solar irradiance capture by continuously adjusting panel orientation in two dimensions throughout the day. Lu and Ajay [8] designed an arithmetic optimization algorithm-based dual-axis solar tracker that demonstrated superior energy yields over fixed systems. Their system dynamically adjusted the orientation using real-time sensor input and algorithmic correction, optimizing panel positioning throughout the day.

This approach showcased the potential for mathematical optimization techniques to enhance tracking precision and energy capture efficiency. Anshory et al. [9] explored passive dual-axis tracking using thermal intensity control, eliminating electronic components and improving reliability in remote setups. This innovative approach addresses one of the primary concerns with active tracking systems the potential for electronic component failure in harsh environmental conditions. Pardosi et al. [10] presented a low-cost dual-axis system built with Arduino microcontroller technology, achieving a 14% performance improvement compared to basic LDR-based trackers. Precise sun tracking can be achieved through sensor-driven methods utilizing components such as LDRs and photodiodes, or through astronomical algorithms based on sun position calculations. The choice between these approaches often depends on factors such as cost, accuracy requirements, and environmental conditions. A low-cost photodiode-based single-axis tracker was introduced, demonstrating a 20% improvement in output and validating its utility for off-grid applications. Ponce-Jara et al. [11] compared astronomical and sensor-based tracking systems in equatorial Ecuador, providing valuable insights into the performance of different tracking approaches under specific geographical conditions. Their findings indicated that both systems outperform static photovoltaic installations, with the sensor-based system performing slightly better due to its ability to respond dynamically to cloud movement and partial shading conditions. Xu et al. [12] developed a multi-sensor tracking unit that reduced sun-angle error, demonstrating the benefits of sensor fusion approaches in improving tracking accuracy. This work highlights the importance of precision in solar tracking applications and the potential for advanced sensor technologies to enhance system performance. Artificial intelligence, machine learning, and fuzzy logic are being increasingly applied to enhance tracking performance, reduce mechanical wear, and enable predictive operation capabilities. [13] implemented a fuzzy logic

controller to dynamically adjust tracker angles in real time, accommodating for wind and ambient temperature variations. This approach demonstrates the potential for intelligent control systems to adapt to changing environmental conditions and optimize system performance under varying operational scenarios. The same research group used fuzzy logic to manage actuator movement in extreme climates, increasing both energy gain and structural stability [13]. This work addresses critical concerns about system durability and performance in challenging environmental conditions. Ibrahim et al. [12] developed a hybrid maximum power point tracking (MPPT) and tracking algorithm that minimized orientation delay and improved power extraction. This integrated approach demonstrates the potential for combining multiple optimization strategies to achieve superior system performance. [14] introduced edge-AI approaches through an IoT-enabled embedded system capable of on-device inference, reducing communication delays. This work represents an important advancement in the integration of modern computing technologies with solar tracking systems, enabling more responsive and autonomous operation. [15] applied AI prediction models integrated with real-time sensors to adapt panel orientation under complex irradiance patterns. This predictive approach allows systems to anticipate changing conditions and adjust accordingly, potentially improving energy capture efficiency. [16] implemented a Long Short-Term Memory (LSTM)-based predictive tracker using weather forecasts to proactively adjust panel positions, resulting in improved output during cloudy intervals. This work demonstrates the potential for machine learning techniques to enhance system performance by incorporating weather prediction data. [17] employed reinforcement learning in a simulated solar tracker, enabling self-learning orientation control based on real-time environmental data. This approach represents a significant advancement in autonomous system operation, allowing trackers to continuously improve their performance through experience. [18] focused on predictive

maintenance using machine learning algorithms to identify actuator faults before system failure. This proactive approach to maintenance can significantly reduce system downtime and maintenance costs, improving the overall economic viability of solar tracking installations. The adoption of solar tracking systems depends not only on energy performance but also on return on investment, particularly in carbon-conscious systems. Ge et al. [12] examined fixed, single-axis, and dual-axis configurations within an urban microgrid context, incorporating carbon credit schemes. They concluded that although dual-axis systems have higher initial costs, they become financially viable when carbon savings and hydrogen storage are factored into the economic analysis. The same study emphasized that dual-axis systems are cost-effective when combined with carbon trading schemes, highlighting the importance of considering environmental benefits in economic evaluations of solar tracking technologies. Comparative studies have provided valuable insights into the relative performance of different tracking systems. Research has reported up to 57% higher energy efficiency with dual-axis trackers in stable irradiance environments, but noted diminished returns under variable conditions. This finding emphasizes the importance of considering local environmental conditions when selecting appropriate tracking technologies. Sun et al. [19] tackled the problem of actuator longevity, proposing a horizontal tracker that reduced movement frequency by 71.7%, which lowers maintenance costs and extends system life key factors in levelized cost of energy (LCOE) calculations. This work addresses one of the primary concerns with active tracking systems: the potential for increased maintenance requirements due to mechanical complexity.

1.3. Research Gap

A detailed analysis of the recent literature reveals persistent limitations in terms of sensing precision, tracking intelligence, energy efficiency, and system connectivity. (1) Most solar tracking systems rely on LDRs as their primary sensing element due to their low cost and simplicity.

However, LDRs have major drawbacks, including poor performance in low-light conditions (sunrise/sunset), limited angular precision, and vulnerability to environmental factors like dust and shading. (2) The unnecessary power consumption during non-tracking hours, particularly overnight, most tracking controllers remain active throughout the night in anticipation of sunrise. This results in continuous energy drain, particularly problematic in off-grid or battery-limited environments. None of the examined designs incorporated low-power sleep modes, RTC-triggered wake-up routines, or predictive sunrise algorithms. (3) Most existing solar tracking systems operate as standalone units without remote monitoring, data logging, or cloud connectivity. Only two studies integrated basic IoT features using ESP32, limiting real-time performance evaluation and smart grid compatibility.

1.4. Research contribution

In response to the limitations identified in current solar tracking systems, we propose a novel, intelligent solar tracking system that introduces several key contributions. (1) Our system replaces traditional light sensor-based tracking with a GPS-driven approach. By using the GPS, the tracker obtains precise geolocation data (latitude, longitude, and time) to calculate the sun's azimuth and elevation using astronomical algorithms. (2) The system leverages astronomical equations to align the panel based on time and location, ensuring accurate positioning even in low-irradiance conditions. This addresses the gap related to the absence of solar path prediction in earlier designs. (3) We introduce a scheduling mechanism based on a watchdog timer. The GPS module is activated once daily to synchronize time and location data. Outside active hours, the main tracking unit enters a low-power sleep mode, while a lightweight watchdog program remains active to detect sunrise. (4) Our system includes IoT capabilities for real-time monitoring and performance evaluation. This enables remote system diagnostics, data logging, and predictive maintenance, filling the gap in communication

and edge intelligence observed in most standalone systems.

2. Research methodology and materials

2.1. Research methodology

The proposed solar tracking system utilizes geospatial data obtained from a Global Navigation Satellite System (GNSS) through a GPS receiver to optimize the tracking accuracy of photovoltaic (PV) panels. As illustrated in Figure 1, the GPS module provides precise latitude (φ), longitude (λ), and Coordinated Universal Time (UTC), which are fundamental inputs for astronomical computations. These data are processed to determine the sun's geographic position in terms of solar Azimuth (Φ) and Elevation (α) angles. The elevation and azimuth angles are the two essential outputs of the astronomical model that govern the mechanical

motion of the solar tracking system. Together, these parameters allow the dual-axis tracker to continuously align the PV panel surface perpendicular to the incoming solar rays. Using the geographic coordinates and UTC data from the GPS receiver, the system computes elevation and azimuth at regular intervals (the optimum interval selected based on the evaluation). The results are stored in a 24-hour tracking dataset, which is used throughout the day to control the panel's movement without requiring continuous recalculations. This predictive approach ensures that the solar incidence angle remains as close as possible to 90° , maximizing energy absorption efficiency. Moreover, this method significantly reduces computational load and power consumption, enhancing the reliability and responsiveness of the tracker under both daily and seasonal variations.

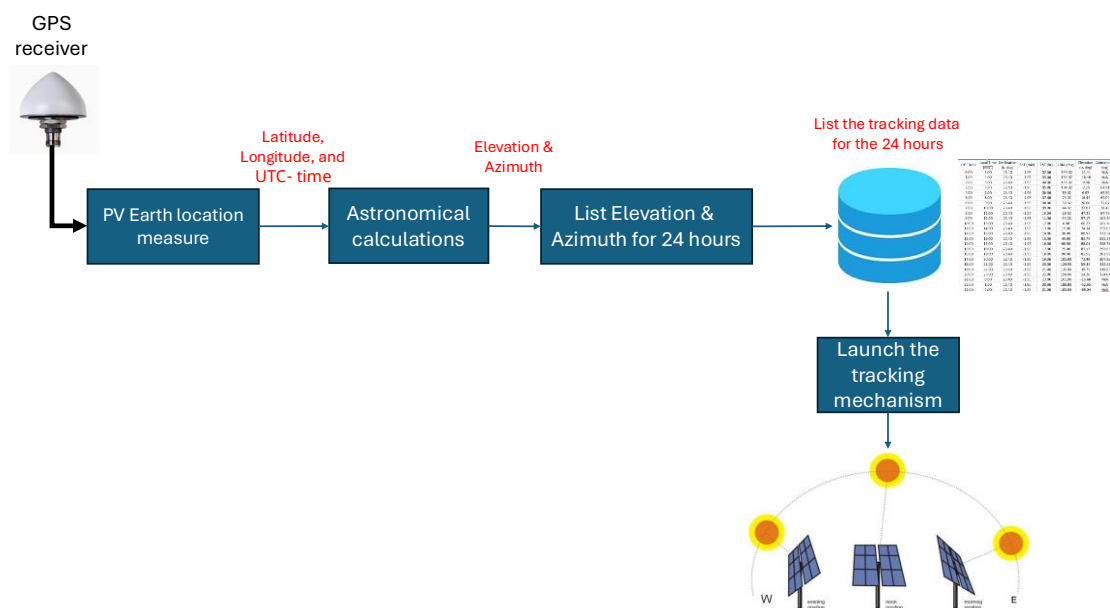


Figure 1. the proposed research methodology

2.2. System modeling

Astronomical Equations

To determine the apparent position of the sun relative to the solar tracking system, a series of fundamental astronomical equations are applied in certain sequence as presented in Figure 2. These equations use inputs from the GPS receiver, including the latitude (φ), longitude (λ), and UTC, in addition to the day of the year (n).

