

Solar Tracker System Study
Technical Executive Study for a Dual-Axis Solar Tracker
Prepared by Engineer Mahmoud Ali Hassan
Supervised by Dr. Marcel Salameh
Directed to Venusian Chinese Industrial Investment Company

Table of Contents

- 1. Introduction
- 2. Objectives and Scope
- 3. Detailed Component Analysis and Functions
 - 3.1 Light Sensors (LDR)
 - 3.2 Actuators (Linear & Servo)
 - 3.3 Embedded Control Unit
 - 3.4 MPPT Module
- 4. Signal–Flow Block Diagram
- 5. Electrical Panel & Wiring Diagrams
- 6. Functional Diagrams and UML Models
 - 6.1 Use Case Diagram
 - 6.2 Sequence Diagram
 - 6.3 Activity Diagram
- 7. MPPT Algorithm Implementation

8. System Simulation and Validation Example
9. HMI/SCADA Interface Design
10. Code Examples – Tracking & MPPT
11. Complete Control Program Code
12. Communication Configuration (Modbus/CAN)
13. Cybersecurity & Data Integrity Policies
14. Implementation Plan & Detailed Schedule
15. Conclusion
16. Appendices

1. Introduction

English: This report delivers a comprehensive technical executive study of a dual-axis solar tracker system that optimizes photovoltaic panel orientation throughout the day. It integrates light-dependent resistors (LDRs) for sun-position sensing, linear actuators or servo motors for panel rotation, and an embedded MPPT algorithm for maximum energy harvest.

German: Dieser Bericht liefert eine umfassende technische Ausführungsstudie eines zweiachsigen Solar-Trackers, der die Ausrichtung von Photovoltaikmodulen im Tagesverlauf optimiert. Er integriert lichtabhängige Widerstände (LDR) zur Sonnenstandserfassung, Linearantriebe oder Servomotoren für die Modulrotation und einen eingebetteten MPPT-Algorithmus zur maximalen Energiegewinnung.

2. Objectives and Scope

English:

Objectives

- Achieve continuous sun-tracking on two axes to maximize daily energy yield.
- Incorporate an embedded MPPT routine for real-time power optimization.
- Provide robust control hardware and a user-friendly monitoring interface.

Scope

- Selection and specification of LDR sensors, actuators, and control electronics.
- Development of MPPT firmware and real-time tracking logic.
- Design of control cabinet wiring and communication network.
- Implementation of cybersecurity measures and data backup procedures.

German:

Ziele

- Kontinuierliches Verfolgen der Sonne auf zwei Achsen zur Maximierung des täglichen Energieertrags.
- Integration einer eingebetteten MPPT-Routine zur Echtzeit-Leistungsoptimierung.
- Bereitstellung robuster Steuerungshardware und einer benutzerfreundlichen Überwachungsschnittstelle.

Umfang

- Auswahl und Spezifikation von LDR-Sensoren, Aktoren und Steuerelektronik.
- Entwicklung von MPPT-Firmware und Echtzeit-Tracking-Logik.
- Auslegung der Schaltschrankverkabelung und des Kommunikationsnetzwerks.
- Umsetzung von Cybersicherheitsmaßnahmen und Datensicherung.

3. Detailed Component Analysis and Functions

3.1 Light Sensors (LDR)

English:

| Component | Function | Interface |
|-----------------|--------------------------------------|-----------|
| ----- | ----- | ----- |
| LDR Array (4×) | Differential light intensity sensing | Analog In |
| Temperature RTD | Ambient temperature compensation | 4–20 mA |

German:

| Komponente | Funktion | Schnittstelle |
|----------------|-----------------------------------|---------------|
| ----- | ----- | ----- |
| LDR-Array (4×) | Differenzielle Lichtstärkemessung | Analog In |
| RTD-Temperatur | Umgebungstemperaturkompensation | 4–20 mA |

3.2 Actuators (Linear & Servo)

English:

| Actuator Type | Function | Drive Interface |
|-----------------|---------------------------|-----------------|
| ----- | ----- | ----- |
| Linear Actuator | Elevation axis adjustment | PWM / 0–10 V |
| Servo Motor | Azimuth axis rotation | PWM (Standard) |

