TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Daniel Poljanski 232074IACM

Modernization and Automation of a Reliability Tester for Industrial Fans

Master's thesis

Supervisor:	Kristina Vassiljeva
	PhD
Co-Supervisor:	David Santos Lopez
	M. Sc

Tallinn 2025

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Daniel Poljanski 232074IACM

Tööstusventilaatorite Töökindluse Testeri Moderniseerimine ja Automatiseerimine

Magistritöö

Juhendaja: Kristina Vassiljeva
PhD
Kaasjuhendaja: David Santos Lopez
M. Sc

Tallinn 2025

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Daniel Poljanski

12.05.2025

Abstract

This thesis focuses on the modernization and automation of an industrial fan reliability tester at an electronics company in Estonia. The original tester, which relied on separate manual control components, suffered from issues such as lack of remote access, unreliable data collection, inefficient temperature regulation, and absence of real-time error detection. The objective was to enhance testing efficiency and reliability by integrating a Programmable Logic Controller and developing a purpose-built control program.

The new system enables remote access, automates data storage in a centralized database, separates the control of fans and heaters for improved efficiency, and includes a permission system for issue detection. Results demonstrate improved operation convenience, reduced manual effort, and significant time savings. The thesis also includes analysis of PLC programming languages defined within IEC 61131-3 standard and examines Accelerated Life Testing practices, aligning the solution with industry standards such as IPC-9591.

The work addresses the main flaws of the original tester: lack of remote control, risk of data loss, inefficient energy management, and limited error analysis. The PLC-based system manages several test cabinets simultaneously and created graphical user interface allows intuitive control and real-time feedback, reducing operator workload. Workflow comparisons show that the new system saves approximately 39 hours annually per cabinet, making the testing process faster and more reliable. The company is now capable of conducting more advanced ALT tests and explore new testing settings.

This thesis is written in English and is 58 pages long, including 10 chapters, 20 figures and 4 tables.

Annotatsioon

Tööstusventilaatorite Töökindluse Testeri Moderniseerimine ja Automatiseerimine

Käesolev lõputöö keskendub tööstusliku ventilaatori töökindluse testija moderniseerimisele ja automatiseerimisele Eesti elektroonikafirmas. Esialgne tester, mis tugines eraldi manuaalsetele juhtimiskomponentidele, kannatas selliste probleemide all nagu kaugjuurdepääsu puudumine, ebausaldusväärne andmete kogumine, ebaefektiivne temperatuuri reguleerimine ja reaalajas vea tuvastamise puudumine. Eesmärk oli suurendada testimise tõhusust ja usaldusväärsust, integreerides programmeeritava loogilise kontrolleri ja töötades välja selleks otstarbeks loodud kontrolliprogrammi.

Uus süsteem võimaldab kaugjuurdepääsu, automatiseerib andmete salvestamist tsentraliseeritud andmebaasis, eraldab ventilaatorite ja kütteseadmete juhtimise tõhususe parandamiseks ning sisaldab lubade süsteemi probleemide tuvastamiseks. Tulemused näitavad paremat kasutusmugavust, väiksemat manuaalset tööd ja märkimisväärset ajasäästu. Lõputöö sisaldab ka IEC 61131-3 standardis määratletud PLC programmeerimiskeelte analüüsi ja uurib kiirendatud eluea testimise tavasid, viies lahenduse vastavusse tööstusstandarditega, nagu IPC-9591.

Töös käsitletakse algse testeri peamisi puudusi: kaugjuhtimise puudumine, andmete kadumise oht, ebaefektiivne energiahaldus ja piiratud veaanalüüs. PLC-põhine süsteem haldab korraga mitut katsekappi ning loodud graafiline kasutajaliides võimaldab intuitiivset juhtimist ja reaalajas tagasisidet, vähendades operaatori töökoormust. Töövoogude võrdlused näitavad, et uus süsteem säästab aastas umbes 39 tundi ühe kabineti kohta, muutes testimisprotsessi kiiremaks ja usaldusväärsemaks. Ettevõte on nüüd võimeline teostama keerukamaid ALT-katseid ja uurima uusi testimisseadistusi.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 58 leheküljel, 10 peatükki, 20 joonist, 4 tabelit.

List of abbreviations and terms

AC	Alternating Current
ADC	Analog-to-Digital Converter
ALT	Accelerated Life Testing
CFC	Continuous Function Chart
CNC	Computer Numerical Control
DAQ	Data Acquisition
DC	Direct Current
EUT	Equipment Under Test
FBD	Function Block Diagram
GVL	Global Variable List
HMI	Human-Machine Interface
IA	Department of Computer Systems
IDE	Integrated Development Environment
I/O	Input/output
LD	Ladder Diagram
MUX	Multiplexer
NFC	Near-field communication
PID	Proportional-Integral-Derivative
PLC	Programmable Logic Controller
PN-IO	PROFINET-I/O
POU	Program Organization Units
PROFINET	Process Field Network
PVL	Persistent Variable List
ST	Structured Text
TCP/IP	Transmission Control Protocol/Internet Protocol
V	Volts
Ω	Ohms

Table of contents

List of figures

List of tables

1 Introduction

Industrial fans play a critical role in various applications, ranging from ventilation systems to complex industrial processes and components. To ensure their reliability and durability, many companies, like fan manufacturers themselves and other industrial companies, that use these fans, perform various reliability tests, which often simulate harsh operating conditions and extended usage. This is done by companies to claim the expected lifespan of a fan or a product containing a fan to their customers and partners or to verify claims of their suppliers. At the electrical company in Estonia that the author of this thesis is working with, the existing fan reliability tester relies on a mostly manually operated setup. This tester uses some inefficient and unreliable control components like physical temperature controllers, and several manual counters for cycle tracking. In addition to that, it also has problems with suboptimal wiring and soldering solutions. Despite serving its fundamental purpose, the existing setup presents several limitations: the testing process cannot be remotely controlled, data collection and saving is entirely manual and can be lost in case of component failure or software glitch, and problem detection is almost non-existent and can pose safety issues.

The main goals of this master's thesis are to modernize and automate the fan tester by applying physical upgrades like integrating a Programmable Logic Controller (PLC) instead of the most previously used manual components, as well as to develop a dedicated PLC control program for automation of the process and finally, creating a graphical user interface (visualization) for easy and intuitive test control and real-time monitoring. This upgrade aims to resolve the identified issues by enabling remote process control and monitoring, automating data collection with centralized data storage, and implementing real-time error detection. These enhancements seek to improve test accuracy, reduce manual effort, and enhance operational efficiency, updating the system to meet modern industry standards.

The company's fan testing system is based on the principles of Accelerated Life Testing (ALT), which aims to speed up the aging process of devices by exposing them to environmental stress conditions such as elevated temperatures and repeated thermal and power cycling. The testing process follows industry guidelines, including the IPC-9591 standard, which defines reliability requirements for air-moving devices. While the current solution already aligns with these principles, the new system developed in this work will aim to be even better. It will offer improved flexibility and expanded functionality, enabling more efficient testing and allowing the exploration of new testing methods and strategies. Additionally, the system will be designed to integrate seamlessly into the company’s existing infrastructure and to work reliably with the current testing equipment.

The work is structured as follows: Chapter 2 explores ALT principles and their application in the company, referencing IPC-9591 and comparing internal practices with its suppliers. Chapter 3 introduces PLCs, their advantages, IEC 61131-3 standard and the choice of the programming language for the program. Chapter 4 details the current tester setup, while Chapter 5 analyzes its key issues. Chapter 6 proposes the new PLC-based solution, including system architecture and hardware upgrades. Chapter 7 describes the PLC program design, Chapter 8 covers the visualization interface, and Chapter 9 evaluates the results, focusing on workflow improvements and time savings. Chapter 10 summarizes the findings and contributions.

2 Accelerated Life Testing for Fan Reliability

This chapter describes how Accelerated Life Testing (ALT) is used by the company to evaluate the reliability of fans. It includes a general overview of the ALT method, reference to the IPC-9591 standard, and a comparison between supplier tests and the company's internal procedures.

2.1 Accelerated Life Testing

Accelerated Life Testing (ALT) is a reliability engineering technique used to estimate a product's lifespan by subjecting it to elevated stress conditions, such as increased temperature, voltage, vibration, or humidity to cause failures more rapidly than under normal operating conditions. This approach enables manufacturers to predict long-term performance and identify potential failure modes within a compressed timeframe, which is particularly valuable for components with inherently long service lives [1].

The fundamental principle of ALT involves applying stressors that accelerate the aging process of materials or components without introducing new failure mechanisms. By collecting failure data under these elevated conditions, engineers can model the relationship between stress and failure time, often using statistical models such as the Arrhenius, Eyring, or inverse power law models. These models facilitate the extrapolation of test results to predict performance under normal use conditions [2], [3].

2.2 Purpose of the Test from the Company's Perspective

The company develops and delivers a wide range of products and systems for external clients. For this reason, it is essential to provide reliable estimates for the expected lifetime of the product itself and its components. To obtain this data, accelerated life testing is performed under controlled conditions.

There are several scenarios where such testing becomes necessary:

- New Product Development: When launching a new product or prototype, the company must validate how long it can reliably operate before degradation or failure occurs.

- Component Verification: Many products include third-party components that are not manufactured in-house. For example, fans used in various systems are sourced from external suppliers. While these suppliers may provide their own lifetime estimates based on their own testing, the company cannot rely solely on those claims. To ensure quality and consistency, independent testing is conducted in-house.
- Supplier Evaluation and Design Changes: When introducing new suppliers, it is necessary to validate the reliability of their components before integrating them into production. Similarly, if an existing supplier makes design or material changes to a component, it must be retested to verify that the lifetime and performance remain within acceptable limits.

By performing its own accelerated tests, the company ensures that only validated and reliable components are used in its products, reducing the risk of failures in the field and leaving customers satisfied.

2.3 IPC-9591 Overview and Application

The reliability testing of fans conducted by the company is based on the industry standard specification IPC-9591, which sets requirements for the design, manufacturing, and testing of air moving devices (fans). This specification is widely used not only by the company itself, but also by several of its fan suppliers, making it a consistent and recognized system for lifetime testing.

Reliability Assessment Criteria

According to IPC-9591, a fan is considered to have failed if any of the following conditions are met:

- Rotational Speed Reduction: A drop of ≥15% from the original measured RPM.
- Increased Current Consumption: An increase of ≥15% from the original measured input current.
- Acoustic Noise Increase: More than a 3 dB increase in sound level under the same operating conditions.
- Electrical Malfunction: Any incorrect or erratic operation of the electronic interface.

- Physical Damage: Visible cracks in the housing, impeller, or guards.
- Lubrication Issues: Visible grease or oil leakage, or loose bearing seals.

Test Chamber Requirements

The standard requires that:

- The average air temperature in the test chamber must be used when reporting results.
- The air temperature at the inlet of each fan must be within ±3°C of the average chamber temperature.
- Fans must operate in free delivery conditions, and the chamber must be large enough to prevent airflow interference between test units.

Orientation Testing

All specified fan orientations must be tested. The supplier must provide test data for each orientation listed in the product datasheet. Unverified orientations cannot be claimed as acceptable.

Temperature Stress and Acceleration Factor

The standard defines a simplified acceleration model. An acceleration factor of 1.5 per every 10°C increase in test temperature is used to predict equivalent failure rates under normal conditions. This model assumes that failure mechanisms accelerate with increased temperature and allows high-temperature testing to approximate real-world performance.

Test Modes

Two modes of operation are supported:

- Continuous Operation
- Periodic Start-Stop Operation – includes a daily power-off cycle (typically 5 minutes), followed by a restart, to simulate grease degradation and cold-start conditions. Acoustic measurements can be taken during these events. [4]

2.4 Comparison with Supplier Testing Practices

Based on documentation provided by fan manufacturers, most suppliers generally follow the IPC-9591 standard. However, in many cases, certain test conditions are applied differently. For example, some suppliers do not use some acceleration factors defined in the standard, or their lifetime calculations deviate from the standard.

These differences can affect the final lifetime estimates given by the suppliers. Because of this, the company’s own internal testing, which follows a defined and consistent method, is very useful and important. It allows the company to verify the actual reliability of the components and ensures that all products meet internal quality standards, regardless of what is stated by the supplier.

3 Introduction to PLCs and Programming Language Selection

This section introduces Programmable Logic Controllers (PLCs) and their role in automating the fan testing system. It explains what PLCs are, why they are commonly used in industrial environments, and how they support reliable long-term operation. The section also outlines the IEC 61131-3 standard and its supported programming languages. Each language is briefly described, and the reasoning behind the language selected for this project is discussed in detail.

3.1 Definition of a PLC

A Programmable Logic Controller (PLC) is a specialized industrial computer used for automation of electromechanical processes. PLC is defined as a solid-state control system with a user-programmable memory for storing instructions to implement specific functions such as logic, sequencing, timing, counting, control of machines and processes [5]. In general, it is an industrial digital computer, adapted for control tasks in manufacturing or other industrial environments that require high reliability and real-time operation. A PLC continuously monitors the state of input devices (such as sensors and switches) and performs logic or mathematic operations in a cyclic manner to update output devices (such as actuators, motors and valves) accordingly [6]. PLC is a real-time system, meaning it must produce control outputs within a limited time in response to input conditions to avoid unintended operation. The fundamental purpose of a PLC is to automate industrial control: replacing hard-wired relays and timers with a single unit capable of performing the same control functions with better flexibility and reliability.

3.2 Advantages of PLC

PLCs gained popularity because they offer several key advantages over traditional hard-wired control, including:

Reliability: PLCs are designed to operate continuously in harsh industrial environments with minimal downtime. They have no mechanical wear-out parts (unlike relays with moving contacts) and are built to handle vibration, electrical noise, temperature extremes, and moisture. This robust design leads to high mean-time-between-failure. It is common for a PLC to run 24/7 in a factory for many years before requiring replacement. The reliability of PLC-based control minimizes unexpected plant shutdowns.

Flexibility: A major benefit of PLCs is the ease with which control logic can be modified. Changing the behaviour of a relay-based system required broad rewiring and physical changes, while a PLC-based system only requires modifying the software program. This means the same hardware controller can be reprogrammed for a wide range of tasks or updated as process requirements evolve. Reprogramming is much faster and less labour-intensive than rewiring, allowing quick implementation of improvements or new product recipes.

Modularity: PLC hardware is typically modular, meaning systems can be expanded or adapted by adding plug-in modules rather than replacing the entire controller. Most PLCs consist of a base chassis or rack in which different I/O modules (for digital/analog inputs and outputs), communication modules, and special function modules can be installed. It also simplifies maintenance: if one module fails (e.g. an output card), it can be swapped out without replacing the whole PLC.

