MiraTherm Radiator Thermostat Software Specification

Requirement Specification for a Master Project

Fulda University of Applied Sciences
Department of Electrical Engineering

submitted by:	Alexander Menzel
Degree Program:	M.Eng. Embedded Systems (PO 2020)
Matriculation Number:	1316814
Advisor:	Prof. Dr. Uwe Werner

October 30, 2025

Contents

Change Log

Table 1: Document Change Log

№	Date	Version	Changed Chapters	Change Type	Editor
1	30.10.2025	1.0	All	Initial version	A. Menzel (AM)

Change testtext

Add test

Test note

Test highlight

[AM 1] v1.0: Example change

[AM 2] v1.0: Example addition

[AM 3] v1.0: Example comment

[AM 4] v1.0: Example highlight

1 Introduction

In this chapter, the purpose of this document and the context of the project are described.

1.1 Document Purpose

This document provides a requirements specification for software of a Micro Controller Unit (MCU) based radiator thermostat, which will be developed as part of a master project at Fulda University of Applied Sciences. This software should implement basic consumer functions and could be used as a base for research, development and production of smart heating controllers or thermostats.

1.2 Project Context

The master project will be realized as part of a bigger interdisciplinary development named “MiraTherm Radiator Thermostat”, which includes the following areas:

- **Mechanics**: Development of the thermostat’s power transmission mechanism for proper function with commonly used radiator valves, followed by the design of an enclosure.
- **Control algorithms**: Engineering of control algorithms to be used by the thermostat.
- **Electronics**: Development of the thermostat’s Printed Circuit Board (PCB) and its integration with mechanical components.
- **Software**: The subject of this work, development of the thermostat’s software and its integration with PCB components.

2 Concept

In this chapter, the overall concept and approach for the development of the radiator thermostat software is described. Additionally, the required hardware for development and testing is outlined.

2.1 Solution approach

In this project, a basic software for the device should be implemented including hardware drivers and general program logic. The description of the eQ-3 eqiva Bluetooth from [1] will be used as a reference for defining the functional scope of the software to be developed.

The software should be designed ready for prospective integration of the control algorithms and Matter-over-Thread standard. (For further details about this standard see [2].) The device is supposed to be used for smart home applications, specifically for the integration of the thermostat in Home Assistant, apps like Google Home and/or custom Application Programming Interfaces (APIs).

2.2 Hardware requirements

To ensure a degree of independence from the PCB design and mechanics, the software will be developed using a hardware set that resembles the final thermostat in terms of components and interfaces. This approach enables early software development and testing before a hardware prototype is available.

A block diagram of the hardware setup for software development and testing is shown in Figure 2.1.

For the software development and testing, the following hardware components are required:

- **P-NUCLEO-WB55** - MCU development board with Matter-over-Thread standard support
- **eQ-3 eqiva Model N** - Radiator thermostat with a C300 3V motor and gear box for disassembly (available)

- **DRV8833** - Motor driver module
- **Shunt resistor** - For current measurement of the motor
- **1.3” OLED Display incl. SH1106** - Display with an embedded driver
- **KY-040** - Rotary encoder
- **Connecting wires**
- **Breadboard(s)**
- **Power supply** - laboratory power supply or batteries