The process begins with calculating the declination angle (δ) and the equation of time (EoT), which represent the angular position of the sun and the correction between solar and clock time, respectively. Using these, the local solar time (LST) and hour angle (HRA) are derived, leading finally to the computation of the elevation (α) and azimuth (Φ) angles that define

the sun's position in the sky. The overall model can be summarized as follows:

declination angle (δ): declination angle represents the angular distance of the sun from the celestial equator and calculated as following.

$$\delta = 23.45 \sin[(360/365) \times (n - 81)] \quad (1)$$

equation of time (EoT): The EoT corrects discrepancies between solar and mean time due to the Earth's orbital eccentricity and axial tilt.

$$\text{EoT} = 9.87 \times \sin(2\beta) - 7.53 \times \cos(\beta) - 1.5 \times \sin(\beta) \quad (2)$$

Where

$$\beta = (360/365) \times (n - 81) \quad (3)$$

Local Solar Time (LST): provides the local solar reference for subsequent time-based calculations.

$$\text{LST} = \text{UTC} + (\varphi/15) + (\text{EoT}/60) \quad (4)$$

Hour angle (HRA): The HRA represents the angular distance between the sun and the local meridian.

$$\text{HRA} = 15 \times (\text{LST} - 12) \quad (5)$$

Elevation Angle (α): The elevation angle indicates the sun's height above the horizon.

$$\sin(\alpha) = \sin(\delta) \times \sin(\varphi) + \cos(\delta) \times \cos(\varphi) \times \cos(\text{HRA}) \quad (6)$$

Azimuth Angle (ϕ): The azimuth angle specifies the sun's compass direction, measured clockwise from true north.

Azimuth =

$$\arccos\left[\frac{\sin(\delta) \times \cos(\varphi) - \cos(\delta) \times \sin(\varphi) \times \cos(\text{HRA})}{\cos(\alpha)}\right] \quad (7)$$

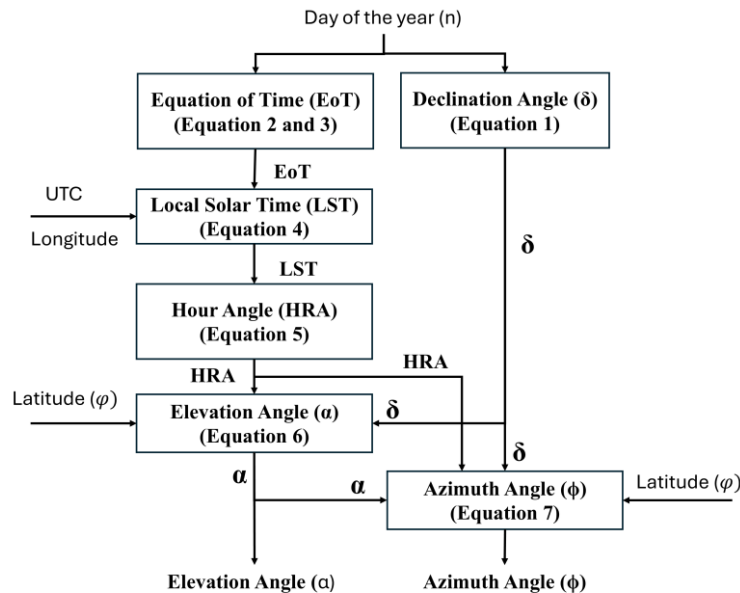


Figure 2. The astronomical equation model used to calculate the solar elevation and azimuth angles based on GPS derived latitude, longitude, and UTC data.

Voltage and current sensor

In the proposed DC measurement front-end, both the PV voltage and current are mapped to the Arduino ADC input through linear conditioning networks. For the voltage path, a resistive divider followed by a unity-gain buffer produces an ADC-level signal proportional to the PV terminal voltage. The resulting total voltage transformation ratio is:

$$K_V = \frac{R_2}{R_1 + R_2} G_V \quad (8)$$

Where R_1, R_2 : voltage divider resistors of the voltage sensor, $G_V \approx 1$: analog gains (op-amp) in voltage paths. For the current path, the PV current is sensed as a voltage drop across a precision shunt R_{sh} and may be buffered or lightly amplified before digitization; the total current transimpedance is:

$$K_I = R_{sh} G_I \quad (9)$$

Where R_{sh} : current shunt resistance, $G_I \approx 1$: analog gains (op-amp) in current paths.

Primary processor ADC

To recover DC voltage and current at the PV terminals from raw ADC outputs, the microcontroller maps integer codes to physical units through the known reference and front-end gains. For an N -bit ADC with reference V_{ref} the ideal code-voltage relation at the ADC pin is $V_{ADC} = \frac{c}{2^N-1} \times V_{ref}$. Combining this with the voltage- and current-path gains (K_V and K_i , Equation 8,9) yields the estimators.

$$V_{PV} = \frac{c_v}{2^N-1} \times \frac{V_{ref}}{K_V} \quad (10)$$

$$I_{PV} = \frac{c_i}{2^N-1} \times \frac{V_{ref}}{K_i} \quad (11)$$

Where $c_v, c_i \in \{0, \dots, 2^N - 1\}$ are the quantized codes measured on the voltage and current channels, respectively. To suppress quantization and electronic noise, each channel is averaged over M consecutive samples prior to conversion, leading to the sample-mean codes.

$$c_v = \frac{1}{M} \sum_{m=1}^M c_v[m] \quad (12)$$

$$c_i = \frac{1}{M} \sum_{m=1}^M c_i[m] \quad (13)$$

Equations (12) and (13) reduce the variance of the estimates in Equations (10) and (11) approximately by a factor $1/M$ under zero-mean noise. Equations (10), (11), and (13) provide a linear, calibration-friendly pathway from raw ADC codes to PV-side voltage, current, and hence power $P_{PV} = I_{PV} \times V_{PV}$, while making the resolution and dynamic-range dependencies on V_{ref}, N, K_V , and K_i explicit.

Tracking mechanism

In each update step, the tracker computes the shortest required motion on both axes by subtracting the old sun pose from the new one and wrapping azimuth to the principal interval. The commanded angle changes are therefore.

$$|\Delta\alpha| = \text{wrap}_{(-180,180)}(\alpha_{new} - \alpha_{old}) \quad (14)$$

$$|\Delta\varepsilon| = \text{wrap}_{(-180,180)}(\varepsilon_{new} - \varepsilon_{old}) \quad (15)$$

The open-loop rotation time needed to achieve a commanded change

$$t = \frac{|\Delta\theta| \times G}{6RPM_{motor}} \quad (16)$$

Where $|\Delta\theta|$ absolute value for the mortar rotation angle (azimuth $|\Delta\alpha|$ or elevation $|\Delta\varepsilon|$), G the gear ratio, and RPM_{motor} is the DC motor rotation/min. the number of motor turns expended in the move is

$$n_{motor} = G \times \frac{|\Delta\theta|}{360} \quad (17)$$

2.3. Experimental Validation

Tracker Hardware Design

To verify the effectiveness of the proposed design and the accuracy of the astronomical tracking algorithm, a real-life GPS-based dual-axis sun-tracking prototype was fabricated and field-tested for three consecutive days under real operating conditions. The prototype integrates sensing, computation, actuation, power management, and backhaul communications in a compact embedded stack (Figures 3). The power unit comprising a 20 W PV module, a 12/24 V MPPT charge controller, and a 12 V lead-acid battery supplies all subsystems. A MAX471 voltage-current module measures the PV electrical variables continuously (V , I , P) and reports them to the controller for performance monitoring. The control architecture follows a split-brain pattern: a Raspberry Pi executes high-level tasks (astronomical ephemerides, logging, network services), while an Arduino Nano handles deterministic low-level I/O and actuation. The Arduino receives geospatial/time data (latitude, longitude, UTC) from a NEO-M8N GPS and analog readings from the MAX471, then exchanges measurements and motion requests with the Raspberry Pi via a UART link. From the computed setpoints, the Pi issues target azimuth (ϕ) and elevation (α), which the Arduino converts into motor steps and timing to drive an L298N H-bridge powering two JGY-370-1285 DC geared motors (azimuth/elevation). The Pi is fed from the 12 V battery through an LM2596S buck (regulated 5 V) and provides remote access via a 4G-LTE IoT gateway (Ethernet). To extend battery autonomy,

