Greenhouse Environmental Control (Siemens TIA) – Detailed Design Report

1. Project Overview

Goal: Maintain optimal temperature, humidity, and light in a greenhouse using a Siemens PLC programmed primarily in **Structured Text (ST)** with **minimal Ladder (LAD)** just for orchestration and I/O mapping.

Core idea:

- A **Function (FC)** scales raw analog inputs into engineering units.
- A **Function Block (FB)** makes decisions (fan/heater/humidifier/grow light) using setpoints, hysteresis, and optional minimum on/off times.
- A **Global Data Block (DB)** holds setpoints, scaled values, enables, timing parameters, and output states (HMI-friendly).
- **OB1 (LAD)** only calls the FC/FB and maps outputs to the digital outputs.

---

2. System Architecture

2.1 Hardware (example)

- **PLC:** Siemens S7-1200 or S7-1500 (any CPU with required AI/DI/DO).
- **Analog Inputs (AI):**
    - Temp sensor (e.g., 0–10 V or 4–20 mA) → IW64
    - Humidity sensor → IW66
    - Light sensor (lux or irradiance) → IW68
- **Digital Outputs (DO):**
    - Fan → Q0.0
    - Heater → Q0.1
    - Humidifier → Q0.2
    - Grow Light → Q0.3

Adjust addresses to your actual I/O modules.

2.2 Software Objects

- **FC_SensorScaling (ST):** Converts 0…27648 raw values to engineering units (°C, %RH, Lux) via linear scaling.
- **FB_ClimateControl (ST):** Implements decision logic (hysteresis, min on/off) for actuators. Uses static memory and TON timers inside the Instance DB.
- **DB_Greenhouse (Global DB):** Setpoints, scaled values, feature enables, min on/off times, and actuator outputs.
- **OB1 (LAD):**

1. Call FC_SensorScaling
2. Call FB_ClimateControl (Instance DB_ClimateControl)
3. Map DB outputs to Q outputs

---

3. Functional Requirements

3.1 Inputs

- **Temperature (°C):** derived from TempRaw (IW)
- **Humidity (%RH):** derived from HumRaw (IW)
- **Light (Lux):** derived from LightRaw (IW)
- **Setpoints:** TempSet, HumSet, LightSet (editable at runtime via DB/HMI)
- **Feature enables:** EnTempCtrl, EnHumCtrl, EnLightCtrl
- **Min on/off times:** Per-actuator TIME parameters for anti–short-cycling

3.2 Outputs

- **Fan (BOOL)** – cooling/air exchange
- **Heater (BOOL)** – heating
- **Humidifier (BOOL)** – moisture addition
- **GrowLight (BOOL)** – supplemental lighting

3.3 Control Strategy

- **Temperature:** "heat or cool" with deadband
  - If TempValue > TempSet + TempHyst → Fan ON, Heater OFF
  - If TempValue < TempSet - TempHyst → Heater ON, Fan OFF
  - Within deadband → both OFF
- **Humidity:** "humidify if low" with deadband
  - If HumValue < HumSet - HumHyst → Humidifier ON
  - If HumValue ≥ HumSet → Humidifier OFF
  - Within band → hold last request (prevents chattering)
- **Light:** threshold
  - If LightValue < LightSet → GrowLight ON else OFF
- **Min On/Off times:** Optional timer gating per actuator (TON instances in FB) to meet minimum run/rest durations.

---

4. Detailed Design

4.1 Scaling Function – FC_SensorScaling (ST)

Purpose: Normalize sensor inputs to meaningful engineering units.
Inputs: TempRaw, HumRaw, LightRaw, optional TempMaxEU, HumMaxEU, LightMaxEU.
Outputs: TempValue, HumValue, LightValue.

Logic:
EngUnit := (REAL(Raw) / 27648.0) * SpanEU

- Typical spans: Temp 0–50 °C, Humidity 0–100 %RH, Light 0–1000 Lux (customizable).

Notes:

- Keep math isolated and re-usable (any sensor/channel can reuse this FC).
- Optional checks (e.g., clamp negative raw, handle faulty sensor) can be added here.

4.2 Control Function Block – FB_ClimateControl (ST)

Purpose: Decide actuator states based on scaled values and setpoints with hysteresis and min on/off.

