BSc Thesis Report

Wrepair Powerstation: USB implementation

Bert van Lange [5348285]
Pieter Pronk [5446376]

July 3, 2024

DELFT UNIVERSITY OF TECHNOLOGY

FACULTY OF ELECTRICAL ENGINEERING, MATHEMATICS
AND COMPUTER SCIENCE

ELECTRICAL ENGINEERING PROGRAMME

Abstract

This report focuses on the USB implementation subgroup of the Wrepair Powerstation Project, which aimed to design and develop a versatile USB powerstation capable of supporting both high-power delivery and data transfer through USB Type-C ports. The implementation involved integrating advanced charging protocols, real-time measurements, and power control capabilities into a modular system that can seamlessly fit into existing Wrepair stations. This integration enables the powerstation to feature dynamic power allocation and visualisation of charging performance.

Key tasks of this project included selecting appropriate power delivery integrated circuits (ICs), designing a high-speed USB data hub, and developing control hardware to manage power and data flow efficiently. The USB charging station is designed to support up to 100W per port, with a total available power budget of 300W, distributed across six USB Type-C ports. The powerstation also incorporates safety features to protect against overcurrent, overheating, and short circuits, ensuring user safety and device protection.

The report details the design process, including the selection of components, system integration, and validation procedures. The design process involved evaluating different power delivery systems, including development boards, PD modules, and stand-alone chips, to select the most suitable solution for the project requirements. The chosen solution, the IP2368 PD (Power delivery) module, was initially tested but was found to be inadequate, leading to the exploration of alternative modules such as the SW2303 and MAX25430 development boards.

Furthermore, the design includes a USB data hub sub-module to facilitate high-speed data transfer and connectivity. This hub supports multiple devices and allows for real-time data logging and display, enhancing user interaction and monitoring capabilities. The validation and evaluation phase involved rigorous testing of the USB implementation sub-module, including power delivery, measuring and controlling capabilities, and overall system integration.

The results demonstrate that the USB implementation meets the required specifications, ensuring efficient power distribution and data handling for multiple devices. This work contributes to the overall Wrepair Powerstation Project by providing a robust and scalable USB solution that enhances the functionality and utility of Wrepair stations.

Preface

The completion of this thesis marks the end of our Bachelor's in Electrical Engineering at Delft University of Technology. The objective of the project was to create a functional, reliable, and safe powerstation that meets the high standards of modern electronic devices.

The growing acceptance and adoption of USB Type-C Technology in contemporary electronic devices was a major stimulant to take on the challenge Wrepair provided. From the start, we were motivated by the challenge of creating a reliable and efficient charging solution that could seamlessly integrate into the existing Wrepair stations. The project demanded a deep dive into advanced charging protocols, real-time measurement techniques, and power control systems, which presented unique challenges that tested our problem-solving skills and technical knowledge.

This document presents the findings and outcomes of our project on the development of a USB powerstation for Wrepair stations. The technical and practical knowledge gained through this project has been very valuable, and we are proud of the contributions we have made to the Wrepair Powerstation project.

This thesis would not have been possible without the guidance and support of several individuals. We would like to extend our thanks to our supervisors, Mark van Beusekom and Ali Kaichouhi, for their insights and constant encouragement. Our peers, Alex Gulam, Amin Aynan, Yahya Al-Araij, and Zubair Al-Zubi, provided critical feedback and support throughout the project, for which we are grateful. In addition, we appreciate the assistance and resources provided by Wrepair, which were instrumental in our research and development process.

Bert van Lange
Pieter Pronk
Delft, June 2024

Contents

Chapter 1

Introduction

This chapter covers the motivation for developing a versatile USB Type-C powerstation in section 1.1, an overview of the client Wrepair and their requirements in section 1.2, and the assignment details specifying the powerstation its features in section 1.3. It defines the project its problem and high-level requirements in section 1.4, describes the structured task division among subgroups in section 1.5, compares existing products in section 1.6, highlights the project its contributions in section 1.7, and presents the report's organizational structure in section 1.8.

1.1 Motivation

The increasing prevalence of USB Type-C as a universal charging standard necessitates the development of versatile and high-power charging solutions for powering devices such as laptops, phones, or other devices and tools. Existing solutions often lack the ability to support multiple devices simultaneously and also lack the combination with data transfer capabilities, while ensuring user safety and efficient power distribution.