the prototype now incorporates an Arduino-controlled power relay (Figure 3, red dashed paths). A 5 V relay channel switches the sensor/actuation rail, removing power from the GPS receiver, MAX471 sensor, and L298N motor driver when tracking is idle; a second channel switches the 12 V auxiliary rail to shut down the 4G router overnight. In the implemented watchdog scheme, the Raspberry Pi owns the power-gating of non-essential loads while remaining on a permanently powered 5 V rail; it drives the power-relay control line that feeds the Arduino's 5 V rail and the router's 12 V rail. At sunset detected when solar elevation falls below a small negative threshold for a dwell time the Pi's watchdog put the tracker in the initial position (inclined with angle 45 degree towards the south), and asserts relay-open, which shuts down the Arduino and, consequently, all Arduino-powered peripherals (GPS, MAX471 V-I sensor, L298N motor driver), and simultaneously powers down the 4G router; only

the MPPT and battery remain online to keep the Pi alive for housekeeping. At sunrise, the Pi closes the relay, restoring power to the Arduino (thereby re-energizing GPS, MAX471, and L298N) and the router, after which the system automatically reacquires position/time, resumes astronomical tracking through moving the tracker from the initial position toward the sunrise direction, and continues logging. This addition reduces night-time quiescent draw, improves energy efficiency, and protects the small battery under low-irradiance conditions, while preserving deterministic motor control on the Arduino and concentrating compute and networking on the Raspberry Pi. All electronics were integrated on standoff-separated PCBs (Arduino I/O board with MAX471 and motor interfaces; Raspberry Pi compute board) and deployed on the rooftop prototype; the GNSS front end (NEO-M8N with active patch) was housed in a sealed enclosure to maintain sky view and environmental protection.

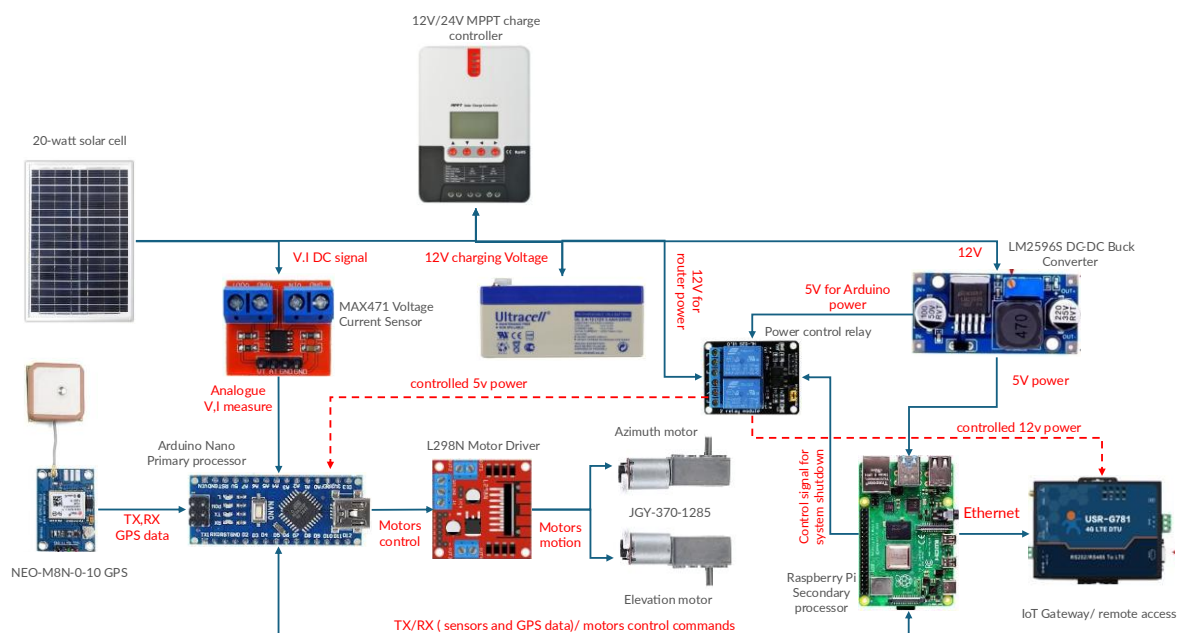


Figure 3. Hardware schematic of the developed GPS-based dual-axis sun-tracking system showing integration of the power, control, actuation, and communication modules.

Tracking Firmware design

As illustrated in Figure 4, the embedded control stack adopts a dual-processor architecture in which time-critical I/O is delegated to a resource-constrained primary processor, while compute-

and data-intensive services execute on a Linux-class secondary processor. The primary firmware comprises three hard-real-time subroutines: (i) a GPS acquisition task that ingests NEO-M8N latitude/longitude/UTC and validates fixes; (ii) a

metrology task that samples the MAX471 voltage–current front-end, performs sample-mean noise suppression, and converts ADC codes to P,I, and V; and (iii) a tracking actuator task that maps target azimuth and elevation setpoints to direction, duration, and enable signals for the dual JGY-370 geared DC motors via an L298N driver. The secondary firmware hosts three higher-level services: an astronomical ephemerides task producing setpoints, a data

archiving task for persistent logging, and an IoT publisher that exposes system state to the gateway. Coordination is enforced by a watchdog main routine: heartbeat supervision and timing checks ensure freshness of GPS and metrology data and completion of motor moves. This separation of concerns yields deterministic sensor/actuator timing on the Arduino, scalable computation and networking on the Pi, and robust, autonomous operation of the tracker [20].

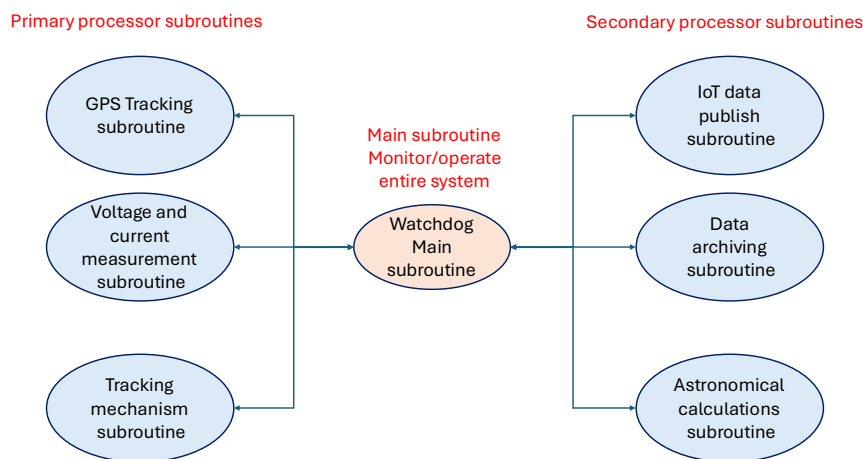


Figure 4. subroutines forming the tracking operating firmware

Primary processor Firmware

The primary processor (Arduino Uno) executes a cyclic pipeline comprising (i) GPS/time acquisition (Figure 5), (ii) PV electrical measurement (Figure 6), and (iii) tracker actuation (Figure 7). Each stage packages its outputs into compact serial frames and hands them to a secondary processor for logging/supervision. The loop repeats at a fixed tracking interval (or on command from the secondary). On boot, the firmware initializes the serial port and the GPS UART. If initialization succeeds, the code enters a bounded retry/timeout sequence: a 60-s window is allowed for valid NMEA sentences; a small counter limits the number of initializations attempts before issuing an Alarm frame. When a valid fix is available, the parser extracts (i) geographic position, $Loc = (Lat, Long)$; UTC time $T = HH:mm:ss$; and (iii) day-of-year (DoY). These fields are assembled into a Data packet (Lat, Long, DoY, T). and sent to the secondary over

the system serial link. If validity times out, an Alarm packet (type = “Warning”) is emitted to indicate loss of GPS service, after which the system reattempts acquisition on the next cycle. The measurement task begins by loading the ADC parameters (N, V_{ref}) and the analog front-end gains (K_v, κ_i) defined by the voltage divider and shunt paths (Eqs. (8)–(9)). To mitigate initial bias and sensor settling, the firmware enforces a warm-up interval (≈ 10 s) during which dummy readings are discarded. For each acquisition window, the ADC collects M samples per channel $c_v[m]$, $c_i[m]$ and computes their arithmetic means using Eqs. (12) (13). The averaged codes are converted to PV-side quantities via the linear ADC/model relations via equation (10) and (11). Instantaneous DC power is then computed as $P_{PV} = I_{PV} \times V_{PV}$. The task timestamps the record, forms a Measurement packet P_{PV}, I_{PV}, V_{PV} , and T , transmits it, and then idles awaiting the next measure command from the secondary. For mechanical actuation, the

firmware loads the known gear ratio G and the rated motor RPM and reads the previous and current sun poses α_{new} , α_{old} , ε_{new} , and ε_{old} the shortest angular deltas are computed using Eqs (14) and (15). while the signs $\text{sign}(\Delta\alpha)$ and $\text{sign}(\Delta\varepsilon)$ select the H-bridge direction pins. With only G and RPM known, the corresponding open-loop durations to realize the commanded

rotations are calculated from Eq (16). the expected motor turns are given by Eq (17), The firmware then drives each axis simultaneously, holding a constant PWM setpoint and stopping each motor when its timer elapses. A Status packet ("Tracking Done", timestamp) is emitted at completion, and the pose (α_{new} , α_{old}) is updated to the new values for the next cycle.