Real-time operation: PLCs execute control tasks in real time with a scan cycle. They are optimized for rapid I/O sampling and logic execution, allowing them to respond to input changes within milliseconds. This real-time performance is very important for industrial control – for example, the moment a safety sensor is triggered or a part reaches a position, the PLC can instantly update outputs to stop a machine or trigger a device. PLCs run a continuous loop where they read inputs, execute the user program, and update outputs, all under a limited timeframe.

Together, these features make PLCs a core technology in modern industrial automation, known for their consistent performance, ease of use, and low total cost over long lifecycles [7], [8].

3.3 PLC Programming Standard IEC 61131-3

As PLC usage spread globally, the need for standardization in programming became needed – early PLCs from different vendors each had proprietary languages and tools. The IEC 61131-3 standard addresses this by defining a unified set of programming languages and a basic software model for PLCs. The main objective of IEC 61131-3 is to enable portability and consistency in PLC software: engineers should be able to learn a set of standard languages and use them on any compliant PLC, and potentially even reuse control programs between different platforms with minimal changes.

To achieve this, the standard specifies the syntax and semantics of five programming languages and defines rules for structuring PLC programs. It introduces the concept of Program Organization Units (POUs) – such as programs, function blocks, and functions – which are building blocks for PLC software architecture. It also standardizes data types (elementary types like BOOL, INT, REAL, etc., and structures) and how variables are declared and used, which helps in catching errors early (preventing type mismatches).

By providing vendor-neutral languages and structuring guidelines, IEC 61131-3 greatly improved the portability of PLC programs and the training of staff. A control engineer can move from one brand of PLC to another and still use the same IEC 61131-3 languages rather than learning an entirely new programming method. It also supported the development of independent programming environments (PLC Open editors, CODESYS, etc.) used to program different brands of PLCs. Importantly, the standard does not mandate that a PLC support all languages, but most major PLC systems support at least the core ones.

3.4 Supported Programming Languages

IEC 61131-3 specifies five programming languages for PLCs. Each has its strengths, and the standard allows mixing them (writing different parts of a project in different languages). The five languages are:

Ladder Diagram (LD): Ladder Diagram is a graphical programming language that represents the control logic in a format similar to an electrical schematic diagram. Visually, it is drawn as vertical rails (representing supply power) with horizontal “rungs” of logic connecting them, therefore looking like a ladder. Each rung typically equates to a logical statement: it consists of input conditions (contact symbols) on the left, which must all be logically true (closed) to energize the output coil on the right. The symbols for inputs and outputs in LD directly map to real I/O or internal bits. Ladder logic was designed to be intuitively understood by electricians and technicians, since it mirrors relay logic diagrams, they were already familiar with [9].

Advantages: Ladder is very visual and easy to follow for Boolean logic. It’s excellent for discrete control and interlocking logic. Many basic patterns have direct ladder implementations.

Disadvantages: Ladder can become bulky for complex arithmetic or data manipulation, representing mathematical formulas or is possible but not very convenient. The language also lacks high-level structuring beyond dividing into multiple ladders or subroutines, so large programs with thousands of rungs can be harder to maintain.

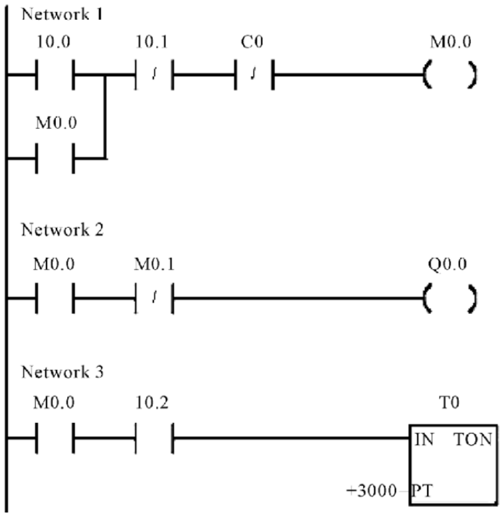


Figure 1. Ladder Diagram example [10]

Function Block Diagram (FBD): Function Block Diagram is a graphical language that models the program as a network of interconnected blocks, each block representing a functional element (such as a logic gate, a timer, a math operation, or an entire control algorithm). It is similar to drawing a signal flow or electronic circuit diagram: blocks have inputs and outputs, and lines (wires) connect outputs of one block to inputs of another. Complex control algorithms can be built by wiring together many function blocks.

Advantages: FBD promotes reusable, modular design – each function block can be tested and reused. It’s very good for parallel logic and multi-step computations on a single diagram. Many PLCs come with extensive libraries of function blocks that can be drag-and-dropped, speeding up development. It’s also ideal for complex control structures that would require many rungs in ladder.

Disadvantages: Its flexibility in layout can lead to inconsistent programming styles – each engineer might wire blocks differently, making it hard to standardize and read someone else’s FBD program. Large FBD programs may span multiple sheets, if not documented well, it can be tricky to follow signals across sheets, leading to confusion. Additionally, for very simple logic, FBD might be overkill compared to a simple ladder rung.

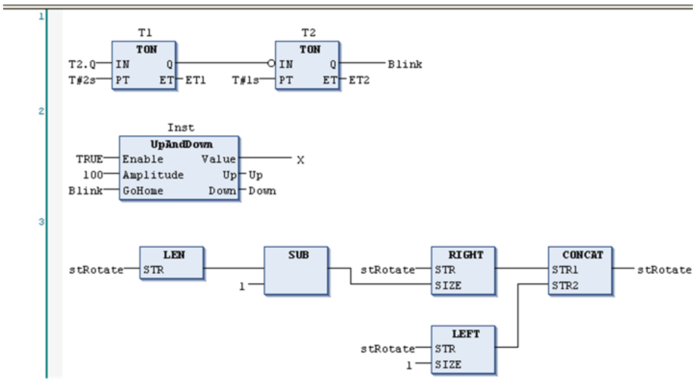
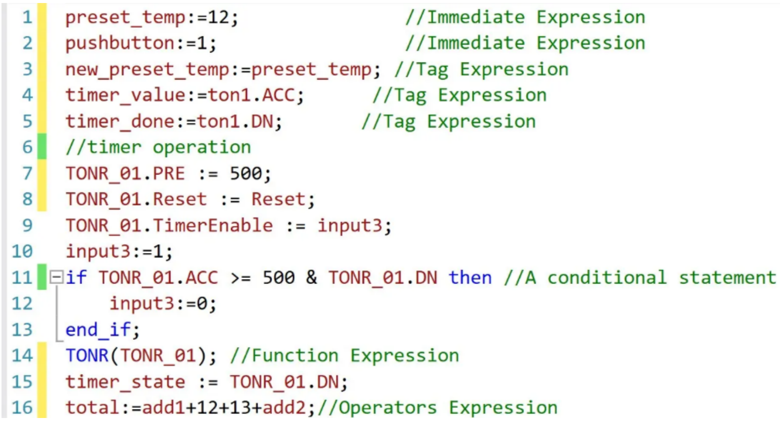


Figure 2. Functional Block Diagram Example [11]

Structured Text (ST): Structured Text is a high-level, textual programming language for PLCs. It resembles traditional programming languages like Pascal. An ST program is essentially a block of code with statements that can include assignments, conditional IF…THEN…ELSE, loops (FOR, WHILE), function calls, etc [12], [13].

Advantages: ST is very powerful for implementing complex algorithms, calculations, and array or table processing. It allows use of modern programming constructs: loops for repetitive operations, making it easy to iterate through an array of inputs, which would be tedious in ladder. It also supports user-defined functions and function blocks in code form, enabling structured programming. ST tends to be more compact and easier to modify for large programs. It is also the most similar to standard computer programming, so engineers with a software background or coming from languages like C, Python, or Java find it relatively easy to adopt.

Disadvantages: The flip side is that ST, being text-based, is less immediately transparent to maintenance staff who are not trained in programming. A ladder diagram shows relay logic in a visual form that a technician can grasp, while an ST code block might appear as something complex. Troubleshooting in ST might require reading and understanding code, which can be hard.



```
preset_temp:=12;               //Immediate Expression
pushbutton:=1;                 //Immediate Expression
new_preset_temp:=preset_temp; //Tag Expression
timer_value:=ton1.ACC;      //Tag Expression
timer_done:=ton1.DN;       //Tag Expression
//timer operation
TONR_01.PRE := 500;
TONR_01.Reset := Reset;
TONR_01.TimerEnable := input3;
input3:=1;
if TONR_01.ACC >= 500 & TONR_01.DN then //A conditional statement
    input3:=0;
end_if;
TONR(TONR_01); //Function Expression
timer_state := TONR_01.DN;
total:=add1+12+13+add2;//Operators Expression
```

Figure 3. Structured Text Example [14]

Sequential Function Chart (SFC): Sequential Function Chart is a graphical language for structuring programs that must follow discrete steps or states. SFC is more of a graphical flowchart of the sequence of operations than a programming language for individual logic. A SFC consists of Steps (states) represented by rectangles and Transitions (conditions to move to next step) represented by horizontal lines between steps [15]. Each Step can have associated actions, and each Transition has a guard condition (a logical condition that, when true, causes the chart to move to the next step). SFC allows programming of complex sequential behaviours in a very organized manner [16].

Advantages: SFC is good for batch processes, machine step sequences, or any control that can be broken into discrete modes or phases. It provides a high-level overview of the operation, making it easier to visualize where in the sequence the process is. This simplifies both programming and troubleshooting. Maintenance of sequence logic is easier – inserting a new step or changing a condition is straightforward.

Disadvantages: SFC by itself doesn’t execute control logic, it relies on underlying actions typically written in LD, ST, or FBD. Therefore, one must be proficient in both SFC and another language. If a process has many parallel branches, the SFC can become complex to design and maintain. [17]

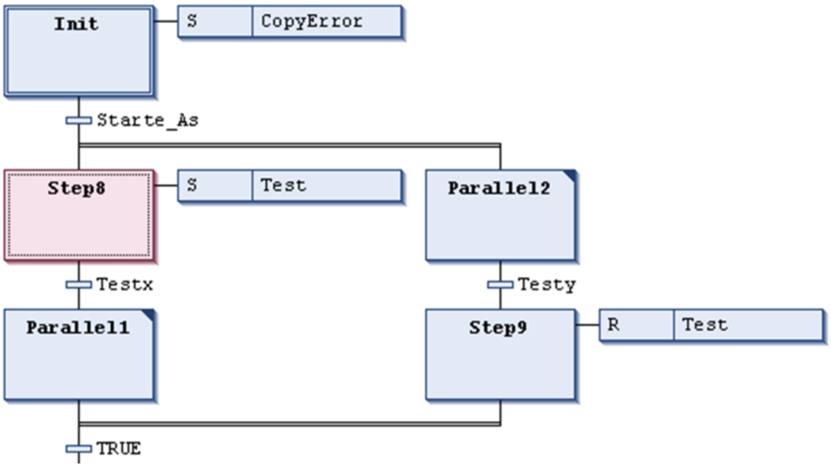


Figure 4. Sequential Function Chart Example [18]

Continuous Function Chart (CFC): Continuous Function Chart is not an official IEC 61131-3 language, but it is a closely related programming approach supported by some vendors (notably in the CODESYS environment and Siemens). CFC can be viewed as an extension of FBD: it also involves connecting function blocks with wires, but unlike standard FBD, CFC is free-form and not constrained to a left-to-right, top-to-bottom data flow on a fixed grid. Essentially, CFC allows a more continuous placement of logic – multiple control loops or logic flows can coexist on the same page and the execution order is determined automatically rather than strictly by layout.

Advantages: CFC’s main advantage is flexibility in program layout. Engineers can organize the control blocks in a way that logically groups related functions (even if they execute in different order) on one sheet. It is particularly useful for implementing complex interrelated analog control, where outputs of some blocks feed back into others in a loop.

Disadvantages: Because CFC is not formally standardized, its availability and behaviour can differ, only certain platforms (like CODESYS or Siemens) support it, limiting portability. Moreover, the free-form nature can lead to messy diagrams if not well managed – without the left-to-right convention, wires can cross and tangle, and execution order might not be immediately obvious. Many see CFC as a convenience for advanced users rather than a beginner’s tool [19].

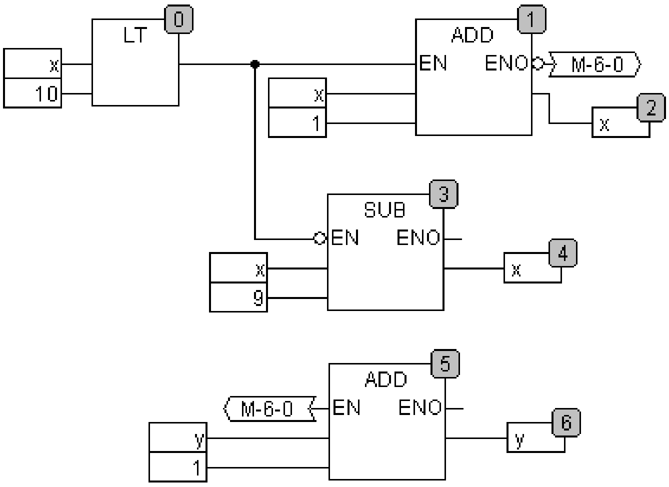


Figure 5. Continuous Function Chart Example [20]

3.5 Language choice for the program

Following a careful comparison of the programming languages defined in IEC 61131-3, the author chose Structured Text (ST) for developing the PLC tester control program for several reasons. First, ST uses a syntax that closely resembles common high-level programming languages, which makes it easier to write well-organized and modular code. Second, ST is already widely used within the company's software environment. This compatibility ensures it works well with other systems and makes it easier for colleagues familiar with ST to share knowledge and offer support. As a result, teamwork becomes smoother, and development can proceed faster with fewer delays. Additionally, the author had already gained basic experience with ST during earlier university studies, which made it easier to get started and reduced the time needed to learn. Overall, choosing ST offers the right mix of technical capability and practical benefits, making it the best fit for both short-term project success and long-term maintenance in an industrial context.

4 Overview of the Existing Test Setup

This chapter outlines the structure and functionality of the current industrial fan testing setup used in the electronics company. It describes the physical design of the test cabinets, and the key subsystems involved in Accelerated Life Testing (ALT), such as power distribution, temperature control, airflow, cycle counting, and data acquisition.

4.1 Test cabinets general overview

The current fan testing installation consists of seven identical test cabinets, each of which acts as a thermally insulated metal chamber housing multiple fans (referred to as Equipment Under Test, or EUT). Inside each cabinet, there are four stands designed to hold fans of different diameters. On one of the outer walls of the cabinet, there are two electronics compartments, that hold all the electronics needed for conducting the test.