Currently, most charging stations lack the ability to have an interface for the visualization of the charging station performance as well as the capability of power allocation and data transfer through the ports. Another insight that is missing most of the time, is measurement data of voltage and current.

This project addresses these inabilities by designing a powerstation (as seen in Figure 1.1) that not only supports fast charging for a variety of devices and data transfer capability but also integrates seamlessly into the Wrepair ecosystem, enhancing the utility and functionality of the Wrepair repair station, which is shown in Figure 1.2.

Figure 1.1: Powerstation stand alone

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

1.2 Wrepair

The client for this project is Wrepair [1], 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 ports Figure 1.2.

Figure 1.2: Wrepair repair station [1]

The idea for their products comes from the fact that they had trouble with messy work areas, finding tools, and knowing which screws went where. This made them want to find better ways, so they made new solutions such as the Wrepair station.

The Wrepair stations are the ultimate solution for efficient tool organization, thereby upgrading your workspace. They are compact, durable and have space for multiple tools. They are USB powerport, hub, and NUC ready.

1.3 Assignment

The task at hand is to design a powerstation that can fit into existing Wrepair stations. It must support the current power protocols, ensuring safety for the user, external devices, and the powerstation itself.
Moreover, the Wrepair powerstation should be able to display relevant charging information of the connected USB ports. As an added bonus, the device could pass through or detect data from the connected device to an external computer.

In the original BAP proposal the following properties were stated:

1. Charging smart devices like Smartphones, tablets, laptops, gaming devices, and wearables
2. Displaying Watt, Voltage, and Ampere on a touchscreen
3. USB-A and USB-C ports (6 total)
4. Standard version: Charging ONLY
   Plus version: Data Transfer 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 output: MAX 140W
   USB-A ports output: MAX 36W
7. Security: 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 and maximum output: to be determined later
10. 230V external power adapter

1.4 Problem Definition

The high-level requirements for the project, presented in Table 1.1 (next page), 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 (which is elaborated in section 2.3), backward compatibility (as discussed in section 2.2), and alignment with current industry standards for charging. This simplification also made the design easier to implement. The output of these USB Type-C ports was adjusted to 100W after consultation with Wrepair, as this power level is more commonly supported and practical for the intended applications (as stated in subsection 2.3.1).
- **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 its 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 blue 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.

Here is the revised list of high-level requirements, reflecting the project adjustments:

Mandatory requirements or constraints
The product must have six USB-C charging ports
The product must charge with at least 100W.
The product must be operating at a maximum of 300W.
The product must be safe for users and devices to use.
The product must be suitable for a lab environment.
The product must be user-friendly.
The product must be able to measure current, voltage, power per USB Type-C port
The product must display the current, voltage, and power used by the USB Type-C port.
The product must have a selling price that is targeted at €300,
so price of whole system must be below €200 to have margin.
The product must fit in the Wrepair station.
The product must be able to be used as stand-alone powerstation
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 be connected 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 Type-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

Table 1.1: High-level requirements of the project

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. The product must feature six USB Type-C charging ports with support for various charging protocols, operate at a maximum of 300W, and display current, voltage, and power metrics. To efficiently address these requirements, the project was divided into several sub-modules, each focusing on a specific aspect of the product:

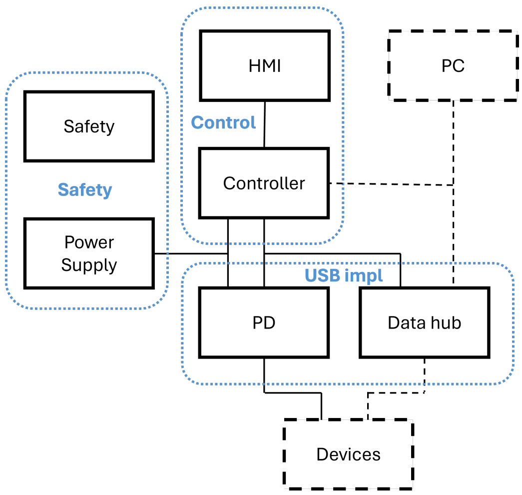


Figure 1.3: Overview of whole system

- **PD (Power Delivery)**: This module is essential for managing the power delivery protocols required for the USB Type-C ports, ensuring proper charging performance and compatibility with various devices.
- **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 300W with 20V for the whole system and realizes the lower voltage levels for the HMI and the controller.
- **Safety of the Entire Powerstation**: 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, dynamically 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 current, voltage, and power metrics, and possibly providing graphical representations and other user interactions.

