Wrepair Powerstation: Advanced HMI and Microcontroller Integration

HMI and Control

EE3L11: Bachelor Graduation Project

Zubair Al-Zubi
Alexander Gulam

Delft University of Technology

Wrepair Powerstation: Advanced HMI and Microcontroller Integration

HMI and Control

by

Zubair Al-Zubi
Alexander Gulam

Student Name	Student Number
Zubair Al-Zubi	5342007
Alexander Gulam	5411963

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended publicly on Thursday June 27, 2024 at 12:00 PM.

Project supervisors:	M. van Beusekom & A. Kaichouhi
Project duration:	April 22, 2024 – June 28, 2024
Faculty:	Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology

Cover: *Using modern technologies* by Sergey Nivens [50]

Abstract

This thesis explores the design and implementation of a Human Machine Interface (HMI) and microcontroller system for the Wrepair Powerstation, a charging and data transfer station for smart devices. The project focuses on developing an intuitive touchscreen interface for real-time monitoring and control of power allocation across multiple USB-C ports. The system allows users to adjust the power output of each port while respecting the limitations of connected devices and the overall power budget of the charging station.

The adjustable power allocation system improves the user experience by enabling users to prioritize certain devices for faster charging. This optimization enhances charging efficiency and provides greater flexibility in managing power distribution among connected devices.

The thesis also discusses the selection and integration of a suitable microcontroller to manage communication between power delivery modules, the touchscreen display, and other system components. The goal is to create a user-friendly and reliable Powerstation that meets the needs of professionals in the electronics repair industry.

Preface

This thesis is submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical Engineering at Delft University of Technology.

We would like to express our gratitude to our supervisors, Mark van Beusekom and Ali Kaichouhi, for their invaluable guidance and support throughout this project.

We also extend our thanks to the faculty and staff of the Electrical Engineering department for providing us with the resources and knowledge necessary to undertake this project.

Zubair Al-Zubi
Alexander Gulam
Delft, June 2024

Contents

Nomenclature

List of Abbreviations

Abbreviation	Definition
ADC	Analog-to-Digital Converter
DSI	Display Serial Interface
DP	DisplayPort
EEPROM	Electrically Erasable Programmable Read-Only Memory
ESD	Electrostatic Discharge
FPGA	Field-Programmable Gate Array
GUI	Graphical User Interface
HDL	Hardware Description Language
HDMI	High-Definition Multimedia Interface
HMI	Human Machine Interface
IT	Information Technology
I^2C	Inter-Integrated Circuit
LCD	Liquid-Crystal Display
micro-LED	Micro Light-Emitting Diode
MCU	Microcontroller Unit
CPU	Microprocessor
mini-LED	Mini Light-Emitting Diode
OLED	Organic Light-Emitting Diode
PC	Personal Computer
PD	Power Delivery
PCB	Printed Circuit Board
RAM	Random-Access Memory
ROM	Read-Only Memory
RTC	Real-Time Clock
SCL	Serial Clock Line
SPI	Serial Peripheral Interface
SoC	System on Chip
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
UX	User Experience
UI	User Interface
VHDL	VHSIC Hardware Description Language
WYSIWYG	What You See Is What You Get

1

Introduction

This chapter provides an overview of the Wrepair Powerstation project, starting with the motivation behind the initiative (section 1.1). It introduces the client for the project (section 1.2), outlines the assignment (section 1.3), and defines the problem we aim to solve (section 1.4). The chapter also describes the subdivision of the project group (section 1.5), presents a state-of-the-art analysis (section 1.6), and concludes with a thesis overview (section 1.7).

The rapid evolution of smart devices has highlighted the need for efficient and versatile charging solutions. The Wrepair Powerstation project aims to address this need by designing a power station that fits seamlessly into existing Wrepair Stations.

1.1. Motivation

The increasing popularity of USB Type-C as a universal charging standard necessitates the development of versatile and powerful charging solutions for devices such as laptops, phones, and other tools. Existing solutions often lack the ability to support multiple devices simultaneously and do not offer data transfer capabilities while ensuring user safety and efficient power distribution.

Currently, most charging stations lack an interface for visualizing the charging station's performance, as well as the capability for power allocation and data transfer through the ports. Additionally, voltage and current measurement data is typically absent.

This project addresses these shortcomings by designing a charging station (see Figure 1.2) that not only supports fast charging for various devices and data transfer but also seamlessly integrates into the Wrepair Powerstation. This integration enhances the utility and functionality of the Wrepair repair station, as shown in Figure 1.1.

A power station refers to a modular charging hub designed to provide reliable power and data transfer for various electronic devices.

1.2. Wrepair

The client for this project is Wrepair [70], a company that specializes in manufacturing and distributing tools for the repair industry and lab environments. One of their products is the Wrepair station, which is specifically designed to hold the necessary tools and parts for repairing mobile phones. This station is ESD safe and currently supports charging via USB-A ports, as shown in Figure 1.1.

Figure 1.1: Wrepair Station Model 40+ [70]

Figure 1.2: The stand-alone Powerstation

1.3. Assignment

The task at hand is to design a power station that can fit into existing Wrepair stations, capable of powering various devices through USB-C. It must support current power standards, ensuring safety for the user, external devices, and itself.

Moreover, the Wrepair Powerstation should be able to display relevant charging information of the connected USB-C ports. As an added bonus, it would be great if the device could pass through or detect data from the connected device to an external computer.

In the original project proposal [8], the following properties were stated:

1. Charging smart devices like smartphones, tablets, laptops, and wearables.
2. Displaying voltage, current, and power on a touchscreen.
3. USB-A and USB-C ports (6 in total).
4. Standard version: charging only.
   Plus version: data transfer capability in addition to charging.
5. Compatible with Wrepair stations through a docking station.
   Can be built-in to all existing Wrepair stations.
   Can also be used standalone.
6. USB-C ports max output: 140 W.
   USB-A ports max output: 36 W.
7. Safety: protection against overloading, overcurrent, overheating, and short-circuits.
8. Innovations such as: multi-device charging, USB power delivery, fast charging, etc.
9. Power division over a maximum output: to be determined later.
10. 230 V external power adapter.

1.4. Problem Definition

The high-level requirements for the project, presented in Table 1.1, follow directly from the properties specified in the initial assignment by Wrepair (section 1.3).

To refine these requirements into a more focused project scope, several key decisions were made:

- **USB Port Selection:** Initially, the design included both USB Type-A and USB Type-C ports. The decision was made to use only USB Type-C ports due to their superior fast charging capabilities, backward compatibility, and alignment with current industry standards. This simplification also made the design easier to implement. The output of these USB-C ports was adjusted to a maximum of 100 W after consultation with Wrepair, as this power level is more commonly supported and practical for the intended applications.

- **Scope Adjustment:** The original project scope aimed at delivering a final product, which included designing, building, and testing a custom PCB. However, after reviewing the feasibility and time constraints, the scope was shifted to creating a proof of concept. This change was necessary to first validate the project's unique aspects and manage the project within the available eight-week timeframe. As a result, some requirements were either removed or reclassified as trade-offs, depending on their relevance to the proof of concept.
- **Requirement Changes:** Due to the scope adjustment, several requirements were either removed or reclassified. These changes are highlighted in the requirement table:
  - Requirements highlighted in red were removed due to the change in project scope and time constraints.
  - Requirements highlighted in green were shifted to trade-off requirements, indicating that while they are desirable, they are not critical for the proof of concept.

Table 1.1 lists the revised list of high-level requirements, reflecting the project adjustments.

Table 1.1: High-level requirements of the project

Mandatory Requirements or Constraints
The product must have six USB-C charging ports.
The product must provide a maximum output power of 100 W per USB-C port.
The product must operate at a maximum of 300 W.
The product must ensure safety for both users and connected devices during operation.
The product must be designed to operate reliably and efficiently in an electronics lab environment, including resistance to electromagnetic interference and durability for frequent handling.
The product must be user-friendly, featuring a straightforward and intuitive interface that allows users to easily navigate and operate all functions with minimal training.
The product must be able to measure current, voltage, and power per USB-C port.
The product must display the current, voltage, and power used by the USB-C port.
The product must be able to be used as a stand-alone Powerstation.
The product must fit in the Wrepair station.
The product must have a selling price that is targeted at €300, so the price of the whole system must be below €200 to have margin.
The product must be producible in eight weeks.
Trade-Off Requirements or Objectives
The product should use modern charging protocols.
The product should be able to support real-time data logging.
The product should be able to connect to an external device for measurement logging and USB data hub connectivity.
The product should be able to be used as a USB data hub.
The product should use the highest speed for USB data transfer.
The product should be able to turn off the data connection of an individual port.
The product should be able to control the power delivered through the USB-C port.
The product should display power information graphically on the display.
The product should display charge/data protocol to the user.
The product could detect malicious user/activities.
The product could be controlled via an external PC.
The product could measure external voltages like a multimeter.

1.5. Subgroup Division

Based on the high-level requirements of the project, a structured system overview (Figure 1.3) and subdivision (Table 1.2) were developed to ensure comprehensive coverage of all mandatory and trade-off requirements.

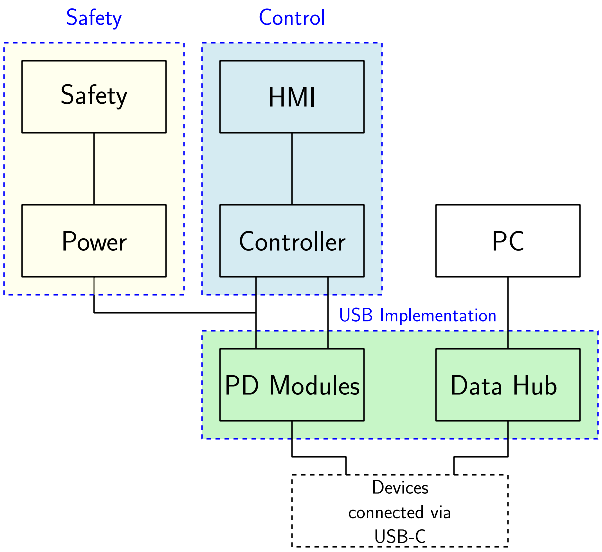


Figure 1.3: Overview of the whole system and division of each submodule

To efficiently address these requirements, the project was divided into several submodules, each focusing on a specific aspect of the product:

- **PD Modules:** These are essential for managing the power delivery protocols required for the USB-C ports, ensuring proper charging performance and compatibility with various devices.
- **USB/Data Hub:** This module handles data transfer and connectivity, allowing the product to function as a USB data hub and support high-speed data transfer.
- **Power Supply:** This module ensures there is 300 W for the whole system and realizes the lower voltage levels for the HMI and MCU.
- **Safety of the Entire Power Station:** This module focuses on ensuring the product is safe for users and devices, incorporating necessary safety features and protections.
- **Controller:** The controller is responsible for controlling the power output per port, managing the power to ensure it does not exceed the maximum limit.
- **HMI (Human-Machine Interface):** This module is responsible for the user interface, displaying measurements for current, voltage, and power and possibly providing graphical representations and other user interactions.

To efficiently address these submodules, the project was subdivided into three main categories, with each category assigned to a dedicated team, as shown in Table 1.2.

Table 1.2: Task distribution

Category	**Safety (O1)**	**Control (O2)**	**USB Implementation (O3)**
Team members	(Amin Aynan & Yahya Al-Araji)	(Alex Gulam & Zubair Al-Zubi)	(Bert van Lange & Pieter Pronk)
Module 1	Safety of Entire Power Station	Controller	PD Modules
Module 2	Power Supply	HMI	USB/Data Hub

1.5.1. Responsibilities of The Control Subgroup

The control subgroup is primarily responsible for the development and integration of the controller unit and the human-machine interface (HMI) for the Wrepair Powerstation. Their key responsibilities include:

- **Controller Selection:** Choosing a suitable controller that meets the system requirements and integrating it with the power delivery modules, touchscreen display, and other system components. This involves ensuring seamless communication between the controller and the peripherals using communication protocols.
- **Screen Selection:** Choosing a suitable touchscreen that meets size, resolution, and screen type/technology requirements.
- **HMI Design:** Create an intuitive, user-friendly GUI for real-time monitoring and power adjustment.
- **Power Division:** Implement algorithms to be able to manually allocate the input power across multiple USB-C ports.

1.6. State-Of-The-Art Analysis

The market for charging and data transfer stations is diverse, with numerous products that meet the needs of various users. An analysis and comparison of different products similar to the target design will provide insights into suitable components for the Control submodule. The focus will be on different criteria for the products, as listed below:

- Number of ports
- Port type
- Screen type
- Power allocation
- Visualization for each port
- GUI type

The comparison will focus on four charging stations: UGREEN Nexode 300W, MaAnt 8-Port 128W, ChargeAsap 280W ZEUS, and a generic 300W charger, as depicted in Figure 1.4.