German:

| Aktortyp | Funktion | Ansteuerschnittstelle |
|---------------|-------------------------|-----------------------|
| ----- | ----- | ----- |
| Linearantrieb | Höhenachsen-verstellung | PWM / 0–10 V |
| Servomotor | Azimet-achsen-drehung | PWM (Standard) |

3.3 Embedded Control Unit

English:

- Microcontroller: ARM Cortex-M4 (STM32F4 series)
- I/O: 8×Analog In, 4×PWM Out, 6×Digital In/Out
- Communications: CAN 2.0, Modbus RTU over RS-485, Ethernet (optional)
- Real-time OS: FreeRTOS for deterministic scheduling

German:

- Mikrocontroller: ARM Cortex-M4 (STM32F4-Serie)
- I/O: 8×Analog In, 4×PWM Out, 6×Digital In/Out
- Kommunikation: CAN 2.0, Modbus RTU über RS-485, Ethernet (optional)
- Echtzeit-OS: FreeRTOS für deterministisches Scheduling

3.4 MPPT Module

English:

- Algorithm: Perturb & Observe with adaptive step size
- Sampling: DC bus voltage and panel current at 1 kHz
- Outputs: Duty cycle for buck-boost converter control

German:

- Algorithmus: Perturb & Observe mit adaptiver Schrittweite
- Abtastrate: DC-Bus-Spannung und Modulstrom bei 1 kHz
- Ausgänge: Tastverhältnis für Buck-Boost-Wandlersteuerung

4. Signal–Flow Block Diagram

English:

``plaintext

[LDR Array] --> Analog In --> [MCU FreeRTOS]

| \

[Temp RTD] --> 4–20mA In -----+ > MPPT Routine --> PWM Out --> [Power Converter]

| /

[Azimuth Servo] <-- PWM Out <--+

[Elevation Linear] <-- PWM Out-+

German:

[LDR-Array] --> Analog In --> [MCU FreeRTOS]

| \

[RTD-Temperatur] --> 4–20mA In --+ > MPPT-Routine --> PWM Out --> [Leistungswandler]