To efficiently address these sub-modules, the project was subdivided into three main categories, with each category assigned to a dedicated team:

Table 1.2: Task Distribution

Category	**USB implementation**	**Safety**	**Control**
Assigned Team	***Bert van Lange*** ***Pieter Pronk***	***Amin Aynan*** ***Yahya Al-Araij***	***Alex Gulam*** ***Zubair Al-Zubi***
Module 1	PD	Safety of Entire Powerstation	Controller
Module 2	Data Hub	Power Supply	HMI

Responsibilities of the USB implementation subgroup
This report focuses on the USB implementation subgroup, which is responsible for the USB charging and USB data capabilities. It is also responsible for creating the drivers to control the hardware, allowing the control group to focus solely on the software implementation for controlling the USB Type-C ports.

1.6 State of the Art

In this section, a good look will be taken at products that already exist and their features that will be interesting for the USB implementation subgroup. This will help create an understanding of the possibilities of the project. It is also necessary to ensure the product design is unique and does not already exist.

Different charging stations are mentioned. The main functionality of these devices is to charge a product. Some of them have added features that can be useful. The following stations are used for the comparison:

- PinePower desktop power supply(Figure 1.4) [2]
- MaAnt multifunctional charging station(Figure 1.5) [3]
- Chargeasap ZEUS USB-C GaN CHARGER (Figure 1.6) [4]
- UGREEN Nexode USB-C GaN (Figure 1.7) [5]

The criteria on which the products will be compared in Table 1.3:

- Amount of power of the whole station and per port.
- Visualisation of charging station performance.
- Special features that are not common in a charging station.

Figure 1.4: PinePower

Figure 1.5: MaAnt

Table 1.3: Comparison of existing charging stations

Specifications	**PinePower**	**MaAnt**	**Chargeasap**	**UGREEN**
Power & ports	• 120W • 1 USB-C (65W) • 4 USB-A (3x15W, 1x18W)	• 1 USB-C (20W) • 7 USB-A (6x15W, 1x18W)	• 280W • 3 USB-C (2x140W, 1x100W) • 1 USB-A (65W)	• 300W • 4 USB-C (1x140W, 2x100W, 1x45W) • 1 USB-A (22.5W)
Visualisation	V & I per port	V & I per port	V, I & P per port	No display
Special features	Wireless charging	• Wireless charging • Multi-meter	Power prioritization for 140W ports	Power dispenser-system

This analysis of existing products shows that these products have fast charging ports, but the term fast charging has frequently been used differently. In some cases, like the MaAnt, 20W is already considered fast charging. However, this is not the case when you compare it to the other charging stations and the USB charging standards mentioned earlier.
Furthermore, most of the stations have voltage and current measurements to get insight into the performance of the charging.
The higher total power charging stations also have built-in power management systems, to efficiently manage the power when reaching the limit.

Figure 1.6: Chargeasap

Figure 1.7: UGREEN

The product that will be built for Wrepair has also these capabilities but with improved functionality like manual control of power allocation and more fast charging ports.

1.7 Contribution

This project contributes to the field of power electronics by developing a comprehensive solution for high-power USB-C charging and data transfer. Key contributions include:

- The design and implementation of a modular power delivery system that can be easily integrated into existing Wrepair stations.
- Integrate with data transfer and connectivity, allowing the product to function as a USB data hub and support high-speed data transfer next to the fast charging capability.
- The development of a robust safety and data security mechanism to protect users and devices.
- The incorporation of real-time data logging and display features for better user interaction and monitoring.

In this report, the focus is set on the first two points. For which different solutions are elaborated and validated.

1.8 Organization

The report is structured as follows:

1. **Introduction**: Provides an overview of the project, including the motivation, assignment, and problem definition.
2. **Technical Review**: Discusses the history, connectors, charging protocols, and data transfer speeds associated with USB technology.
3. **Design Description**: Details system requirements, subsystem specifications, and the design choices for the USB powerstation.
4. **Validation and Evaluation**: Describes the testing procedures and results for the implemented USB modules and the overall system integration.
5. **Conclusion**: Summarises the project outcomes, provides recommendations, addresses any challenges faced during the project, and future work.
6. **Bibliography**: Lists all the references and sources used throughout the project.

