ECE 445 Senior Design Laboratory
Design Document

Intelligent Net-Energy-Optimized Solar Tracker

Team #48
Minghao Fang
Ziru Niu
Yifei Liu
Yikai Zhang
Advisor: Ruisheng Diao

April 1, 2026

Contents

1 Introduction **2**
1.1 Problem Statement . . . 2
1.2 Solution Overview and Visual Aid . . . 2
1.3 High-Level Requirements List . . . 3

2 Design **3**
2.1 Block Diagram . . . 3
2.2 Mechanical Design . . . 4
2.2.1 Overall Structural Configuration . . . 4
2.2.2 Dual-Axis Motion Mechanism . . . 5
2.2.3 Multi-Panel Solar Arrangement and Sensing Structure . . . 6
2.2.4 Structural Design Considerations . . . 6
2.3 Sensing Subsystem . . . 6
2.4 Control and Communication Subsystem . . . 7
2.5 Power Subsystem . . . 8
2.6 User Interface and One-Button Operation . . . 10
2.7 Tolerance Analysis . . . 11

3 Cost **12**

4 Schedule **12**

5 Ethics and Safety **13**
5.1 Ethics . . . 13
5.2 Safety . . . 13

6 References **14**

1 Introduction

1.1 Problem Statement

Small photovoltaic platforms lose energy when a fixed panel does not face the dominant light direction. A simple motorized tracker does not automatically solve that problem, because each movement consumes electrical energy and introduces mechanical wear. Under weak sunlight, cloud cover, or fast-changing illumination, a tracker can easily spend more energy turning the panel than it gains from the improved panel angle. Most low-cost student solar trackers are designed to maximize motion toward light, but not to determine whether that motion produces positive net energy gain.

This project addresses that gap with a dual-axis solar-tracking test platform that explicitly measures both generated panel power and motor-consumption power. The system uses a lower-level STM32 controller for sensing, actuation, and safety, and a Raspberry Pi for online inference and logging. The goal is not only to point the panel toward brighter light, but to decide when movement is worth the energy cost.

1.2 Solution Overview and Visual Aid

The project is organized around the top-level hardware chain. A small solar panel is connected through an `INA219` monitor to a dummy load so the panel output can be measured without using the panel to power the controller. A second `INA219` measures the motor-power branch. A `BH1750 + I2C mux + 16-sensor` light ring samples directional light around the panel. A custom STM32-centered control board reads the sensors, refreshes an OLED, executes homing and motion commands, and drives two `A4988` stepper-driver stages connected to two `NEMA17` motors for yaw and pitch motion. A Raspberry Pi exchanges state and target-angle commands with the STM32 over UART and serves as the online inference and logging node. Offline model training is performed on a PC or cloud host and the trained model is later exported to `ONNX` for Raspberry Pi deployment.

This architecture matches the project objective especially well because it makes net-energy accounting measurable. The panel test path, motor branch, light ring, and control path are all visible and instrumented. The system can therefore explain why it moved, why it stayed still, and whether the resulting action increased or decreased net energy gain.

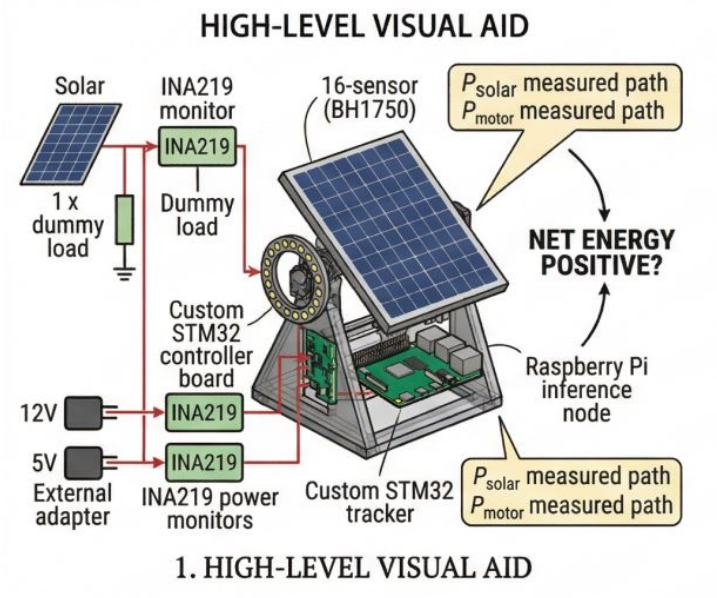


Figure 1: High-level visual aid derived from the current top-level hardware design. The figure shows the dual-axis tracker context, the panel measurement path, the STM32 lower-level controller, the Raspberry Pi inference node, the I2C sensing network, and the split 12 V and 5 V power architecture.