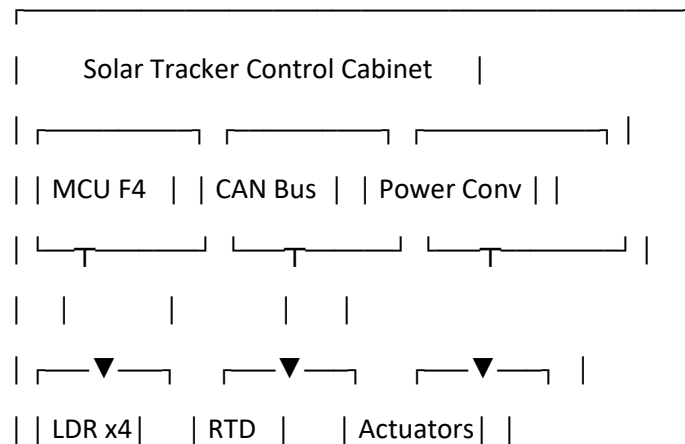
| /

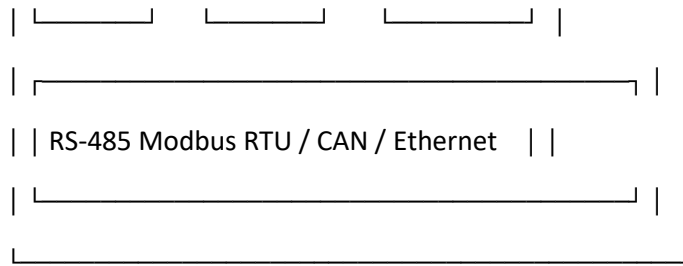
[Azimut-Servo] <-- PWM Out <----+

[Höhen-Linear] <-- PWM Out -----+

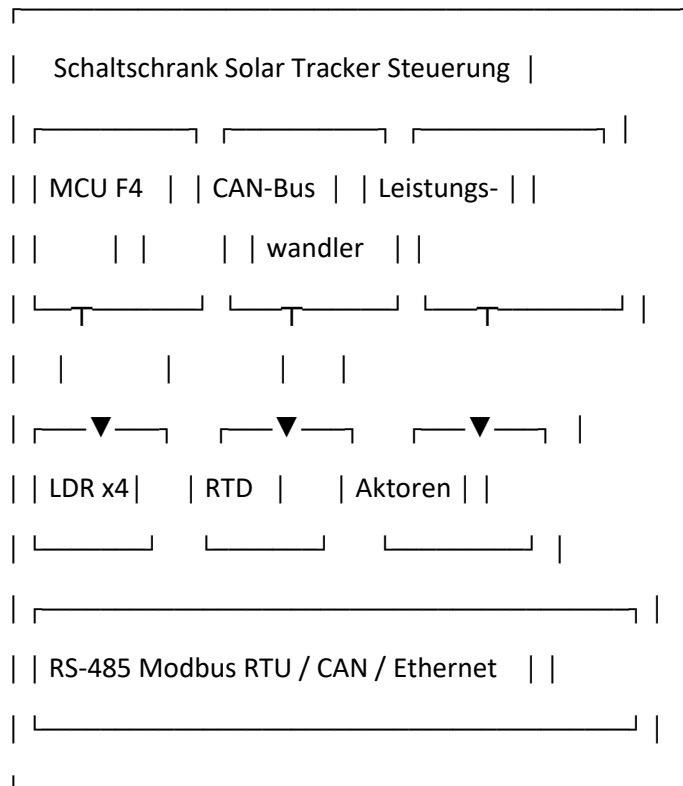
5. Electrical Panel & Wiring Diagrams

English:





German:



6. Functional Diagrams and UML Models

6.1 Use Case Diagram

English:

```
usecaseDiagram
```

```
actor "Operator" as Op
```

```
actor "Maintenance" as Maint
```

actor "Monitor" as Mon

Op --> (Start Tracker)

Op --> (Set Parameters)

Maint --> (Calibration)

Mon --> (View Status)

German:

usecaseDiagram

actor "Bediener" as Op

actor "Wartung" as Maint

actor "Überwacher" as Mon

Op --> (Tracker starten)

Op --> (Parameter einstellen)

Maint --> (Kalibrierung)

Mon --> (Status anzeigen)

6.2 Sequence Diagram

English:

sequenceDiagram

participant UI

participant MCU

participant Sensor as LDR

participant Actuator

UI->>MCU: Start Tracking

MCU->>Sensor: Read Light Levels

Sensor-->>MCU: Light Data

MCU->>MCU: Compute Az/EI Angles

MCU->>Actuator: Move Motors

Actuator-->>MCU: Ack

MCU->>UI: Update Dashboard

German:

sequenceDiagram

participant UI

participant MCU

participant Sensor as LDR

participant Aktor

UI->>MCU: Tracking starten

MCU->>Sensor: Lichtwerte lesen

Sensor-->>MCU: Lichtdaten

MCU->>MCU: Az/EI Winkel berechnen

MCU->>Aktor: Motoren bewegen

Aktor-->>MCU: Bestätigung

MCU->>UI: Dashboard aktualisieren

6.3 Activity Diagram

English:

flowchart TD

A[Power On] --> B{Read Sensors}

B --> C[LDR Differential]

B --> D[RTD Temp]

C & D --> E[Compute Sun Vector]

E --> F[MPPT Routine]
F --> G[Generate PWM]
G --> H[Drive Actuators]
H --> I[Update UI & Log]
I --> B

German:

flowchart TD

A[Power On] --> B{Sensoren lesen}
B --> C[LDR-Differenz]
B --> D[RTD-Temperatur]
C & D --> E[Sonnenvektor berechnen]
E --> F[MPPT-Routine]
F --> G[PWM erzeugen]
G --> H[Aktoren ansteuern]
H --> I[UI & Log aktualisieren]
I --> B

7. MPPT Algorithm Implementation

English:

- Initialize PV voltage and current sampling at 1 kHz.
- Apply Perturb & Observe: adjust converter duty by ΔP sign.
- Adapt step size based on power gradient to reduce oscillation.
- Limit duty cycle between 0–100 % and handle boundary conditions.