(a) UGREEN Nexode 300W [66]

(b) MaAnt 8-Port Multifunctional 128W Charging Station [42]

(c) ChargeAsap 280W ZEUS USB-C GaN charger [1]

(d) Generic 300W GaN charger [2]

Figure 1.4: Various charging stations

Table 1.3 provides a detailed comparison of the four mentioned chargers.

Table 1.3: Comparison of charging stations

Feature	UGREEN Nexode 300W	MaAnt 8-Port	ChargeAsap 280W ZEUS	Generic 300W GaN	Wrepair Powerstation
Number of Ports	5	8	4	4	6
Port Type	USB-C (4x)/USB-A (1x)	USB-C (1x)/USB-A (7x)	USB-C (3x)/USB-A (1x)	USB-C (3x)/USB-A (1x)	USB-C (6x)
Screen Type	None	Small LCD	1.8-inch LCD	Small LCD	2.4-inch touchscreen
Power Allocation	Fixed	Fixed	Fixed	Fixed	Adjustable
Visualization for each port	None	Readings (V & I)	Readings (V, I & P)	Readings (V & P)	Readings + Graph (V, I & P)
GUI Type	None	Basic	Basic	Basic	Interactive

The Wrepair Powerstation (Figure 1.2) stands out by providing unique features such as six USB-C ports, a larger 2.4-inch touchscreen for user convenience, interactive interactions, and a versatile approach. Additionally, it offers the option to adjust the power output for each port, which sets it apart from the competition.

Although the analyzed charging stations offer fast charging and some of them have a screen for monitoring, they do not provide the same level of control, convenience, and interactivity as the Wrepair Powerstation. Furthermore, the GUI implementations in these products are often basic and limited. This makes the Wrepair Powerstation a superior choice for users who need a high degree of flexibility and detailed power management for multiple devices.

1.7. Thesis Overview

This thesis is organized to cover the research and development of the Wrepair Powerstation's HMI and controller integration in a clear and structured way. Each chapter focuses on a different part of the project.

In chapter 1, an overview of the project, including the motivation, client background, assignment, problem definition, and the organization of the project team is provided. It also includes a state-of-the-art analysis and an overview of the thesis structure. User experience (UX) design principles and their application to the Wrepair Powerstation are the focus of chapter 2. It covers the design of the user interface, including navigation, display organization, and data entry facilitation. Furthermore, it also explains the specific components of the GUI. The program of requirements for the Wrepair Powerstation is outlined in chapter 3. It includes system requirements, GUI requirements, screen requirements, and controller requirements. Design choices made during the development of the Wrepair Powerstation are examined in chapter 4. It addresses the design and selection of the screen, controller, and GUI. It also explains how these parts are combined into a working system. After that, chapter 5 evaluates the effectiveness of the GUI, power management strategies, and the results from user experience testing. It shows the feedback received and the changes made based on the feedback. Finally, chapter 6 summarizes the main findings of the thesis, discusses challenges faced during the project, and provides recommendations for future improvements.

2

User Experience

This chapter provides an overview of User Experience (UX) design. It starts by introducing the fundamentals of UX (section 2.1), emphasizing the integration of user state, system characteristics, and context. It then explores key principles and guidelines, focusing on navigating the interface (subsection 2.1.1), display organization (subsection 2.1.2), and data entry facilitation (subsection 2.1.3). The chapter also examines GUI components and their impact on UX (subsection 2.1.4). Finally, it discusses various UX testing methods, including usability testing, A/B testing, surveys, eye-tracking studies, remote user testing, and observational methods (section 2.2).

2.1. User Experience Design

User experience, UX, is mainly a combination of the user's internal state, the characteristics of the designed system and the context of use within which the interaction takes place. These three elements form the basis of a user experience [7]. There are various disciplines that show different perspectives on UX. However, the three main perspectives are IT, design, and psychology [27]. By integrating various perspectives, it can lead to a deeper understanding of UX and its components. Based on the literature review used in [27], a framework has been built, which shows the three elements and perspectives. This framework is shown in Table 2.1.

Table 2.1: UX elements and perspectives [27]

Perspectives **Elements**	**Design**	**IT**	**Psychology**
Product	Novelty	Mobility, adaptivity	Hedonic, embodied values
User	Ergonomic issues	Quality in use, usefulness	Needs, self-expression
Interaction	Branding	Rich, engaging	Satisfying, rewarding, emotionally fulfilling

As can be seen from Table 2.1, the three elements are shown on the first column, and the three perspectives are shown in the first row. When an element intersects with a design, a sub-element is shown for that particular perspective. For this application, the design perspective is crucial for the user experience. Specifically, identifying what constitutes a good design for embedded systems with a touch screen is of particular interest.

The aim of UX design is to be so seamless that users do not even notice it, ensuring smooth functionality. Design elements work like visual language, where images combine to communicate a clear message [58]. The following book explains the eight most important principles called 'golden rules' for interface design, which are listed in Table 2.2.

Table 2.2: Golden rules for interface design [60]

No.	Principle	Description
1	**Strive for consistency**	Actions should follow the same sequence in similar situations. This should be applied to prompts, menus, and help screens, and ensure colors, layouts, fonts, and other visual elements are consistent throughout.
2	**Cater to universal usability**	Designing interfaces that are accessible and usable by a diverse range of users, regardless of their expertise, age, and so on. The aim is to create a flexible design that can adapt to various user needs.
3	**Offer informative feedback**	Providing users with clear responses to their actions. The system should indicate whether an action was successful, if an error occurred, or what the next steps are. This helps users understand what is happening in the system, reduces confusion, and guides them through their tasks smoothly.
4	**Design dialogs to yield closure**	Interactions should be organized such that the user knows when a task is completed. Each dialog should have a beginning, middle, and end, with feedback at the end to indicate that the task is completed. This helps the user to feel a sense of satisfaction and lets the user know that they can proceed to the next task.
5	**Prevent errors**	The system should be designed such that it minimizes the chances of the user making a mistake. These mistakes can be avoided by predicting where errors might occur. If the user does make an error, then the interface should detect it and offer instructions to recover from it.
6	**Permit easy reversal of actions**	The user should easily undo their actions. It helps users recover from mistakes without significant stress. It makes the system more user-friendly and forgiving.
7	**Support internal locus of control**	The system should be designed in a way such that the user feels that they have control over their interactions. Users should feel that their actions affect the outcomes of the system. This increases user satisfaction as they feel that they are in charge rather than being controlled by the system itself.
8	**Reduce short-term memory load**	The interface should be designed to minimize the amount of information users need to remember while using the system. Keep the display simple by showing important information, providing clear instructions, and reducing the number of steps users need to recall. The goal is to make the system easier to use by relying more on intuitive design and less on the user's memory.

The aforementioned principles are essential for designing a well-organized interface. They provide a foundation for developers to use in their own designs. Additionally, guidelines for interfaces have been derived, which can be used to design the interface for embedded systems. The guidelines cover navigating the interface, organizing the display, and facilitating data entry [60]. Each guideline has its own principles, which will be explained in their respective sections.

2.1.1. Navigating the Interface

This guideline shows how the user can interact with different elements that are provided by the software application. The principles for this guideline are focused on the design interface for an embedded system. These are listed in Table 2.3.

Table 2.3: Principles for navigating the interface [60]

Principle	Description
Standardize task sequences	Allows the user to perform tasks in a consistent sequence across the interface, reducing the need to remember different sequences for similar tasks.
Use unique and descriptive headings	Headings should be distinct and explain immediately what the related content is about, helping users find information quickly and improve overall content organization.
Use check boxes for binary choices	Check boxes can be provided for options that have two states, such as ”on” or ”off,” making it easy for users to understand their selection.

2.1.2. Organizing the Display

For an embedded system, organizing the display is of great importance, as the usability and overall experience of the interface will be clear. The principles for this guideline are shown in Table 2.4.

Table 2.4: Principles for organizing the display [60]

Principle	Description
Consistency of data display	Maintain consistency in terminology, abbreviations, formats, colors, and other elements to help users recognize and understand interface elements, reducing confusion and errors.
Efficient information assimilation by the user	The display format should be familiar to the user and relate to the tasks they need to perform, enabling efficient task performance through effective information assimilation.
Minimal memory load on the user	Design the interface to minimize the need for users to remember information, structuring tasks to require only a few steps, and providing a standard format to reduce reliance on memory.
Compatibility of data display with data entry	Ensure that the format of displayed information is clearly linked to the format for data entry, with output fields also functioning as input fields to help users understand the relationship between displayed data and data entry.
Flexibility for user control of data display	Allow users to customize the display of information to fit their specific tasks, providing a more personalized and user-friendly experience.

2.1.3. Facilitating Data Entry

For user interface design to facilitate data entry is crucial. By structuring the data entry effectively, it can reduce the user's frustration, prevent errors and reduce the cognitive load. Table 2.5 lists the essential principles to achieve this.

Table 2.5: Principles for facilitating data entry [60]

Principle	Description
Consistency of data-entry transactions	Use similar sequences of actions under all conditions to help users predict and understand the process, reducing errors.
Minimal input actions by user	Reduce the number of actions required for input to increase productivity and reduce errors, using simple input methods like a finger press.
Minimal memory load on users	Users should not have to remember long lists of codes or commands; display only necessary information and use intuitive features to reduce memory load.
Flexibility for user control of data entry	Allow users to control the sequence in which they enter information, balancing flexibility with the need for consistency.

These guidelines form a framework for designing user-friendly data entry interfaces, by ensuring that the interface is both efficient and effective in reducing errors and improving productivity.

2.1.4. GUI Components

By following the golden rules and guidelines of UX design, a GUI specifically for embedded systems with touchscreens can be developed that significantly improves the overall user experience. In Table 2.6, the GUI elements are listed.

Table 2.6: GUI components [44] [58]

Component	Description
Pages	It separates the screen into individual areas, enabling the execution of different programs, the display of directories, or the visualization of various plots.
Icons	Small graphics or pictures representing tasks, programs, folders, files, and windows.
Buttons	Push buttons present the user with discrete choices, triggering actions immediately upon being pushed.
Sliders	Sliders are mainly used for navigating through the pages and adjusting values.
Swipe	Swipe is an example of a gesture and can be used to navigate between pages and scroll through content.

The UX strategy uses these principles to design a user-friendly GUI for the application. Figure 2.1 shows the strategy to design a GUI for such embedded systems with touchscreens.

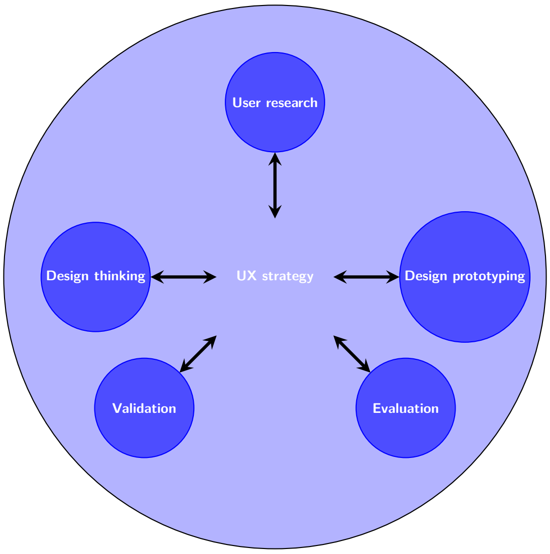


Figure 2.1: UX strategy [58]

2.2. UX Testing Methods

User Experience (UX) testing is essential to ensure that a digital interface is intuitive, efficient, and meets user needs [49] [48]. Various UX testing methods can be employed to gather feedback and improve interface designs [41].

- **Usability Testing:** This method involves real users performing specific tasks on the interface to identify usability issues. Observations and metrics such as task completion rate, time on task, and error rate are collected to assess the interface's effectiveness and efficiency [48].
- **A/B Testing:** Different versions of the GUI are presented to users to determine which design performs better [57]. By comparing user interactions with each version, designers can identify the most user-friendly elements and make informed decisions about design choices.
- **Surveys and Questionnaires:** After interacting with the interface, users provide feedback through structured surveys or questionnaires. This method gathers qualitative data on user satisfaction, perceived usability, and overall experience, helping to understand user sentiments and preferences [41].
- **Eye-Tracking Studies:** By tracking where users look while interacting with the interface, designers can understand which elements draw attention and how users navigate the interface [22] [41] [23]. This data is valuable for optimizing layout and design to enhance user engagement.
- **Remote User Testing:** Users test the interface in their own environment, providing feedback through online tools [41]. This method is useful for gathering diverse user perspectives and identifying issues that may arise in different settings, proving to be a fast and cost-effective approach [38] [20].
- **Observations and Timings:** Observing users as they interact with the interface and recording the time taken to complete tasks can provide quantitative data on efficiency and user behavior [64] [26]. These observations help identify bottlenecks and areas where users struggle.