As illustrated in Figure 6 (where different stands can be seen) the stands inside of a chamber are arranged so that an airflow loop circulates throughout the chamber, driven by the EUT fans themselves. This continuous loop ensures that the hot and cool air mix evenly, enabling consistent temperature for the duration of the test. According to one of the requirements specified in IPC-9591, the fans need to be tested at different orientations. Therefore, each of the four stands in a single cabinet is mounted in a unique position:

- **Stand 1:** Horizontally oriented, with fans blowing upward
- **Stand 2:** Horizontally oriented, with fans blowing downward
- **Stand 3:** Vertically oriented, with fans blowing to the left
- **Stand 4:** Vertically oriented, with fans blowing to the right

This arrangement enables the simulation of various installation scenarios and provides better assessment of fan reliability under real-world conditions.

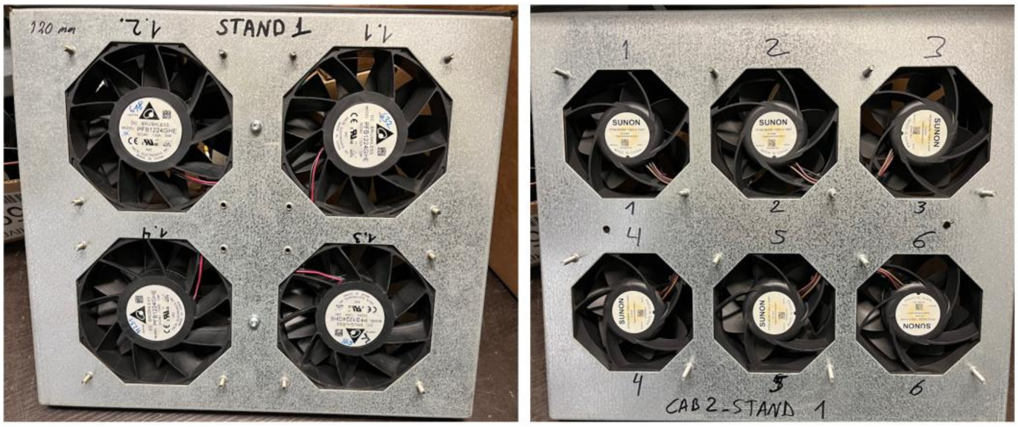


Figure 6. Stand with 120 mm fans (left) and stand with 80 mm fans (right)

In the lower portion of each test cabinet, there are eight 3.3 Ω resistors connected in series to 24 V and distributed along the two side walls (four per wall). These resistors serve as heating elements, heating the cabinet’s internal temperature as part of the accelerated life test protocol. By supplying power to these resistors, it is possible to achieve the high temperatures required to stress the EUT fans.

To balance out the generated heat and maintain a temperature setpoint, two cooling fans are installed at the top of each cabinet. These fans draw cool ambient air across the internal metal walls, lowering the cabinet’s internal temperature when needed. Figure 7 provides a visual representation of the airflow pathways, the location of the stands, and the arrangement of the heating resistors and cooling system. Although the fans successfully reduce internal temperatures, the existing design forces the heating elements to remain active, even during cooling phases, leading to energy inefficiency.

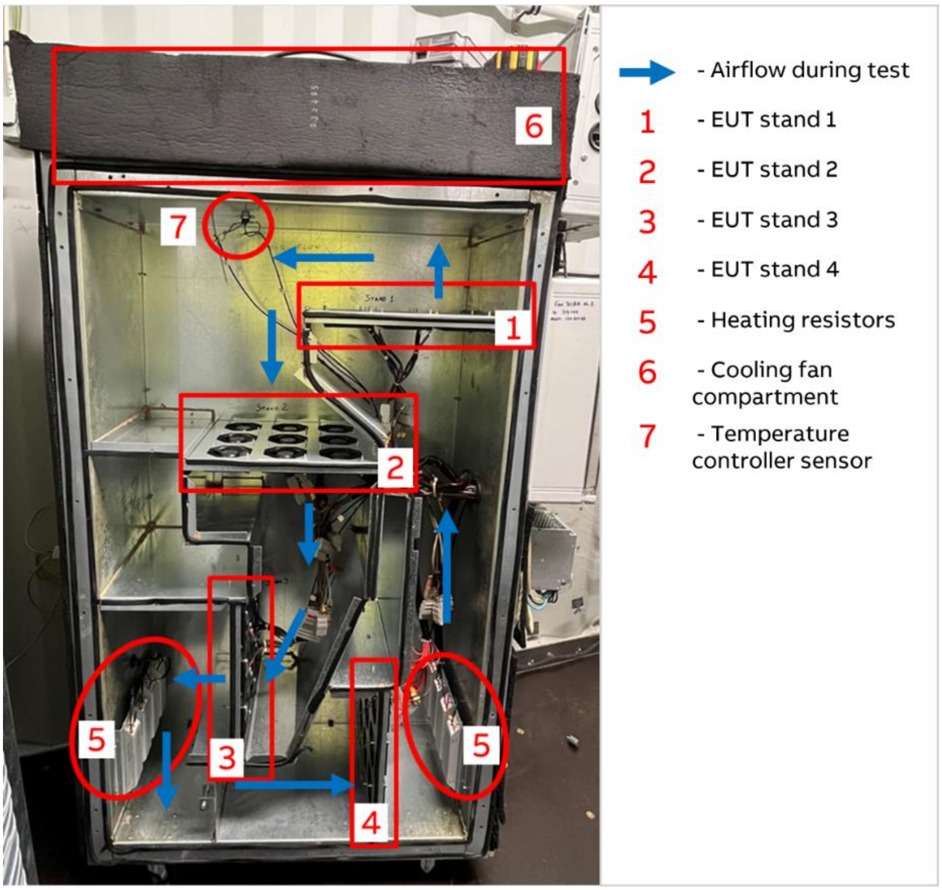


Figure 7. Tester cabinet setup inside

4.2 Test cabinet electrical compartment overview

Each test cabinet is powered by three separate 24 V DC power supplies.

1. **Control and Auxiliaries (3.5 A):** One supply provides power for all the test's control electronics – such as relays, counters, and timer modules. By isolating sensitive devices from high-current components, the setup avoids electrical noise and overload issues.
2. **Heaters and Cooling Fans (20 A):** A second supply runs the heating resistors and the cooling fans. Keeping these elements on a dedicated line prevents fluctuations in current draw from affecting the control or measurement systems.
3. **EUT Fans (20 A):** The final supply is dedicated to the fans being tested. This isolation is important for measuring fan current, as it ensures that the power baseline remains stable, improving measurement accuracy.

In addition to improved measurement quality, using multiple power supplies allows to balance loads, protect individual circuits from short circuits or overloads, and perform targeted diagnostics more efficiently.

For the chamber temperature regulation, a West 6100+ PID temperature controller is used. It is powered by 220 V AC. To obtain the current temperature inside a cabinet, it uses a single thermocouple placed inside the chamber, located at the top part of it (see Figure 7). Based on the user-defined setpoint (commonly around 85°C), the controller activates the cooling fans whenever the measured temperature exceeds the setpoint.

Two pulse counters are used to monitor the number of cycles completed during the testing process (see Figure 8). Each counter is a small digital device capable of incrementing its displayed value when receiving an electrical signal. The first counter is dedicated to thermal cycling. A thermal cycle is defined as one complete transition between a heating and cooling phase inside the test cabinet. The second pulse counter tracks power cycles of the EUT fans. A power cycle is counted every time voltage is applied to the fans. These counters are functional but purely manual – operators must be physically present to read the displayed values, and there is no automatic digital logging or remote visibility of the count.

Figure 8. Cycle counter [21]

Thermal and power cycle timings are controlled by a smart timer relay, which determines how long the cabinet remains in a heating or cooling state before switching (see Figure 9). Once the relay’s internal timer finishes a cycle, it toggles the heating elements accordingly and sends a pulse to the relevant counter. This programmable time-delay relay equipped with NFC (Near Field Communication) capabilities, which allows operator to configure timing parameters via a smartphone app and then send it directly to the relay. Users can define time intervals for which the output should remain on (heating phase) and off (cooling phase), effectively creating a repeating thermal cycle. Once the configured cycle begins, the relay energizes or de-energizes the heating resistors based on the elapsed time.

Figure 9. Finder Smart Timer Relay [22]

4.3 Data Acquisition (DAQ) Systems

To monitor the performance of each EUT fan, the testing setup uses Keysight Data Acquisition (DAQ) units, such as 34972A LXI and DAQ970A (see Figure 10). These devices can measure multiple voltage inputs and thermocouple channels simultaneously. A key feature of this system is its use of multiplexers (MUX), which allow a single DAQ device to sequentially read from multiple input channels using one or more shared analog-to-digital converters (ADCs). The multiplexers can be seen on the right side of Figure 10. This method is highly efficient, as it reduces the need for dedicated circuitry for each signal. For example, DAQs used in this tester can be equipped with up to three plug-in MUX modules, each supports 20 two-wire inputs. During operation, the system cycles through connected sensors one by one, switching between them rapidly enough to provide near real-time updates. This multiplexing capability is particularly useful in the fan testing setup, where a single DAQ unit needs to measure both the voltage across multiple shunt resistors and the readings from several thermocouples.

In this test, the DAQs serve two primary functions:

1. **Fan Current Measurement:** Direct current measurement is not possible through the DAQ terminals, so a 1 Ω shunt resistor is placed in series with each EUT fan. By measuring the voltage across the resistor, the fan current can be easily calculated based on Ohm’s law (in this case, current is the same as voltage in value). Monitoring

current is crucial for detecting fan failures – if a fan locks up or malfunctions, the current typically drops to zero or deviates significantly from the expected range.

2. **Temperature Monitoring:** Four channels on the DAQ are allocated to thermocouples, each mounted after every stand in the cabinet. This arrangement allows to track the temperature of the airflow passing through each set of fans, which is needed to comply with IPC-9591.

The DAQ devices support remote data access over a local network, enabling off-site monitoring of the test and data gathering.

Figure 10. Keysight DAQ970A Data Acquisition System with MUX modules (on the right) [23]

4.4 System Architecture

The initial test system architecture was designed to be as simple and straightforward as possible. It consists of a dedicated tester computer, 7 test cabinets and 7 data loggers (for each cabinet), all interconnected via a local network. A schematic representation of this system can be seen in Figure 11.

Tester Computer: An important part of the system is the tester computer, which is used for data processing and storage. This machine is responsible for running the software necessary for communicating with DAQs and collecting data from each of them. The data is then stored on the computer for further analysis. The tester computer is connected to the facility's local Ethernet network, allowing it to communicate with all data loggers.

Test Cabinets: Test cabinets are already described earlier in Section 4.1. They store EUT fans inside and all the other necessary equipment to conduct the test. Cabinets are connected to the rest of the system only via dataloggers, that collect test data from the cabinets.

Data Loggers: Each test cabinet is paired with a dedicated data logger. These data loggers are also connected to the same local Ethernet network, enabling communication with the tester computer. The primary functions of each data logger include:

- Reading internal temperature values from the respective test cabinet
- Monitoring the current of each of the cabinet's fans

This data is collected via multiplexers installed within each data logger, allowing multiple signals to be read through a single interface.

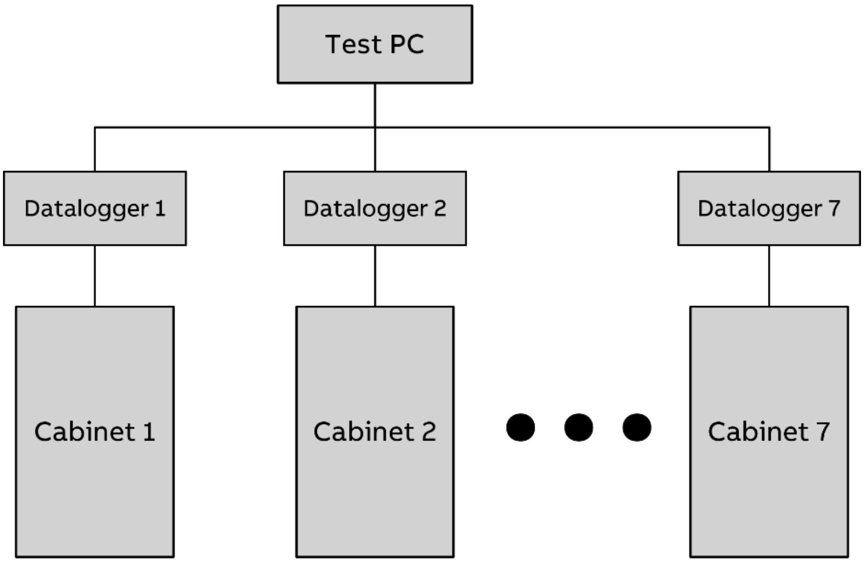


Figure 11. System architecture of the original system

5 Current Setup Issues

This chapter outlines the main issues and limitations of the current fan testing setup. Each problem is examined in terms of its technical nature, operational impact, and relevance to test reliability and efficiency. The analysis shows how the fragmented architecture, reliance on manual processes, and limited remote access capabilities slow back both efficiency and maintainability.

5.1 Issues description

Requirement for Physical Presence: One of the biggest drawbacks of the existing tester is the necessity for an operator to be physically present to control the test. Although remote access through DAQ devices was possible, it only gave a possibility to monitor the test – not control it, and in practice, proved to be highly unreliable. Multiple DAQs, combined with numerous fans connected to them, led to frequent software crashes after extended operation, and the computer connected to the DAQs was occasionally powered off as well. These conditions made remote access difficult and time-consuming, requiring on-site presence for even minor adjustments. This combined with lack of centralized data management, led to test data loss.

Combined Equipment Under Test and Heaters: The present tester design merges the EUTs and the heating elements onto the same power circuit. As a result, the heaters are turned on all the time, even when cooling is needed. This arrangement makes temperature cycle control highly inefficient, unsustainable and complicates scenarios where only cooling is needed. This makes precise temperature management almost non-existent and disrupts continuous testing, leading to suboptimal or incomplete test results.

Limited Temperature Control and Risk of Overheating: Temperature in the old tester is regulated by a physical temperature controller. Adjusting the setpoint, changing thermal cycle settings can only be done on-site. Additionally, the user interface of this temperature controller is limited to a small screen based on 7-segment display technology and a series of front panel buttons. The problem with that is the wide variety of settings this controller offers and the very limited and complicated user interface it has for these options. That makes setting this controller incredibly complicated and needing to read a manual every time. For example, understanding what the display is showing is almost impossible without consulting the manual first. Furthermore, if the temperature controller malfunctions or is incorrectly configured, it may continue heating indefinitely. Incidents have been happening where the test chamber temperature exceeded 100°C because the controller failed to shut off due to some kind of malfunction. Moreover, only one temperature sensor was used for both temperature control and safety shutdown with no redundancy. If this sensor failed or provided incorrect readings, the system risked overheating.

