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

November 4, 2025

Contents

Change Log

Table 1: Document Change Log

№	Date	Version	Changed Chapters	Change Type	Editor
1	4.11.2025	0.1	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 a device prototype 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.

Due to time constraints, the implementation of control algorithms, wireless connectivity, advanced features, and energy management will be considered out of scope for this project. Instead, the focus will be on developing a solid software foundation that can be extended in the future.

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 Required Hardware

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

A block diagram of the prototype hardware for software development and testing is shown in Figure 2.1. The following components are required:

- **P-NUCLEO-WB55** - MCU development board with Matter-over-Thread standard support

- **eQ-3 eqiva Model N** - Radiator thermostat for disassembly and reuse of its C300 3V motor, gear box and valve connector
- **DRV8833** - Motor driver module
- **Shunt resistor** - 1Ω shunt resistor for current measurement of the motor
- **1.3" 128 × 64 Organic Light-Emitting Diode (OLED) Display incl. SH1106** - Display with an embedded driver
- **KY-040** - Rotary encoder
- **Connecting wires**
- **Breadboard(s)**
- **Power supply** - laboratory power supply or batteries

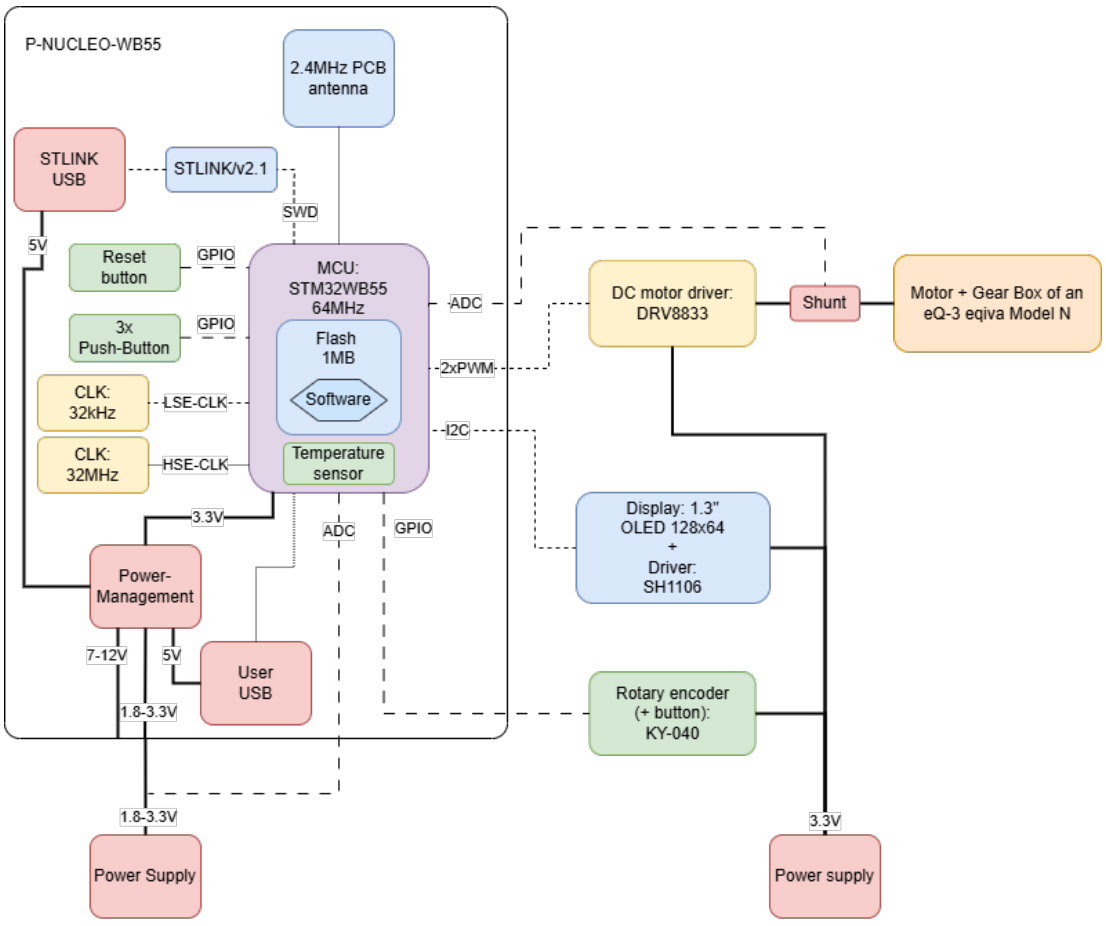

Figure 2.1: Block diagram of the prototype hardware for software development and testing

3 Requirements

3.1 Functional Requirements

REQ 1: Display

The system shall integrate a display.