1.3 High-Level Requirements List

The final system shall satisfy the following high-level requirements:

(1) **Autonomous startup:** After one button press, the system shall complete self-check and homing, then enter autonomous tracking mode without requiring a tethered computer.

(2) **Tracking and net-gain behavior:** Under stable lighting, the platform shall track the brightest direction with steady-state pointing error less than 5°. Under weak or rapidly changing light, it shall avoid unnecessary actuation and demonstrate non-negative net energy gain relative to a naive always-move baseline during the same test window.

(3) **Telemetry and visibility:** The OLED and host log shall update at least twice per second and report operating mode, panel voltage, panel current, motor-branch current, and current yaw/pitch state.

2 Design

2.1 Block Diagram

The complete system contains six functional blocks: sensing, embedded control, online inference, actuation, power, and local user interface. The STM32F407ZGT6 is the hardware hub. It reads the shared I2C sensor network, drives the two motor axes, enforces homing and fault behavior, and exchanges data with the Raspberry Pi over UART. The Raspberry Pi receives current light-ring readings and current yaw/pitch state, runs the deployed model, and returns the next target angles. The two-axis mechanical structure converts these target angles into panel motion through `A4988`

stepper drivers and NEMA17 motors. The power system keeps the motor and logic branches separate so motor transients do not collapse the logic rail.

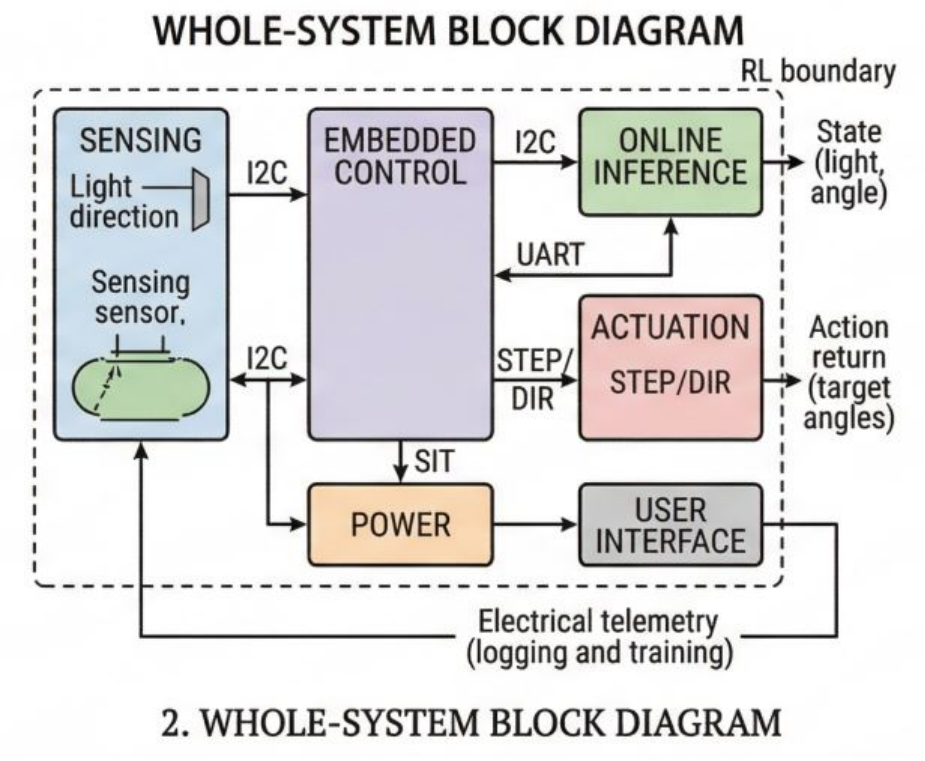


Figure 2: Whole-system block diagram based on the current top-level design. The solar panel feeds a measured dummy-load path, the STM32 reads the light ring and telemetry devices over I2C, the Raspberry Pi exchanges commands and state over UART, and the split 12 V and 5 V rails isolate motor power from logic power.