Thermocouple Connections via DAQs: Thermocouples used to measure temperatures inside the test chamber were wired into the same DAQs that are used to monitor EUTs. As mentioned earlier, reliance on multiple DAQs increased the likelihood of data loss, software glitches, and connectivity failures. When these devices became unresponsive, temperature data was no longer recorded, making the test results incomplete or invalid. Operators needed to be on-site to troubleshoot, further increasing downtime and complicating the tests.

Fan Current Measurements: Each fan is monitored by measuring voltage across a 1 Ω resistor to calculate current draw and confirm the fan's operational status. In the existing system, these resistors and their connections were manually soldered in a suboptimal way. Over time, weak solder joints, heat and physical disturbances caused connections to degrade or break. Additionally, the proximity of these resistors to each other's heat sinks and other components occasionally led them to overheat and fail. Repairs required stopping testing, physically resoldering the connections, and revalidating the circuit – all of which contributed to unnecessary downtime.

Reliance on a Smart Timer: A Time Delay Relay was used for controlling power cycle and thermocycle intervals. While it allowed the user to configure timings via a smartphone, this design introduced safety risks and practical limitations. Programming the relay required physical proximity to place the phone against the relay, which was supplied with 230 V, posing electrical hazards. The relay's application-based programming also meant that updating or troubleshooting required a phone with NFC connectivity and a specialized app. Furthermore, the relay served as a point of failure – if it malfunctioned or lost its configuration, the entire cycle logic would be disrupted.

Counter Reliability and Lack of Centralized Cycle Logging: Two physical counters incremented with each power or thermal cycle. Although these counters provided a record of test progress, their readings were only visible on-site, and data could not be transferred or stored in a centralized database. The lack of data logging and remote access has caused problems for quality control, analysing trends, and studying product reliability.

Cooling Challenges and Inefficiency: Another inefficiency was the cooling fan power supply. Legacy design utilized two cooling fans on top of the enclosure, that required 5 V control voltage in addition to 24 V powering voltage. This was implemented by dropping a 24 V supply to 5 V through a pair of resistors, which is inefficient and generated excessive heat loss. Moreover, cooling fans could only run together, limiting cooling control and leading to energy waste when only one fan might have been enough for cooling.

Complexity of Maintenance and Multiple Failure Points: The old system is composed of numerous independent components – DAQs, a temperature controller, a smart relay, multiple physical counters, resistors with sub optimally soldered measurement points, and complicated wiring. This arrangement introduces a range of potential failure points, making troubleshooting difficult and time-consuming. Technicians must be familiar with each piece of hardware, and a fault in one device can lead to a partial or total test failure.

5.2 Summary of Key Issues

The following Table 1 summarizes the issues identified in the current industrial fan testing system. Each problem is categorized by its nature, type, operational impact, and level of urgency. The table distinguishes between hardware and software limitations, safety concerns, data reliability issues, and energy inefficiencies, all of which have a direct influence on the overall effectiveness and maintainability of the testing process.

Table 1. Summary of key tester issues

Issue	**Nature of Problem**	**Impact**	**Type**	**Urgency**
Remote Access	Reliance on on-site monitoring and inefficient remote connections via multiple DAQs	Inefficient workflows, increased labour costs, risk of data loss	Hardware + Software	High
Combined EUTs & Heaters	EUT and heaters share connection, forcing simultaneous operation or complete shutdown	Limited temperature control flexibility, potential for improper cycling	Hardware	High
Temperature Regulation	Single sensor, manual setpoint changes, vulnerable temperature controller prone to failures	Overheating risks, lack of redundancy, risk of damage to tested devices	Hardware + Software	High
Current Measurement	Fragile solder joints and resistor-based measurement prone to breakage or overheating	Frequent hardware failures, manual repairs, inconsistent test data	Hardware	Medium
Smart Timer Relay Limitations	NFC-based relay requires physical proximity, exposing user to 230 V and complicating reprogramming	Safety hazards, lack of remote reconfiguration, potential single point of failure	Hardware + Software	Medium
Data Logging & Counters	Physical counters only visible on-site, DAQs can lock up, losing cycle data	Incomplete test records, limited historical data for analysis, reliance on manual data entry	Hardware + Software	High
Cooling Fans	Inefficient 24 V to 5 V conversion using resistors, all fans run at once, minimal individual control	Unnecessary energy consumption, inability to selectively cool specific parts of the test environment	Hardware	Medium
Maintenance Complexity	Multiple standalone components (DAQs, timer, temperature controller, counters) create confusion	Prolonged downtime, complex troubleshooting	Hardware + Software	High

6 New System Solution

This section outlines the modifications made to address the issues identified in Section 5. It presents proposed improvements to enhance the tester’s functionality and reliability, with a focus on selecting an appropriate PLC configuration for the system's specific requirements. The final system architecture is introduced, including communication between the PLC and remote I/O modules. Key hardware upgrades, such as resistor housing improvements and test chamber updates, are also described to improve system stability and maintainability.

6.1 Proposed Solutions to Existing Issues

To address the limitations identified in the original test system design, a series of improvements were introduced in the new version of the tester. The most significant upgrade, capable of resolving multiple issues simultaneously, is the integration of a PLC as the central control unit. The inclusion of a PLC brings a higher level of automation, modularity, and reliability to the system. One of the biggest advantages of the PLC-based solution is the ability to monitor and control the test process remotely. Through a suitable network configuration, the PLC allows operators to track test progress and interact with the system in real time, without requiring physical presence near the hardware.

Another key improvement is the transition of temperature control from the standalone PID controller to the PLC program. This change increases the flexibility and reliability of the thermal system. With this change, users can not only change the temperature control settings, but also remotely modify the setpoints and control parameters, which is not possible with the standalone PID unit.

The smart timer relay previously responsible for thermal and power cycle timings will be replaced by the program within the PLC as well. This removes the need to have physical proximity to program the relay, which previously posed a risk of electric shock. Integrating timing control into the PLC makes the test sequence easier to adjust.

Both pulse counters, used in the original setup to count thermal and power cycles, will also be replaced by internal counters within the PLC program. This allows remote monitoring of the number of completed cycles, without the need to be physically present to read the values from the counters' screens.

Another issue addressed by the addition of PLC is the unreliable storage of test data on the DAQ system. While the DAQ unit will remain part of the system for current acquisition, it will no longer be responsible for long-term data storage. Instead, the PLC will read data from the DAQ in real time and transfer it directly to a centralized database, the same database that most other test setups within the company use. This greatly increases data safety and consistency.

The previous system also had a limitation where the EUT fans and heating resistors were controlled together, which made temperature cycling inefficient. This issue will be resolved by rewiring the heating resistors to dedicated relays, which will also be controlled by the PLC. This configuration allows the heaters and EUTs to be controlled independently, making it possible to cool the chamber while the heaters remain off, significantly improving thermal control efficiency.

The current measurement points on the fans should also be redesigned. While the same principle remains – measuring voltage across shunt resistors connected to the fan EUTs, the current hand-soldered connections will be replaced with dedicated terminal blocks. This makes the setup not only more reliable, but also easier to service, especially when a resistor needs to be replaced.

Finally, the inefficient voltage conversion method previously used for cooling fans will be eliminated. These fans will be replaced with units that operate natively on 24V and do not need a 5V control connection, removing the need for additional converters. Furthermore, each cooling fan will be controlled independently by the PLC, enabling precise thermal control inside the chamber.

Together, all previously mentioned improvements not only add new functionality to the system and enhance ease of use, but also significantly reduce maintenance demands. By moving control and monitoring logic into the PLC and improving physical wiring and connection points, the system becomes more robust, modular, and maintainable, minimizing points of failure and increasing overall test reliability.

6.2 PLC Configuration Choice

Typically, in test systems at the Electronics Company, all control hardware is housed in a single centralized enclosure. It contains components such as circuit breakers, relays, terminal blocks, network switches, and the main PLC CPU along with its local I/O modules. From this central panel, all required connections are routed to the individual test elements, allowing the PLC to control and monitor the test process. However, this centralized architecture may not be the most suitable for this test system, as it consists of several independent thermal cabinets, each with an identical internal layout and fully self-contained wiring. For this system three potential approaches were considered to implement PLC-based control while preserving the structure of each cabinet:

Centralized PLC CPU with Multiple Local I/O Modules

In this configuration, a single PLC CPU is installed in the main electrical cabinet, along with enough local I/O modules to support all required input and output signals from every cabinet. This setup follows the standard practice used across test systems at Electronics Company. From this centralized control panel, wiring is extended to each test cabinet individually, allowing the PLC to manage the entire system from a single point.

The main advantage of this approach is cost-effectiveness and component availability. Since this configuration is widely implemented within the company, it takes advantage of commonly used components, easier implementation, and easy access to spare parts. Additionally, local I/O modules are typically less expensive than remote I/O systems and do not require additional network components or protocols for integration.

However, this solution has significant drawbacks in terms of wiring complexity and maintenance limitations. Routing all control signals from multiple independent cabinets to a single PLC needs extensive cabling that must go through the entire facility. This results in a large volume of wires going into one location, which is not only aesthetically cluttered, but can also become overcomplicated, unreliable, and potentially hazardous if not managed correctly. Furthermore, if one of the test cabinets needs to be relocated or temporarily disconnected, its wiring must be physically removed and later reconnected, a process that is both time-consuming and tedious.

Individual PLC CPU per Cabinet

An alternative approach involves equipping each test cabinet with its own dedicated PLC CPU and local I/O module. In this setup, every cabinet functions as an entirely autonomous unit, running its own control PLC program and operating independently of the others. The only shared requirement is a connection to the local network to enable communication with other systems and data logging.

The main advantage of this solution is the complete independence of each test station. All wiring related to the PLC is contained within the cabinet itself, eliminating the need for long cable runs. Furthermore, having a separate CPU for each unit allows the control program to be lightweight and optimized, as it only needs to manage the processes of a single test cabinet. Another benefit is fault isolation: if one cabinet has a failure, it does not affect the operation of the others.

However, this option also has drawbacks, primarily in terms of cost. Supplying each of the seven test cabinets with its own PLC CPU and I/O module is significantly more expensive than using shared components. Additionally, while independent programs offer flexibility, they also introduce complexity in software management. Any change to the test logic must be applied individually to each cabinet. From a networking standpoint, having multiple PLC CPUs also means more IP addresses are consumed – something that can be a limitation within the network infrastructure of a large organization. Moreover, each CPU must be connected separately to monitor or modify test parameters, which is not convenient.

Centralized PLC CPU with Distributed Remote I/O Modules

This approach combines centralized control with decentralized signal acquisition. A single common PLC CPU is used to manage all test cabinets, like in option one, but instead of installing all I/O modules in the main electrical cabinet, remote I/O modules are installed directly inside each test cabinet. These remote modules communicate with the central PLC via a single cable using PROFINET connection, enabling clean, reliable, and modular connectivity.

The advantages is that all internal wiring between the sensors, actuators, and the I/O module happens within the cabinet, which simplifies installation and improves signal integrity. To connect from the cabinet to the main PLC, only a single Ethernet cable is required, which reduces cable clutter. If a test cabinet needs to be relocated, only one cable needs to be unplugged (other than the power connection), making the system highly modular and easy to maintain.

There are, however, certain drawbacks. Remote PROFINET I/O modules are generally more expensive than usual local I/O modules. In addition, to establish a proper PROFINET connection, the PLC must either natively support PROFINET or be equipped with an additional PROFINET communication module, which increases the overall hardware cost.

After evaluating the available options, the third configuration (centralized PLC CPU with distributed remote I/O modules) was chosen as the most practical solution. It offers a good balance between centralized logic and simplified wiring, reducing installation complexity and supporting modularity and scalability. Despite slightly higher hardware costs, the benefits in flexibility, maintainability, and expandability make it the optimal choice for a multi-cabinet test system.

6.3 System Architecture

The system architecture of the new version of the tester is more complex compared to the previous version. This is necessary to support additional functionality, such as remote access, data management, and distributed control. To enable these features, several new subsystems had to be added, including a centralized database, PLC, and remote PROFINET I/O modules. This architecture should be able to manage multiple test cabinets simultaneously. The new system consists of the following key subsystems:

- Test PC
- Central Database
- PLC Master (CPU)
- Ethernet Switch
- PLC Slaves (7 unit)
- Data Loggers (7 unit)
- Test Cabinets (7 units)

Main part of this setup is the PLC Master (CPU), which runs the main control logic for the entire testing process. This CPU is responsible for executing the test sequence, communicating with the database server, connecting with all seven test cabinets, and receiving measurement data from each data logger. The remote I/O architecture is implemented using PROFINET: remote I/O modules are installed inside each test cabinet and connected to the central PLC via an Ethernet switch. These modules provide the necessary digital and analog I/O channels for controlling test operations and collecting feedback from sensors and relays. This architecture optimizes the physical layout by reducing cabling complexity and allows for modularity and easy system maintenance. Each cabinet operates as a fully independent unit, while still being integrated into the central control system.

The structure of the full system is illustrated in Figure 12, which presents the relationship between all subsystems and highlights the communication links between the central PLC, I/O modules, and external interfaces.

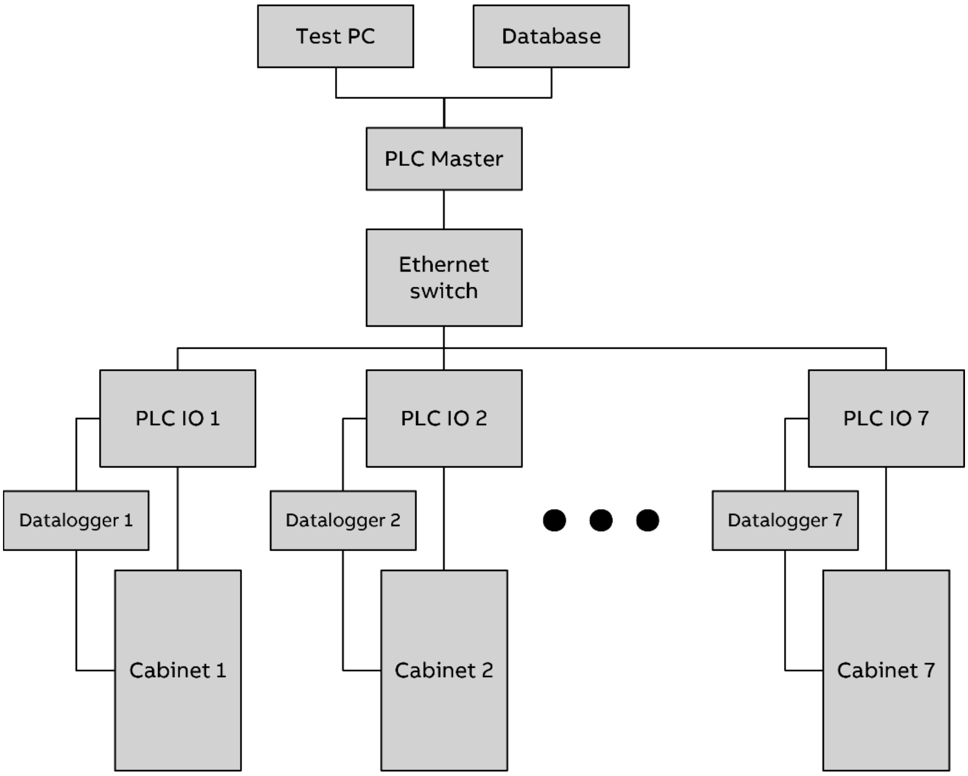


Figure 12. Overall system architecture of the new tester setup

6.4 PLC Communication with Remote I/O and DAQ Systems

An unusual network setup is used to enable simultaneous communication with both PROFINET-based remote I/O (PN-IO) modules and a standard TCP/IP datalogger. The main PLC (CPU) in the system is equipped with two Ethernet ports. One of these ports is connected directly to the facility's local network, which provides PC connection for programming and monitoring. In addition to the PLC itself, a PROFINET communication module is installed directly to the PLC CPU, which has two RJ45 ports identical to standard Ethernet, but they are dedicated to PROFINET connections.