REQ 1.1: Prototype Display

In the prototype version, a 1.3” 128×64 OLED display with SH1106 controller over the Inter-Integrated Circuit (I2C) protocol shall be used.

REQ 2: Rotary Encoder

The system shall integrate a rotary encoder as a control wheel via Quadrature Decoder (QDEC) interface.

REQ 2.1: Prototype Rotary Encoder

In the prototype version, a KY-040 rotary encoder module shall be integrated as the control wheel.

REQ 3: Push Buttons

The system shall integrate the following push buttons via GPIOs:

- **Mode** button
- **Centered** button
- **Menu** button

REQ 3.1: Prototype Push Buttons

In the prototype version, the push buttons SW1 and SW3 from the P-NUCLEO-WB55 and KY-040 shall be used:

- SW1 as **Mode** button
- KY-040 push button as **Centered** button
- SW3 as **Menu** button

REQ 4: Motor Control

The system shall support control of motor driver hardware.

REQ 4.1: Prototype Motor Control

The system shall support control of the eQ-3 eqiva Model N C300 3V motor through a DRV8833 motor driver module using two Pulse Width Modulation (PWM) signals.

REQ 5: Motor Current Measurement

The system shall measure motor current consumption.

REQ 5.1: Prototype Motor Current Measurement

In the prototype version, the system shall measure motor current consumption using an Analog-to-Digital Converter (ADC) channel connected to a 1Ω shunt resistor in series with the motor circuit.

REQ 6: Power Supply Voltage Measurement

The system shall measure the power supply voltage. The measured voltage shall be used to calculate the battery charge level as a percentage.

REQ 6.1: Prototype Power Supply Voltage Measurement

In the prototype version, the system shall measure the power supply voltage using an ADC channel.

REQ 7: Motor Current Extremes Definition

During the software development phase, the motor current extremes shall be defined to detect maximal and minimal valve pin positions. This definition shall be based on empirical measurements and analysis of the motor's current consumption during operation.

REQ 8: Temperature Measurement

The system shall measure the ambient temperature using a digital temperature sensor.

REQ 8.1: Prototype Temperature Sensor

In the prototype version, the system shall use a temperature sensor integrated into the STM32WB55 MCU of the P-NUCLEO-WB55 development board.

NOTE 1: Pin Position Extremes Definition

After a successful adaptation procedure:

- Maximal valve pin position corresponds to the fully open valve position.
- Minimal valve pin position corresponds to the fully closed valve position.

REQ 9: Configuration Routines

The system shall provide the following configuration routines for initial setup:

- **Configuration via App (CVA)**: Partially automatic, wireless configuration via Matter standard and/or Thread protocol.
- **Configuration on Device (COD)**: Manual configuration via buttons on the device without wireless connectivity.

The setup process shall run after battery insertion if no complete configuration exists in persistent storage. Otherwise, the system shall resume normal operation with existing settings.

REQ 9.1: Configuration Routine Choice

The user shall be able to choose between the CVA ("Configuration via App") and COD ("Configuration on Device") before the initial setup process.

REQ 9.2: Prototype Configuration

In the prototype version of the thermostat software, only the COD shall be implemented. Wireless connectivity and the CVA shall not be supported.

REQ 9.3: Configuration Via App Blocking

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

REQ 10: Date and Time Configuration

The system shall request date and time configuration as the first step in the COD, or each time if no wireless connection is configured. The user shall be able to set the year, month, day, hour, and minute through the control wheel interface with confirmation capability via the centered button. After setting the date and time, the system shall ask whether the user wants to enable automatic summer/winter time switching.

REQ 11: Installation Command

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

REQ 12: Valve Adaptation

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

1. Display “Adaptation...” indicator page during the procedure.
2. Move the motor to the maximal valve pin position.
3. Move the motor to the minimal valve pin position.
4. Incrementally move the motor to the maximal valve pin position.
5. Calculate the valve stroke range.
6. Validate that the measured travel distance matches the expected 4.3mm linear travel.
7. If valve characteristics are outside acceptable ranges, display the 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, allow reversal of the adaptation run by pressing the centered button, returning to the waiting state for the installation command.