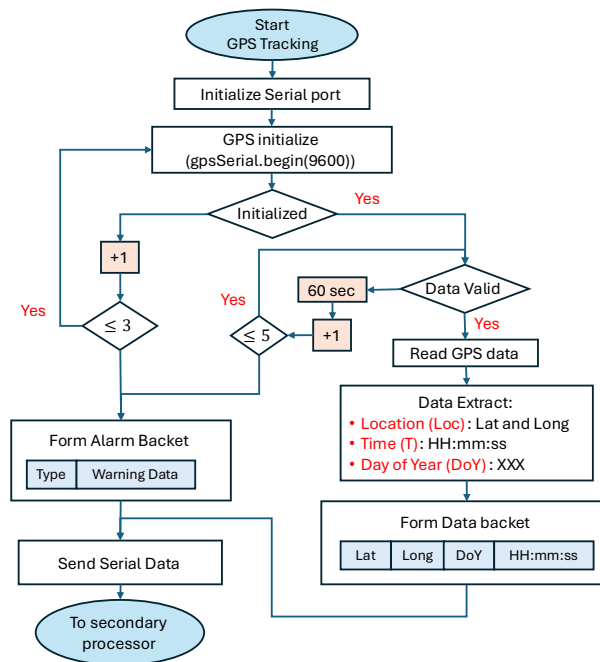


Figure 5. GPS/time acquisition subroutine

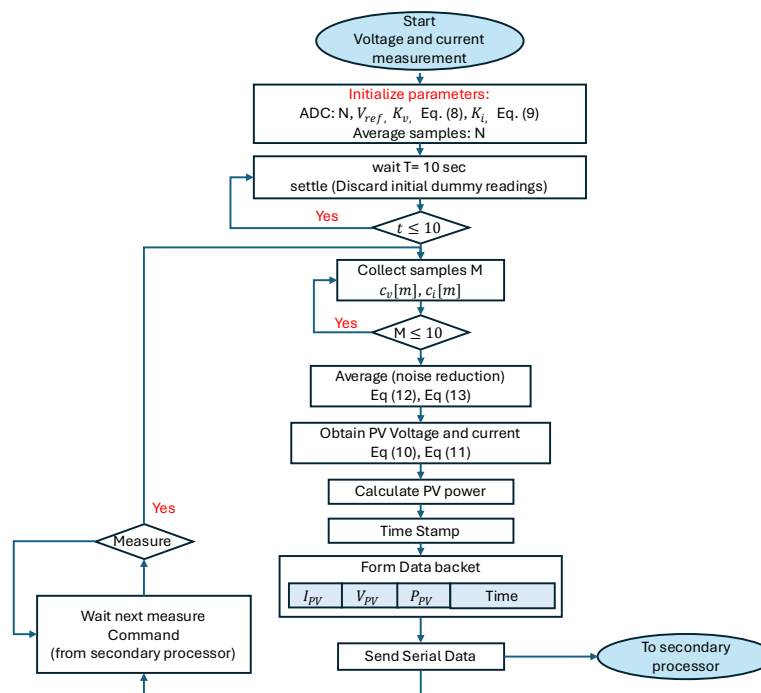


Figure 6. Current and voltage measure subroutine

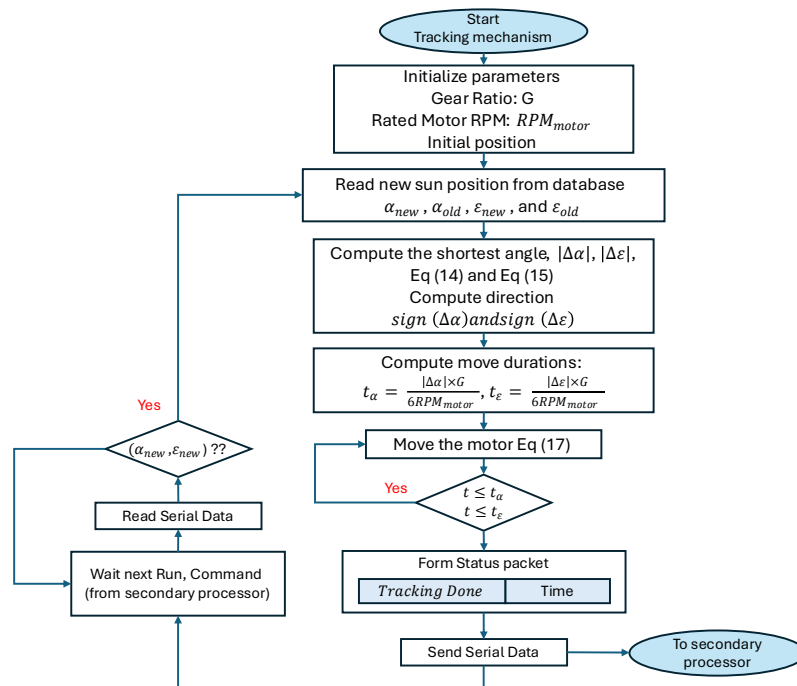


Figure 7. Tracking system movement subroutine

Secondary processor firmware

The Raspberry Pi hosts four cooperating subroutines: Watchdog, Astronomical Calculations, Data Archiving, and IoT Publish with the Watchdog acting as the orchestrator that schedules, supervises, and power-gates the stack (Figures 8–11). At boot, the Watchdog establishes system time, opens the UART link to the primary processor, and enters a cyclic supervisor loop. Each cycle it: (i) checks sunrise/sunset state from the last computed solar elevation (α); if α for a dwell window the Pi opens the power-relay, depowering the Arduino's 5 V rail (thereby GPS, MAX471, L298N) and the router's 12 V rail; if α at a scheduled wake, it closes the relay and waits for the Arduino to announce "ready"; (ii) polls the UART for packets and classifies them as GPS, sensor, or status; (iii) when a valid GPS fix arrives, it tags the day as "started," launches the Astronomical Calculations task, and arms the other two subroutines; (iv) when a measure/status request is due, it issues a command to the Arduino, waits for reply (with timeout), and then fans out the resulting record simultaneously to Data Archiving and IoT Publish. The Watchdog maintains heartbeats for all tasks (freshness

timers), escalates faults (restart subroutine, re-initiate UART, and relay cycle), and logs reason codes in the daily file. It also handles GPS power optimization: once the daily location/time is confirmed, it instructs the Arduino to shut the GPS rail to save power.

Given ϕ , λ , and DoY from the morning GPS fix, this task selects the update time (5 min) using the update time found optimal in the experimental study. It computes the number of setpoints per day $\text{samples} = (86\,400 / \text{update_time})$, then evaluates the sun coordinates for each timestamp using Eqs. (1)–(7): declination δ (Eq. 1), equation of time EoT (Eq. 2–3), local solar time LST (Eq. 4), hour angle HRA (Eq. 5), elevation α (Eq. 6), and azimuth φ (Eq. 7). The computed φ_k and α_k sequence is buffered in temporary memory and, when complete, saved as the "day plan" (CSV/DB) that will drive the primary tracker. The Watchdog exposes the next setpoint to the Arduino on demand; if the update time or location changes, the table is regenerated.

At day start it creates an empty CSV file and writes a header describing the schema: Time, DoY, Lat, Long, Azimuth, Elevation, I, V, P, error/status. For every packet delivered by the

Watchdog, it forms a validated archive row (time-aligned to the Pi's clock; GPS time used to check drift) and appends it to the file. The subroutine flushes to disk periodically to avoid data loss and rolls the file at day end (or on fault rollover), closing the handle and moving the artifact to a dated archive directory. It records status codes for exceptional events (no-GPS, no-sensor, UART timeout, relay cycle, etc.) so that post-processing can filter or impute missing values. This task publishes a compact JSON document for each record: {time, DoY, lat, lon, azimuth,

elevation, I, V, P, status}. On start it loads broker parameters (IP/port, credentials), forms topics, and establishes a PAHO-MQTT session. If the connection test fails, it retries with a bounded back-off (e.g., every 60 s up to 5 attempts) and exposes a "No IoT connection" alarm to the Watchdog. On success it streams packets with acknowledgment (QoS per deployment policy), and maintains a small local queue so brief link outages do not drop data. All publish outcomes are mirrored to the archive as confirmation fields.