In a typical PROFINET configuration, the master (PLC) connects to one or more slave (I/O modules) via one of these ports, usually through an unmanaged Ethernet switch to support multiple nodes. Each slave also has two PROFINET ports – one for communication with the master and one for daisy-chaining to additional PROFINET devices. In this case, the first port of the I/O module is connected to the PROFINET master, while the second port is used to connect a data logger. However, since the data logger does not support the PROFINET protocol and only operates on a standard TCP/IP network, it cannot communicate with the PLC through the PROFINET network. To work around this, a bridge is created by physically connecting the second Ethernet port of the PLC (part of the local network) to the second port on the PROFINET master communication module. Although this may seem wrong – to mix PROFINET and TCP/IP networks, it works because PROFINET is built on standard Ethernet infrastructure and supports mixed traffic at the physical level. This physical connection allows the non-PROFINET device (the datalogger) to access the same Ethernet layer as the PLC's local network, enabling normal TCP/IP communication.

This bridging configuration merges the PROFINET and standard Ethernet segments into a single logical network, enabling both communication for PLC I/O via PROFINET and data transfer for datalogger via TCP/IP within the same wiring infrastructure. This eliminates the need for separate connections for the DAQs, conveniently utilizing the free port on the remote I/O module. This setup is illustrated in Figure 13, which shows the connections between the PLC, the PROFINET module, the remote I/O slave, and the datalogger.

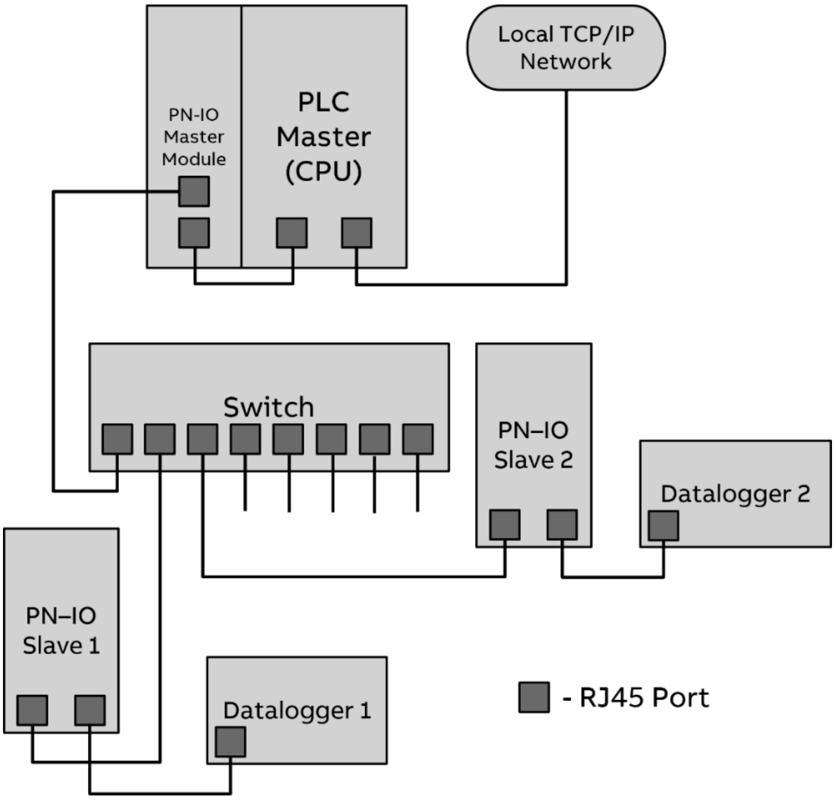


Figure 13. Network topology illustrating the connection of PROFINET devices and a TCP/IP data logger through dual Ethernet links.

6.5 Overview of PROFINET and Comparison with Other Industrial Communication Protocols

PROFINET (Process Field Network) is an open industrial Ethernet standard developed by PROFIBUS & PROFINET International for real-time data communication in automation systems. It uses standard Ethernet technologies to allow high-speed, deterministic communication between controllers and field devices, including I/O modules, sensors, and actuators. PROFINET supports various network topologies and offers features like device configuration, diagnostics, and seamless integration with IT systems. [24].

A key advantage of PROFINET over traditional fieldbus systems is its ability to handle both real-time and non-real-time data on the same network infrastructure. This integration simplifies network architecture and reduces installation and maintenance costs. PROFINET defines different conformance classes (CC-A to CC-D) to cater to varying real-time requirements, from standard cyclic data exchange to isochronous real-time communication for motion control applications [25].

In contrast, PROFIBUS (Process Field Bus) is a serial communication protocol that has been widely used in industrial automation. While PROFIBUS provides reliable communication, it operates at lower data rates and lacks the flexibility and scalability offered by Ethernet-based systems like PROFINET. Additionally, PROFIBUS networks are more challenging to integrate with modern IT infrastructures, limiting their applicability in today's interconnected industrial environments [26].

Other industrial Ethernet protocols, such as EtherCAT and Modbus TCP, also offer real-time communication capabilities. However, PROFINET's widespread adoption, comprehensive feature set, and strong support from major automation vendors make it a leading choice for modern industrial automation systems. Its compatibility with standard Ethernet hardware and protocols ensures seamless integration with existing networks, facilitating the convergence of operational technology and information technology systems.

6.6 Resistor Redesign

The original solution with directly soldered shunt resistors was replaced with a more versatile and more maintainable solution. Terminal blocks from Phoenix Contact were added into the circuit to provide a modular and reliable interface. These blocks are equipped with component connectors, also supplied by Phoenix Contact, into which the shunt resistors used for current measurement across the fans are easily inserted.

This improved connection method (illustrated in Figure 14) significantly increases the mechanical reliability of the setup and simplifies maintenance. If a resistor needs to be replaced, it can now be done quickly and without the need for desoldering. Additionally, this design gives a cleaner and more professional looking installation, adding the visual and functional quality of the setup.

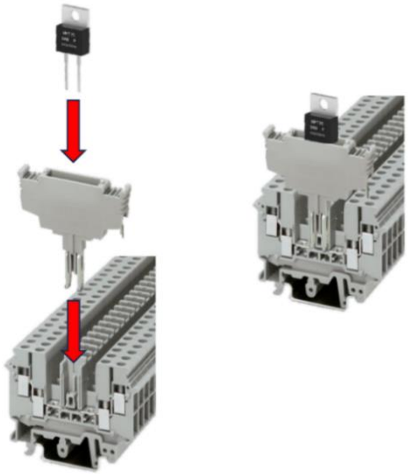


Figure 14. Resistor connection redesign scheme

6.7 Temperature Sensor Change

Originally the temperature at each stand in the test chamber was measured using thermocouples connected directly to the datalogger. The datalogger would periodically sample the voltage signals from these thermocouples to determine the air temperature passing through the EUTs. However, it was decided to move temperature measurement from the datalogger to the PLC to improve the reliability of the control system, as the PLC now is the one to control the temperature in the chamber and it needs a more reliable and quicker source of the temperature, which datalogger sampling would not achieve. It will also free up the data logger channels for other measurement tasks. Thermocouples, while accurate and widely used, produce very low voltage signals in the millivolt range, which require specialized high-impedance analog inputs with built-in cold junction compensation. The standard analog inputs available on the PLC were not designed to handle such low-level signals with sufficient accuracy and adding separate hardware for reading thermocouples would not be worth it. As a result, the thermocouples were replaced with PT100 resistance temperature detectors. PT100 sensors are more compatible with PLC analog input modules, as they operate based on changes in resistance rather than millivolt-level voltage and provide more stable and noise-resistant readings. PT100 are also widely available withing the company. This modification allowed for more direct integration of temperature data into the PLC control logic, improving both system simplicity and data integrity, as it does not need to go through DAQ to PLC but goes straight to PLC.

6.8 Electronics Compartment Redesign

The electronics compartment of the tester was redesigned to hold the additional components introduced in the updated system architecture. In the original setup, the compartment consisted of two separate electrical boxes, with two 20-amp power supplies mounted externally due to limited internal space. In the new design, the compartment was expanded to include three enclosures, allowing for all components, including all power supplies, to be installed internally. The addition of a third box not only provided the necessary space for the new control electronics, but also enabled a better layout of components, reducing heat buildup and minimizing the risk of accidental disconnections or shorts during maintenance processes. The added internal space also improved the routing wiring, making installation cleaner and more serviceable.

A comparison between the original and redesigned electronics compartment is shown in Figure 15, illustrating the improvements in layout, accessibility, and available space. Note: Wiring in the new version has not yet been finalized, as the system is currently in the prototyping phase.

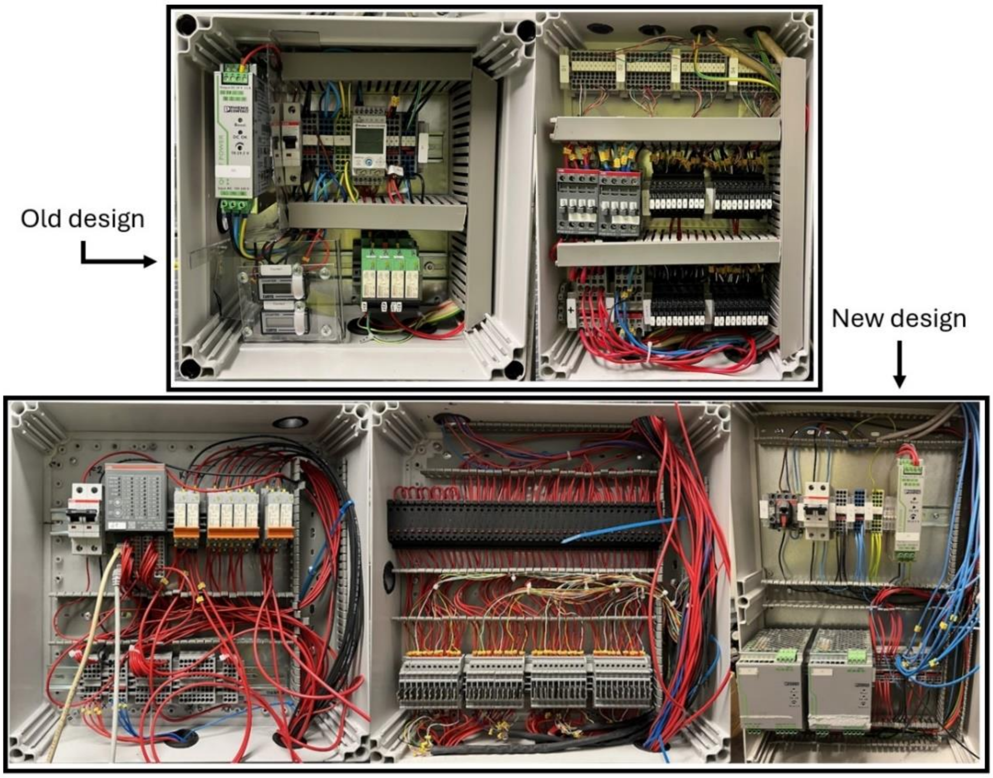


Figure 15. Comparison of the electronics compartment: original version (top) vs. redesigned version (bottom).

7 PLC Program

This section discusses the requirements for the PLC program, the overall program structure, the I/O mapping, the main functional blocks, and the principle of operating multiple cabinets under a single program.

7.1 Purpose and Objectives of the Program

The main purpose of this program is to fully automate the fans testing process across multiple independent test cabinets. The program must have all previously available functionality while introducing several new features to enhance reliability, data management, and user interaction.

The core functional requirements of the system:

- Starting and stopping tests
- Temperature control and monitoring
- Setting and controlling thermal and power cycles
- Logging of test results and EUT data into a centralized database
- Accessing stored data from the database when needed
- Collecting EUT current from dataloggers
- Detecting errors or anomalies during testing and react if needed

The PLC program is written entirely in Structured Text. Development is made in an IDE based on CODESYS – a widely adopted platform for PLC programming. CODESYS provides a complete programming environment that fully complies with IEC 61131-3 standards, enabling both classic PLC programming like LD, ST and FBD and modern object-oriented programming across all standardized languages. It also offers seamless configuration and commissioning support for major industrial I/O and fieldbus systems, such as PROFIBUS, CANopen, EtherCAT, EtherNet/IP, and PROFINET. Additionally, CODESYS can integrate optional advanced components like visualization, motion control, CNC, robotics, and functional safety, allowing users to develop complete automation solutions within a single platform [27].

7.2 Program Structure

This section describes the program structure, focusing on the use of function blocks, structured data types, and global variable lists.

7.2.1 Overall Structure

As described in Chapter 6.3, the system architecture is based on a single central PLC that manages all seven test cabinets. Therefore, the control software is implemented as a single unified program that can operate all cabinets independently and simultaneously. This is achieved by containing each part of the program into function blocks. These function blocks are then called within indexed loops, cycling through the cabinet array. Each cycle provides a needed cabinet index, allowing the same function block to execute separately for each cabinet with its own set of data. This structure removes the need for duplicating the same code across multiple instances. It simplifies maintenance and makes it easily scalable, because any change to the program logic automatically applies to all cabinets. It also ensures consistent behavior across all cabinets and keeping them fully independent.

7.2.2 Data Structures and Variables

To make the program scalable, modular, and allow code reuse, data in the system is organized in a structured way. All cabinet-specific variables are contained within dedicated structures, while global control variables and constants are kept in a common global list.