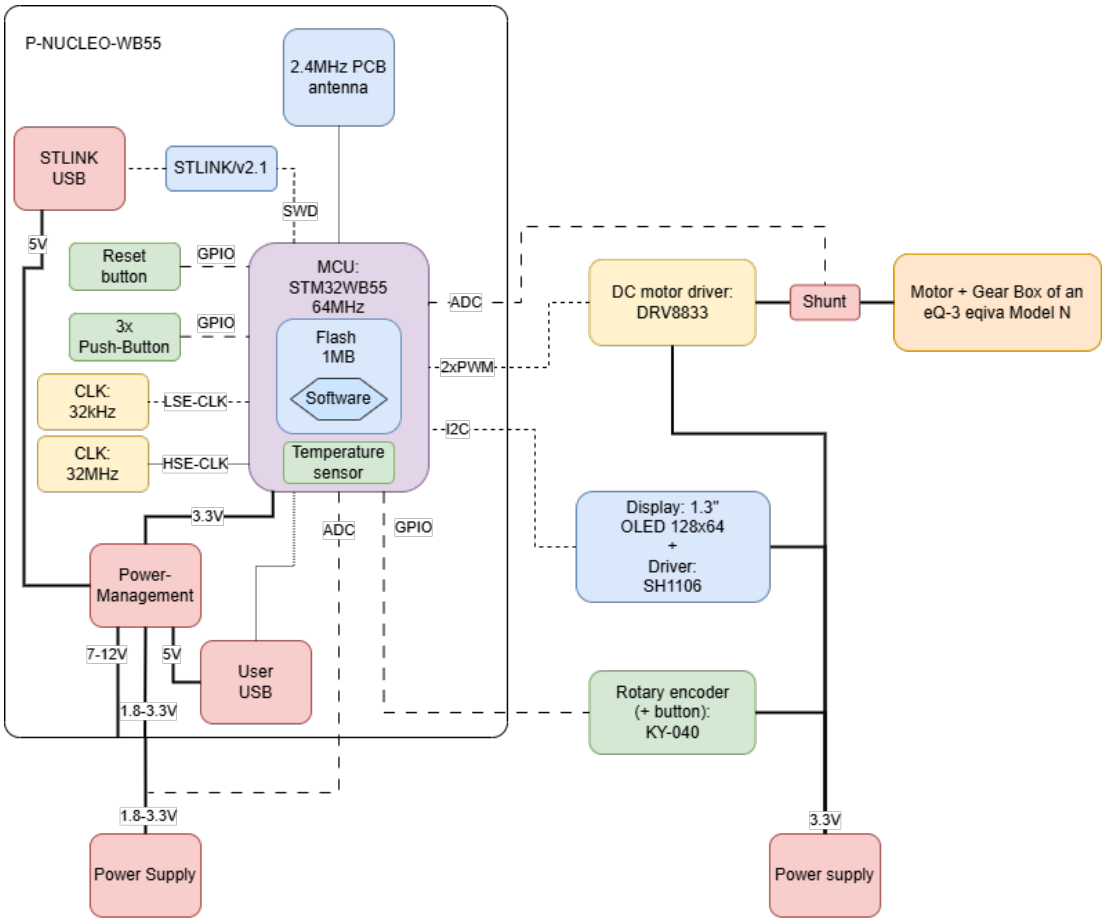

Figure 2.1: Hardware block diagram of the hardware set for software development and testing

3 Requirements

3.1 Functional Requirements

REQ 1: Configuration Routines

The system shall provide following configuration routines for initial setup:

- **Basic Configuration Routine (BCR)** Manual configuration via control wheel and buttons on the device without wireless connectivity.
- **Advanced Configuration Routine (ACR)** Particularly automatic wireless configuration via Matter standard and/or Thread protocol.

REQ 2: Prototype Features

In the prototype version of the thermostat software, only the BCR shall be fully supported. Wireless connectivity and the ACR shall not be implemented.

REQ 3: Configuration Routine Choice

The user shall be able to choose between the BCR (“Manual Configuration”) and ACR (“Automatic Configuration”) during the initial setup process. The setup process shall be run after battery insertion if no previous configuration exists, otherwise the system shall resume normal operation with existing settings.

REQ 4: Advanced Configuration Routine Blocking

The system shall block access to the ACR in the prototype version of the thermostat software.

REQ 5: Date and Time Configuration

The system shall request date and time configuration as the first step in the BCR. The user shall be able to set the year, month, day, hour, and minute through the control wheel interface with confirmation capability by the control wheel button. The system shall ask whether the user wants to activate automatic summer/winter time switching after setting the date and time.

REQ 6: Installation Command

The system shall wait for an installation command from the user as the second step in the BCR. The text “Begin Installation?” shall be displayed. The user shall initiate the installation command by pressing the control wheel button, which will start the valve adaptation procedure.

REQ 7: Valve Adaptation

The system shall perform an automatic adaption run as the third step in the BCR to detect and adapt to the specific valve characteristics. The adaptation procedure shall:

1. Display the message “Adaptation...” during the procedure
2. Move the motor to the fully opened position
3. Move the motor to the fully closed position
4. Incrementally move the motor to the fully open position
5. Calculate the valve stroke range
6. Validate that measured travel distance matches expected 4.3 mm linear travel
7. If valve characteristics are outside acceptable ranges, the system shall display following error messages:
    - **F1**: Valve drive sluggish (motor movement is impeded or extremely slow)
    - **F2**: Actuating range too wide (measured valve stroke exceeds expected parameters)
    - **F3**: Adjustment range too small (measured valve stroke is below acceptable minimum)

8. If an error occurs, the system shall allow reversal of the adaption run by pressing the control wheel button, returning to the waiting of the installation command. Otherwise, the system shall proceed to the main display state.