REQ 13: Daily Schedule Configuration

The system shall request daily schedule configuration as the fourth step in the COD. The user shall be able to set three time slots with corresponding target temperatures for the day using the control wheel interface with confirmation capability via the centered button. The first time slot shall always start at 00:00 and the last time slot shall always end at 23:59. After completing the daily schedule configuration, the system shall proceed to the main display page.

REQ 14: Main Display Page

On the main display page, the system shall display at least the following information:

- Current time in hours and minutes.
- Current temperature.
- Target temperature.
- Current time slot.
- Operational mode indicator.
- Battery charge as a percentage.

REQ 15: Configuration Menu

The system shall provide a configuration menu page accessible from the main display page by pressing the menu button. The menu shall list the following configurable options:

- Temperature offset configuration.
- Factory reset function.

REQ 16: Inactivity Timeout

If no user interactions occur for 30 seconds, the system shall turn the display off. The system shall return to the currently active page upon the next user interaction via any button.

REQ 17: Operational Modes

The system shall support the following operational modes:

- **Manual Mode**: The user can manually set the target temperature.
- **Auto Mode**: The system follows the heating program and sets the target temperature according to the current time slot.
- **Boost Mode**: Described in REQ 19.

REQ 18: Switching Operational Modes

The system shall allow switching between Manual Mode and Auto Mode using the mode button.

REQ 19: Boost Mode

The system shall provide a boost mode that immediately opens the heating valve to 80% for 5 minutes after double-pressing the control wheel button. Then the system shall display the remaining time instead of the target temperature on the main display page. 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 20: Temperature Range and Resolution

In 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 the following special states:

- **CLOSED state**: When the user sets a target temperature below 5.0°C, the valve shall be fully closed. Then the system shall display “CLOSED” instead of the target temperature on the main display page.
- **OPEN state**: When the user sets a target temperature above 30.0°C, the valve shall be fully opened. Then the system shall display “OPEN” instead of the target temperature on the main display page.

REQ 21: 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 22: Automatic Summer/Winter Time Switching

The system shall automatically switch between summer and winter time.

REQ 23: Persistent Settings Storage

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

REQ 24: Factory Reset

The system shall provide a factory reset function that clears all user settings and returns to default configuration. A confirmation prompt page (“Confirm Factory Reset?”) shall be displayed to prevent accidental data loss.

REQ 25: Automatic Descaling Routine

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

- Display “Maintenance...” indicator page.
- Ignore any user inputs during the routine.
- Move the motor through its full stroke range at maximum speed.

The calcification protection routine shall continue running in all operational modes except Boost Mode. If Boost Mode is active, the system shall wait until it is deactivated before beginning the descaling routine.

3.2 Non-Functional Requirements

REQ 26: Energy Efficiency

The system shall be designed for low power consumption to maximize battery life. Deep sleep modes and efficient peripheral management shall be implemented to minimize energy usage during idle periods.

REQ 27: User Interface Responsiveness

The system shall provide immediate visual feedback for all user interactions via the control wheel and buttons, 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.
- CW 49: Motor Current Extremes Definition.
- CWs 50-52: Implementation and tests of program logic.
- CWs 01-03: Paper writing.
- CWs 04-05: Final review and submission of the paper.

Each calendar week will approximately consist of $\frac{150\text{h}}{14} \approx 10.7$ 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 Figures

List of Abbreviations

CW	Calendar Week
MCU	Micro Controller Unit
PCB	Printed Circuit Board
API	Application Programming Interface
COD	Configuration on Device
CVA	Configuration via App
OLED	Organic Light-Emitting Diode
QDEC	Quadrature Decoder
GPIO	General Purpose Input/Output
I2C	Inter-Integrated Circuit
PWM	Pulse Width Modulation
ADC	Analog-to-Digital Converter

List of Symbols

Symbol	Meaning	Units
D	Distance	mm
U	Electric voltage	V
R	Electrical resistance	Ω
d	Diagonal	” (inches)
T	Temperature	°C
t	Time	ms, s
v	Valve position	%