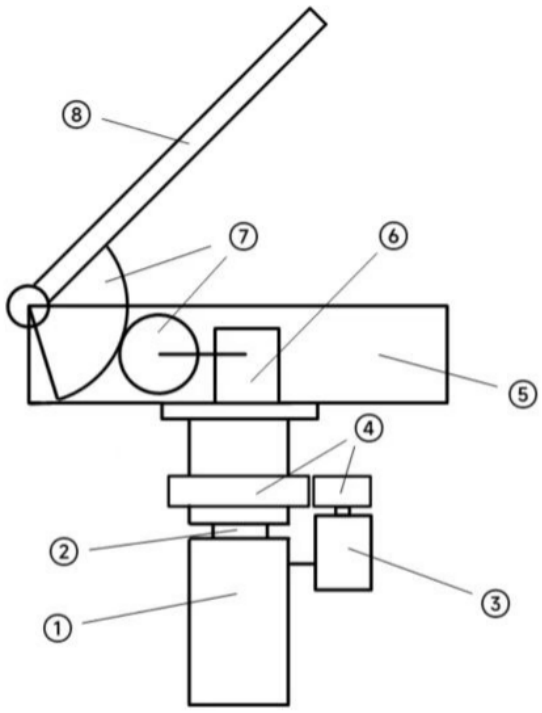
2.2 Mechanical Design

2.2.1 Overall Structural Configuration

The mechanical structure of the proposed solar tracking system is designed as a compact dual-axis tracking platform, enabling independent control of azimuth and elevation angles. The system adopts a stacked configuration, where rotational motion in the horizontal plane is combined with tilting motion in the vertical plane.

The overall structure consists of a base, a rotational support layer, and an upper tilting assembly. Two stepper motors are used to actuate the two rotational degrees of freedom, ensuring precise angular control and compatibility with digital control algorithms.

The mechanical design emphasizes structural stability, modularity, and ease of integration with sensing and control subsystems.



1. base 2. sleewing bearing 3. NEMA17 stepper motor (azimuth axis) 4. gear set
5. solar panel support frame 6. NEMA17 stepper motor (elevation axis)
7. gear reduction set 8. solar panel module

Figure 3: Mechanical subsystem placeholder. Replace with a dual-axis assembly drawing showing the yaw and pitch frame, motor placement, panel mount, and homing-switch locations.

2.2.2 Dual-Axis Motion Mechanism

The system realizes solar tracking through two independent rotational mechanisms: azimuth rotation and elevation adjustment.

Azimuth Rotation The azimuth motion is driven by the stepper motor, which is mounted on the base. The motor transmits torque through a gear set, and the slewing bearing serves both as a load-bearing structure and as a rotational interface, allowing the upper assembly to rotate smoothly around the vertical axis. This configuration ensures stable rotation under load while maintaining positioning accuracy.

Elevation Adjustment The elevation motion is driven by a second stepper motor, which is mounted on the rotating support frame. The motor drives a gear reduction set, which controls the rotation of the solar panel from 0 to 90 degrees around a horizontal hinge.

Compared with linear actuation mechanisms, this rotational design provides higher control precision, improved mechanical simplicity, and better integration with stepper motor control.

The two stepper motors cooperate to control the azimuth and elevation angles, allowing the solar panel module to align with the vertical direction of incoming light.

2.2.3 Multi-Panel Solar Arrangement and Sensing Structure

The solar panel module adopts a customized multi-panel configuration to simultaneously achieve energy harvesting and directional light sensing.

At the center, a primary photovoltaic panel is used for energy generation. Surrounding this central panel are eight auxiliary panels arranged in the following directions: up, down, left, right, upper-left, upper-right, lower-left, and lower-right.

Each auxiliary panel is mounted at a fixed inclination angle of 30° relative to the central panel. Each auxiliary panel is equipped with a light sensor. Due to the different orientations of the panels, the incident light intensity varies across them. By comparing these intensity values, the system can determine the direction of the light source.