3

Program of Requirements (PoR)

This chapter gives the general description of the product (section 3.1) and outlines the comprehensive Program of Requirements (PoR) for the Wrepair Powerstation, encompassing the system (section 3.2), graphical user interface (GUI) (section 3.3), screen (section 3.4), and controller (section 3.5).

3.1. General Description of Product

The Wrepair Powerstation is a charging and data transfer station for smart devices such as smartphones and tablets. The Powerstation will be mainly used by professionals. The control subgroup focuses on the controller and the human machine interface, which is primarily a touchscreen. The following subsection will describe the requirements for the control and human machine interface parts.

3.2. System Requirements

1. The system must be able to display voltage, current, and power on the screen.
2. The system must present waveforms consisting of voltage, current, and power over time.
3. The system must support real-time data logging.
4. The system must provide an intuitive user interface without having prior knowledge.
5. The system must provide functionalities for data transfer.
6. The system must be able to integrate with other modules.

3.3. Graphical User Interface (GUI)

The GUI for the Wrepair Powerstation is designed to be intuitive, to visualize real-time data and to customize settings to ensure a good user experience. The following requirements describe the GUI.

3.3.1. Overall requirements GUI

1. The GUI must be intuitive and easy to use, especially for new users without proper experience of repair stations.
2. The interface must display the logo of Wrepair prominently on the home page.
3. The interface will include virtual buttons on the main screen, one for each port, to provide quick access to port-specific settings and information.
4. The user must be able to see the voltage, current and power readings directly on each button associated with a port, ensuring that the readings are updated in real-time.
5. The font size and style across the GUI should be clear and readable on each page.
6. Each page dedicated to an individual port must include a power adjustment feature, allowing users to modify the output power to meet specific requirements, with safeguards in place to ensure the power limit is not exceeded.

3.3.2. Functional requirements GUI

1. The interface must display voltage, current, and power readings on their respective buttons for each port.
2. Voltage, current, and power readings should be presented in a real-time waveform graph to visualize changes over time.
3. The interface must be user-friendly and adaptable to various lighting conditions, enabling users to operate comfortably in their preferred lighting environment, such as electronics labs.
4. There must be an option to adjust the brightness of the display, with the system remembering the user's last setting across sessions and page transitions.
5. Each page of each port must contain a control mechanism, allowing the user to adjust the output power of a certain port to meet their needs.

3.3.3. Non-Functional Requirements GUI

1. The graphs displaying voltage, current, and power must update continuously in real-time, ensuring no delays in data visualization.
2. The graphical elements of the GUI must be designed to ensure accessibility for users with visual impairments. This includes using high-contrast color schemes and patterns to accommodate colorblind users, and ensuring that text is readable with sufficient size and font choice.
3. The interface should provide stable operation, free from freezing, crashes, or graphical glitches, and all graphical elements must render correctly and consistently without visual artifacts or delays.

3.4. Screen

The screen is used to interact with Powerstation and to monitor the readings for each port in the Wrepair Powerstation. The screen must be of high quality, as it is the primary interface for user interaction. The following requirements are related to the screen.

3.4.1. Overall Requirements Screen

1. The screen must be a touchscreen.
2. The screen's dimensions must not exceed 7.5 cm in length and 4.5 cm in width to ensure a proper fit within the designated space on the top layer of the Wrepair Powerstation casing.
3. The screen must be suitable for a lab environment.
4. The screen must operate within an appropriate voltage range that is compatible with the main controller.
5. The screen must be responsive and quick.
6. The screen must have support for hardware communication protocols, such as UART, I^2C or SPI to ensure compatibility with the main controller.
7. The screen must have sufficient onboard flash memory to store graphical elements, icons, fonts, and other visual assets required for the user interface.
8. The screen must have sufficient RAM to store instructions, variables, and temporary data during operation.
9. The screen must be securely attachable to the Wrepair casing, using appropriate mounting mechanisms or adhesive.

3.4.2. Non-Functional Requirements Screen

1. The screen must be durable and capable of withstanding frequent use.
2. The screen must have an active area large enough to allow users to comfortably interact with the interface elements, ensuring optimal usability and minimizing the risk of accidental touches.
3. The screen must have a matte finish to minimize glare and reflections under varied lighting conditions, particularly in brightly lit lab environments.

4. The screen must maintain operational reliability within a temperature range of -20°C to 70°C and relative humidity of 20% to 80% [35].
5. The screen's brightness should automatically adjust to ambient lighting conditions to optimize visibility.

3.5. Controller

The controller is the main computing hardware of the system, connecting all submodules. The requirements for the controller will be described in the following sections.

3.5.1. Overall Requirements Controller

1. The dimensions of the controller unit must not exceed 8 cm x 8 cm x 4 cm to ensure it fits within the Wrepair Powerstation.
2. The controller's input voltage range must be compatible with the screen to ensure seamless integration and prevent electrical damage.
3. The controller must be able to consume power as efficiently as possible.
4. The controller must be able to execute the division of power of up to 300 W across six ports.
5. The controller must have an interface with a screen that displays real-time data of voltage, current, and power and their corresponding waveforms over time.
6. The controller must support communication protocols like UART, I^2C, or SPI to interface with the screen and Power Delivery (PD) modules.
7. The controller must have sufficient memory capacity to manage, store, and process real-time data, historical logs, and configuration settings.

3.5.2. Functional Requirements Controller

1. The controller must be able to allocate up to 300 W of power across six devices, adjusting the power delivery to each port based on the specific requirements of the connected devices and the available power.
2. The controller must prevent the total power consumption from exceeding 300 W.
3. The controller must continuously measure and update voltage and current readings in real-time, providing accurate and up-to-date information to the user through the screen interface.
4. The controller must store the data in order to display real-time waveforms of voltage, current, and power for each connected device.

3.5.3. Non-Functional Requirements Controller

1. The controller must be reliable and operate continuously without failures or crashes.
2. The controller should be protected against electrical surges, overvoltage, and other potential hazards to ensure the safety of the system and connected devices.

4

Design

This chapter examines the choices made in the Wrepair Powerstation development, specifically the choice of the screen (section 4.2), controller (section 4.4), and GUI (section 4.3).

4.1. Design Overview

This chapter explores the design choices made in the development of the Wrepair Powerstation, focusing on the selection of the controller, screen, and GUI. These submodules must integrate to ensure the system operates as expected.

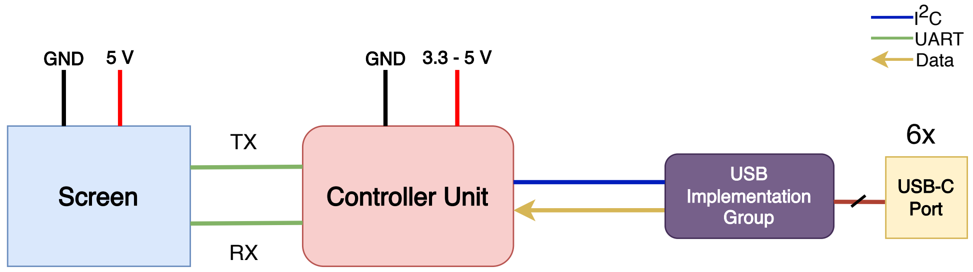


Figure 4.1: System design overview

As shown in Figure 4.1, the components are interconnected. The controller receives data from the PD modules (USB Implementation submodule), and this information must be displayed on the screen. The following sections will provide a detailed explanation of each submodule.

4.2. Screen

The screen serves as the primary interaction device between the user and the Powerstation. The screen's functionality extends beyond simple display capabilities. It integrates touch technology to allow users to perform intuitive actions such as tapping and swiping. This section discusses the various screen types, screen technologies, touchscreen's capabilities, resolution, graphical user interface elements, which are designed to optimize usability and accessibility, types of display interfaces, and the screen interface development. Based on these functionalities combined with the requirements listed in section 3.4, a suitable touchscreen will be chosen.

4.2.1. Screen Types

There are mainly four types of screens: Liquid Crystal Display (LCD), Organic Light-Emitting Diode (OLED), mini-LED, and micro-LED, out of which LCD and OLED are the most popular today [30] [28].

Each technology offers unique benefits, catering to different needs based on display quality and energy efficiency.

LCDs utilize a backlight to illuminate their pixels, making them exceptionally bright and readable even in well-lit environments [11]. This is particularly advantageous for use in laboratories, where varying lighting conditions can affect visibility. LCDs are also more cost-effective than OLEDs and newer technologies like mini-LED and micro-LED [34]. Furthermore, LCDs do not suffer from burn-in issues, a common problem in OLED displays where images are retained as ghost images on the screen from displaying static graphics over long periods [53] [36]. An example of such a burn-in is shown in Figure 4.2.

Figure 4.2: Example of OLED burn-in [32]

When it comes to power efficiency, OLED displays are generally more power efficient when displaying black images, whereas LCDs are more efficient when displaying bright images [11].

Lastly, there are mini-LED and micro-LED, which are emerging next-generation displays [30]. They both offer long lifetimes and high luminance, however they are significantly more expensive due to their complex manufacturing process [72].

A comparison of various screen types is given in Table 4.1.

Table 4.1: Comparison of screen types [52]

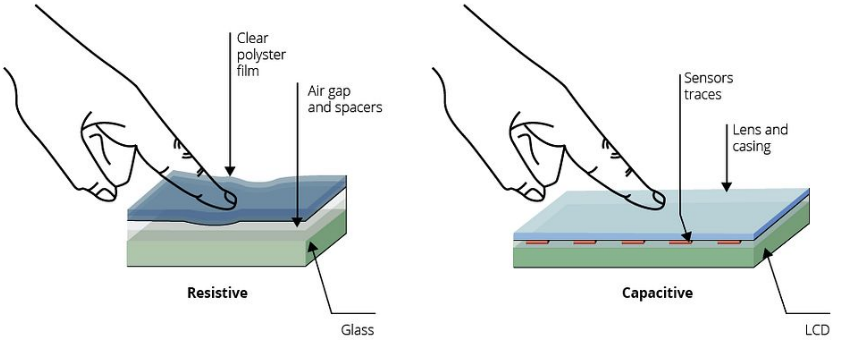
	LED-backlit LCD	**Mini-LED**	**OLED**	**Micro-LED**
Display type	Backlit	Backlit	Emissive	Emissive
Contrast	Low to medium	Medium	High	High
Response time	Low (ms)	Low (ms)	High (μm)	Very High (ns)
Power efficiency	Medium	Medium-high	Medium	High
Only lit pixels draw power	No	No	Yes	Yes
Lifetime	Long	Long	Medium	Long
Sunlight visibility	Medium	Medium	Medium	High
Operating temperature	$\sim$ -20 to 80°C	$\sim$ -20 to 80°C	$\sim$ -30 to 70°C	$\sim$ -100 to 120°C
Need for encapsulation	No	No	Yes	No
Brightness	Medium	Medium	Low-Medium	High
Flexibility	Low	Low	High	Medium-high
Viewing angles	Low	Low	High	High
Tech maturity	High	Medium-High	Medium-High	Low

4.2.2. Touchscreen Technologies

Touchscreen technology can be broadly categorized into resistive and capacitive types (Figure 4.3). Each type offers unique characteristics suited to different application environments.

Resistive touchscreens operate by registering pressure applied to the surface, which brings two conductive layers into contact, thereby creating an electrical circuit that is registered as a touch event [29]. Resistive touchscreens are highly durable and can generally be operated with any object, such as a stylus or even with gloved hands, making them suitable for industrial environments, such as electronics repair labs [29].

In contrast, capacitive touchscreens rely on the electrical properties of the human body to detect touches [47]. They support multitouch input, allowing for gestures like pinching and swiping. Capacitive screens offer a higher clarity and sensitivity than resistive screens and are better suited for consumer devices that prioritize smooth user experiences and high interaction fidelity [4] [47].



» Incubent touch technology. Resistive and Capacitive

Figure 4.3: Resistive vs capacitive touchscreens [40]

4.2.3. Types of Display Interfaces

The primary serial communication protocols used for display interfaces are UART, I^2C, and SPI, each with unique characteristics that suit different requirements. These protocols transmit data one bit at a time over a single wire (or a pair of wires for differential signaling). They are often used for simpler applications or when minimizing the number of wires is important.

- **UART (Universal Asynchronous Receiver-Transmitter):** UART is a serial communication protocol that enables asynchronous transmission [10] [73]. This means that data is sent one bit at a time at a specific baud rate, which is the rate at which information is transferred in bits per second (b/s or bps). UART is widely used for simple communication tasks where high-speed and high-reliability are not critical, but it's particularly favored for debugging and simple text data communication between the controller and computer or other devices [10].
- **I^2C (Inter-Integrated Circuit):** I^2C is a multi-master, multi-slave, packet-switched, single-ended, serial communication bus [43]. This means multiple devices can be connected to the same bus and communicate with each other without the need for more than two wires (power and ground), reducing the complexity of the wiring significantly. I^2C is beneficial for connecting low-speed peripherals like small displays and sensors in close proximity within the same device due to its moderate speed and excellent support for complex communications with multiple devices over a single channel [43] [33].
- **SPI (Serial Peripheral Interface):** SPI is another serial communication protocol used to communicate in short distances at high speeds [69]. It operates in a master-slave setup, where the master device controls the clock signal that dictates the data transmission rate. SPI is faster than I^2C and UART, making it ideal for applications like full-color video streaming or high-speed data logging in screen interfaces where quick data refresh rates are crucial [12] [21].