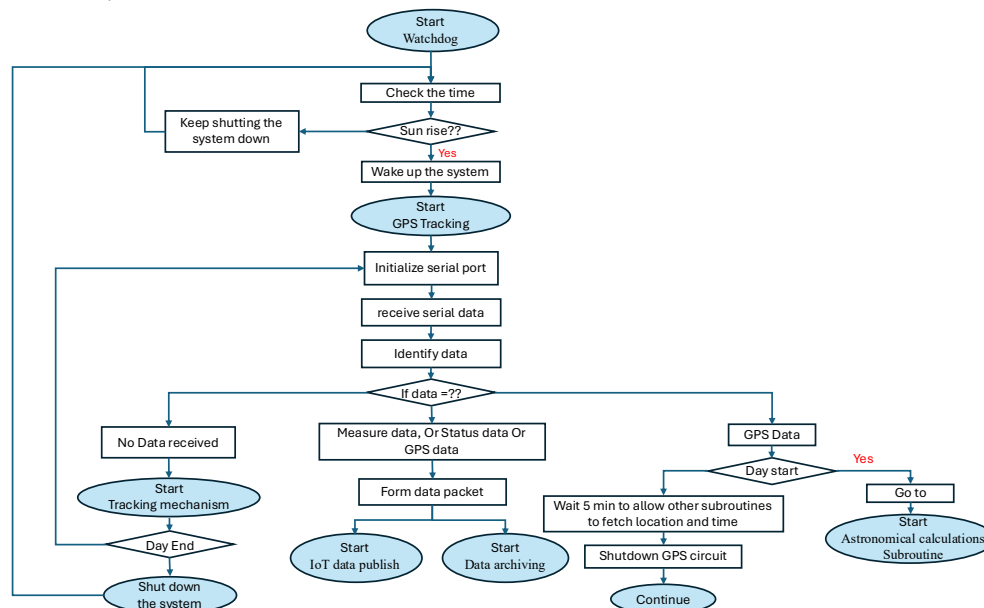


Figure 8. the watchdog main subroutine

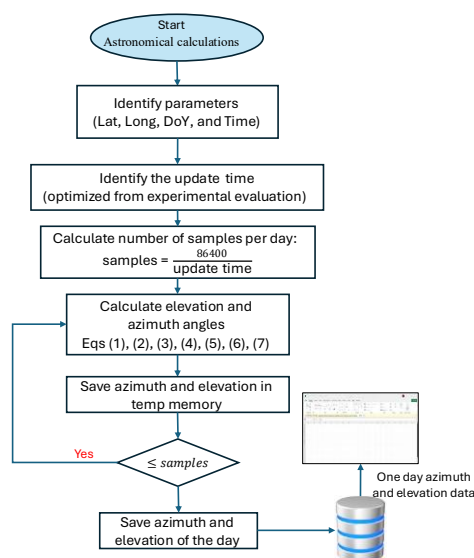


Figure 9. Azimuth and elevation calculation subroutine

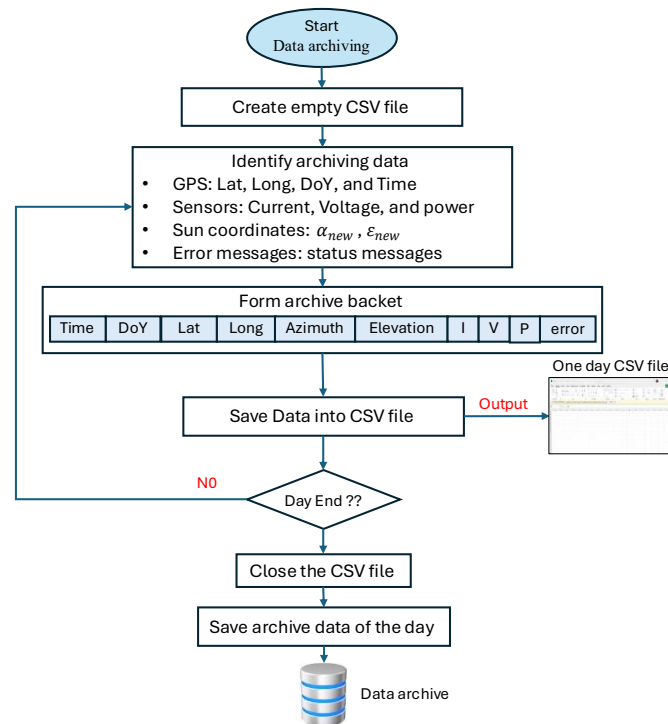


Figure 10. data archiving subroutine

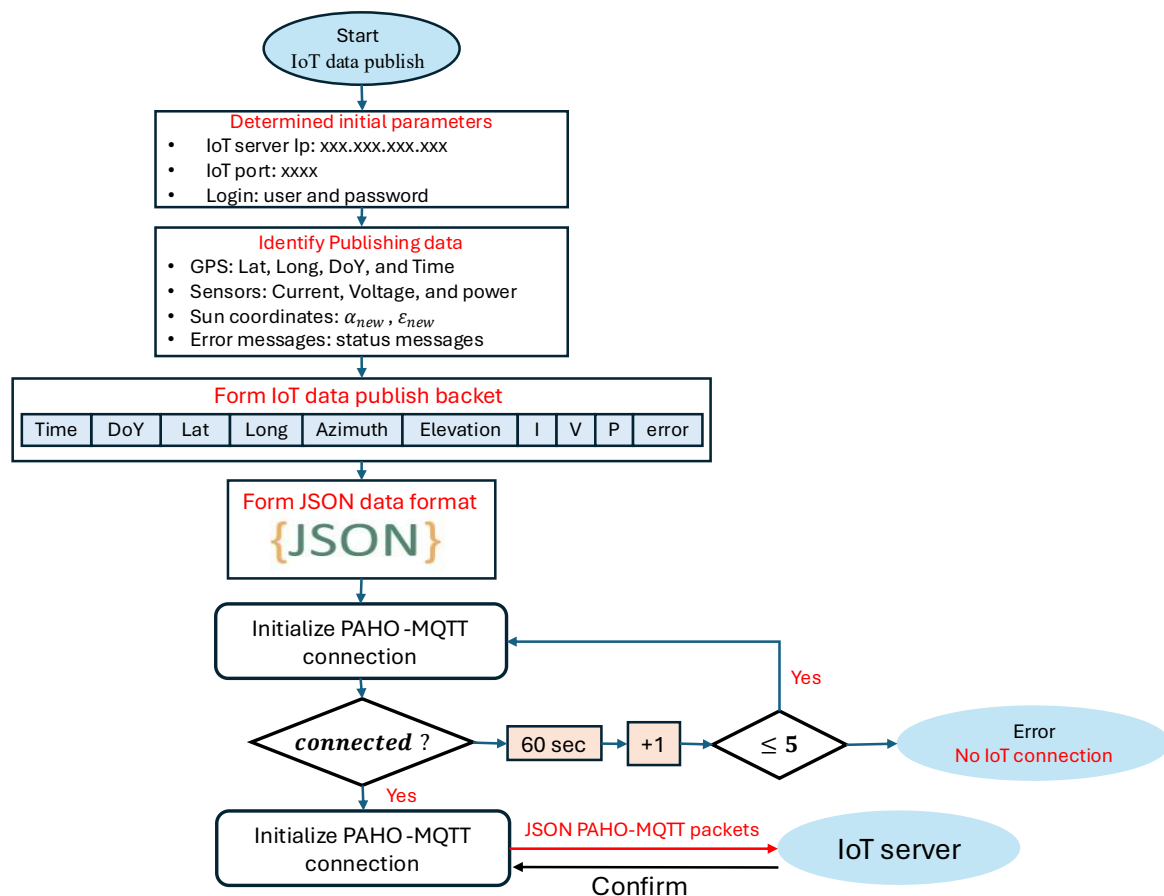


Figure 11. IoT data publishing subroutine

3. Evaluation and Results

Testing Case study

The field evaluation was conducted on the rooftop of the Egyptian National Seismic Network (ENSN), National Research Institute of Astronomy and Geophysics (NRIAG) facility as shown in Figure 12, where the prototype solar tracker was installed at geographic coordinates 29.8635° N, 31.3428° E. The test campaign spanned three consecutive days (15–17 June

2025), capturing complete diurnal cycles to assess start-up behavior, steady tracking performance, and late-afternoon operation. The deployment location is shown in Figure 13, and the site metadata used throughout the analysis are summarized in Table 6. These coordinates served as inputs to the astronomical model for computing solar azimuth and elevation, and the resulting setpoints were used to command the tracker during the measurement campaign.

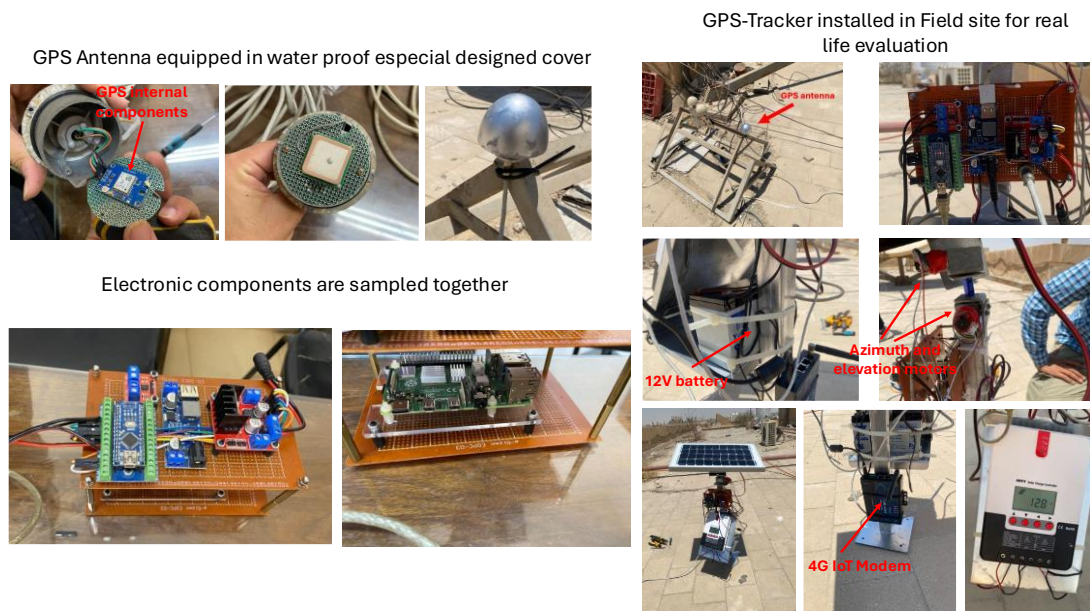


Figure 12. Field installation test and evaluation



Figure 13. real life test and evaluation location

Over the three consecutive test days we generated time-stamped solar geometry tables at 5-min resolution (local time, UTC+3) as shown in Figure 14. Each record provides azimuth $\phi(t)$

and elevation $\alpha(t)$. Sunrise and sunset were near 04:53 and 18:56–18:57 local time, respectively, bounding the interval of useful tracking on all three days.