This design forms a distributed sensing structure, enabling real-time estimation of light direction without relying on external solar position models.

2.2.4 Structural Design Considerations

To ensure reliable operation, several design considerations are incorporated.

Stability The use of a slewing bearing improves load distribution and enhances structural rigidity.

Transmission Efficiency The use of gear reduction improves torque output and minimizes motor load, which is critical for energy-aware operation.

Modularity Each component can be independently assembled and replaced.

Control Compatibility Stepper motor-based actuation enables accurate angle control required by advanced control strategies.

2.3 Sensing Subsystem

The sensing subsystem combines directional sensing and electrical measurement. A `BH1750 + mux + 16-sensor ring` samples incident light around the panel so the controller can estimate the dominant light direction. The STM32 polls the mux and reads one channel at a time over I2C. `INA219_1` measures the solar-panel test path into a dummy load, and `INA219_2` measures the external 12 V motor branch. This combination gives the project both the directional information needed to track light and the power data needed to reason about net gain.

The sensor set follows the exact intent of the top-level design. The project does not assume that the brightest direction is always the correct decision. Instead, it measures whether moving the panel improves generated power enough to justify the motor energy required to do so.

Subsystem requirements

1. The STM32 shall sample the light ring and both `INA219` channels at an effective rate of at least 10 Hz.

2. The panel measurement path shall resolve voltage and current changes accurately enough to distinguish useful changes in generated power during controlled tracking experiments.

3. Sensor processing shall include filtering or decision deadband so transient shadows and measurement noise do not cause rapid oscillation.

Verification

1. Sweep a directional light source around the panel and verify that the light-ring distribution shifts consistently with source position.

2. Compare both `INA219` channels against a bench multimeter at multiple operating points and confirm the error is small enough for design-level net-power comparison.

3. Inject noisy or rapidly changing light conditions and verify that the filtered light estimate and motion decision remain stable.

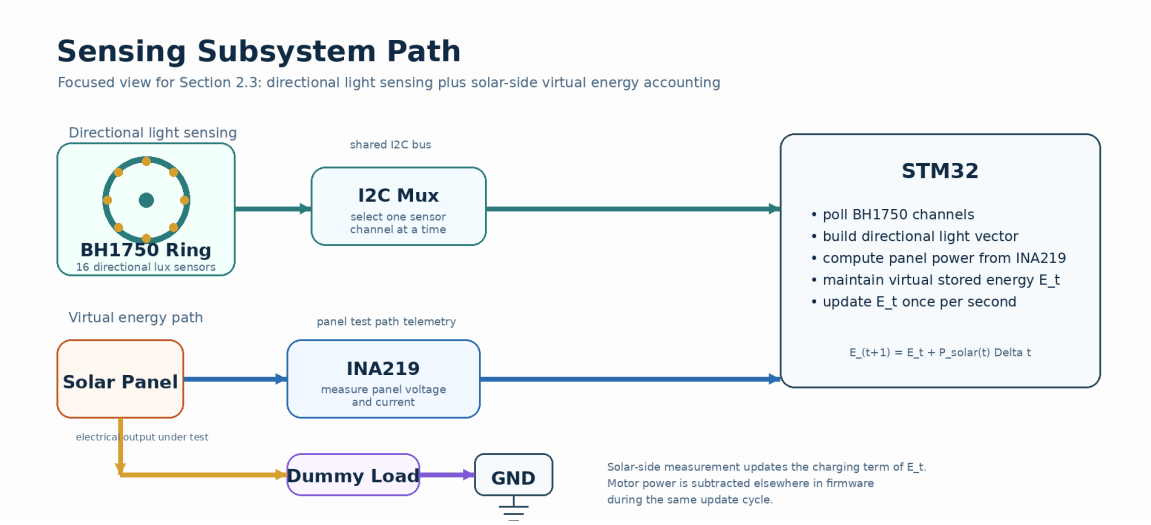


Figure 4: Simplified sensing-subsystem view with two coupled paths. The upper path shows directional light sensing through the `BH1750` ring and I2C mux into the STM32. The lower path shows the solar-panel test branch through `INA219` and the dummy load. Together, these measurements support both light-direction estimation and the STM32's virtual energy-state update.