A comparison of these serial communication protocols is given in Table 4.2.

Table 4.2: Comparison of UART, I2C, and SPI serial communication protocols [67]

	UART	I^2C	SPI
Number of Pins	2	2	>3
Baud Rate (b/s)	Up to 115,200	Up to 400,000	Up to a few megabits
Communication Type	Point to Point	Multi Master - Multi Slave	One Master - Multi Slave
Half and Full Duplex	Full Duplex	Half Duplex	Full Duplex
Synchronous or Asynchronous	Asynchronous	Synchronous	Synchronous
Maximum Number of Devices on the Bus	2	Up to 128	Theoretically Infinite
Complexity	Low	High	Medium
Cost	Low	Medium	High

Another category of display interface are the so-called high-performance display interfaces, which are designed to handle the demands of modern displays, such as high resolutions, fast refresh rates, and rich color depths. They often use more complex signaling schemes and require more wires than serial protocols.

- **DSI (Display Serial Interface):** DSI is a high-speed serial interface primarily used in mobile devices for connecting displays to processors. It supports high-resolution displays and fast refresh rates, making it suitable for modern smartphones and tablets [39].
- **HDMI (High-Definition Multimedia Interface):** HDMI is a widely-used interface for transmitting high-definition video and audio from a source device to a display. It is commonly used in televisions, monitors, and projectors [65].
- **DP (DisplayPort):** DisplayPort is a digital display interface primarily used to connect a video source to a display device. It supports high resolutions, multiple displays, and high refresh rates, making it ideal for computer monitors and high-end displays [61].

While high-performance display interfaces like DSI, HDMI, and DisplayPort are essential for modern, high-resolution, and high-refresh-rate displays, they are not very relevant to our application. Our project involves a relatively small touchscreen, as outlined in the program of requirements in section 3.4. For this project, the simplicity, cost-effectiveness, and sufficient performance of serial communication protocols, particularly UART, make them more suitable. Thus, the primary focus is on these three serial communication protocols to ensure efficient and effective communication with the small touchscreen display.

4.2.4. Screen Interface Development

The development of the GUI for the Wrepair Powerstation is an important process in ensuring that the device is both functional and user-friendly. The choice of GUI design method, particularly the use of a What You See Is What You Get (WYSIWYG) environment, plays a crucial role in this process. WYSIWYG editors allow developers to visually design the interface by dragging and dropping elements, adjusting their properties, and seeing the results in real-time. This approach is significantly more efficient and saves time, especially given the project's timeframe of just eight weeks, as opposed to coding-based GUI editors where each graphical element must be manually coded, which is inefficient and time-consuming.

4.2.5. Screen Selection

Three touchscreen options are considered in this section. The main criterion for the touchscreen is that it should fit within the Wrepair Powerstation (i.e., not exceed the dimensions), as mentioned in the program of requirements in section 3.4. The three main touchscreen options are shown in Figure 4.4.

(a) Nextion NX3224K024 resistive touchscreen [51]

(b) DWIN DMG32240C024 resistive touchscreen [68]

(c) ESP32-2432S024C capacitive touchscreen [17]

Figure 4.4: Three 2.4-inch touchscreen options

The resolution of 320x240 pixels, also known as QVGA (Quarter Video Graphics Array), has been chosen for the touchscreen display in the Wrepair Powerstation project. This resolution is well-suited to the project's requirements, particularly given the 2.4-inch screen size, providing a clear and readable display at approximately 166 PPI.

The screens can also be rotated to function in portrait or landscape mode. For the Wrepair Powerstation, the screen will be used in portrait mode to maximize the available space and not exceed the Powerstation's dimensions.

The Nextion NX3224K024 (Figure 4.4a) and DWIN DMG32240C024 (Figure 4.4b) are both 2.4-inch resistive touchscreens, while the ESP32-2432S024C (Figure 4.4c) is a 2.4-inch capacitive touchscreen. Resistive touchscreens are known for their durability and compatibility with various input methods, including gloves and styluses, making them suitable for a lab environment (requirement 3.4.1.3 from subsection 3.4.1). Capacitive touchscreens, on the other hand, offer higher clarity and sensitivity but are less durable and may not work well with gloves.

The Nextion and DWIN screens have similar specifications in terms of display type and interface options (UART). However, the Nextion screen stands out due to its easy-to-use WYSIWYG editor, which significantly simplifies the GUI development process. This editor allows for easy creation and customization of the interface without requiring extensive coding knowledge, aligning with the project's time constraints. Additionally, the availability of the DWIN is an issue, which poses a risk to the project's timeline. This screen is less favorable due to these constraints.

The ESP32 touchscreen, while offering the advantages of capacitive technology such as higher clarity and sensitivity, lacks a WYSIWYG editor. This absence means that developing the GUI requires more technical expertise and manual coding, which can be time-consuming and complex. Furthermore, the capacitive nature of the ESP32 makes it less suitable for lab environments where users might wear gloves, which is more compatible with resistive touchscreens. This lack of adaptability and the additional development complexity make the ESP32 touchscreen less efficient and practical for the intended application in the Wrepair Powerstation project. Table 4.3 summarizes the specifications of thee three touchscreens.

Table 4.3: Comparison of 2.4-inch touchscreens

	Nextion NX3224K024	**DWIN DMG32240C024**	**ESP32-2432S024C**
Display Type	Resistive	Resistive	Capacitive
Interface	UART	UART	UART, SPI, I^2C
Resolution	320x240	320x240	320x240
WYSIWYG Editor	Yes	Yes (complex)	No
Flash Memory	16 MB	16 MB	4 MB
Price (approx.)	€30	€20	€25

The Nextion NX3224K024 2.4-inch resistive touchscreen is highly suitable for the Wrepair Powerstation for several reasons. First, its dimensions do not exceed the maximum dimensions of the top layer of the Wrepair Powerstation, ensuring a good fit. Second, the GUI design process is both easy and efficient with this display compared to similar displays, thanks to its WYSIWYG editor environment (Figure 4.5). This eliminates the need for coding each GUI element, which is quite inefficient. Additionally, this Nextion screen has 16 MB of flash memory, which is sufficient to store GUI elements such as fonts and images, making it an excellent choice for the Wrepair Powerstation. It meets all the requirements for functionality, durability, and ease of use in a professional lab setting.

Encoding:iso-8859-1 | Model:NX3224T024_011 inch:2.4(240X320) Flash:4M RAM:3584B Frequency:48M Coordinate X:414 Y:184

Figure 4.5: Nextion Editor [6]

4.3. GUI Design

The Graphical User Interface (GUI) is an important component of the system, providing users with an intuitive and interactive way to monitor and control various system functions. By communicating with the controller, the GUI ensures that all data is presented clearly and accurately. The interface will be designed to prioritize readability and ease of use, featuring intuitive controls and adjustable settings to allow users to optimize the system's performance efficiently. Based on the literature on UX and the requirements for GUI, listed respectively in chapter 2 and section 3.3 an initial design for the GUI was made.

4.3.1. Initial GUI

The initial GUI design was based on the review of existing literature on GUI development and specific requirements for the application. These insights were carefully considered during the design process. The Nextion editor, with its WYSIWYG environment, allowed the developer to view and design the interface as it would appear on the actual device. By incorporating UX design principles, essential GUI components, and specific requirements, the initial GUI design was created, as shown in Figure 4.6.

(a) Initial GUI homepage

(b) Initial GUI settings page

Figure 4.6: The initial GUI illustrated

Figure 4.6 displays two key pages of the system: the initial homepage and the settings page, which are respectively shown in Figure 4.6a and Figure 4.6b. Upon powering on the Powerstation, the user is greeted with the homepage, which features the Wrepair logo at the top. The homepage includes six virtual buttons, each representing a connected device. To ensure consistency, these buttons are uniformly styled and positioned, following the same sequence across the interface in similar situations. This consistency extends to the use of colors, layouts, and fonts, improving user familiarity and ease of navigation. These buttons display real-time measurements of voltage, current, and power, updated every 500 ms due to the chosen sampling frequency of 2 Hz.

Additionally, a button allows users to toggle between dark and light modes, improving universal usability by enabling adjustments to the interface based on visual preferences and ambient lighting conditions. This feature ensures the interface remains accessible to a diverse range of users.

The interface design focuses on reducing memory load for users. Instead of remembering complicated steps or commands, users can recognize familiar icons and buttons. For example, the settings cogwheel and the 'Back' button are well-known symbols that make the interface easier to use and more user-friendly.

Pressing this icon leads to the settings page, shown in Figure 4.6b. The settings page allows the user to adjust the screen's brightness. This can be done adaptively using a Light Dependent Resistor (LDR) that senses the ambient light intensity [16], or manually by using a slider to increase or decrease the brightness level. These options offer flexibility for user control of data display, allowing users to customize their experience based on their needs. The use of a slider for brightness control provides an intuitive way for users to make adjustments, following the guideline for facilitating data entry with minimal input actions. To return to the homepage, the user can press the 'Back' button, which supports the principle of designing dialogs to yield closure by providing a clear way to navigate back to the main interface.

4.3.2. UX Testing Method

The initial design was created and immediately tested using a UX testing method. The UX testing was conducted with 8 participants to evaluate the usability and effectiveness of the GUI. The testing method included observations and timing, with an average navigation time of 40 seconds. The feedback received was both positive and constructive, leading to several improvements.

- **Negative Feedback:**
    - Buttons were too small (62.5% of participants).
    - Button icons needed improvement (37.5% of participants).
    - Color contrast should be increased (25% of participants).
    - Font size was too small (50% of participants).

- **Positive Feedback:**
    - The GUI is clear and easy to use (37.5% of participants).
    - The two themes were well-received, catering to different user preferences (50% of participants).
    - Real-time readings for each port were considered highly useful (37.5% of participants).
    - Brightness is a useful addition (37.5% of participants).

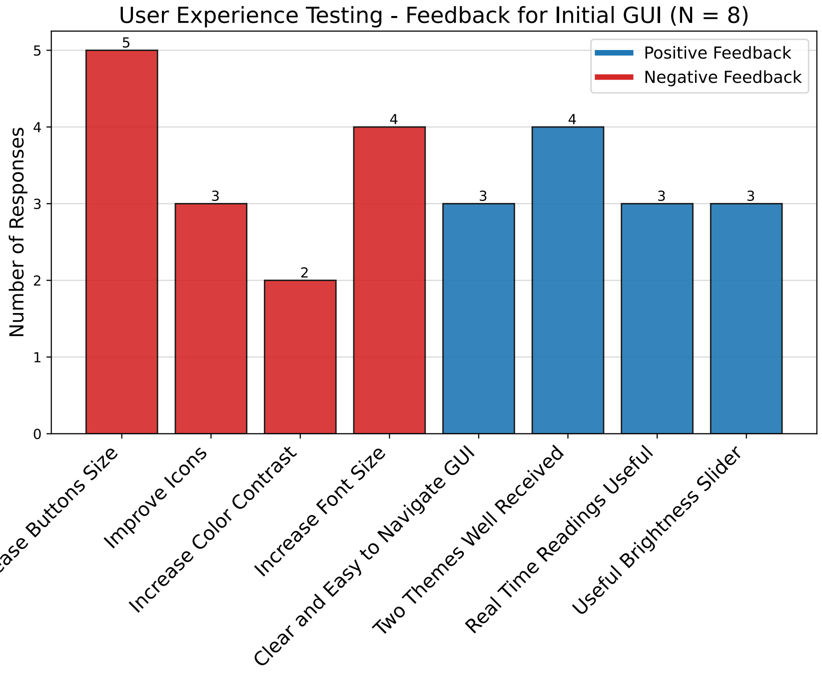


Figure 4.7: Bar chart illustrating both positive and negative feedback received from user experience testing for the initial GUI

The feedback from the initial GUI testing has shown that several areas require improvements to make the interface more intuitive and user-friendly. One major issue was the size of the buttons, which were found to be too small by 62.5% of participants. Increasing the button sizes is crucial to make them easily tappable. This adjustment will reduce errors and make the interface more accessible to individuals with different physical abilities.

Furthermore, the button icons also need improvement by 37.5% of participants. The icons need to be redesigned to be more intuitive and descriptive. Using well-known symbols can make it easier for users to quickly understand and use the interface. This helps the user to navigate and know the purpose of each button without getting confused.

Moreover, the need for increased color contrast and larger font sizes were necessary. Increasing the color contrast will improve readability, allowing users to see different elements on the screen. Similarly, larger font sizes will help those who struggle with smaller text, making the GUI more accessible.