This structure ensures a logical flow of information, guiding through the project its development from inception to conclusion.

Chapter 2

Technical Review

This chapter provides a concise overview of key aspects of USB technology relevant to the project. It covers the history and evolution of USB technology in section 2.1, different types of USB connectors and wiring in section 2.2, and USB charging protocols in section 2.3. It also discusses USB data transfer capabilities in section 2.4, the design of USB multi-device connectivity hubs in section 2.5, and methods of inter-chip communication in section 2.6. Lastly, it explains voltage and current measurement techniques for USB devices in section 2.7.

2.1 History of USB

The Universal Serial Bus (USB) technology was developed in the mid-1990s as a standardized way to connect peripheral devices to computers. It was a collaborative effort by several tech companies, including IBM, Intel, and Microsoft (see Figure 2.1), to replace the existing serial and parallel ports with a simpler and faster interface [6]. The collaborating companies are the founders of the USB Implementers Forum (USB-IF), an organization to maintain USB.

The first USB specification (USB 1.0) was introduced in 1996, followed by USB 1.1 in 1998 [7]. However, the adoption of USB was initially slow until Apple its iMac in 1998, which used only USB ports, driving other manufacturers to adopt the standard.
USB 2.0, released in 2000, brought a significant speed increase, with data transfer rates up to 480 Mbps [8]. This paved the way for the widespread adoption of USB flash drives, which were first introduced in 1999 by companies like M-Systems, Phison Electronics, and Trek 2000 International [9].

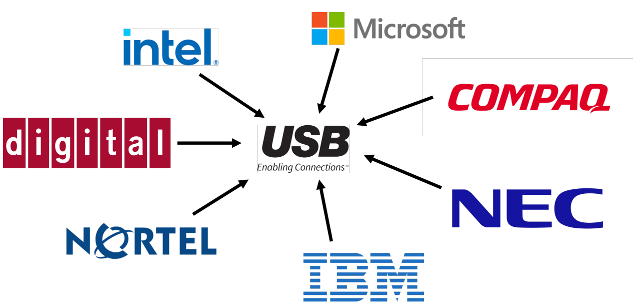


Figure 2.1: The founding companies of USB-IF

Further improvements came with USB 3.0 in 2008, offering transfer speeds up to 5 Gbps, and USB 3.1 in 2013, with speeds up to 10 Gbps. The latest USB4 specification, released in 2019, supports data transfer rates up to 40 Gbps and is based on the Thunderbolt 3 protocol.

Over the years, USB has become the standard interface for connecting a wide range of devices to computers, including printers, scanners, keyboards, external storage devices, and many others, revolutionizing the way we interact with technology.

2.2 USB Connectors and Wiring

USB technology offers various types of connections, including the USB Type-A and USB Type-C ports. Both of these connectors are capable of transmitting data and providing power. However, the USB Type-C port has more advanced features than the older USB Type-A standard. It is also backwards compatible with the older standard, as shown in Figure 2.2.
The Type-C connector has a wide range of capabilities, from data transfer and display connectivity to power delivery. All of this is made possible due to the many pins available on the connector.

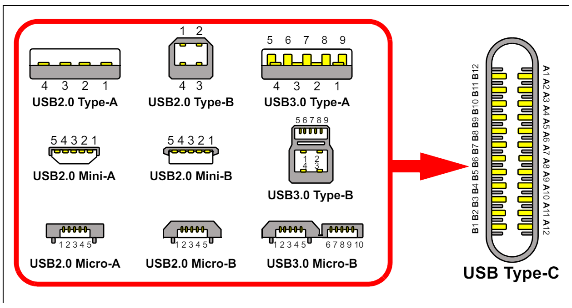


Figure 2.2: USB connector pin out [10]

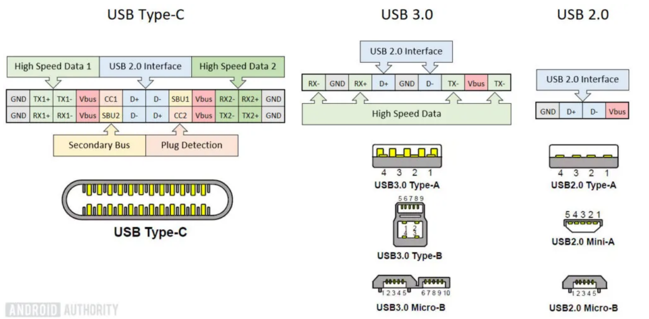


Figure 2.3: USB connection standard [11]

USB Type-C is a connector standard that is compatible with older USB types. This means that devices that use the old USB 3.0 Type-A or even the USB 2.0 Type-A connector can use the USB Type-C connector with a simple converter. This is possible because additional communication buses have been added to the standard while the old ones are still being reused, as seen in Figure 2.3.