The system uses three main global variable lists:

- **GVL (Global Variable List):** Contains the main "cabinet" structure and other global variables not tied to any specific cabinet. These include general test flags, control parameters, and network configuration.
- **PVL (Persistent Variable List):** Contains the "cabinet persistent" structure. Variables defined here are saved in PLC memory, ensuring data is saved across PLC reboots or power losses.
- **DB_GVL (Database Global Variable List):** Contains the "database" structure used for database operations, as well as additional parameters related to database access, such as login credentials and connection settings.

Cabinet Structure: The “cabinet” structure serves as a container for all variables related to an individual test cabinet. This includes information such as the current test state (e.g., active or inactive), control elements status (heaters and cooling fans), temperature measurements, data for communication with dataloggers, permissions, and other operational parameters. In the Global Variable List, this structure is defined as an array of seven elements, with each array element representing one physical test cabinet. This provides an identical but fully independent set of variables for each cabinet. The cabinet structure also includes several nested structures that group related data types. These structures are described in more detail below.

EUT Structure: This structure is defined as an array within the “cabinet” array and represents the EUTs assigned to a specific cabinet. Each EUT structure contains information such as serial number, position, dimensions, and other data for test identification and result logging. The system supports up to 40 EUTs per cabinet. This structure is primarily used both for configuration and result logging to the test database.

Cycle Step Structure: This structure is defined as an array in “cabinet” as well and represents individual steps of a thermal cycle. Each step includes configurable parameters such as step duration, target temperature, power cycling settings, and active fan stands. The number of steps is dynamic and depends on the specific test settings applied.

Database Structure: This structure is used for storing database related variables that are needed for establishing connection with the database and writing/reading data. Since each cabinet independently logs data to the database, this structure is also defined inside the “cabinet” array to isolate database operations.

Cabinet Persistent Structure: In addition to the regular “cabinet” structure, the system includes a structure called “cabinet persistent”, which is also defined as an array of seven elements – one per cabinet. This structure is used to store cabinet-specific variables that must be retained after a power cycle or system reboot. Normally, when the PLC loses power or when the software is updated, all variables are cleared. To preserve important values, CODESYS suggests using Persistent Variable List (PVL), which works similarly to a GVL but ensures that variables are stored in a PLC memory. This is the list, where “cabinet persistent” structure is defined, allowing variables to be saved even after restarts.

7.3 PLC I/O Mapping

This section explains the I/O mapping of the PLC system. The mapping defines how sensors and actuators are physically connected to the remote PLC I/O modules. Each test cabinet has an identical I/O structure, connected through dedicated PROFINET remote I/O modules. The purpose of this mapping is to unify signal assignments across cabinets.

Table 2 presents an example of the variable names, signal types, and their corresponding functions for test cabinet number one. Each cabinet follows the same mapping scheme, with variables adapted for their cabinet numbers.

- Analog Inputs: Continuous signals typically used for sensors such as PT100 temperature sensors.
- Digital Outputs: ON/OFF signals used for controlling actuators such as heaters and fans.

Table 2. PLC I/O mapping table for cabinet 1

Variable Name	**Type**	**Description**
cabinet_1_temp_1	Analog Input	Reads the temperature from PT100 sensor at stand 1
cabinet_1_temp_2	Analog Input	Reads the temperature from PT100 sensor at stand 2
cabinet_1_temp_3	Analog Input	Reads the temperature from PT100 sensor at stand 3
cabinet_1_temp_4	Analog Input	Reads the temperature from PT100 sensor at stand 4
cabinet_1_heating_resistor_1	Digital Output	Controls the first group of heating resistors
cabinet_1_heating_resistor_2	Digital Output	Controls the second group of heating resistors
cabinet_1_fan_1_control	Digital Output	Turns ON/OFF the fan stand 1
cabinet_1_fan_2_control	Digital Output	Turns ON/OFF the fan stand 2
cabinet_1_fan_3_control	Digital Output	Turns ON/OFF the fan stand 3
cabinet_1_fan_4_control	Digital Output	Turns ON/OFF the fan stand 4
cabinet_1_cooling_fan_1_control	Digital Output	Turns ON/OFF the first cooling fan
cabinet_1_cooling_fan_2_control	Digital Output	Turns ON/OFF the second cooling fan

7.4 Main Program Blocks

This section provides an overview of the main functional blocks that make up the PLC program. Each block is responsible for a specific aspect of the fan testing process, such as system initialization, cycle management, temperature and power control, database communication, data logging, and permission checking.

7.4.1 System Initialization

The system initialization block is responsible for preparing all necessary variables and settings before the test process can start. During initialization, the PLC I/O variables are assigned to the corresponding fields within the “cabinet” structure for each cabinet individually. These variables include temperature measurements, stand control signals, cooling fan control, and heating elements. For certain signals, additional processing is done, such as the conversion of raw analog input values into real-world units (e.g., °C for temperatures). This ensures that all measurements are accurate and ready for use in visualization and control logic. In addition to variable assignments, the initialization block also handles technical processes such as setting IP addresses for data logger communication, initializing internal flags, and preparing database connection parameters.

7.4.2 Main Cycle

The Main Cycle program is responsible for starting several subprograms that manage the operational state of the test system. These subprograms are:

Average Temperature Calculation: This subprogram calculates two types of average temperatures within each cabinet:

1. The average temperature based on all four stand temperature sensors
2. The average temperature calculated only from active stands (i.e., stands currently participating in the test, as some stands may be excluded from the test by user configuration). This is needed for more accurate test results on specific stands.

Cycle Template Program: This subprogram manages the system of cycle templates for thermal cycle settings. It allows the operator to save their configured thermal cycle parameters and later restore them when needed. The template system utilizes the Persistent Variable List, described in Section 7.2.2, ensuring these configurations can be preserved even in the event of an unexpected PLC reboot. This feature reduces downtime and enables a quick recovery of an ongoing test.

Cycle Start Program: The Cycle Start subprogram is responsible for initializing and managing the execution of the test cycle itself. Its functionality is described in detail in the next section.

7.4.3 Cycle Start

The Cycle Start program is the main component of the test system, responsible for controlling both the thermal and power cycling processes. It is one of the most important blocks within the entire tester program. When a start signal is received, the program first performs a complete permission check. If all permissions are okay, the system sets the "Test Active" flag, starts the overall test timer, and verifies that the test is at the initial cycle state (State 1). If the system is in State 1, the program performs the following actions:

- Check if all cycle steps have been completed. If so, the step counter is reset to the first step, ensuring the continuity of the cycle.
- It loads the parameters for the current step, including target temperature (setpoint), interval settings for the power cycle and active test stands for this step.
- After loading the parameters, the system transitions to State 2.

In State 2, the program activates two control blocks:

- **Temperature Cycle Control:** Regulates heating and cooling systems to maintain the setpoint temperature.
- **Power Cycle Control:** Manages the on/off switching for the test stands based on the configured intervals.

The step timer is starting as well. When the step timer finishes (indicating the end of the current step), the system deactivates the timer, resets relevant flags, and transitions to State 3.

In State 3, the program increments the current cycle step number and transitions back to State 1, initiating the next cycle step.

If the user manually stops the test at any time, the program stops all active timers, clears all status flags, disables heating and cooling systems and resets the cycle state and step counters. Cycle Start block flowchart can be seen on Figure 16.

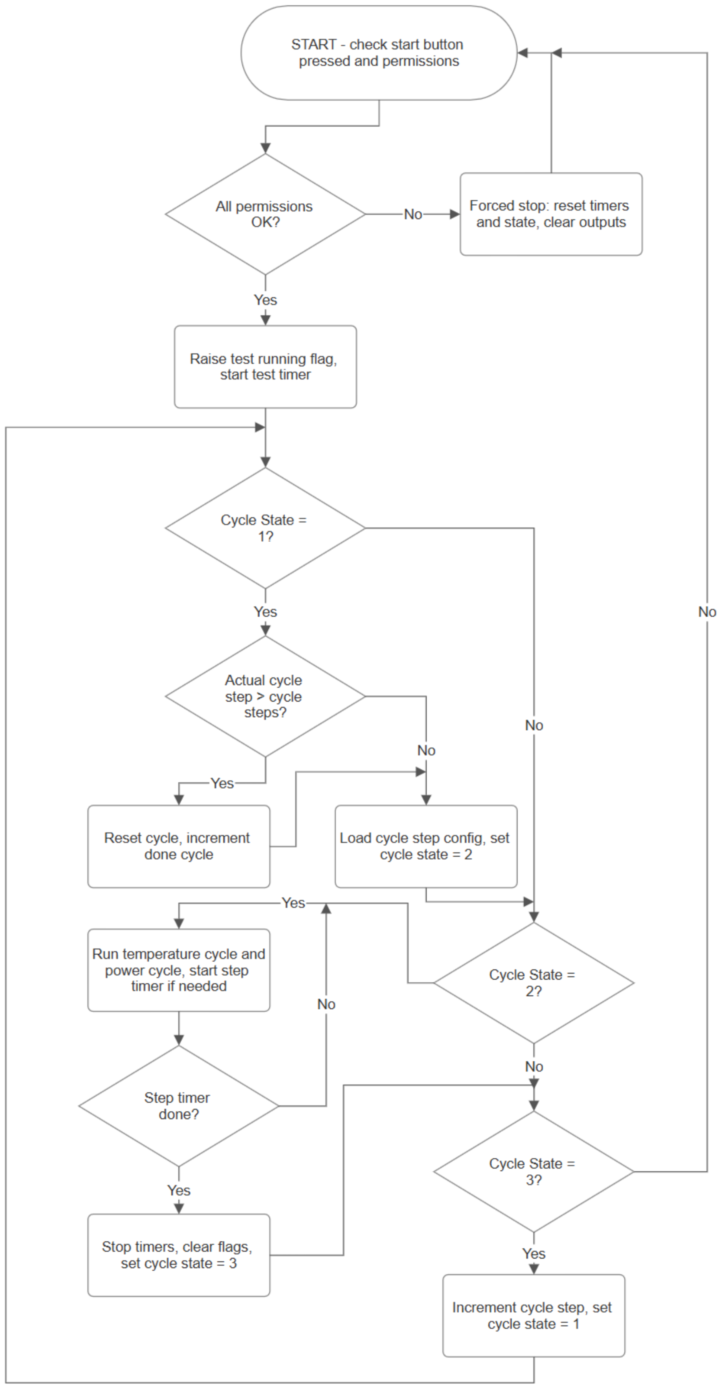


Figure 16. Cycle Start block flowchart

7.4.4 Temperature Cycle

The Temperature Cycle program is responsible for regulating the internal temperature of each test cabinet during the testing process. This task, which was previously handled by a separate hardware temperature controller, has now been fully integrated into the PLC software. The program monitors the average temperature of the currently active stands within each cabinet and compares it to the defined temperature setpoint. Based on the temperature, the program adjusts the heating and cooling systems to maintain the test conditions needed.

The control logic works as follows:

- If the measured temperature is significantly below the setpoint, both heating resistor groups are activated.
- If the temperature deviation is moderate, only one heating group is activated.
- If the temperature is within the acceptable range, no heating is applied.

The same logic is used for cooling.

This control flexibility is made possible due to the independent wiring of the two heating groups and two cooling fans, allowing better adjustments to the temperature. The specific temperature deviation thresholds that determine whether zero, one, or two heating or cooling elements are activated can be configured by the user through the graphical user interface, as later detailed in Section 8.

7.4.5 Power Cycle

The Power Cycle program is responsible for managing the on/off cycling of the EUTs during the testing process. It provides control over the power states of the fans based on user-defined parameters. For each step of the thermal cycle, the user can configure which stands will participate in the power cycle, the interval during which the EUTs remain powered ON and the interval during which the EUTs remain powered OFF. If power cycling is not required for a particular test, the system allows all EUTs to remain continuously powered throughout the thermal cycle.

7.4.6 Permission System

The program also has a Permission System, designed to ensure that all necessary safety and operational requirements are met before a test can work. Each permission block verifies specific conditions within the system. Based on the severity of the detected issue, permissions are categorized into three types:

- **Critical:** If the condition fails, the test is immediately stopped and prevented from restarting.
- **Blocking:** If the condition fails, the test cannot start, but an ongoing test is not interrupted.
- **Warning:** If the condition fails, a warning is displayed, but the test is allowed to start and continue.

The following permissions are implemented:

Temperature Permission: This permission checks the temperature conditions inside the test chamber according to three specific rules:

- **Sensor Failure Detection:** Checks if the sensors on Stands 1 and 2 provide readings outside the normal operational range. Needed to ensure that in case of a malfunction at any stand, the primary sensors on Stands 1 and 2 are working fine, and difference in top and bottom temperature is going to be detected.
- **Temperature Imbalance Detection:** Checks if temperatures on stands 1 and 2 are lower than 60% average of stand 3 and 4. Needed to avoid situations when hot temperature stays on the bottom due to malfunction of Stands 1 and 2.
- **Sensor Consistency Check:** Checks if half of the sensors detect low temperature (difference 20 °C) from other sensors. Works when it is not in the active cooling or heating stages. Needed to assure that more than half of the sensors are working fine. It allows only 1 failure sensor when 4 are working.

Permission type: Blocking.

Current Permission: This permission verifies that the current draw of each fan matches its operational status. If a fan is running, it must have a measurable current value, within the allowed range. If a fan is stopped, the current must be near zero. The data is collected from datalogger.

Permission type: Warning

Stand Enabled Permission: This permission ensures that at least one stand is selected and active for the test.

Permission type: Critical.

Cycle Table Permission: This permission validates the data entered in the cycle settings table: each step must have both a temperature setpoint and a duration (or neither), the cycle table must not be empty.

Permission type: Critical.

EUT Table Permission: This permission checks the correctness of the information provided in the EUT data table. Required fields include fan serial numbers, stand installation, dimensions, cycle target information.

Permission type: Critical.

Cabinet Online Permission: This permission verifies that the communication with the remote I/O module of the corresponding cabinet is active and stable.

Permission type: Critical.

Database Permission: This permission checks whether the program has successfully connected to the database server.

Permission type: Warning.

7.4.7 Datalogger Block

The Datalogger Block is responsible for communication with the datalogger assigned to each test cabinet. The block establishes a network connection to the datalogger using its IP address, which is configured during the system initialization phase. At regular intervals, the Datalogger Block sends read requests to the datalogger and retrieves current measurement data for all EUTs associated with the cabinet. Once the data is received, it is processed and distributed to the appropriate internal variables within the program. This acquisition of current is needed for monitoring the operational status of each EUT and powering the permission system that checks for abnormal current values.