2.4 Control and Communication Subsystem

The control subsystem is centered on the STM32F407ZGT6. It handles sensor polling, OLED refresh, state-machine logic, homing, motor-command execution, and safety enforcement. The current board-level interface assignment is consistent with the hardware notes: `PB6` and `PB7` are used for I2C `SCL/SDA`; `PA8` and `PA9` are used for yaw and pitch `STEP`; and `PB0/PB1` and `PB10/PB11` provide the direction and enable-style signals for the two axes. The Raspberry Pi is connected over a simple UART header and acts as the formal online inference node.

At the RL boundary, the STM32 uploads only the online state needed for deployment: current light-ring readings and current yaw/pitch state, where current angle is defined as the previous accepted target angle after homing. The Raspberry Pi runs inference and returns the next target

angles. Logged electrical telemetry remains available for reward design, offline analysis, and evaluation, but is not required by the online model input. This division keeps the live interface small and feasible while preserving richer data for later training.

Subsystem requirements

1. The shared I2C bus shall support the OLED, both `INA219` devices, and the mux controller without address conflict.
2. The STM32 shall enforce safe motor behavior even if the Raspberry Pi stalls, disconnects, or returns invalid commands.
3. The UART protocol shall support deterministic state upload and target-angle return with bounded timeout behavior.

Verification

1. Scan the I2C bus and verify that the OLED and both `INA219` devices respond at their intended addresses while the mux remains selectable.
2. Disconnect or pause the Raspberry Pi during active operation and verify that the STM32 transitions to a safe stop or hold mode.
3. Timestamp sent and received UART frames and confirm that the STM32-to-Pi-to-STM32 loop remains within the chosen control-period budget.

2.5 Power Subsystem

The current top-level design uses two external supplies. A 12 V adapter powers the motor branch and both `A4988` drivers. A separate 5 V rail powers the Raspberry Pi and the STM32 logic branch. The solar panel is not used as the controller power source; instead it is treated as a measured source under test. Its output flows through `INA219_1` into a dummy load so the system can measure generated electrical power independently of the control electronics. The external motor branch is monitored by `INA219_2`. This arrangement makes the energy accounting much clearer during verification and avoids unstable bring-up caused by trying to self-power the logic from a small test panel.

The same measured power paths support a virtualized energy-management layer inside the STM32. Rather than charging a real battery in the present prototype, the firmware maintains a virtual stored-energy variable E_t that represents the energy that would be available in a rechargeable buffer if panel generation were treated as charging input and motor consumption were treated as discharge. This state may be initialized at startup to E_0, then updated at a fixed interval, currently intended to be once per second.

The charging term is computed from the solar-panel branch using the panel-side monitor:

$$P_{\text{solar}}(t) = V_{\text{panel}}(t) I_{\text{panel}}(t) \tag{1}$$

The discharge term is computed from the externally powered motor branch:

$$P_{\text{motor}}(t) = V_{\text{motor}}(t) I_{\text{motor}}(t) \tag{2}$$

The STM32 then updates the virtual energy state according to

$$E_{t+1} = E_t + \left(P_{\text{solar}}(t) - P_{\text{motor}}(t)\right) \Delta t \tag{3}$$

with $\Delta t = 1$ s in the current implementation plan. If E_t is increasing, the recent operating window is net positive in energy terms. If E_t is decreasing, the system is spending energy faster than it is effectively recovering it. This gives the controller a compact power-aware state for deciding whether a proposed movement is worth its energy cost.

The PCB should expose clear test points for `12V`, `5V`, `3V3`, `GND`, I2C, UART, and motor supply nodes. Strong-current and weak-signal regions should be laid out separately so motor noise does not corrupt the sensor bus.

Subsystem requirements

1. The logic supply shall remain within tolerance during worst-case motor startup and reversal events.
2. The panel path and motor path shall both provide telemetry sufficient for net-power calculation during experiments.
3. The STM32 shall maintain and log the virtual energy state E_t at a fixed update period during experiments.
4. The 12 V motor branch and 5 V logic branch shall be partitioned and protected against reverse polarity, short circuit, and wiring faults.