Additionally, the slider for manually adjusting the brightness was positioned in a confusing manner. Most participants were used to a horizontal slider, similar to those used in smartphones, rather than a vertical slider. Furthermore, some participants felt that the GUI background was too basic and lacked visual appeal.

All this feedback was taken into consideration, and the GUI needs to be changed. These changes are reflected in the final GUI, which is explained in section 5.1.

4.4. Controller

The controller is the most important part of the embedded system. The controller communicates with both the power delivery modules and the display screen. It should obtain real-time voltage and current measurements from the power delivery system. These measurements are already converted to digital signals and from the controller it will be written to the screen. Based on the requirements for the controller, which are listed in section 3.5, an appropriate controller will be chosen, which can be an FPGA, SoC, or an MCU.

4.4.1. Comparison between FPGA, SoC and MCU

Table 4.4 presents a comparison between FPGAs, SoCs, and MCUs based on several features that are of interest for this particular application. It provides the definitions, the type of programming languages used, and the programming difficulty for each technology. The table also compares their flexibility, performance, cost, power consumption, and use cases. Additionally, it shows the complexities of each technology and their disadvantages.

Table 4.4: Comparison of FPGAs, SoCs, and MCUs

Feature	FPGA	SoC	MCU
Definition	Programmable logic device with reconfigurable logic and interconnects [56]	Integrated circuit combining CPU, memory, I/O ports, etc. onto a single chip [59]	Embedded system integrating a microprocessor, memory, I/O interfaces, and other components [24]
Programming Language	HDLs (Verilog, VHDL) [31]	Mix of hardware and software programming [59]	Various high-level programming languages like C/C++ [37]
Programming Difficulty	High, requires skilled programmer [25]	Complex, requires understanding of whole system [59]	Moderate, relatively simpler to program [37]
Flexibility	High, reprogrammable to implement different digital circuits and functions [13]	Configurable, combines programmable and hardwired units [59]	Moderate, designed for specific tasks [71]
Parallel Processing	Yes, can execute multiple tasks simultaneously [13] [31]	Yes, efficient and adaptable [3]	No, designed for single-threaded tasks [71]
Performance	High due to parallelism [13]	High performance and scalable on-chip communication [3]	Moderate, suited for specific and simpler tasks [71]
Cost	High, especially for large-scale productions [15]	Variable, but can be high due to integration of multiple components	Low, cost-effective for small-scale applications [37]
Power Consumption	Moderate to high	Variable, depends on components and configuration	Low, designed for low power consumption [71]
Use Cases	Custom digital circuits, parallel processing tasks [31]	Devices like cars, watches, cellphones, household appliances [59]	Specific embedded applications, control systems, simple tasks [24]
Complexity	High [25]	High, requires understanding of both hardware and software [59]	Low, simpler integration and operation
Suitable for	Applications requiring custom and parallel processing [31]	High-performance and scalable systems with complex integration [3]	Applications needing efficient, low-power, and cost-effective solutions [55]
Disadvantages	Difficult to program, time-consuming, expensive for large-scale use [25]	Complex design and implementation, requires in-depth system knowledge [59]	Unsuitable for complex computational tasks, limited to specific functions [37]

Considering the different types of controllers, a choice needs to be made based on the functionalities required for the application. Since FPGAs have HDL languages, which are difficult to program, it will be impractical for our use. While SoCs offer high performance and scalability, their complexity and the need for a thorough understanding of both hardware and software may be challenging to manage within the project's time constraints. In contrast, an MCU is much simpler and can execute specific tasks efficiently, making it the right fit for the requirements. Therefore, the type of controller chosen for our application is the MCU.

4.5. Microcontroller

In the preceding section, the decision to use an MCU was made. This section identifies the specific MCU that best fits the application. The success of the application greatly depends on selecting the right MCU [14], as system performance, design, features, and specifications are all influenced by this choice [9]. The key components relevant to the application are listed below:

- Input-output lines
- Data bus
- Memory
- Communication protocol
- Clock frequency
- ADC

These components will be discussed in their respective sections.

4.5.1. Input-Output Lines

The amount of peripherals that needs to be connected to the MCU will determine the amount of input output lines [9]. The devices that are connected to the MCU are the screen and the six development boards. However, only the development boards use the I/O lines, meaning that only six lines are required for the MCU. The screen itself is connected to the screen via a communication protocol, which will be explained in its respective subsection.

4.5.2. Data Bus

In our application, the CPU will execute the instructions and tasks essential for the operation of the entire system. To execute these instructions, it needs to read data from the PD modules, which consist of digital signals representing voltage and current measurements. These digital signals will be sent to the MCU, each occupying one byte, as specified in the PD modules datasheet [5]. Therefore, the CPU requires at least a 16-bit data bus [9].

4.5.3. Memory

The MCU's memory architecture comprises various types of memory, including RAM, ROM, flash, EEPROM, and cache. For our application, the focus is specifically on SRAM (static RAM) and flash memory. SRAM, a type of volatile memory, is used to store data temporarily. In contrast, flash memory is non-volatile and used to store program codes, ensuring data retention even when the power is off. To display a waveform illustrating voltage, current, and power over time, data storage is essential. Therefore, SRAM will be utilized to store data transmitted from the PD modules, while flash memory will store the program code [46] [54]. The data stored in SRAM will be used to generate a waveform display on the screen. The amount of data that can be stored is calculated as follows:

$$\text{Data storage} = 6\ \text{devices} \cdot \frac{\text{bytes}}{\text{sample}} \cdot \frac{\text{samples}}{\text{second}} \cdot \text{Time} \tag{4.1}$$

In our application, each sample will include voltage and current measurements, each requiring one byte, totaling two bytes per sample. To ensure that the user can effectively perceive the readings on the screen, the sampling rate must be compatible with the display capabilities. When testing with simulated data, a delay of 500 ms was sufficient to display the readings on the screen, updating twice per second. Therefore, a sampling rate of 2 Hz is adequate for the application, allowing the user to see the readings update every half a second. Moreover, while charging the devices, users can extract data to an external device. Thus, the charging data needs to be stored for a specified time interval. It was decided to make this interval dynamic, based on battery capacity and charging technology. However, obtaining such information is not feasible within an 8-week time constraint. Therefore, a fixed time interval of 1 hour, considered an average charging time, was chosen and implemented. Given these parameters, the data storage requirements for the SRAM can be calculated as follows:

$$\text{Data storage} = 6\ \text{devices} \cdot \frac{2\ \text{bytes}}{\text{sample}} \cdot \frac{2\ \text{samples}}{\text{second}} \cdot 1\ \text{hour} \cdot 3600\ \text{seconds} = 86.4\ \text{kB} \tag{4.2}$$

Besides SRAM, flash memory is also of significant importance. Flash memory is used to store the program instructions that need to be executed. Consequently, the capacity of the flash memory depends on the size of the application program. Currently, the program requires only a few kilobytes, indicating that the flash memory will not be a limiting factor, as many MCUs have a flash memory that is larger than the SRAM.

4.5.4. Communication Protocol

The communication protocol required for the application depends on the modules connected to the MCU. As previously shown in Figure 4.1, the MCU is directly connected to both the screen and the PD modules. According to the datasheet of the PD modules [5], communication can be achieved using I^2C. The I^2C communication protocol is a serial data communication method. It operates on a two-wire, bidirectional serial bus that facilitates data exchange based on the master-slave principle [19]. For this application, the MCU will function as the master, while the six PD modules will serve as slaves. The data rate will be determined by the MCU's clock. This setup enables the MCU to both send data to and receive data from the PD modules. On the other hand, communication with the screen will be done via UART, as explained in section 4.2. Therefore, the MCU must support both I^2C and UART communication protocols to integrate effectively with the other modules.

4.5.5. Clock Frequency

Any MCU needs a clock frequency to operate. The operating speed of the MCU is dependent upon the clock frequency. For our application, determining the clock frequency depends on the SCL needed to communicate with the PD modules via I^2C communication protocol. From the datasheet [5], it is shown that the maximum SCL frequency required for the PD modules is 1 MHz. Therefore, the clock frequency of the MCU needs to be at least 1 MHz. However, our application demands real-time processing, high accuracy and precision, fast computations, and the ability to perform multiple tasks simultaneously. These requirements demand a higher computing power [9]. To ensure smooth and reliable operation at all times, the clock frequency of the MCU has been chosen to be in the range of 10 – 20 MHz. Moreover, the required memory size for the MCU is usually not paired with low frequency ranges. These MCUs typically have clock frequencies higher than the specified range. Additionally, accurately determining the clock frequency is somewhat challenging at this stage, as testing has been conducted through simulations rather than full integration.

4.5.6. ADC

To effectively communicate with the PD modules, the MCU must be equipped with an ADC that has a resolution of at least 10 bits. According to the datasheet [5], the MAX25432B development board (responsible for Power Delivery) is equipped with a 10-bit ADC. Therefore, to ensure compatibility and maintain consistent resolution, the MCU's ADC should also be at least 10-bit.

4.5.7. Comparison between MCUs

Based on the requirements and considerations discussed in this section, the MCUs can be filtered to the STM32F401 [62], STM32F411 [63], ESP32-D0WD [18], and the ATmega328P [45]. These MCUs will be compared with each other based on the requirements for the application and the additional features of the MCUs themselves.

Table 4.5: Comparison of MCUs

	STM32F401	STM32F411	ESP32-D0WD	ATmega328P
Input-Output Lines	Up to 51 GPIOs	Up to 50 GPIOs	39 GPIOs	14 digital I/O, 6 analog I/O
Data Bus	32-bit ARM Cortex-M4	32-bit ARM Cortex-M4	32-bit dual-core Xtensa LX6	8-bit AVR
Memory (SRAM)	64 kB	128 kB	520 kB	2 kB
Memory (Flash)	256 kB	512 kB	None	32 kB
Communication	I²C: 3, UART: 4	I²C: 3, UART: 4	I²C: 2, UART: 3	I²C: 1, UART: 1
Clock Frequency	Up to 84 MHz	Up to 100 MHz	Up to 240 MHz	16 MHz
ADC	12-bit, up to 16 channels	12-bit, up to 16 channels	12-bit, up to 18 channels	10-bit, 6 channels
Cost	€6.36	€6.70	€7.70	€3

Table 4.5 presents the specifications for the application. Based on these specifications, only two MCUs are suitable for the Wrepair Powerstation: the STM32F411 and the ESP32-D0WD. However, the ESP32-D0WD lacks internal flash memory. By using the ESP32-WROVER-B module, which integrates the ESP32-D0WD, this limitation is addressed with the inclusion of 4 MB of external flash memory. Nevertheless, the ESP32-WROVER-B module has additional features, such as wireless capabilities, which are excessive for this particular application. Therefore, the most suitable MCU for the Wrepair Powerstation is the STM32F411 (Figure 4.8).

During the project, the Arduino UNO, featuring the ATmega328P MCU, was utilized for testing purposes because it was readily available at the start. The Arduino UNO is user-friendly with its integrated development environment (IDE) and is more than sufficient for testing. However, for full integration, the STM32F411 will be used.

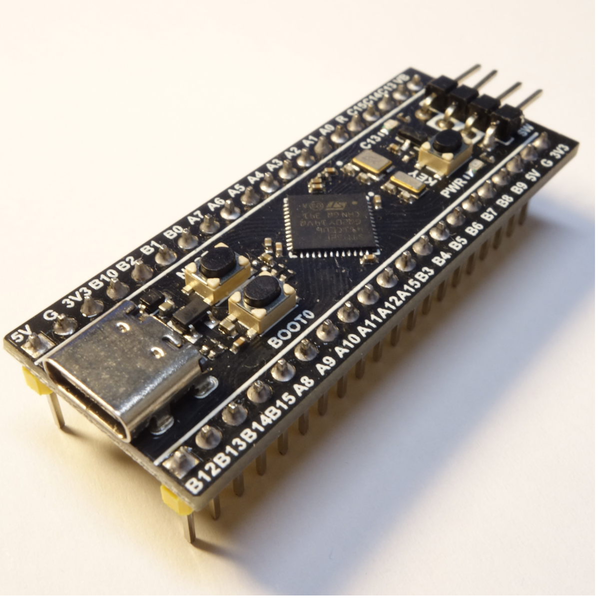


Figure 4.8: DFRobot DFR0864 ”Black Pill” development board featuring the STM32F411CEU6 chip

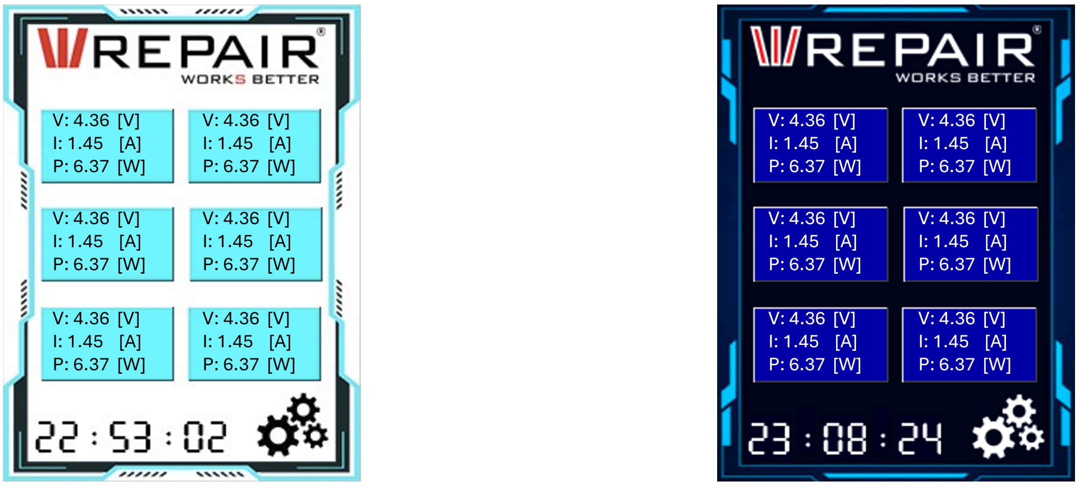
5

Evaluation/Validation

This chapter discusses the evaluation and validation of the Wrepair Powerstation, focusing on the results of the GUI (section 5.1), both static and adjustable power division (section 5.3), and user experience testing (section 5.2).