7.4.8 Database Main

The Database Main block is responsible for all interactions between the tester system and the company's centralized internal database. This block coordinates three key subprograms: Summary Read, Summary Write, and Measurement Write. For PLC and database communication a special SQL library is used. Each of these subprograms establishes a connection to the database server and sends queries for read or write operations necessary for the correct management of test data.

- **Summary Read:** This subprogram is used to upload EUT information entered by the user into the database. Details like stand assignment, physical position, serial number, and dimensions of each EUT. The Summary Read program takes this information and writes it to a dedicated Summary Table within the database.
- **Summary Write:** This subprogram performs the reverse operation: it retrieves previously stored EUT information from the database and fills back tester's EUT Table. This feature allows for easy restoring of previously prepared test setups.
- **Measurement Write:** During the test, this subprogram continuously logs process data into a separate table within the database. It records critical parameters such as the temperature of the stand associated with each EUT, and the electrical current drawn by each fan. This is needed for future test data analysis.

Database Structure

The company's database system is based on Microsoft SQL Server and is hosted on a centralized server shared among all testers. Each tester is assigned its own dedicated database on the server, and all tester databases follow an identical structure design to maintain consistency.

The primary tables used in each tester's database are:

- **Measurements Table:** Stores test results, including temperature and current readings for each EUT during active testing.
- **Summary Table:** Stores static data about each EUT, including serial numbers, assigned stands, physical dimensions, and testing objectives.

In addition to the centralized database, each tester also has a local database of the same structure stored on a tester computer. The local database allows the tester to continue operating and saving data even if the connection to the central server is interrupted. Additionally, the local database contains an additional table:

- **Measurements Commission Table:** This table has the same structure as the standard Measurements Table but is intended for commissioning tests, system verification, and debugging. Data recorded in the Measurements Commission Table is not synchronized with the central database.

Within the tester software, a switch is implemented that allows the user to select whether test results are saved into the standard Measurements Table (for official test runs) or into the Measurements Commission Table (for commissioning and internal checks).

7.5 Multi-Cabinet Implementation and Comparison of Alternative Solutions

Since a single centralized PLC program manages all test cabinets, it is important to organize the program in a way that allows each cabinet to operate independently and at the same time. To achieve this, a modular approach was implemented using function blocks and indexed execution within loops.

Final Solution: Indexed Function Block Execution

All functions of the program, such as cycle control, temperature regulation, power management, database communication, and permission checking, were implemented as independent Function Blocks.

A for-loop runs from 1 to 7, corresponding to the seven test cabinets. During each loop:

- The current cabinet index is passed to the function block.
- The function block internally uses the index to select the correct data structure (e.g., cabinet[7]), ensuring that each instance of the block operates on its own specific set of variables.

This is made possible by storing all cabinet-specific data in an array of structures (cabinet[7]), where each element contains all the necessary variables for one cabinet. As a result, every cabinet operates independently, while sharing a common code.

Alternative Methods

During the design phase, several alternative solutions were considered but later rejected:

- **Separate Programs for Each Cabinet:** One possible approach was to create fully separate program instances for each cabinet. While straightforward, this method would lead to significant code duplication. Any update or correction in the logic would require modifying all seven copies individually, increasing maintenance effort and the risk of introducing inconsistencies.
- **Shared Code Without Function Blocks:** Another considered approach was to move the content of the programs directly into a for-loop without using function blocks. However, this caused conflicts, because the local variables inside the loop body would be shared across all iterations, leading to unpredictable behavior and data corruption. Proper instance separation could not be achieved without containing the logic inside function blocks.

Therefore, using function blocks with indexed execution was chosen as the most reliable, maintainable, and scalable solution for this program.

7.6 Summary of PLC Program Implementation

The PLC program developed for this tester successfully combines all required functions into a single system. It preserves the full functionality of the previous version and introduces important new features such as automated cycle management, database communication and detailed permission checks. Thanks to its modular design and the use of function blocks, the system is easy to maintain and extend. All seven test cabinets operate independently but are managed by the same program, which simplifies updates and ensures consistency.

8 Visualization

This section discusses the requirements for the tester visualization, provides an overview of the main interface layout, and presents a more detailed review of its main components.

8.1 Visualization Purpose and Requirements

In addition to the program itself, the system needs to have an intuitive graphical user interface (HMI – Human-Machine Interface, also referred to as visualization). The interface should allow operators to control and view test progress and adjust test parameters. There also should be a separate view of each cabinet, even though all cabinets are managed through a single PLC CPU.

The main visualization requirements include:

- Switching between individual test cabinets
- Displaying temperatures from each stand within one cabinet
- Providing feedback on the status of heaters and cooling fans
- Allowing configuration of thermal and power cycle parameters
- Inputting and displaying each EUT information (to be written to the database)
- Indicating system status, active warnings, and errors
- Displaying real-time current received from DAQs

8.2 Instruments Used

To create the visualization, the built-in HMI editor in CODESYS was used. This solution is convenient because everything is done in one place – the visualization is directly connected to the PLC program, making it easy to link variables, update settings, and maintain the project. It also follows the company's internal standards, where all testers use CODESYS for both programming and visualization. Normally, HMI systems are shown on separate touch panels or operator terminals. In this case, however, the visualization is simply run on the main laptop of the tester. This allows operators to control the system both locally at the test site and remotely, by connecting to the laptop through Remote Desktop Connection.

8.3 Interface Overview

The entire tester interface, including inputs, indicators, and control elements, is organized within a single visualization page per cabinet. Each test cabinet has its own dedicated page, and operators can switch between cabinets using a navigation bar located at the top of the interface. The interface is divided into nine main functional areas, which help organize the information and controls clearly. These areas are:

1. Cabinet Switcher
2. Main Test Control Block
3. Temperature Control Status Block
4. Stand Control Block
5. Permissions Block
6. Temperature Bands Block
7. Cycle Table Block
8. Temperature Block
9. EUT Table Block

An overview of the main interface layout is shown in Figure 17 below, where the numbered blocks correspond to the list above.

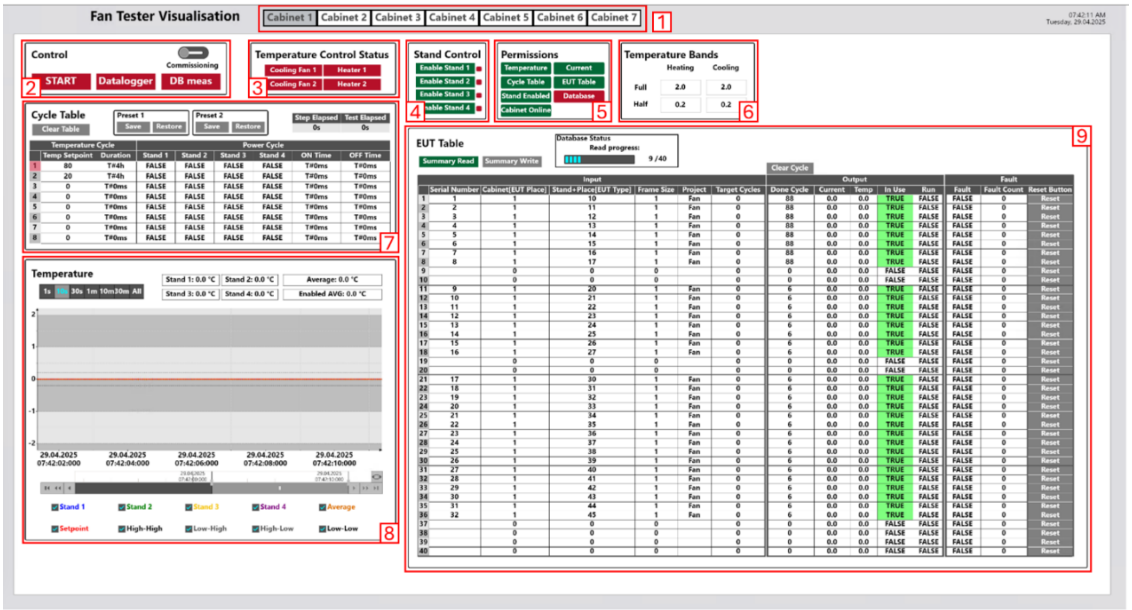


Figure 17. Tester Main Interface Overview

8.4 Interface Parts

This section describes each main functional part of the interface in more detail.

Cabinet Switcher (1)

The Cabinet Switcher allows the user to switch the interface view between different cabinets for control. The currently selected cabinet is highlighted with a grey button. Additionally, the outline of a cabinet button glows green when a test is running on the corresponding cabinet, making it easy to see which cabinets are active at the moment.

Main Test Control Block (2)

This block contains the Start, Datalogger, and DB Meas buttons, as well as the Commissioning switch.

- Start button: Starts the test and activates the Datalogger button as well. When stopped, it disables Datalogger and DB Meas Buttons.
- Datalogger button: Enables reading current measurements from the DAQ for each EUT.
- DB Meas button: Activates writing of measurement data into the database.
- Commissioning switch: Selects which database table to use for storing measurements. If enabled, the program writes measurements into the commissioning database. Otherwise, data is written into the main production database.

Temperature Control Status Block (3)

This block indicates the status of the temperature control elements – heating resistors and cooling fans. An element is shown in green if it is ON and red if it is OFF.

Stand Control Block (4)

The Stand Control buttons allow the operator to enable or disable individual fan stands for testing. Each enable button has a small indicator next to it that shows the real-time status of the corresponding stand: green if the stand is currently running, red if it is not.

Permissions Block (5)

This block displays the status of all system permissions. Green – permission is ok, Red – permission is not ok. The permission system is described in detail in Section 7.4.6.

Temperature Bands Block (6)

This block allows the user to configure how the temperature control stages behave based on the deviation from the temperature setpoint. The detailed logic of temperature control is explained in Section 7.4.4.

Cycle Table Block (7)

The Cycle Table is used to configure both the thermal and power cycles within a single thermal step. The table supports up to eight thermal cycle steps. For each step, the operator can set the temperature setpoint, define the duration of the step, choose which stands participate in the power cycle and set ON/OFF intervals for the power cycle.

At the top of the table, there are two preset buttons for saving and restoring configurations, a step timer, a total test timer, and a clear table button.

Temperature Block (8)

This block displays the temperatures from each stand individually, the overall average temperature and the average temperature calculated only from active stands.

EUT Table Block (9)

The EUT Table is divided into three sections: Input, Output, and Fault.

- **Input:** The operator enters data for each EUT, such as serial number, stand assignment, and dimensions.
- **Output:** Displays real-time test results like temperature and current for each EUT.
- **Fault:** Shows the number of detected faults for each EUT and provides a reset button.

Additionally, above the table, there is a Database Status window displaying the status of the database connection, a button for Resetting Done Cycles and Summary Write and Summary Read buttons to synchronize EUT information with the database

9 Results and Evaluation

This section gives an overview of what was achieved with the new system, what improvements were made compared to the old version, and how it affects the daily workflow. It also makes an analysis on how much time can be saved thanks to the new design.

9.1 System Overview and Achievements

The main objective of this project was to improve and automate work of the industrial fan tester. The new system shows improvements that make the testing process faster, more flexible, and more reliable. Table 3 demonstrates the key improvements for the tester that were achieved and how that affects the testing process:

Table 3. Main improvements of the tester achieved

Improvement	**Description**
Remote Access	Test control and configuration can now be made remotely through the visualization, removing the need for most physical control.
Separated EUTs and Heaters	Control of EUTs and heating elements is now independent and allows for more flexible test configurations and more sustainable energy management.
Intuitive Graphical Interface	The visualization provides centralized control of the test process and real-time feedback.
Database Integration	All test and EUT data is saved to the common database for more convenient and reliable data storage.
Advanced Step Configuration	More precise test cycle step configuration allows experimenting with more complex and customized test programs that could be better than current solution.
Permission System	A permission system verifies that the system is ready for the test and identifies errors as they arise, which was very lacking before.

These improvements significantly reduce manual workload, lower the risk of configuration errors, and make the system easier to maintain. In addition, the modular structure of the PLC code ensures that the solution can be extended to support additional cabinets or new test scenarios in the future.

9.2 Workflow Comparison

This section shows a comparative analysis of the typical workflow of fan testing before and after the implementation of the new control system. The testing process is broken down into three main stages:

- New Test Setup
- Running Test Maintenance
- Test Completion

Each stage is visualized using a workflow diagram comparing step-by-step process for old and new systems.

The diagrams represent ideal conditions, assuming everything goes as intended without any errors. In practice, the old system often required significantly more time due to unexpected issues – such as unstable data exports, DAQ software being slow, or the need for physical presence to resolve some configuration changes which requires walking back and forth between the office and the test area, which takes additional time. The new system minimizes these risks by enabling remote access with real-time monitoring and database usage.

9.2.1 New Test Setup Workflow

Figure 18 shows a comparison between the old and new workflow for the process of setting up a new test, which needs to be performed every month.