Verification

1. Monitor the 5 V rail during aggressive motion tests and verify that no brown-out or logic reset occurs.
2. Compare sensor-based power estimates against external instrument measurements under several light and load conditions.
3. Run a scripted experiment with known intervals of panel illumination and motor actuation, then verify that the logged E_t curve increases during net-positive intervals and decreases during net-negative intervals.
4. Validate the staged bring-up sequence by checking each rail and interface at dedicated test points before full electromechanical integration.

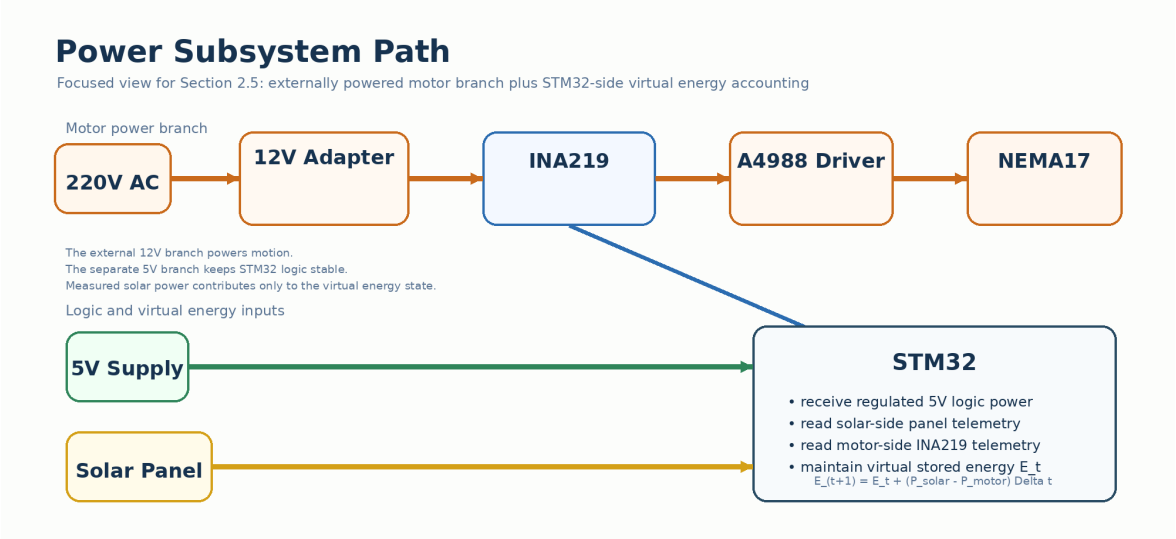


Figure 5: Power-subsystem view centered on the two external rails and the measured energy paths. The upper branch shows mains-powered actuation from `220V AC` through the `12V` adapter, motor-side `INA219`, `A4988` driver, and `NEMA17` motor. The lower branch shows the separate `5V` logic supply into the STM32 and the solar-panel test path used as virtual charging input for the STM32's energy-accounting state.

2.6 User Interface and One-Button Operation

The local user interface is intentionally simple. A single button initiates startup. A 0.96-inch I2C OLED displays operating mode, panel voltage and current, motor branch telemetry, and current yaw/pitch state. This allows the system to be demonstrated without a laptop connected to the board.

The intended startup sequence is:

1. Power-on and self-test
2. Home the yaw and pitch axes
3. Start sensor polling and telemetry refresh
4. Enter normal tracking or energy-saving hold mode

This directly supports the project requirement that the final platform behave as an autonomous device after one button press.