5.1. GUI Evaluation

The Wrepair Powerstation's GUI was designed to be clear, intuitive, and user-friendly, offering two themes: light and dark. The GUI displays real-time readings for each port, providing essential information such as voltage, current, and power. Figure 5.1a and Figure 5.1b illustrate the homepage in both themes.



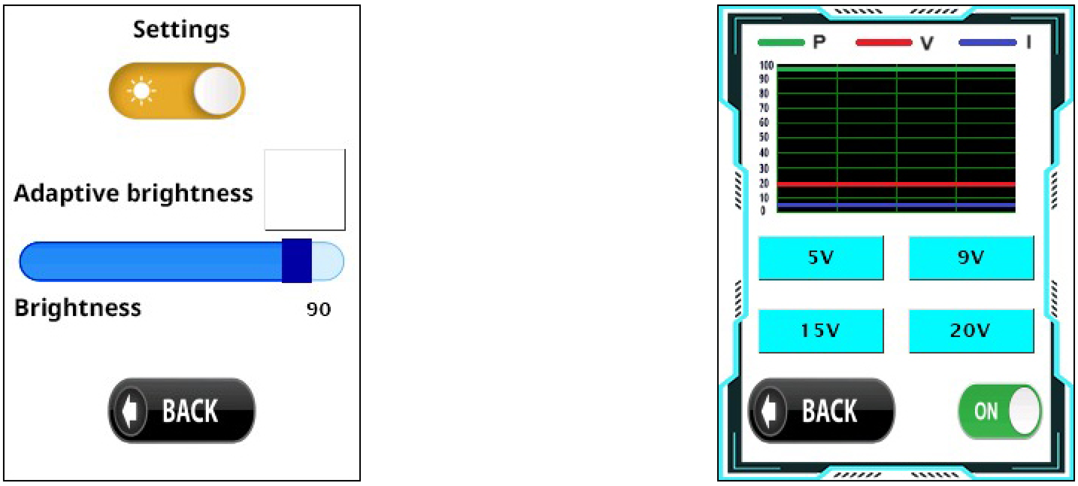
(a) Homepage in light theme

(b) Homepage in dark theme

Figure 5.1: The homepage illustrated in the two themes

The GUI also includes a settings page (Figure 5.2a) and individual graphs for each port (Figure 5.2b). The settings page is easily accessible by tapping the cog-wheel icon on the home page, allowing users to quickly adjust various settings. Users can access the graph page for any specific port by tapping on its corresponding rectangle on the home page. On the graph page, users have the option to limit the power output of the selected port or turn it off entirely.

The settings page enables adjustments for brightness, theme selection, and adaptive brightness, which sets the screen brightness according to the surrounding lighting conditions. Additionally, the homepage features a real-time clock (RTC), allowing users to quickly tell the time at a glance. Meanwhile, the graph page provides a visual representation of power usage over time, helping users monitor and manage power distribution effectively.



(a) Settings Page **(b)** Graph page for each port

Figure 5.2: Settings page and graph page for each port

A tree-view diagram of the GUI structure is depicted in Figure 5.3. This diagram outlines the hierarchical organization and navigation flow of the GUI, demonstrating its clear, intuitive design and the ease with which users can quickly navigate between different pages.

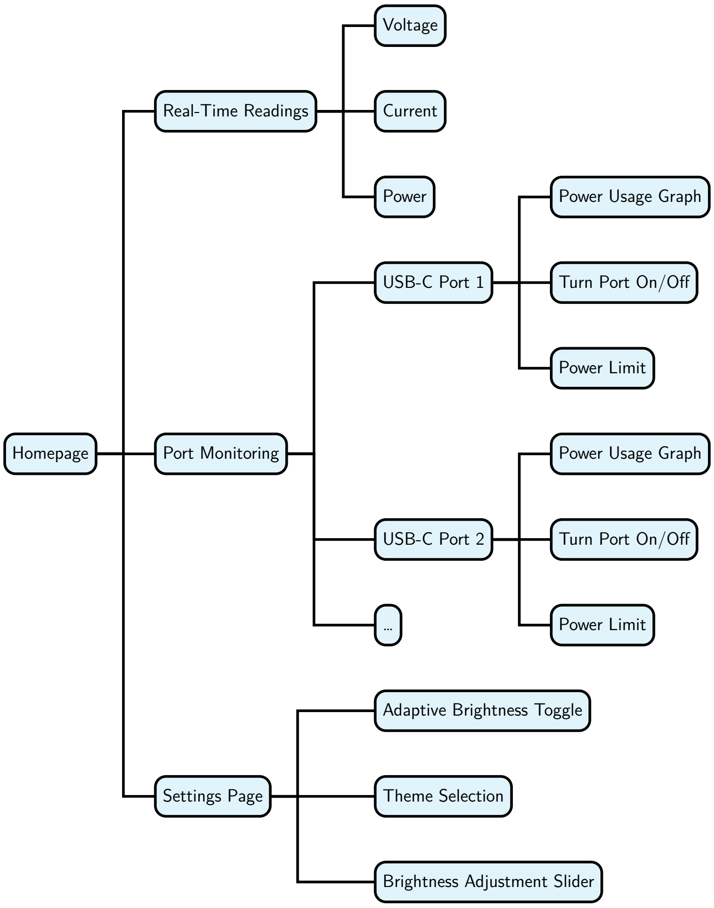


Figure 5.3: Tree-view diagram of the GUI structure

5.2. User Experience Testing

User experience testing was once again conducted with 12 participants to evaluate the usability and effectiveness of the current GUI, which incorporated the feedback received from the first test. The testing method included observations and timing, with an average navigation time of 36 seconds.

- **Negative Feedback**:
    - Buttons were too small (8.3% of participants).
    - Button icons needed improvement (8.3% of participants).
    - Color contrast should be increased (16.7% of participants).
    - Font size was too small (8.3% of participants).
- **Positive Feedback**:
    - The GUI is clear and easy to use (75% of participants).
    - The two themes were well-received, catering to different user preferences (66.7% of participants).
    - Real-time readings for each port were considered highly useful (41.6% of participants).
    - Brightness is a useful addition (33.3% of participants).

The feedback is visually summarized in the bar chart shown in Figure 5.4.

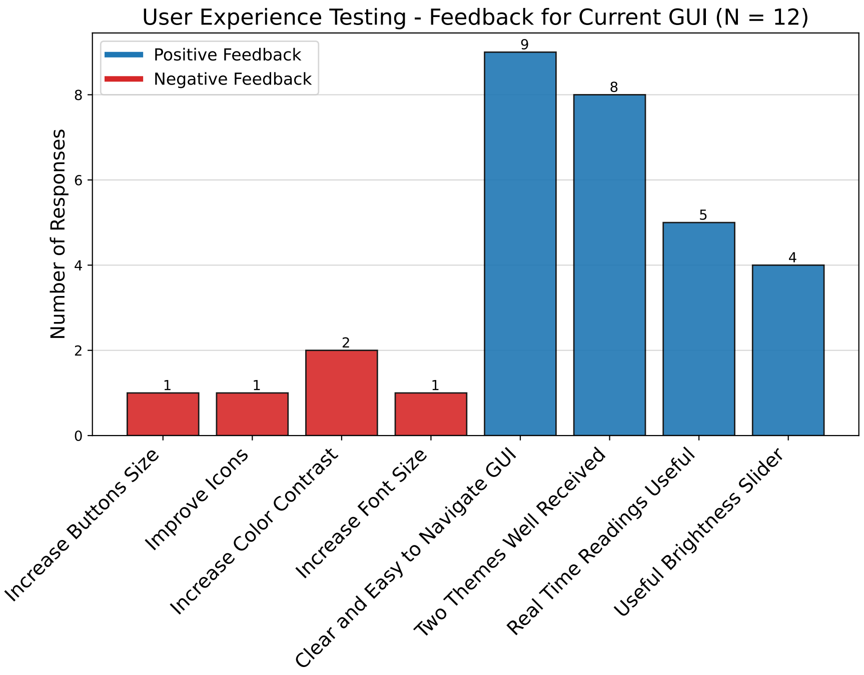


Figure 5.4: Bar chart illustrating both positive and negative feedback received from user experience testing for the current GUI

5.3. Power Division Evaluation

This section evaluates the two power division implementations of the Wrepair Powerstation: static and adjustable. Proper power management is essential for distributing the 300 W input power across the six USB-C ports. It ensures efficient charging by allocating power based on device needs and user preferences, preventing overload, and optimizing the charging process.

Both methods are designed to enhance the user experience by offering flexibility and simplicity in power management, thereby maximizing the utility and efficiency of the Powerstation.

5.3.1. Static Power Division

The static power division implementation consists of a simple on/off digital toggle switch for each port. When the switch is on, the connected device receives the default configuration of the PD module, which is 5 V, provided that enough power is available. When the switch is off, the port is completely disabled, ensuring neither power is supplied to the connected device nor data transfer can take place through this port. This approach is ideal for users who prefer simplicity and direct control without dealing with the complexity of adjusting power levels. The static power division implementation is visually explained in Figure 5.5.

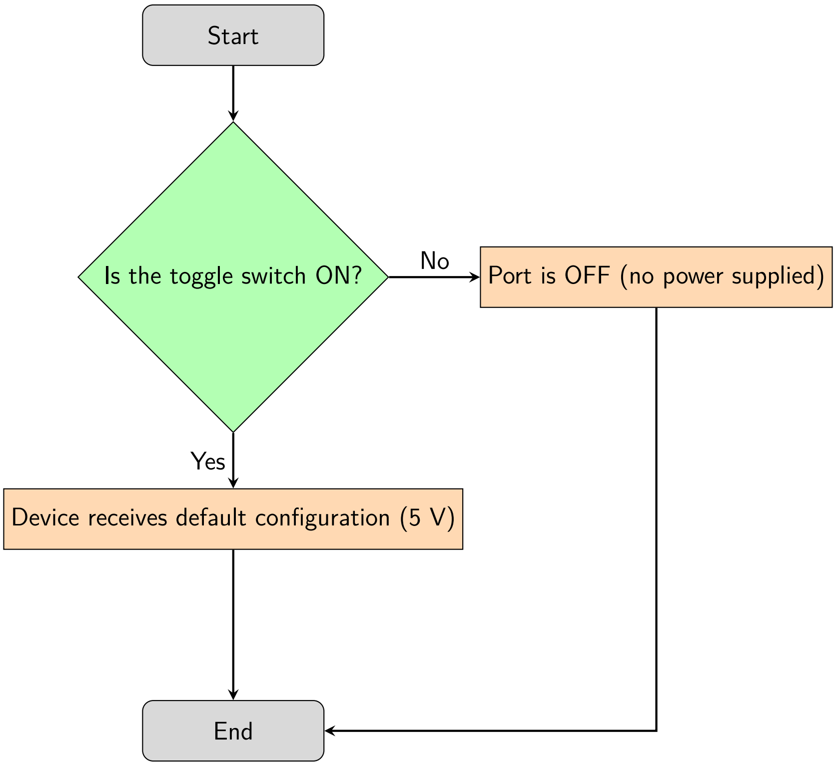


Figure 5.5: Flowchart explaining the static power division process

The static power division, i.e., an on/off toggle switch, is extremely handy and perhaps even crucial in certain scenarios. Take, for example, when a user connects their USB-C soldering iron to the station but does not want to use it yet or turn on the power. By turning off the power for that specific port beforehand, they can avoid potential burn injuries or dangerous situations.