Variable scopes in the FB:

- **VAR_INPUT:** TempValue, HumValue, LightValue, TempSet, HumSet, LightSet, enables, min on/off TIME parameters.
- **VAR_OUTPUT:** Fan, Heater, Humidifier, GrowLight.
- **VAR (Static):**
  - Hysteresis values: TempHyst (°C), HumHyst (%RH)
  - Request latches: ReqFan, ReqHeater, ReqHumid, ReqLight
  - TON timer instances: tOn*, tOff* per actuator
  - Latched states: FanStateLatched, HeaterStateLatched, HumidStateLatched, LightStateLatched
  - These persist cycle-to-cycle in the **Instance DB**.
- **VAR_TEMP:** Scratch booleans WantFan, WantHeater, WantHumid, WantLight.

Execution flow:

1. **Raw decision** per variable (apply hysteresis logic).
2. **Store requests** into static Req*.
3. **Apply min on/off gating** with TON timers to avoid short cycling.
4. **Drive outputs**: Output := LatchedState AND Request.

Why this pattern?

- Separates **intent** (Want/Req) from **permission** (min-time gates + latched state), producing stable outputs.

4.3 Global Data Block – DB_Greenhouse

Role: Single source of truth for HMI and diagnostics.

Suggested tags:

- **Setpoints:** TempSet: REAL := 25.0, HumSet: REAL := 60.0, LightSet: REAL := 500.0
- **Scaled values (from FC):** TempValue, HumValue, LightValue
- **Enables:** EnTempCtrl := TRUE, EnHumCtrl := TRUE, EnLightCtrl := TRUE
- **Min times:** TIME tags (e.g., MinOnSec_Heater := T#10S)
- **Outputs (from FB):** Fan, Heater, Humidifier, GrowLight

Keep all runtime-tunable parameters here to simplify HMI binding and commissioning.

4.4 OB1 – Main Organization Block (LAD, minimal)

Network 1: Call FC_SensorScaling

- Inputs: IW64, IW66, IW68
- Outputs: DB_Greenhouse.TempValue, HumValue, LightValue

Network 2: Call FB_ClimateControl with **Instance DB** DB_ClimateControl

- Map inputs from DB_Greenhouse (values, setpoints, enables, min times)
- Map outputs back to DB_Greenhouse (Fan/Heater/Humidifier/GrowLight)

Network 3: Physical I/O mapping

- Q0.0 := DB_Greenhouse.Fan
- Q0.1 := DB_Greenhouse.Heater
- Q0.2 := DB_Greenhouse.Humidifier
- Q0.3 := DB_Greenhouse.GrowLight

---

5. I/O List (example)

Signal	Address	Range / Type	Description
Temp Raw	IW64	0–27648 (INT)	Temperature AI
Hum Raw	IW66	0–27648 (INT)	Humidity AI
Light Raw	IW68	0–27648 (INT)	Light/Lux AI
Fan	Q0.0	BOOL	Fan contactor/relay
Heater	Q0.1	BOOL	Heater contactor

Signal	Address	Range / Type	Description
Humidifier	Q0.2	BOOL	Humidifier relay
Grow Light	Q0.3	BOOL	Lighting relay

Add inputs for **safety interlocks** (door, E-stop, overtemp thermostat) as DI and gate the coils in OB1.

---

6. Safety & Interlocks

Recommended interlocks (add in OB1 before writing to Qs):

- **Emergency Stop** (DI) → forces all outputs OFF.
- **Overtemperature mechanical thermostat** in series with the heater contactor.
- **Door interlock** for fan/venting if required.
- **Overcurrent** or **thermal trip** feedbacks for motors.

Electrical:

- Use properly rated contactors and overload protection.
- Separate low-voltage controls from power wiring.
- Provide manual override or maintenance mode as needed.

---

7. Commissioning Procedure

1. **Offline checks**
    - Verify I/O addresses and sensor wiring.
    - Confirm DB tags and default setpoints.
    - Build and download the PLC program.
2. **Online monitoring**
    - Watch DB_Greenhouse.TempValue/HumValue/LightValue while changing sensor inputs (simulate if needed).
    - Adjust setpoints and observe FB outputs while in **manual safe environment**.
3. **Actuator verification**
    - Toggle setpoints to force Fan/Heater/Humidifier/Light transitions.
    - Confirm min on/off times behave as configured.
    - Check physical outputs and interlocks work as expected.
4. **Final tuning**

- Adjust TempHyst, HumHyst, and min on/off TIMES to reduce cycling and protect equipment.