German:

- Initialisieren der PV-Spannungs- und Stromabtastung mit 1 kHz.
- Anwenden von Perturb & Observe: Anpassung des Wandlertastverhältnisses nach ΔP -Vorzeichen.
- Anpassung der Schrittweite basierend auf der Leistungsdifferenz zur Reduzierung von Oszillationen.

- Begrenzung des Tastverhältnisses auf 0–100 % und Behandlung von Randbedingungen.

8. System Simulation and Validation Example

English:

- Use MATLAB/Simulink: model PV module, DC–DC converter, and tracker kinematics.
- Simulate a clear-sky day: compare fixed vs. tracked energy yield.
- Validate MPPT response to cloud transients by injecting irradiance step changes.

German:

- Verwenden von MATLAB/Simulink: Modellierung von PV-Modul, DC–DC-Wandler und Tracker-Kinematik.
- Simulation eines klaren Tages: Vergleich des Energieertrags feststehend vs. nachgeführt.
- Validierung der MPPT-Reaktion auf Wolkentransienten durch Einspeisen von Strahlungsänderungen.

9. HMI/SCADA Interface Design

9.1 Dashboard Screen

English:

- Real-time sun-position vector display (azimuth/elevation gauges).
- Instantaneous panel power and energy yield graphs.
- MPPT status indicator and converter duty slider.

German:

- Anzeige des Sonnenpositionsvektors in Echtzeit (Azimut-/Elevationanzeigen).
- Graphen für Momentanleistung und Energieertrag des Moduls.
- MPPT-Statusanzeige und Schieberegler für Wandlertastverhältnis.

9.2 Trend & Data Logging Screen

English:

- 24-hour plots for irradiance, panel current, and voltage.
- CSV export and remote query via Modbus/CAN.

German:

- 24-Stunden-Diagramme für Einstrahlung, Modulstrom und -spannung.
- CSV-Export und Remote-Abfrage über Modbus/CAN.

9.3 Alerts & Maintenance Screen

English:

- Threshold alarms (overcurrent, overtemperature, mechanical stall).
- Acknowledge and clear options with timestamped log.

German:

- Schwellenalarme (Überstrom, Übertemperatur, mechanischer Stillstand).
- Bestätigungs- und Löschoptionen mit Zeitstempel-Log.

10. Code Examples – Tracking & MPPT

10.1 Pseudocode for Sun-Vector Calculation

:

```
// Compute Sun Azimuth & Elevation
read LDR[4];
delta_x = LDR[0] - LDR[1];
delta_y = LDR[2] - LDR[3];
azimuth  = atan2(delta_y, delta_x);
elevation = f(east_west_avg, temp_comp);
```

10.2 MPPT Perturb & Observe Routine

:

```
oldP = Vpv * Ipv;
Vpv += step * direction;
delay(sample_interval);
Ipv = readCurrent();
```

```

Vpv = readVoltage();
newP = Vpv * Ipv;
if(newP < oldP) direction = -direction;
oldP = newP;
duty += direction * step;
duty = clamp(duty, 0, 100);

```

11. Complete Control Program Code

11.1 Main Control Loop (FreeRTOS Task)

:

```

void TrackerTask(void *pvParameters) {
    while(1) {
        readSensors();
        computeSunVector();
        mpptRoutine();
        updateActuators();
        logData();
        vTaskDelay(pdMS_TO_TICKS(20));
    }
}

```

11.2 Peripheral Initialization

```

void initPeripherals() {
    ADC_Init(LDR_CHANNELS, RTD_CHANNEL);
    PWM_Init(AZIMUTH_PIN, ELEVATION_PIN);
}