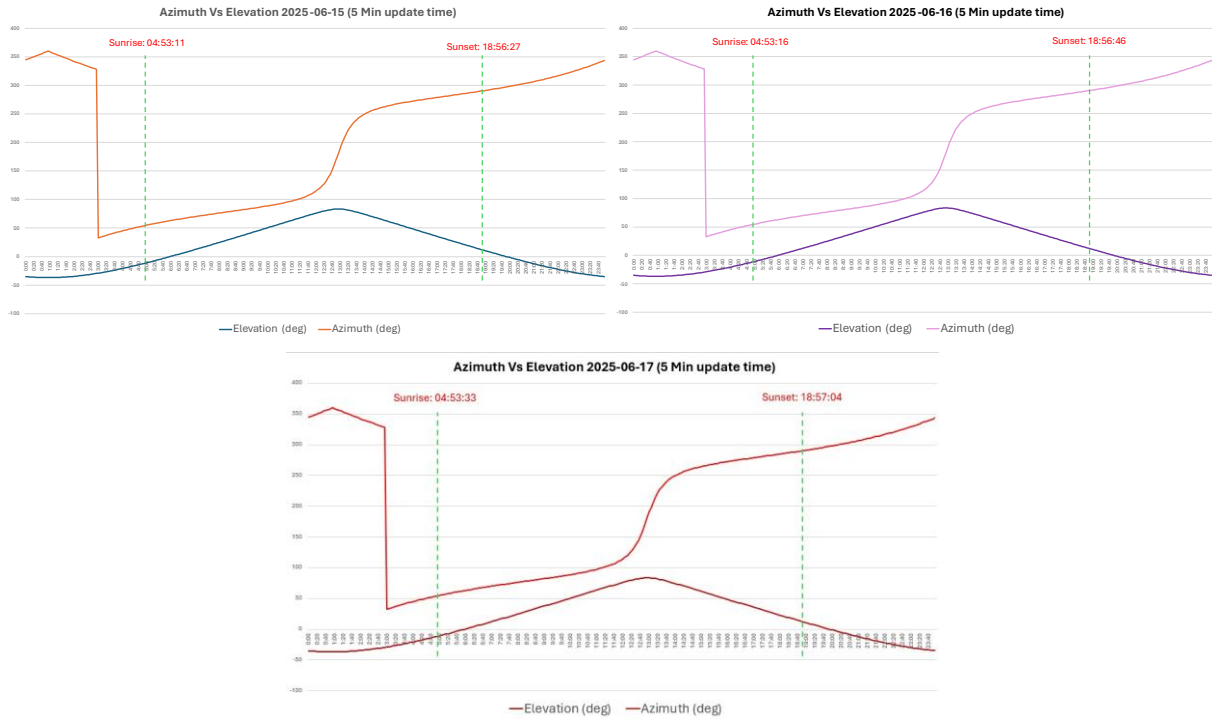


Figure 14. 24-hours Azimuth and elevation database for 15–17 June 2025 three consecutive days

Selecting best update time

During operation, the primary processor reads the next scheduled pair (ϕ, α) from the database, computes the shortest signed axis deltas $\Delta\alpha, \Delta\phi$ (Eqs. 14–15), and commands the geared motor for a duration (t) (Eq. 16) to realize the step. This produces a sample-and-hold orientation: panel pose is updated at discrete times and held constant between updates. we compute the daily sum of measured power samples under each update-time and compare:

$$E_{5min} = \sum P_{PV}(t_i)\Delta t \quad (18)$$

$$E_{other\ interval} = \sum P_{PV}(t_i)\Delta t \quad (19)$$

Our decision criterion is:

$$\Delta E = \frac{E_{5min} - E_{other\ interval}}{E_{5min}} \times 100 \quad (20)$$

We employ a silicon 18 V / 10 W mini-module. This module suit for tracking test (Electrically modest: ≤ 22 V, ≤ 0.6 A) since it easy to condition for Arduino-class ADC modeling (Eqs. 8–13). The voltage divider for the voltage sensor ($R_1 = 82k\Omega, R_2 = 18k\Omega$) gives $K_V = 0.18$ Eq (8) which gives $V_{ADC} = 3.9V$ at $V_{PV} = 21.6V$, which save for the 5V reference voltage. To quantify the impact of the update time on harvested energy, we used the 5-min data stream as the reference experiment and generated seven-time update schedules by down-sampling the same day's set of set-points and measurements to 10, 20, 30, 40, 50, and 60 mins. with V_{PV}, I_{PV} obtained from the MAX471/ADC relations given earlier. The daily energy for update time T is then evaluated by rectangular integration over the day given by Eqs (18-19). The 5-min data provide a high-fidelity baseline; each update time case reuses the same day's measurements but with update time 10, 20, 30, 40, 50, and 60 min,

energy obtained by time-domain accumulation. This protocol allows a fair comparison of total daily energy for 5, 10, 20, 30, 40, 50, and 60-mins update strategies.

For each update time, the system generated time series of the instantaneous PV voltage at the maximum power point V_{mp} , current at the maximum power point I_{mp} , and electrical power $P_{mp} = I_{mp} \times V_{mp}$. The resulting P-V-I profiles are shown in Figures [15a–15g], where each figure groups the three test days for a single update interval (5, 10, 20, 30, 40, 50, and 60 minutes, respectively). Each curve spans the full diurnal operating window, from initial sun acquisition after sunrise through loss of irradiance near sunset. Across all update times, the evolution of the three signals follows physically consistent behavior. The panel voltage V_{mp} rises sharply from near zero once the tracker first aligns with the sun in the early morning, then remains quasi-plateaued around 16 V for most of the illuminated period. This plateau corresponds to operation near the panel's datasheet maximum-power voltage (approximately 17–18 V at STC). The current I_{mp} ramps up more gradually after sunrise as irradiance increases with solar elevation, reaches a broad midday maximum on the order of 0.8–0.9 A, and then decays symmetrically toward evening. Consequently, the electrical power P_{mp} exhibits the expected bell-shaped profile: low in the early morning, peaking around solar noon at ~8 W, then tapering toward zero before sunset. This dome-like behavior of P_{mp} is visible for all three days and for all tested update times. What changes with the tracking update times is not the basic shape but the duration and smoothness of the productive window. With a fast 5-minute update time (Figure 15a), the tracker continuously reorients toward the sun, so useful power begins shortly after sunrise and persists

until just before sunset. The plateau in V_{mp} is long and nearly flat, and the P_{mp} bell is wide, indicating that the module is held close to normal incidence for most of the day. As the update time is relaxed to 10 and 20 minutes (Figures 15b and 15c), the curves for each day remain almost identical in peak magnitude: the midday P_{mp} peak stays near 8 W and V_{mp} remains near 16 V. This shows that, for mid-June sun in Cairo latitude ($\approx 29.86^\circ$ N), sub-hourly updates are already sufficient to preserve near-optimal pointing during the high-irradiance hours. However, subtle edge effects begin to appear: at 20 minutes and especially 30–40 minutes (Figures 15c and 15d), the onset of power in the morning is delayed and the cutoff in the evening arrives slightly earlier. This appears in the figures as a steeper “shoulder” at the start and end of V_{mp} : instead of ramping in smoothly just after sunrise, the voltage and power remain near zero until the next scheduled orientation step, then jump to the plateau. The same stepping is visible before sunset, where power drops to zero one update time interval earlier than in the 5-minute case. By 50- and 60-minute cadence (Figures 15f and 15g), those shoulders become pronounced. The controller holds the panel at a fixed attitude for up to an hour between pointing updates, so in early morning and late afternoon the sun can drift significantly away from the panel normal before the next correction. In the plots, this manifests as shorter effective “on” duration: the flat V_{mp} plateau starts later (after the first hourly correction) and ends earlier (the tracker does not chase the sun into the very low-elevation angles near sunset). Yet even in the 60-minute case, once the tracker has aligned for the hour that covers local noon, the peak P_{mp} is still close to the ~8 W seen in the faster modes. In other words, coarse updates mainly clip the shoulders of the daily power curve rather than degrading the midday peak.