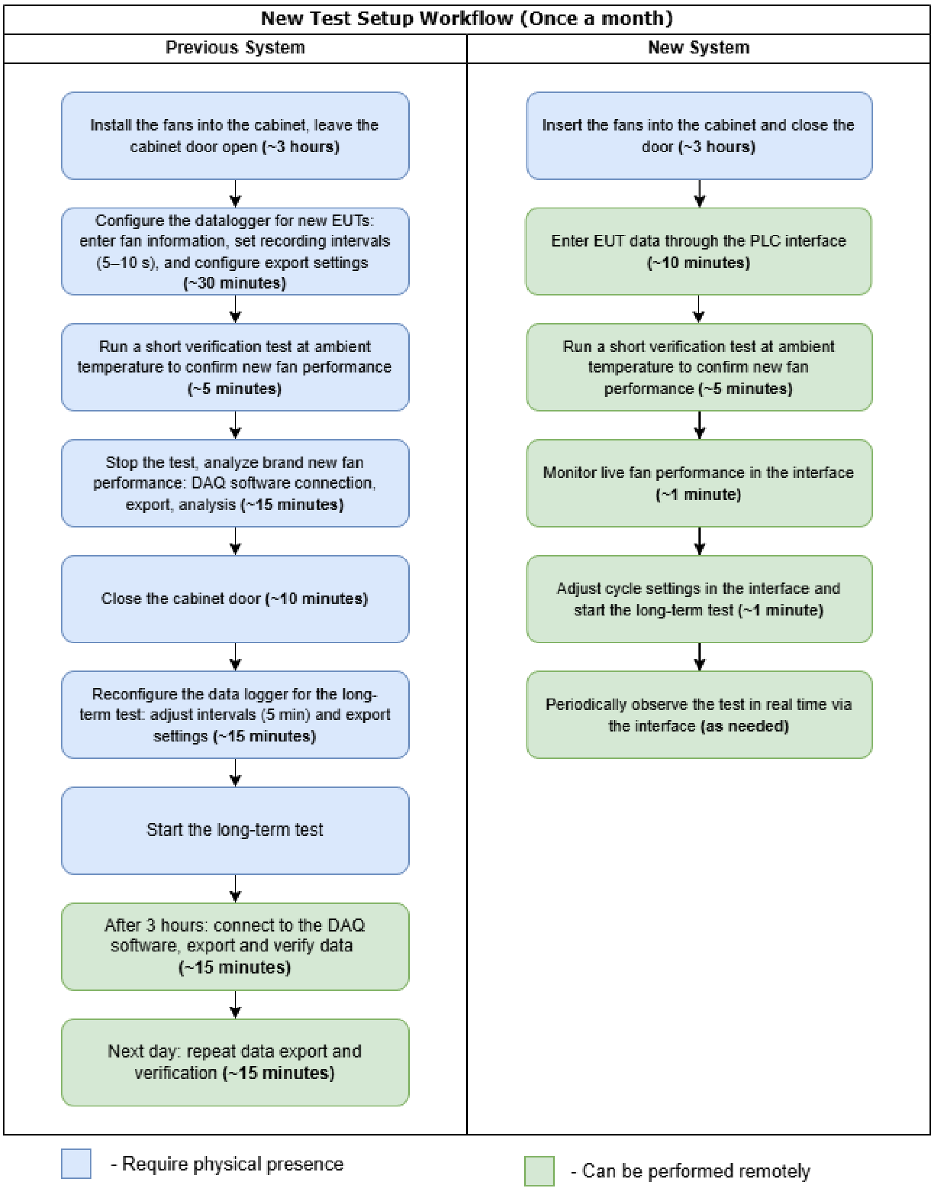


Figure 18. Workflow for preparing a new long-term test

Important improvement in the new system is the ability to start an ambient temperature verification test with the cabinet door closed. Previously, the system did not allow test launch without enabling heaters as well, which made it impossible to run ambient testing with door closed as it would heat up. This forced the operator to stay near the cabinet just to manually close the door later. This task takes a lot of time because of poor door design. Setting up the DAQ software that was used in the old system is very slow and often leads to errors. Now, this process is fully built into the PLC interface and can be done much quicker.

9.2.2 Running Test Maintenance Workflow

Figure 19 shows a comparison between old and new workflow for maintenance of a running test, which needs to be performed every week.

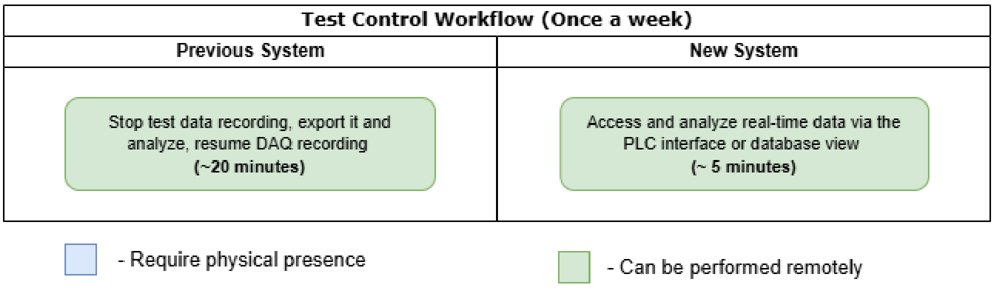


Figure 19. Running test maintenance

In the previous system, the operator was required to stop DAQ recording weekly to export and analyze the data. This was necessary because the software could not handle large file sizes of recorded data. However, this process was very unreliable, and data exports sometimes were getting corrupted, leading to loss of an entire week's worth of test results, which means more lost time. In the new system, all data is continuously written to a centralized database, and there is no need to interrupt the test for regular monitoring and exports. While periodic checks are still recommended, they can now be performed in real time using visualization.

9.2.3 Test Completion Workflow

Figure 20 shows a comparison between old and new workflow for a process of test completion, which needs to be performed every 3-6 months.

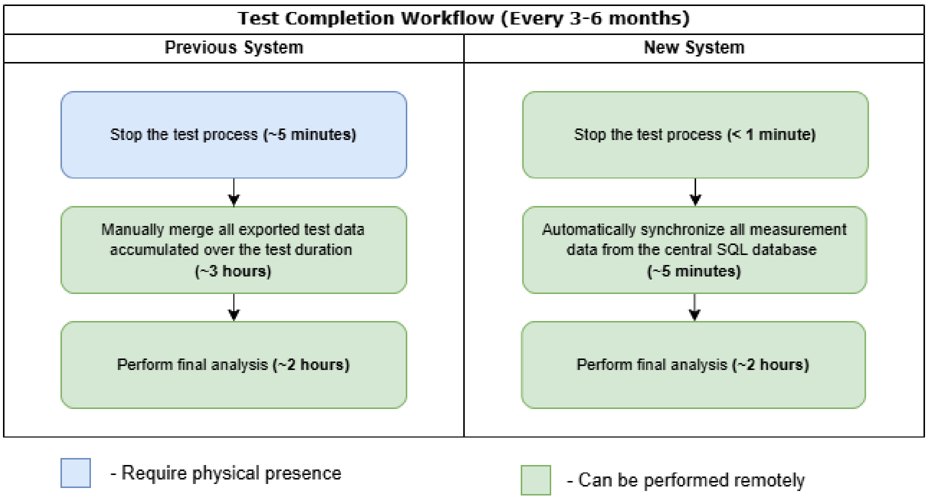


Figure 20. Workflow of test completion process

When the test is completed, the old workflow required operators to manually merge all previously exported data. This process was very time-consuming and introduced a risk of mistakes or missing data. Now using the new system, entire test data can be automatically synchronized from the database to an external file, which then could be used for analyzing. This reduces the amount of effort and time a lot. In addition to the reduced time required for working with the tester, it is important to note that many stages of the workflow in the new system can now be performed remotely without the need to be physically present at the testing site. This saves time and reduces the effort required, as there is no need to walk to a testing location in a different building.

9.3 Time Savings and Process Efficiency

The following time saving analysis (see Table 4) is based on the workflows discussed in the previous section, comparing the old and new system for the tasks needed for operating the test setup. The table presents approximate durations for each stage in hours, and an estimate of annual time saved based on a workflow frequency.

Table 4. Estimated time savings between the previous and new system

Stage	**Previous System (h)**	**New System (h)**	**Time Saving (h)**	**Frequency**	**Annual Saving (h/year)**
New Test Setup	~ 4.75	~ 3.28	~ 1.5	Every month	~ 18
Running Test Maintenance	~ 0.33	~ 0.08	~ 0.25	Every week	~ 13
Test Completion	~ 5	~ 2	~ 3	Every 3-6 months	~ 8
Total	~ 10	~ 5.33	~ 4.75	-	~ 39

These values represent ideal conditions. In reality, unexpected issues often occur during the testing process, such as DAQ connection problems, corrupted data or tester physical issues. This could increase the time required for certain steps, especially in the previous system. It's also important to note that the estimates reflect time calculations and savings per one cabinet, and total time spent depends on how many cabinets are actively in use at the given time. Since the number of tests running can be very different, calculating exact time savings for the entire system is difficult. However, even with one cabinet calculation, the new system can reduce workload by approximately 39 hours per year (about one full working week). With more cabinets working this number increases.

10 Summary

The goal of this thesis was to modernize and automate an industrial fan reliability tester at an electronics company in Estonia, addressing limitations of the existing system. The original tester, based on separate and manual control components, had problems such as lack of remote access, unreliable data collection, suboptimal wiring leading to energy waste, and no real-time problem detection. Work aimed to enhance testing efficiency, ease of use, and scalability through automation.

The solution involved integrating a PLC and developing a dedicated control program in Structured Text, which was chosen within the PLC programming languages analysis. This upgrade resolved the issues effectively. Remote access was achieved through a centralized PLC with distributed I/O modules via PROFINET communication, allowing operators to monitor and control tests remotely. Data collection was automated and made more reliable utilizing a centralized database, eliminating data loss risks. Independent control of EUT fans and heating elements was implemented, improving temperature management effectiveness and energy efficiency. A permission system was added to detect and address issues in real time, making it more reliability.

Main results include a fully automated system managing multiple test cabinets simultaneously, with a PLC program that controls the test process and replaces most previously used control systems. The graphical user interface provides intuitive control and real-time feedback. Workflow comparisons demonstrate significant time savings (approximately 39 hours annually per cabinet) due to optimized process. Hardware upgrades, such as redesigned resistor connections improve system stability even more.

In conclusion, all goals and tasks of this thesis were successfully achieved. The tester has been upgraded with physical changes, a new control program, and a user interface. That not only improves efficiency, ease of use, saves time but also enable the company to conduct more complex tests and experiment with cycle configurations. This could potentially lead to discovering more effective testing methods, improving the company’s capabilities in reliability testing.

References

[1] ScienceDirect, “Accelerated Life Test.” Accessed: May 02, 2025. [Online]. Available: https://www.sciencedirect.com/topics/engineering/accelerated-life-test

[2] Ansys, “What is Highly Accelerated Life Testing (HALT)?” Accessed: May 02, 2025. [Online]. Available: https://www.ansys.com/blog/planning-a-halt

[3] Relyence, “An Introduction to ALT Analysis.” Accessed: May 04, 2025. [Online]. Available: https://relyence.com/2024/02/20/an-introduction-to-alt-analysis/

[4] IPC International, “IPC-9591,” 2006.

[5] Computer Security Resource Center, “PLC Definition.” Accessed: Apr. 15, 2025. [Online]. Available: https://csrc.nist.gov/glossary/term/programmable_logic_controller

[6] PLC Table, “Use PLC in Industries.” Accessed: Apr. 15, 2025. [Online]. Available: https://www.plctable.com/use-plc-in-industries/

[7] IndMALL Automation, “What Are The Advantages Of PLC?” Accessed: Apr. 15, 2025. [Online]. Available: https://www.indmallautomation.com/faq/what-are-the-advantages-of-plc/

[8] MARIAN, “PROGRAMMABLE LOGIC CONTROLLERS (PLC).” Accessed: Apr. 15, 2025. [Online]. Available: https://www.marian.ac.in/public/images/uploads/pdf/online-class/eee/MODULE%206iia.pdf

[9] AutomationDirect, “History of the PLC.” Accessed: Apr. 15, 2025. [Online]. Available: https://library.automationdirect.com/history-of-the-plc/

[10] Mohammad Abdelkareem Alia, “Ladder Diagram example,” ResearchGate. Accessed: Apr. 15, 2025. [Online]. Available: https://www.researchgate.net/figure/The-PLC-ladder-diagram_fig3_272666766

[11] Schneider Electric, “Functional Block Diagram Example.” Accessed: Apr. 15, 2025. [Online]. Available: https://product-help.schneider-electric.com/Machine%20Expert/V1.1/en/SoMProg/SoMProg/FBD_LD_IL_Editor/FBD_LD_IL_Editor-4.htm

[12] Process Solutions, “Basic Architecture of a Programmable Logic Controller.” Accessed: Apr. 15, 2025. [Online]. Available: https://processsolutions.com/basic-architecture-of-programmable-logic-controller/

[13] International Society of Automation, “What You May Not Know About SoftPLCs.” Accessed: Apr. 15, 2025. [Online]. Available: https://www.isa.org/intech-home/2022/february-2022/departments/what-you-may-not-know-about-softplcs

[14] Tamim Ahmmad, “Structured Text Example,” Medium. Accessed: Apr. 15, 2025. [Online]. Available: https://tamim-ahmmad.medium.com/why-structured-text-st-in-plc-programming-is-case-insensitive-cb57186789ef

[15] ABB, “Overview of the IEC 61131 Standard.” Accessed: Apr. 15, 2025. [Online]. Available: https://library.e.abb.com/public/81478a314e1386d1c1257b1a005b0fc0/2101127.pdf

[16] Vladimir Romanov, “Top 5 Most Popular Types of PLC Programming Languages,” SolisPLC. Accessed: Apr. 15, 2025. [Online]. Available: https://www.solisplc.com/blog/plc-programming-languages

[17] International Electrotechnical Commission, “IEC 61131-3,” 2013.

[18] Schneider Electric, “Sequential Function Chart Example.” Accessed: Apr. 15, 2025. [Online]. Available: https://product-help.schneider-electric.com/Machine%20Expert/V1.1/en/SoMProg/SoMProg/SFC_Editor/SFC_Editor-3.htm

[19] P. E. Gary L. Pratt, “Which IEC 61131-3 Programming Language is best? Part 2,” Control Engineering. Accessed: Apr. 15, 2025. [Online]. Available: https://www.controleng.com/which-iec-61131-3-programming-language-is-best-part-2/

[20] Beckhoff, “Continuous Function Chart Example.” Accessed: Apr. 15, 2025. [Online]. Available: https://infosys.beckhoff.com/english.php?content=../content/1033/tcplccontrol/925312139.html

[21] Farnell, “Curtis Pulse Counter.” Accessed: Apr. 15, 2025. [Online]. Available: https://de.farnell.com/en-DE/curtis-instruments/210-1m073-001a-k01-12-48vdc/counter-totaliser/dp/516272

[22] Farnell, “Finder Smart Timer Relay.” Accessed: Apr. 15, 2025. [Online]. Available: https://fi.farnell.com/finder/84-02-0-024-0000/smartimer-nfc-prog-dpdt-16a-24v/dp/2707375

[23] Keysight, “DAQ970A / DAQ973A Data Acquisition System.” Accessed: Apr. 15, 2025. [Online]. Available: https://www.keysight.com/zz/en/products/modular/data-acquisition-daq/digital-acquisition-benchtop-system/daq970a-data-acquisition-system.html

[24] Rheonics, “PROFINET fundamentals.” Accessed: May 02, 2025. [Online]. Available: https://support.rheonics.com/en/support/solutions/articles/81000410678-profinet-fundamentals

[25] Texas Instruments, “How to Select the Right Industrial Ethernet Standard: PROFINET,” 2023. Accessed: May 02, 2025. [Online]. Available: https://www.ti.com/lit/ta/ssztbz0/ssztbz0.pdf

[26] Jared Worth, “What is PROFINET? Everything you need to know,” Engineer’s Toolbox. Accessed: May 12, 2025. [Online]. Available: https://toolbox.igus.com/9233/what-is-profinet

[27] CODESYS, “Why CODESYS?” Accessed: May 02, 2025. [Online]. Available: https://www.codesys.com/ecosystem/discover-codesys/why-codesys/

Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I Daniel Poljanski

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “Modernization and Automation of a Reliability Tester for Industrial Fans”, supervised by Kristina Vassiljeva
   1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;
   1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.
3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

12.05.2025

---

[1] The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.

The generative AI tool ChatGPT was used during the writing process to assist with translation and to improve the clarity and readability of the text.