---

8. Test Plan & Acceptance Criteria

Functional tests:

- **Temp High:** Raise TempValue > TempSet + TempHyst → Fan ON, Heater OFF within one scan; min times respected.
- **Temp Low:** Lower TempValue < TempSet - TempHyst → Heater ON, Fan OFF; min times respected.
- **Temp Band:** TempSet - TempHyst ≤ TempValue ≤ TempSet + TempHyst → both OFF.
- **Humidity Low:** HumValue < HumSet - HumHyst → Humidifier ON; turns OFF once HumValue ≥ HumSet.
- **Light Low:** LightValue < LightSet → GrowLight ON; OFF when above setpoint.
- **Enables:** Disabling a control forces the related output OFF.
- **Interlocks:** E-stop or trip → all outputs OFF immediately (hardwired + software).

Acceptance criteria:

- No rapid cycling; min on/off timings honored.
- Setpoint changes reflect within one scan (subject to min-time gates).
- Interlocks override any command.

---

9. HMI Integration (optional but recommended)

Bind the following to HMI screens:

- **Inputs:** TempValue, HumValue, LightValue (numeric indicators + trends).
- **Setpoints:** TempSet, HumSet, LightSet (numeric entry).
- **Enables:** EnTempCtrl, EnHumCtrl, EnLightCtrl (switches).
- **Timers:** MinOnSec_*, MinOffSec_* (numeric entry with units).
- **Outputs:** Fan/Heater/Humidifier/GrowLight (status lamps).
- **Alarms:** E-stop, overtemp trip, sensor fault (see §10).

---

10. Diagnostics & Fault Handling (extensions)

Sensor plausibility checks (add to FC or a small "Alarm FB"):

- Clamp negative raw values.
- Flag **sensor fault** if raw is below/above realistic thresholds for N seconds.

- On sensor fault → disable relevant control and raise alarm bit for HMI.

Runtime logging (optional):

- Accumulate runtime counters per actuator.
- Store daily max/min for Temp/Hum/Light.

---

11. Parameter Tuning Guidelines

- **Temperature hysteresis (TempHyst)**: start at **2.0 °C**.
  - If cycling is frequent → increase slightly (e.g., 3–4 °C).
- **Humidity hysteresis (HumHyst)**: start at **5 %RH**.
  - Adjust based on actuator response and greenhouse volume.
- **Min On/Off times:**
  - **Heater:** longer minimums (e.g., 10–30 s) to protect elements.
  - **Fan:** short minimums (5–10 s) often sufficient.
  - **Humidifier/Light:** application dependent; lights often 0 s is fine.

---

12. Optional PID Temperature Control

To improve temperature stability, replace the hysteresis heat/cool logic with a **PID** (e.g., PID_Compact / FB41) driving a **modulating output** (valve/VFD) or a staged output. For discrete heater/fan, implement a **time-proportional control** window (e.g., 10–30 s window and duty-cycle the output). Keep humidity/light as threshold control.

---

13. Code Artifacts (summary)

- **FC_SensorScaling (ST):** Raw → Eng units (°C, %RH, Lux).
- **FB_ClimateControl (ST):**
  - VAR_INPUT: values, setpoints, enables, min times
  - VAR_OUTPUT: Fan, Heater, Humidifier, GrowLight
  - VAR (Static): hysteresis, requests, TON timers, latched states
  - VAR_TEMP: transient decision booleans
- **DB_Greenhouse (Global DB):** Parameters + runtime values.
- **OB1 (LAD):** FC call → FB call → Q mapping.

(You already have the full code from earlier messages; paste directly into TIA SCL blocks and create the DBs as specified.)

---

14. Versioning & Maintenance

- Keep code in a Git repo with folders:
  - /plc_src/FC_SensorScaling.scl
  - /plc_src/FB_ClimateControl.scl
  - /plc_src/OB1_LAD.png (screenshot of ladder networks)
  - /db/DB_Greenhouse.udt or export
  - /docs/ (this report, commissioning notes)
- Tag releases upon commissioning.
- Record all parameter changes (hysteresis, min times) during tuning.

---

15. Conclusion

This design cleanly separates **measurement (FC)**, **decision logic (FB)**, **parameters/state (DB)**, and **orchestration (OB1)**. It's production-friendly, HMI-ready, and easy to extend with PID, alarms, and data logging. With minimal Ladder and robust ST structure, it's ideal for both learning and real greenhouse control.