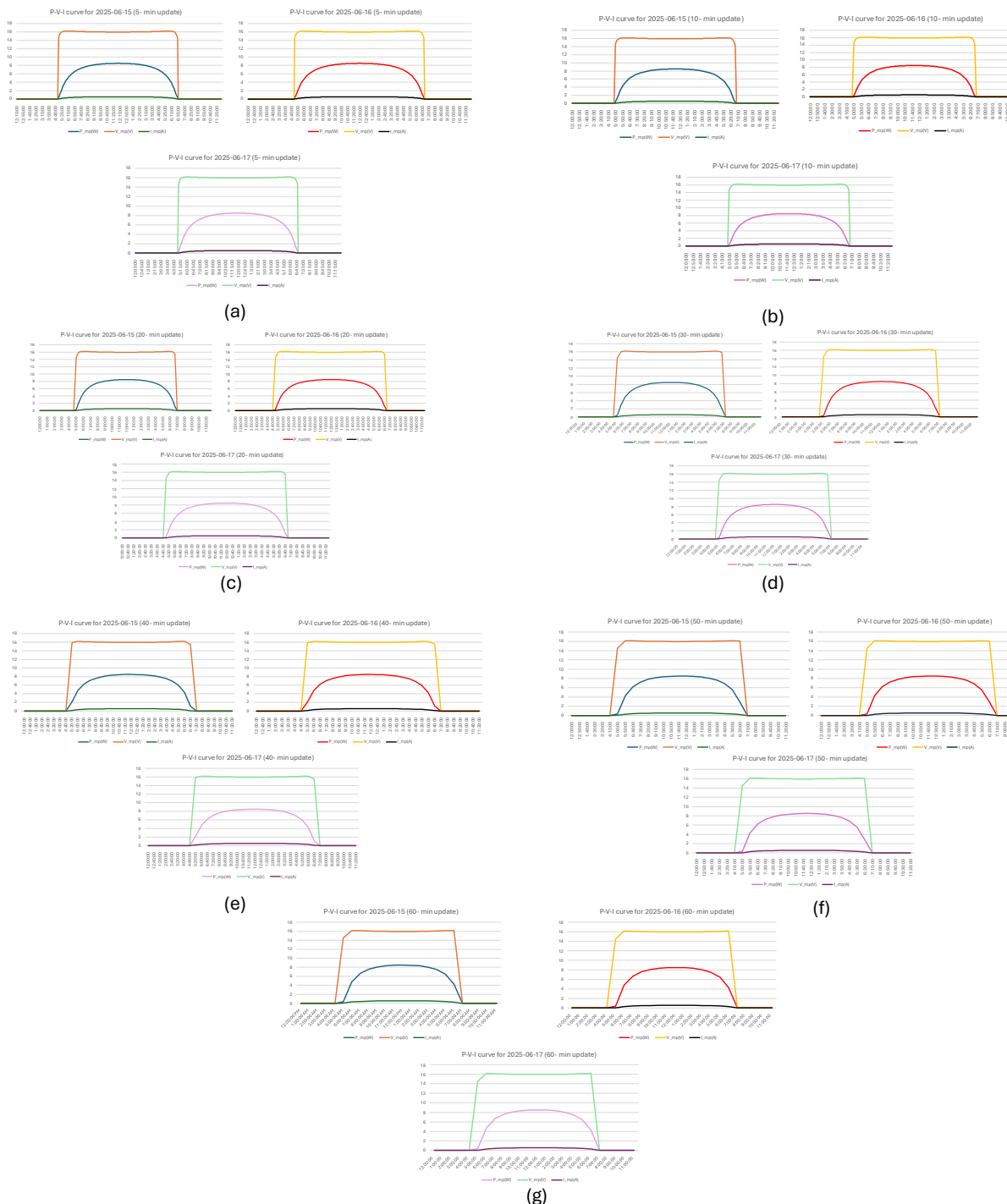


Figure 15. P-V-I curves for the 5,10,20,30-, 40-, 50-, and 60-min update times

The daily energy comparison for all tracking update times is summarized in the bar chart in Figure 16. Each group of bars corresponds to one test day (15, 16, and 17 June 2025), and each bar within the group represents the total harvested energy (Wh) for a different tracking update interval (5, 10, 20, 30, 40, 50, and 60 minutes). Two main observations emerge. First, all update

cadences from 5 minutes up to 50 minutes deliver almost the same daily energy: roughly 96–96.3 Wh per day for this 10 W nominal panel under clear mid-June sun. The difference between “very aggressive” tracking (5-minute updates) and “lazy” tracking (30–40-minute updates) is on the order of a few tenths of a watt-hour, i.e. well below 1%. Only the 60-minute cadence is

consistently lower, dropping to ~ 95.7 Wh/day, which is still within about 1% of the best case. This confirms the trend already seen in the P–V–I curves: slowing down the mechanical updates mainly clips energy at the sunrise and sunset shoulders but barely touches the plateau around local noon, where the irradiance and therefore the contribution to daily Wh is highest. Second, this “flatness” across update times is largely a function of two experimental conditions: (i) the test panel is small (10 W rating, ~ 18 Vmp, ~ 0.56 A at STC), and (ii) the measurements were taken near the summer solstice at latitude $\sim 29.86^\circ$ N, when the sun path is high and the tracker naturally sees long, nearly normal incidence for most of the day. Under these favorable conditions, the marginal gain from chasing the sun every 5 minutes versus every 30 minutes is tiny, because the sun’s azimuth/elevation doesn’t move fast enough to cause a major loss between corrections. In other words, the system is operating in a “slow geometry, high irradiance” regime where coarse steering is already good enough.

This result, while encouraging for low-power autonomous trackers, does not mean cadence is irrelevant for real deployments. In a commercial array with kilowatts or megawatts of PV, even a 0.5–1% gain is economically meaningful over the plant lifetime. Also, our test took place in June. In winter, at higher latitudes, or under partially cloudy skies with strong direct-beam dominance at low solar elevation, the sunrise/sunset shoulders become much more valuable. In those regimes, the difference between re-pointing every 5 minutes versus every 30 or 60 minutes can

translate into a nontrivial number of additional watt-hours harvested per module, especially in the early morning when grid operators actually value ramp-up energy. These two insights motivate a hybrid tracking policy rather than a single fixed cadence. A simple strategy is:

- Use a fast cadence (e.g. 5-minute updates) only during the low-elevation intervals near sunrise and near sunset. In that window, the sun’s angle of incidence changes fastest relative to power level, and every extra watt actually matters because you’re either just starting to generate or about to drop to zero. Fine steering here recovers the shoulder energy that the coarse cadences were missing.
- Switch to a relaxed cadence (e.g. 30-minute updates) for the rest of the day, roughly once the sun is above some elevation threshold and until it drops back below that threshold in the afternoon. In this high-elevation band, our data show that updating every 5 minutes gives essentially the same daily Wh (~ 96.2 Wh) as updating every 30–40 minutes (~ 96.3 Wh). So, we can afford to reduce mechanical motion and controller workload without sacrificing useful energy.

This hybrid mode keeps the mechanical duty cycle and control overhead low for most of the operating hours (good for actuator life, battery budget, and network bandwidth), but still protects the most cadence-sensitive parts of the curve the first and last hour of sun. For a rooftop testbed with two geared DC motors, this directly extends motor lifetime and reduces nighttime wake/power cycles.

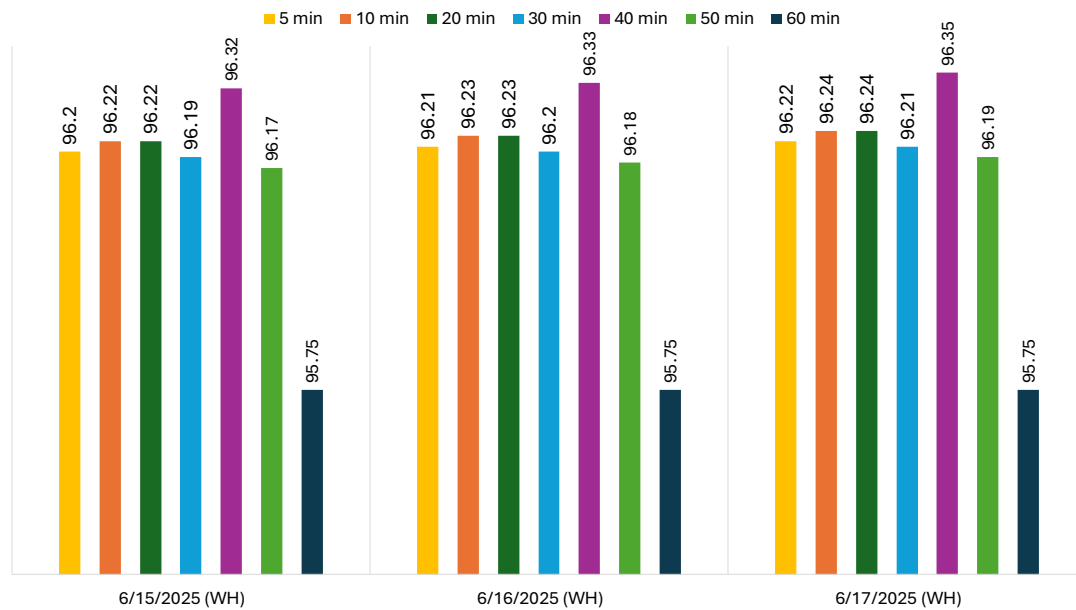


Figure 16. comparative analysis of total energy/day for different update times strategies