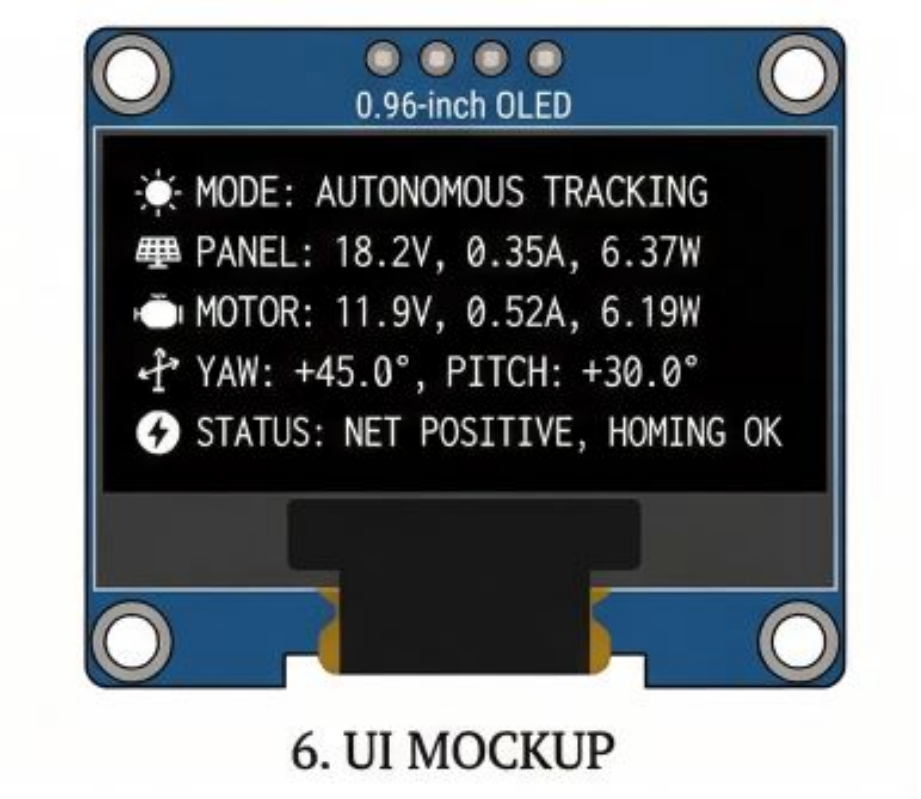


Figure 6: OLED display placeholder. Replace with a screen mockup showing mode, panel power, motor power, yaw angle, pitch angle, and any fault or hold-state indicator.

2.7 Tolerance Analysis

One critical tolerance is panel pointing error. For a flat photovoltaic panel under approximately direct illumination, the captured power can be approximated by

$$P(\theta) = P_{\max} \cos(\theta) \tag{4}$$

where θ is the misalignment between the panel normal and the dominant incoming light direction.

For a 5° error,

$$\frac{P(5^\circ)}{P_{\max}} = \cos(5^\circ) \approx 0.996 \tag{5}$$

which implies only about a 0.4% reduction in ideal captured power. For a 15° error,

$$\cos(15^\circ) \approx 0.966 \tag{6}$$

which corresponds to about a 3.4% reduction.

This result is important for the system-level control decision. Very small angle errors do not justify large motor actions because the recovered solar power may be tiny compared with the cost of moving and holding the panel. The controller should therefore apply a motion deadband and only command new target angles when the expected gain is large enough to overcome the measured motor cost. The exact deadband can be tuned experimentally using the two `INA219` channels and the dummy-load measurement setup.

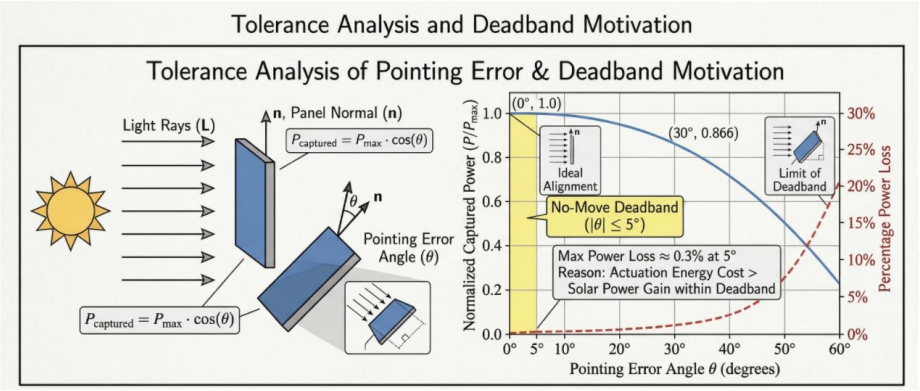


Figure 7: Tolerance-analysis placeholder. Replace with a cosine-loss curve or geometry sketch showing how pointing error affects captured panel power and motivates a no-move deadband.