5.3.2. Adjustable Power Division

The adjustable power division implementation, although significantly more complex, offers detailed control over power allocation for each USB-C port. Unlike the static method, this approach allows users to manually configure the power output by choosing from four different voltage levels: 5 V, 9 V, 12 V, and 20 V. This configuration can be made if there is enough available power to meet the requested setting. For instance, if a high-power laptop and a smartphone are connected simultaneously, the user can manually allocate more power to the laptop (e.g., 20 V) while assigning a lower voltage to the smartphone (e.g., 9 V). This ensures that both devices receive appropriate power levels for efficient charging.

Another practical example might involve a scenario where one is charging a tablet that requires rapid charging while also powering smaller devices like earbuds or smartwatches. The tablet's charging needs can be prioritized by assigning it more power while still providing sufficient power to the smaller devices. Despite the need for manual adjustments, the system has a default configuration similar to the static division: when a device is connected, it initially receives 5 V, provided there is enough available power. This default setting ensures that devices begin charging immediately upon connection, even before any manual adjustments are made. A flowchart of the adjustable power division implementation is depicted in Figure 5.6.

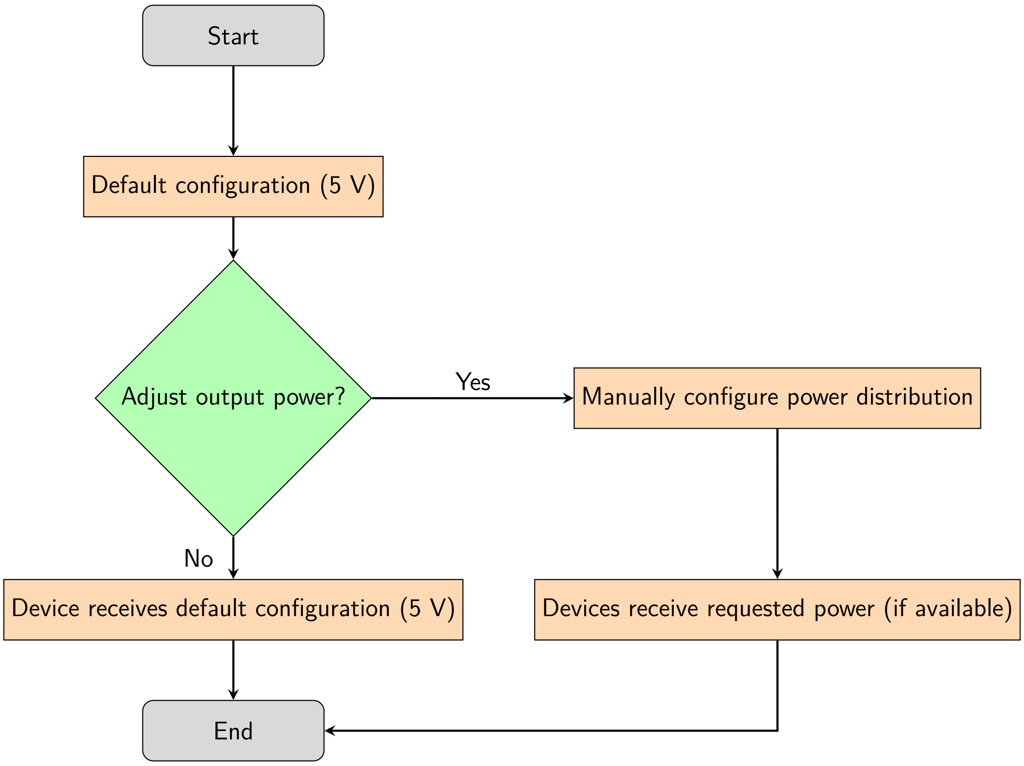


Figure 5.6: Flowchart explaining the adjustable power division process

The ideal implementation would consist of incorporating both the static and adjustable options, giving the user full control and flexibility in limiting and configuring the power division. This dual approach ensures that users can manage power distribution effectively and safely, catering to a wide range of use cases and preferences.

6

Conclusion

This chapter summarizes the main findings of this thesis (section 6.1), talks about the challenges faced during the project (section 6.2), and suggests ways to improve the Wrepair Powerstation in the future (section 6.3).

6.1. Summary

This thesis discussed the design and implementation of a Human Machine Interface (HMI) and microcontroller system for the Wrepair Powerstation, a charging and data transfer station for smart devices. The project focused on developing an intuitive touchscreen interface for real-time monitoring and control of power allocation across multiple USB-C ports. The system allows users to adjust the power output of each port while respecting the limitations of connected devices and the overall power budget of the charging station.

The thesis also addressed the selection and integration of a suitable microcontroller to manage communication between power delivery modules, touchscreen display, and other system components. The Nextion NX3224K024 was chosen for the screen due to its compatibility and ease of use, while the STM32F411 was selected for the controller for its memory capacity and its robust performance. The goal was to create a user-friendly and reliable Powerstation that meets the needs of professionals in the electronics repair industry.

A user-friendly graphical user interface was designed through a rigorous design process, featuring real-time data visualization, adjustable power settings, and intuitive navigation. The GUI was evaluated through user experience testing, and feedback was incorporated to improve its usability and functionality.

In conclusion, the Wrepair Powerstation project demonstrates the potential of integrating HMI and microcontroller systems to create an innovative and user-friendly charging solution.

6.2. Discussion and Recommendation

Both the static and adjustable power division features were successfully Implemented and integrated with the USB implementation submodule. This integration was thoroughly tested, confirming the system's capability to manage power distribution effectively. The system demonstrated the ability to toggle USB-C ports on and off and to change the output voltage as required.

Despite these successful implementations, the project faced limitations due to the availability of only one USB-C PD module. This restriction limited the ability to test multiple ports simultaneously, highlighting the need for more extensive testing to ensure robust performance across all ports.

6.3. Future Work

Future work on the Wrepair Powerstation would focus on several areas to improve its functionality and usability.

Firstly, integrating additional PD modules is crucial for extensive real-world testing and validation. While the project successfully integrated one PD module, obtaining and integrating multiple modules (ideally six) would allow simultaneous testing of all six ports. This extensive testing will ensure the modules function correctly and support both static and adjustable power division across multiple ports.

Additionally, continuously improving the GUI based on user feedback will make sure it remains user-friendly and intuitive. Advanced features such as real-time data logging and export, a toggle between ”charge-only” and ”charge+data transfer” modes, and data activity indicators (e.g., a blinking icon during USB data transfer) could further enhance the user experience.

Moreover, implementing a dynamic power division algorithm will significantly improve the system’s functionality. This intelligent algorithm would automatically adjust the power division across all ports based on the connected devices’ needs, providing optimal power management and user convenience without requiring manual intervention.

By addressing these areas, the Wrepair Powerstation can become a more versatile and reliable tool for professionals in electronics labs, providing greater control and flexibility in managing power distribution for multiple devices.

Bibliography

[1] *280W Zeus USB-C GaN Charger – Chargeasap*. URL: https://chargeasap.com/products/270w-zeus-charger (cit. on p. 5).

[2] *300W GaN Charger*. URL: https://www.alibaba.com/product-detail/300w-gan-charger-300w-adapter-for_1600999490509.html?spm=a2700.7724857.0.0.7241624bWo9Fv8 (cit. on p. 5).

[3] Isiaka A. Alimi et al. "Network-on-Chip Topologies: Potentials, Technical Challenges, Recent Advances and Research Direction". In: *Network-on-Chip - Architecture, Optimization, and Design Explorations*. IntechOpen, 2021. DOI: 10.5772/intechopen.97262. URL: https://www.intechopen.com/chapters/76266 (cit. on p. 23).

[4] Mohammad Alsharif. "Touch Screens Technologies". PhD thesis. Mar. 2013. DOI: 10.13140/RG.2.1.4358.3608 (cit. on p. 17).

[5] ANALOGDEVICES. *MAX25432 Datasheet and Product Info | Analog Devices*. URL: https://www.analog.com/en/products/max25432.html (cit. on pp. 24, 25).

[6] Antonio Bento. "An Experimental Survey with NodeMCU12eShield with Tft Nextion and MAX30102 Sensor". In: *IEE Proceedings* (Dec. 2020), pp. 0082–0086. DOI: 10.1109/IEMCON51383.2020.9284870 (cit. on p. 20).

[7] Aurora Berni and Yuri Borgianni. "FROM THE DEFINITION OF USER EXPERIENCE TO A FRAMEWORK TO CLASSIFY ITS APPLICATIONS IN DESIGN". In: *Proceedings of the Design Society* 1 (July 2021), pp. 1627–1636. DOI: 10.1017/pds.2021.424 (cit. on p. 7).

[8] M. van Beusekom. *WRepair Powerstation*. 2024. URL: https://brightspace.tudelft.nl/content/enforced/595622-EE3L11+2023+4/Files/BSc%20thesis/BAP%20proposals/2023-2024/M_van_Beusekom-WRepair_Powerstation.pdf?isCourseFile=true&ou=595622 (cit. on p. 2).

[9] Ashutosh Bhatt. "How to select a Micro Controller for Particular Application?" In: *International Journal of Engineering Research & Technology (IJERT)* 6 (4 Apr. 2017), p. 6. DOI: 10.17577/IJERTV6IS040735 (cit. on pp. 24, 25).

[10] Lehan Cao, Jianfeng Chen, and Jincheng Li. "Working principle and application analysis of UART". In: *2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*. IEEE. Apr. 2023, pp. 255–259. DOI: 10.1109/EEBDA56825.2023.10090571 (cit. on p. 17).

[11] Hai-Wei Chen et al. "Liquid crystal display and organic light-emitting diode display: present status and future perspectives". In: *Light: Science & Applications* 7 (Mar. 2017), p. 17168. DOI: 10.1038/lsa.2017.168 (cit. on p. 16).

[12] Jinrong Chen and Shengqiu Huang. "Analysis and Comparison of UART, SPI and I2C". In: *2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*. IEEE. 2023, pp. 272–276. DOI: 10.1109/EEBDA56825.2023.10090677 (cit. on p. 17).

[13] Pong P. Chu. *FPGA Prototyping by VHDL Examples: Xilinx MicroBlaze MCS SoC*. 2nd. Wiley Publishing, Oct. 2017. ISBN: 9781119282747 (cit. on p. 23).

[14] GDK Devi and G Swamy. "An Overview of Microcontroller Unit: from Proper Selection to Specific Application". In: 3 (2016), p. 6 (cit. on p. 24).

[15] A.H.G. Al-Dhaher. "Development of Microcontroller/FPGA-based systems". In: *International Journal of Engineering Education* 20 (Jan. 2004), pp. 52–60 (cit. on p. 23).

[16] Jagannath Dixit, Devesh Katiyar, and Gaurav Goel. “Automatic Street Light Controller System Using LDR and PIR Sensor”. In: *International Journal of Creative Research Thoughts (IJCRT)* 9.5 (2021), h66–h78. ISSN: 2320-2882. URL: https://ijcrt.org/papers/IJCRT2105743.pdf (cit. on p. 21).

[17] *ESP32-2432S024C capacitive touch IPS display module supports WiFi and Bluetooth*. URL: https://www.displaysmodule.com/sale-40776756-esp32-2432s024c-capacitive-touch-ips-display-module-supports-wifi-and-bluetooth.html (cit. on p. 19).

[18] Espressif Systems. *ESP32-WROVER-B Datasheet*. 2024. URL: https://www.espressif.com/sites/default/files/documentation/esp32-wrover-b_datasheet_en.pdf (cit. on p. 25).

[19] Bollam Eswari et al. “Implementation of I2C master bus controller on FPGA”. In: Apr. 2013, pp. 1113–1116. ISBN: 978-1-4673-4865-2. DOI: 10.1109/iccsp.2013.6577229 (cit. on p. 25).

[20] Christos Fidas et al. “Remote Usability Evaluation Methods and Tools: A Survey”. In: Jan. 2006 (cit. on p. 11).

[21] Dhananjay V Gadre et al. “Serial Communication: SPI and I2C”. In: *Getting Started with Tiva ARM Cortex M4 Microcontrollers: A Lab Manual for Tiva LaunchPad Evaluation Kit* (2018), pp. 211–238 (cit. on p. 17).

[22] Matías García and Sandra Cano. “Eye Tracking to Evaluate the User eXperience (UX): Literature Review”. In: June 2022, pp. 134–145. ISBN: 978-3-031-05060-2. DOI: 10.1007/978-3-031-05061-9_10 (cit. on p. 11).

[23] Joseph Goldberg and Anna Wichansky. “Eye Tracking in Usability Evaluation”. In: Dec. 2003, pp. 493–516. ISBN: 9780444510204. DOI: 10.1016/B978-044451020-4/50027-X (cit. on p. 11).

[24] Yılmaz Güven et al. “Understanding the Concept of Microcontroller Based Systems To Choose The Best Hardware For Applications”. In: *Research Inventy: International Journal of Engineering And Science* 7 (Dec. 2017), p. 38 (cit. on p. 23).