4. Summary and Discussion

In this work, we evaluated multiple solar tracking update cadences every 5, 10, 20, 30, 40, 50, and 60 minutes and quantified their effect on instantaneous PV power and total daily energy. Across three consecutive test days (15–17 June 2025), the total collected energy per day remained tightly clustered around ~ 96 Wh for all cadences from 5 minutes up to 50 minutes, with only a small drop ($\approx 0.5\text{--}1\%$) at the slowest, 60-minute update interval. This indicates that, under favorable summer conditions with a relatively high sun path and stable irradiance, the system is not highly sensitive to how often the tracker is re-pointed. In other words, once the sun is fairly high in the sky, even “lazy” mechanical repositioning every 20–40 minutes keeps the panel close enough to normal incidence that power loss is marginal. However, even though the curves look similar in this small-scale experiment with a single ~ 10 W panel, the implications for larger systems are not trivial. In a kilowatt- or megawatt-scale farm, a $0.5\text{--}1\%$ daily energy gain is economically meaningful over the asset lifetime. Also, our tests occurred in mid-June at latitude $\sim 30^\circ\text{N}$, where the sun is high and the middle-of-day plateau dominates the daily energy budget. In winter, at higher latitudes, or in partially cloudy conditions dominated by direct

beam at low elevation angles, the “shoulder energy” near sunrise and sunset becomes more important. During those windows, the sun moves quickly in azimuth and elevation relative to the available irradiance. If the tracker only updates once per hour, it can easily “miss” many low-angle photons. Based on these observations, we proposed an adaptive (hybrid) tracking policy rather than a single global cadence. The idea is: (1) Use a fast update rate (e.g. 5 minutes) during sunrise and sunset, when accurate pointing yields relatively large fractional gains and when every watt is valuable. (3) Switch to a slower update rate (e.g. 30 minutes) for the high-sun interval in the middle of the day, when geometry changes slowly and the panel is already near its maximum effective incidence. This hybrid strategy preserves almost all of the energy benefit of dense tracking while reducing actuator wear, power consumption, and communication overhead. It is also seasonally tunable: in winter, the “fast window” can be extended, and in summer it can be shortened. The dual-processor architecture developed in this work (Raspberry Pi for scheduling and decision logic, Arduino Nano for actuation and sensing) is already capable of implementing such a rule. In our field test, the dual-axis GPS-based tracker consistently produced about 96.2–96.35 Wh per day for

reasonable update cadences (5, 10, 20, 30, 40 min). When we slowed the motion to effectively “almost fixed” behavior 60-minute updates, similar to a static south-facing panel at 45° tilt the daily energy dropped to about 95.75 Wh. That corresponds to roughly a 0.5–0.7% gain for active tracking (e.g. 96.33 Wh vs 95.75 Wh). For a single ~10 W panel in midsummer, this looks small in absolute terms, but at larger PV scales or in winter (where a fixed 45° panel is much more misaligned from the sun for longer), that fraction becomes meaningful.

The study still has clear limits and natural next steps. First, our validation window was only three summer days (15–17 June 2025) in clear, high-elevation sun, so we did not capture winter behavior (low sun, fast azimuth change near the horizon) or long-term mechanical effects like backlash growth, dust, wind loading, or thermal drift; we also ran on a single ~10 W module, power model, so the ~96 Wh/day vs ~95.75 Wh/day gains we observed for active tracking versus quasi-fixed pointing should be treated as indicative, not bankable. Second, although we implemented an MQTT-style publisher that formats GPS, DoY, commanded azimuth/elevation, measured V–I–P, and status

into JSON packets, we did not stand up a full IoT back end (broker, database, dashboard); packets were only captured locally, so remote analytics, alarms, and long-term performance dashboards were not demonstrated in the field. Going forward, we plan to (i) close the IoT loop by streaming all telemetry into a persistent time-series database with a web dashboard for health monitoring and historical energy yield; (ii) run continuous multi-season trials to quantify winter gains and mechanical durability; (iii) upgrade cadence control from a fixed rule (“5 min near sunrise/sunset, ~30 min mid-day”) to an adaptive policy that moves only when predicted extra Wh justifies motor wear and energy cost. This work presented the core architecture, control logic, and embedded software of a fully autonomous, GPS-based dual-axis solar tracker, along with an initial energy performance evaluation. Future work will extend beyond the foundational design to long-duration field validation, seasonal energy characterization, and closed-loop power budgeting, including an intelligent energy management layer that dynamically optimizes tracking cadence, power consumption, and harvested energy at system scale.

References

- P. S. Paliyal, S. Mondal, S. Layek, P. Kuchhal, and J. K. Pandey, “Automatic solar tracking system: a review pertaining to advancements and challenges in the current scenario,” *Clean Energy*, vol. 8, no. 6, pp. 237–262, 2024.
- R. K. Ratnesh, R. Kumar, S. Singh, R. Chandra, and J. Singh, “Recent advances in solar cell technology: addressing technological challenges, scenarios, and environmental implications in the development of sustainable energy solutions,” *New Journal of Chemistry*, 2025.
- M. E.-S. M. Essa, A. A. Hassan, E. E. El-Kholy, and M. M. R. Ahmed, “Innovations and advancements in solar tracker systems: A comprehensive review,” *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, vol. 16, no. 1, 2025.
- R. Sadeghi, M. Parenti, S. Memme, M. Fossa, and S. Morchio, “A Review and Comparative Analysis of Solar Tracking Systems,” *Energies (Basel)*, vol. 18, no. 10, p. 2553, 2025.
- A. Tkachenko and V. I. oglu Ismaylov, “Renewable energy innovations: Synergy of technology and sustainable development,” *МЕНЕДЖМЕНТ І БІЗНЕС*, p. 43.
- R. Almutairi, G. Bergami, and G. Morgan, “Advancements and challenges in IoT simulators: A comprehensive review,” *Sensors*, vol. 24, no. 5, p. 1511, 2024.
- Sayeedurrahman, B. Sabir, R. A. Khan, D. Masood, and A. Nat, “A solar tracking system with light-dependent resistors and a stepper motor controlled by a microcontroller,” *2023 International Conference on Computer*,

-
- Electronics and Electrical Engineering and their Applications, IC2E3 2023, 2023, doi: 10.1109/IC2E357697.2023.10262787.
- W. Lu and P. Ajay, "Solar PV tracking system using arithmetic optimization with dual axis and sensor," *Measurement: Sensors*, vol. 33, p. 101089, Jun. 2024, doi: 10.1016/J.MEASEN.2024.101089.
- I. Anshory et al., "Monitoring solar heat intensity of dual axis solar tracker control system: New approach," *Case Studies in Thermal Engineering*, vol. 53, p. 103791, Jan. 2024, doi: 10.1016/J.CSITE.2023.103791.
- C. H. Pardosi, M. Siregar, and L. W. Pandjaitan, "Design and Implementation of A Dual-Axis Solar Tracking System using Arduino Uno Microcontroller," *Jurnal ELTIKOM: Jurnal Teknik Elektro, Teknologi Informasi dan Komputer*, vol. 8, no. 1, pp. 44–56, Jun. 2024, doi: 10.31961/ELTIKOM.V8I1.1105.
- M. A. Ponce-Jara, I. Pazmino, Á. Moreira-Espinoza, A. Gunsha-Morales, and C. Rus-Casas, "Assessment of Single-Axis Solar Tracking System Efficiency in Equatorial Regions: A Case Study of Manta, Ecuador," *Energies* 2024, Vol. 17, Page 3946, vol. 17, no. 16, p. 3946, Aug. 2024, doi: 10.3390/EN17163946.
- Z. Ge, Z. Xu, J. Li, J. Xu, J. Xie, and F. Yang, "Technical-economic evaluation of various photovoltaic tracking systems considering carbon emission trading," *Solar Energy*, vol. 271, p. 112451, Mar. 2024, doi: 10.1016/J.SOLENER.2024.112451.
- W. Batayneh, A. Owais, and M. Nairoukh, "An intelligent fuzzy based tracking controller for a dual-axis solar PV system," *Autom. Constr.*, vol. 29, pp. 100–106, Jan. 2013, doi: 10.1016/J.AUTCON.2012.09.006.
- A. El Hammoumi, S. Motahhir, A. El Ghzizal, and A. Derouich, "Internet of Things-Based Solar Tracker System," *Green Energy and Technology*, pp. 75–95, 2021, doi: 10.1007/978-3-030-64565-6_4.
- E. Ghaderpour et al., "AI-Based Virtual Assistant for Solar Radiation Prediction and Improvement of Sustainable Energy Systems," *Sustainability* 2025, Vol. 17, Page 8909, vol. 17, no. 19, p. 8909, Oct. 2025, doi: 10.3390/SU17198909.
- N. T. Pham, N. Y. Tran-Van, and K. H. Le, "LS-SPP: A LSTM-Based Solar Power Prediction Method from Weather Forecast Information," *Proceedings - 2021 8th NAFOSTED Conference on Information and Computer Science, NICS 2021*, pp. 144–148, 2021, doi: 10.1109/NICS54270.2021.9701529.
- E. Li, B. Li, and H. Li, "Designing and Implementing a Novel Solar Panel Tracker Leveraging Reinforcement Learning Technique," *Journal of Student Research*, vol. 10, no. 4, Nov. 2021, doi: 10.47611/JSRHS.V10I4.2237.
- A. Al-Humairi, E. Khalis, Z. A. Al Hemyari, and P. Jung, "Machine Learning-Based Predictive Maintenance for Photovoltaic Systems," *AI* 2025, Vol. 6, Page 133, vol. 6, no. 7, p. 133, Jun. 2025, doi: 10.3390/AI6070133.
- L. Sun, J. Bai, R. K. Pachauri, and S. Wang, "A horizontal single-axis tracking bracket with an adjustable tilt angle and its adaptive real-time tracking system for bifacial PV modules," *Renew. Energy*, vol. 221, p. 119762, Feb. 2024, doi: 10.1016/J.RENENE.2023.119762.
- R. M. A. Bakir, M. S. Soliman, M. S. M. Elksasy, M. S. Saraya, and M. M. Abdelsalam, "PRO-MATE IOT-based cost-effective Ground motion monitoring acceleration sensor node: hardware and software description," *IEEE Sens. Lett.*, vol. 7, no. 11, pp. 1–4, 2023.
-