REQ 8: Current Extremes Definition

During the software development phase, the current extremes for the motor during the adaptation procedure shall be defined to detect fully opened and fully closed valve positions.

REQ 9: Main Display State

In the main display state, the system shall display at least the following informations:

- Current temperature
- Target temperature
- Time slot of the target temperature
- Time and day of week
- Operational mode indicator
- Battery charge in percentage
- State of the valve in percentage

REQ 10: Operational Modes

The system shall support following operational modes:

- **Manual Mode**: The user can manually set the target temperature.
- **Boost Mode** is described in **??**.

REQ 11: Temperature Range and Resolution

In the Manual Mode, the system shall allow setting target temperatures in the range of $5.0°C$ to $30.0°C$ in increments of $0.5°C$. The system shall support following adjustments:

- **OFF** state, when user tries to set the target temperature below $5.0°C$: Fully closes the valve
- **ON** state, when user tries to set the target temperature above $30.0°C$: Fully opens the valve

REQ 12: Configuration Menu

The system shall provide a configuration menu accessible via the main display state by pressing SW1. The menu shall list the following configurable options:

- Temperature offset configuration
- Inactivity timeout setting
- Automatic summer/winter time switching

REQ 13: Temperature Offset Configuration

The system shall allow setting a temperature offset between $-3.5°C$ and $+3.5°C$ in increments of $0.5°C$. The default value shall be $0.0°C$. The offset shall be applied to the measured temperature to calculate the effective temperature.

REQ 14: Inactivity Timeout

If no interactions by the user occur for a default period of 30 seconds, the system shall turn the display off.

REQ 15: Automatic Summer/Winter Time Switching

The system shall automatically switch between summer and winter time.

REQ 16: Boost Mode

The system shall provide a boost mode that immediately opens the heating valve to 80% for 5 minutes after pressing the control wheel button. The remaining time shall be displayed as a countdown in seconds. The function shall be deactivatable at any time by pressing the control wheel button.

REQ 17: Automatic Descaling Routine

The system shall perform an automatic routine descaling run once a week on Saturday at 12:00 to protect against calcification of the valve. The descaling procedure shall:

1. Display “Maintenance...” indicator on the display to inform the user
2. Move the motor through its full stroke range within a maximal timeframe of 5 minutes

The calcification protection routine shall continue running in all operational modes. User commands shall be ignored during the descaling routine.

REQ 18: Persistent Settings Storage

The system shall persist all user-configured settings in non-volatile memory. These settings shall survive battery removal and replacement.

REQ 19: Factory Reset

The system shall provide a factory reset function (“rES”) that clears all user settings and returns to default configuration, with a confirmation prompt (“COnF”) to prevent accidental data loss.

3.2 Non-Functional Requirements

REQ 20: Power Supply and Battery Life

The system shall operate on 2x 1.5V LR6/mignon/AA alkaline batteries with an expected battery life of approximately 2 years.

REQ 21: User Interface Responsiveness

The system shall provide immediate visual feedback for all user interactions through the control wheel and Mode/Menu button, with display updates occurring within acceptable latency for user perception.

4 Planning

4.1 Time plan

The master project will presumably have the duration of 13 Calendar Weeks (CWs), which are divided into:

- CWs 44-45: Software requirements analysis.
- CW 46: Software architecture design.
- CW 47: Design of software interfaces.
- CW 48: Implementation and tests of software drivers.
- CWs 49-51: Implementation and tests of program logic.
- CWs 52-02: Paper writing.
- CWs 03-04: Final review and submission of the paper.

Each calendar week will approximately consist of $\frac{150\text{h}}{13} \approx 11.5$ hours of work.

4.2 Responsibilities

The whole work will be carried out by Alexander Menzel. The advisor for this master project will be Prof. Dr. Uwe Werner.

E-Mails and questions should be answered within 2 working days by both parties. If any problems arise, the advisor has to be informed as soon as possible.

Bibliography

[1] eQ-3 AG, *Operating Manual BLUETOOTH® Smart Radiator Thermostat UK eqiva CC-RT-M-BLE-EQ*, May 2018.

[2] Wikipedia contributors, *Matter (standard) — Wikipedia, the free encyclopedia*, [Online; accessed 28-October-2025], 2025. [Online]. Available: *https://en.wikipedia.org/w/index.php?title=Matter_(standard)&oldid=1318221979*

List of Abbreviations

CW	Calendar Week
MCU	Micro Controller Unit
PCB	Printed Circuit Board
API	Application Programming Interface
BCR	Basic Configuration Routine
ACR	Advanced Configuration Routine