```

```

CAN_Init(500000);

ModbusRTU_Init(9600);

}

```

12. Communication Configuration (Modbus/CAN)

```

{
  "device": "SolarTracker-MCU",
  "protocols": ["ModbusRTU", "CAN2.0"],
  "modbus": {"baud":9600,"id":5},
  "can":  {"bitrate":500000,"id":0x45}
}

```

13. Cybersecurity & Data Integrity Policies

English:

- Secure boot and firmware signing using SHA-256.
- Encrypted Modbus messages (AES-128) and CAN-auth with HMAC.
- Periodic backup of configuration and log data to external flash.

German:

- Secure Boot und Firmware-Signierung mit SHA-256.
- Verschlüsselte Modbus-Kommunikation (AES-128) und CAN-Auth mit HMAC.
- Regelmäßige Sicherung von Konfigurations- und Logdaten auf externen Flash.

14. Implementation Plan & Detailed Schedule

English:

| Phase | Duration | Deliverables |
|--------------------------------|----------|--------------------------------------|
| 1. Feasibility & Design | 2 weeks | System architecture, component specs |
| 2. Prototype Assembly | 3 weeks | Sensor & actuator bench prototype |
| 3. Firmware & MPPT Development | 4 weeks | Tracking & MPPT firmware |
| 4. Simulation & Validation | 2 weeks | Simulink reports, yield comparison |

| Phase | Duration | Deliverables |
|---------------------------|----------|------------------------------------|
| 5. Control Cabinet Build | 1 week | Cabinet wiring & integration |
| 6. Field Testing & Tuning | 2 weeks | Performance logs, algorithm tuning |

German:

| Phase | Dauer | Ergebnisse |
|---------------------------------|----------|--|
| 1. Machbarkeitsstudie & Design | 2 Wochen | Systemarchitektur, Komponenten-spezifikationen |
| 2. Prototypenaufbau | 3 Wochen | Sensor- & Aktor-Bench-Prototyp |
| 3. Firmware- & MPPT-Entwicklung | 4 Wochen | Tracking- & MPPT-Firmware |
| 4. Simulation & Validierung | 2 Wochen | Simulink-Berichte, Ertragsvergleich |
| 5. Schaltschrankbau | 1 Woche | Schaltschrankverkabelung & Integration |
| 6. Feldtests & Feinabstimmung | 2 Wochen | Leistungsprotokolle, Algorithmus-Feintuning |

15. Conclusion

English: This study outlines a robust dual-axis solar tracking system integrating LDR sensors, linear actuators or servos, and an embedded MPPT algorithm. The architecture ensures maximum energy harvest, real-time control, and secure data handling. The proposed design meets industrial standards and is tailored for Venusian Chinese Industrial Investment Company's renewable energy portfolio.

German: Diese Studie skizziert ein robustes zweiachsiges Solar-Tracking-System mit LDR-Sensoren, Linearantrieben oder Servos und einem eingebetteten MPPT-Algorithmus. Die Architektur gewährleistet maximale Energiegewinnung, Echtzeitsteuerung und sichere Datenverarbeitung. Das vorgeschlagene Design erfüllt Industriestandards und ist auf das Portfolio der Venusian Chinese Industrial Investment Company im Bereich erneuerbarer Energien zugeschnitten.

16. Appendices

English:

- Appendix A: Pinout Diagram of MCU and Sensors
- Appendix B: Detailed Irradiance vs. Yield Charts (CSV)
- Appendix C: Firmware Flowcharts and State Machines
- Appendix D: Sample Field Test Log Template

- Appendix E: User Manual for Control Panel & SCADA Interface

German:

- Anhang A: Pinbelegung von MCU und Sensoren
- Anhang B: Detaillierte Diagramme Einstrahlung vs. Ertrag (CSV)
- Anhang C: Firmware-Flussdiagramme und Zustandsautomaten
- Anhang D: Vorlage für Feldtestprotokolle
- Anhang E: Benutzerhandbuch für Schaltschrank & SCADA-Schnittstelle