[25] Amir HajiRassouliha et al. “Suitability of recent hardware accelerators (DSPs, FPGAs, and GPUs) for computer vision and image processing algorithms”. In: *Signal Processing: Image Communication* 68 (2018), pp. 101–119. ISSN: 0923-5965. DOI: https://doi.org/10.1016/j.image.2018.07.007. URL: https://www.sciencedirect.com/science/article/pii/S0923596518303606 (cit. on p. 23).

[26] Layla Hasan. “The usefulness of user testing methods in identifying problems on university websites”. In: *Journal of Information Systems and Technology Management* 11 (Aug. 2014), pp. 229–256. DOI: 10.4301/S1807-17752014000200002 (cit. on p. 11).

[27] S Hellweger, X Wang, and P Abrahamsson. “The Contemporary Understanding of User Experience in Practice”. In: (2015) (cit. on p. 7).

[28] En-Lin Hsiang et al. “Prospects and challenges of mini-LED, OLED, and micro-LED displays”. In: *Journal of the Society for Information Display* 29 (May 2021). DOI: 10.1002/jsid.1058 (cit. on p. 15).

[29] Ruijie Hu. “The advantages and comparison of resistive touch screens and capacitive touch screens”. In: *Applied and Computational Engineering* 30 (Jan. 2024), pp. 94–103. DOI: 10.54254/2755-2721/30/20230080 (cit. on p. 17).

[30] Yuge Huang et al. “Mini-LED, Micro-LED and OLED displays: present status and future perspectives”. In: *Light: Science & Applications* 9 (Dec. 2020). DOI: 10.1038/s41377-020-0341-9 (cit. on pp. 15, 16).

[31] S Hyder et al. “Field Programmable Gate Array Implementation Technology”. In: *International Journal of Engineering and Advanced Technology* (Oct. 2012), pp. 25–29 (cit. on p. 23).

[32] *Is Your Phone’s Screen At Risk Of Burn-In?* https://www.compareandrecycle.co.uk/blog/is-your-phones-screen-at-risk-of-burn-in (cit. on p. 16).

[33] Alvin Jacob, Wan Nurshazwani Wan Zakaria, and M Tomari MRB. “Evaluation of I2C communication protocol in development of modular controller boards”. In: *ARPN Journal of Engineering and Applied Science* 11.8 (2016), pp. 4991–4996 (cit. on p. 17).

[34] Hepeng Jia. “Who will win the future of display technologies?” In: *National Science Review* 5.3 (Apr. 2018), pp. 427–431. ISSN: 2095-5138. DOI: `10.1093/nsr/nwy050`. eprint: `https://academic.oup.com/nsr/article-pdf/5/3/427/31568279/nwy050.pdf`. URL: `https://doi.org/10.1093/nsr/nwy050` (cit. on p. 16).

[35] K. Kobayashi et al. “7.3: Late-News Paper: IPS mode TFT-LCDs For Aircraft Application”. In: *SID Symposium Digest of Technical Papers* 29.1 (1998), pp. 70–73. DOI: `https://doi.org/10.1889/1.1833859`. eprint: `https://sid.onlinelibrary.wiley.com/doi/pdf/10.1889/1.1833859`. URL: `https://sid.onlinelibrary.wiley.com/doi/abs/10.1889/1.1833859` (cit. on p. 14).

[36] Antti Laaperi. “OLED lifetime issues from a mobile-phone-industry point of view”. In: *Journal of the Society for Information Display* 16.11 (2008), pp. 1125–1130. DOI: `https://doi.org/10.1889/JSID16.11.1125`. URL: `https://sid.onlinelibrary.wiley.com/doi/abs/10.1889/JSID16.11.1125` (cit. on p. 16).

[37] T Lambert. *Introduction to Microcontrollers and Embedded Systems*. 2023. URL: `https://www.eng.auburn.edu/~dbeale/MECH4240-50/Introduction%20to%20Microcontrollers%20and%20Embedded%20Systems.pdf` (cit. on p. 23).

[38] Lars Larsen et al. “Remote User Testing”. In: Aug. 2021, pp. 579–583. ISBN: 978-3-030-85606-9. DOI: `10.1007/978-3-030-85607-6_81` (cit. on p. 11).

[39] Richard Lawrence. *Mobile Displays: Technology and Applications*. John Wiley & Sons, Ltd, May 2008. ISBN: 9780470994641. DOI: `10.1002/9780470994641`. URL: `https://onlinelibrary.wiley.com/doi/book/10.1002/9780470994641` (cit. on p. 18).

[40] Tania Lazaro. *Touchscreen technology*. Jan. 2017. URL: `https://www.mynewsdesk.com/se/vikingen-ab/blog_posts/touchscreen-technology-53645` (cit. on p. 17).

[41] James Lewis and Jeff Sauro. “USABILITY AND USER EXPERIENCE: DESIGN AND EVALUATION”. In: Aug. 2021, pp. 972–1015. ISBN: 9781119636083. DOI: `10.1002/9781119636113.ch38` (cit. on p. 11).

[42] *MaAnt 8 Ports Multifunctional Charging Station*. URL: `https://shop.rewa.tech/detail/maant-8-ports-multifunctional-chargering-station.html` (cit. on p. 5).

[43] Jayant Mankar et al. “Review of I2C protocol”. In: *International Journal of Research in Advent Technology* 2.1 (2014) (cit. on p. 17).

[44] W. L. Martinez. “Graphical User Interfaces”. In: *WIREs Computational Statistics* 3 (2011), pp. 119–133. DOI: `10.1002/wics.150`. URL: `http://dx.doi.org/10.1002/wics.150` (cit. on p. 10).

[45] Microchip Technology Inc. *ATmega328P Microcontroller*. 2024. URL: `https://www.microchip.com/en-us/product/atmega328p` (cit. on p. 25).

[46] S Mishra et al. “FROM APPOSITE PREFERENCE TO SPECIFIC APPLICATION: AN OVERVIEW OF MICROCONTROLLER UNIT”. In: *International Journal of Technical Research & Science* 2 (2017), p. 8 (cit. on p. 24).

[47] Hyoungsik Nam et al. “Review of Capacitive Touchscreen Technologies: Overview, Research Trends, and Machine Learning Approaches”. In: *Sensors* 21.14 (July 2021). ISSN: 1424-8220. DOI: `10.3390/s21144776`. URL: `https://www.mdpi.com/1424-8220/21/14/4776` (cit. on p. 17).

[48] Jakob Nielsen. *Usability engineering*. San Francisco, Calif.: Morgan Kaufmann Publishers, 1994. ISBN: 9780125184069. URL: `https://www.nngroup.com/books/usability-engineering/` (cit. on p. 11).

[49] M Niranjanamurthy et al. “Research study on importance of usability testing/User Experience (UX) testing”. In: *International Journal of Computer Science and Mobile Computing* 3.10 (2014), pp. 78–85 (cit. on p. 11).

[50] Sergey Nivens. *Using modern technologies | Adobe Stock*. URL: `https://stock.adobe.com/images/using-modern-technologies/98279455` (cit. on p. i).

[51] *NX3224K024 – Nextion 2.4” Enhanced Series HMI Touch Display*. URL: `https://itead.cc/product/nx3224k024-nextion-2-4-enhanced-series-hmi-touch-display/` (cit. on p. 19).

[52] *OLED vs MicroLED - a technology comparison | OLED Info*. URL: https://www.oled-info.com/oled-vs-microled-technology-comparison (cit. on p. 16).

[53] Bhrijesh N Patel and Mrugesh M Prajapati. "OLED: a modern display technology". In: *International Journal of Scientific and Research Publications* 4.6 (2014), pp. 1–5. URL: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=404ce87063f98616c42e08b2c969f8d52cfa6a1e (cit. on p. 16).

[54] David A. Patterson and John L. Hennessy. *Computer Organization and Design: The Hardware/Software Interface*. 5th. Morgan Kaufmann, 2021. URL: https://www.elsevier.com/books/computer-organization-and-design/patterson/978-0-12-812275-4 (cit. on p. 24).

[55] I Plauska, A Liutkevičius, and A Janavičiūtė. "Performance Evaluation of C/C++, MicroPython, Rust and TinyGo Programming Languages on ESP32 Microcontroller". In: *Electronics* 12.1 (2023), p. 143. DOI: 10.3390/electronics12010143. URL: https://www.mdpi.com/journal/electronics (cit. on p. 23).

[56] *Principles and Structures of FPGAs*. Springer Singapore, 2018. ISBN: 9789811308246. DOI: 10.1007/978-981-13-0824-6. URL: http://dx.doi.org/10.1007/978-981-13-0824-6 (cit. on p. 23).

[57] Federico Quin et al. "A/B testing: A systematic literature review". In: *Journal of Systems and Software* 211 (2024), p. 112011. ISSN: 0164-1212. DOI: https://doi.org/10.1016/j.jss.2024.112011. URL: https://www.sciencedirect.com/science/article/pii/S0164121224000542 (cit. on p. 11).

[58] Elizabeth Rosenzweig. *Successful User Experience: Strategies and Roadmaps*. 1st ed. Print Pub Date: 2015-10-14. Elsevier Science & Technology, 2015. ISBN: 9780128009857 (cit. on pp. 7, 10, 11).

[59] Homayoun Shahri. "Blurring Lines Between Hardware and Software". In: *ACM Queue* 1.2 (2003). URL: https://queue.acm.org/detail.cfm?id=644267 (cit. on p. 23).

[60] Ben Shneiderman and Erik Eka Putra. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. 4th. Boston, 2009. ISBN: 9780321197856 (cit. on pp. 8–10).

[61] Rudolf Sosnowsky. "DisplayPort". In: *Handbook of Visual Display Technology*. Ed. by Janglin Chen, Wayne Cranton, and Mark Fihn. Cham: Springer International Publishing, 2016, pp. 667–673. ISBN: 978-3-319-14346-0. DOI: 10.1007/978-3-319-14346-0_174. URL: https://doi.org/10.1007/978-3-319-14346-0_174 (cit. on p. 18).

[62] *STM32F401 - STMicroelectronics*. URL: https://www.st.com/en/microcontrollers-microprocessors/stm32f401.html (cit. on p. 25).

[63] STMicroelectronics. *STM32F411 Microcontroller*. 2024. URL: https://www.st.com/en/microcontrollers-microprocessors/stm32f411.html (cit. on p. 25).

[64] S.M. Thompson. "Remote observation strategies for usability testing". In: *Information Technology and Libraries* 22 (Mar. 2003), pp. 22–31 (cit. on p. 11).

[65] Rob Tobias et al. "High Definition Multimedia Interface (HDMI®)". In: *Handbook of Visual Display Technology*. Ed. by Janglin Chen, Wayne Cranton, and Mark Fihn. Cham: Springer International Publishing, 2016, pp. 695–709. ISBN: 978-3-319-14346-0. DOI: 10.1007/978-3-319-14346-0_37. URL: https://doi.org/10.1007/978-3-319-14346-0_37 (cit. on p. 18).

[66] *Ugreen Nexode 300W USB C GaN Charger-5 Ports Desktop Charger | UGREEN US*. URL: https://www.ugreen.com/products/ugreen-300w-desktop-usb-c-gan-charger?variant=40359894515774 (cit. on p. 5).

[67] *What is UART*. https://www.solomotorcontrollers.com/blog/uart/ (cit. on p. 18).

[68] *Wholesale 2.4 inch smart UART Screen DMG32240C024_03W | DWIN*. URL: https://www.dwin-global.com/2-4-inch-smart-uart-screen-dmg32240c024_03wcommercial-grade-product/ (cit. on p. 19).

[69] Cliff Wootton. "Serial Peripheral Interface (SPI)". In: *Samsung ARTIK Reference: The Definitive Developers Guide* (2016), pp. 335–349 (cit. on p. 17).

[70] *Wrepair*. URL: https://www.wrepair.com/ (cit. on pp. 1, 2).

[71] Huanjie Wu, Chun Chen, and Kai Weng. “An Energy-Efficient Strategy for Microcontrollers”. In: *Applied Sciences* 11.6 (Mar. 2021), p. 2581. DOI: 10.3390/app11062581. URL: https://www.mdpi.com/2076-3417/11/6/2581 (cit. on p. 23).

[72] Tingzhu Wu et al. “Mini-LED and Micro-LED: Promising Candidates for the Next Generation Display Technology”. In: *Applied Sciences* 8.9 (2018). ISSN: 2076-3417. DOI: 10.3390/app8091557. URL: https://www.mdpi.com/2076-3417/8/9/1557 (cit. on p. 16).

[73] Morelife Zibayiwa. “A Review on The Inter-Processor Communication: I2C, UART, and SPI interfacing techniques”. In: (Nov. 2021). URL: https://www.researchgate.net/publication/356556000_A_Review_on_The_Inter-Processor_Communication_I2C_UART_and_SPI_interfacing_techniques (cit. on p. 17).