3 Cost

Table 1 summarizes the current estimated project cost based on the hardware system of record in `docs/hardware/`. The design remains below the 1500 RMB budget target and leaves reserve for PCB iteration and mechanical integration.

Table 1: Estimated hardware cost.

Item	Qty	Unit (RMB)	Subtotal (RMB)
STM32F407ZGT6 development board	1	80	80
NEMA17 stepper motor	2	80	160
A4988 stepper driver	2	20	40
INA219 sensor module	2	18	36
BH1750 light sensor	16	8	128
I2C mux module or IC	1	20	20
0.96 in I2C OLED	1	20	20
12 V 5 A adapter	1	45	45
5 V logic supply	1	20	20
Push button, connectors, test points	1 set	35	35
Solar panel	1	80	80
Dummy-load parts	1 lot	25	25
PCB fabrication and passive parts	1 lot	180	180
Mechanical couplings and brackets	1 lot	150	150
Total			1019

The current estimate leaves about 481 RMB of reserve under the 1500 RMB budget cap.

4 Schedule

Table 2 gives a draft milestone schedule aligned with the current team scope split.

Table 2: Draft schedule and ownership.

Week	Main task	Owner
1	Freeze requirements, interfaces, and top-level architecture	Whole team
2	Finalize component selection and cost model	Ziru, Yifei, Minghao
3	Define UART protocol and STM32 pin allocation	Minghao, Ziru
4	Draft schematic and power partitioning plan	Minghao, Yifei
5	Bring up STM32 sensor reads on I2C	Minghao
6	Bring up A4988 motor control and homing logic	Minghao, Ziru
7	Integrate Raspberry Pi UART communication and logging	Minghao
8	Prototype inference pipeline and ONNX deployment	Minghao
9	Build and align dual-axis mechanical frame	Yikai, Whole team
10	Integrate electronics with frame and verify power rails	Whole team
11	Run tracking, power, and net-gain experiments	Whole team
12	Finish document, figures, and final demonstration prep	Whole team

5 Ethics and Safety

5.1 Ethics

The project must present energy-saving claims honestly. It would be misleading to claim that a moving solar tracker always improves performance. The core purpose of this project is to test when motion helps and when motion hurts net energy gain. The team should therefore report both positive and negative cases, clearly distinguish measured net gain from theoretical geometric gain, and state that the present platform is an externally powered test system rather than a self-powered photovoltaic node.

The project should also communicate the RL interface honestly. The online inference path uses only light-ring readings and current yaw/pitch state, while richer electrical telemetry is reserved for logging, training, and evaluation. Hiding that distinction would overstate the capability of the deployed model. These choices align with the IEEE Code of Ethics, especially the responsibility to be honest and realistic in stating claims and to improve understanding of technology and its appropriate application.

5.2 Safety

The project involves rotating mechanical parts, bench power supplies, exposed wiring during bring-up, and a mixed-signal PCB carrying both motor and logic paths. Key safety measures include:

- homing switches and software travel limits to prevent over-rotation,
- current-limited `A4988` configuration and protected motor wiring,
- mechanically secure mounting of the panel and moving frame,

- insulated and strain-relieved power wiring with a controlled common ground,
- staged low-voltage bench testing before full electromechanical operation,
- clear emergency-stop or power-disconnect procedure during motion tests.

During verification, the team should keep hands clear of pinch points, test on a stable bench fixture, and confirm that no rail is energized beyond its intended rating before connecting the Raspberry Pi, STM32, or sensors.

6 References

1. IEEE, "IEEE Code of Ethics," [Online]. Available: `https://www.ieee.org/about/corporate/governance/p7-8.html`
2. STMicroelectronics, "STM32F407xx advanced Arm-based 32-bit MCUs," datasheet.
3. Texas Instruments, "INA219 zerō-drift, bidirectional current/power monitor with I^2C interface," datasheet.
4. Texas Instruments, "DRV/A4988-compatible stepper motor driver module documentation," vendor datasheet or carrier documentation.
5. Rohm Semiconductor, "BH1750FVI digital ambient light sensor IC," datasheet.
6. `TODO`: Add the final I2C mux part number, selected NEMA17 motor datasheet, OLED module reference, and the exact external 12 V and 5 V supply references used in the final build.