USB charging protocols work with a handshake agreement, in this agreement, the device being charged communicates with the charger and they decide the protocol used for charging the device and the direction in which the power is flowing. The power flows through the Vbus pins.
For the USB Type-C and the PD protocol, this communication is done over the CC1 and CC2 pins seen in Figure 2.4. The pins can decide the different voltage levels and max current. The allowed voltages and currents can be seen in Figure 2.5 and will be discussed in section 2.3.
The other standards use the D- and D+ pins seen in Figure 2.4 for detecting which protocol to use, these standards will generally be referenced as Legacy 2.0. Because the D+ and D- pins are also available for the USB type-A connector, as seen in Figure 2.3, these protocols also work on those older ports.

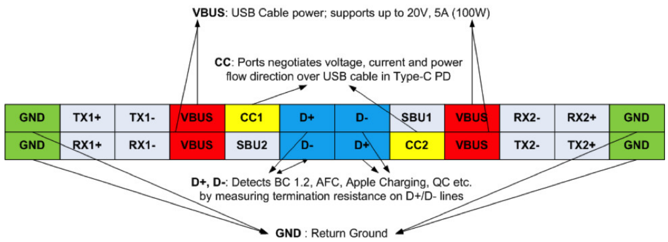


Figure 2.4: Pin description USB Type-C [12]

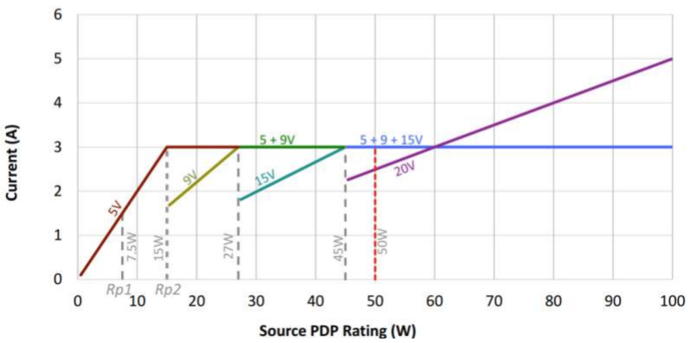


Figure 2.5: charging voltage and current [12]

USB data protocols
USB 2.0 uses two pins for data transfer: D- and D+ for data signals, supporting up to 480 Mbps. USB 3.0 and USB 3.1 add more pins for higher speeds, including TX and RX pairs for full-duplex communication, achieving up to 5 Gbps and 10 Gbps, respectively as can be seen in Table 2.2. USB Type-C features 24 pins, accommodating USB 2.0 and 3.x protocols, power delivery, and alternate modes, with reversible connectors for enhanced versatility. The additional pins in USB 3.x and Type-C allow simultaneous data transmission and reception, significantly increasing data transfer rates.

2.3 USB Charging

USB charging uses different voltage currents to provide the power needed. The charging protocol uses standard Vbus connections where the power will be transmitted.

The USB protocol supports various charging protocols which are utilized by different devices. These protocols can be split into 2 groups, public and private protocols.
The USB port can have a source and sink functionality, where the source provides power to the connected device, while the sink receives power to charge its battery or operate.

Public

USB-IF, the organization that maintains USB, has created a public standard for power delivery, which is widely used in products. Qualcomm, MediaTek, and Samsung (all mobile device CPU designers) have created their own public standard for high-speed charging.
The most used public protocols are:

- USB Power Delivery (USB PD),Table 2.1 Section: USB PD
- Qualcomm Quick Charge, Table 2.1 Section: QC
- MediaTek Pump Express
- Adaptive Fast Charging (AFC), Table 2.1 Section: AFC

Private

Different phone-making companies also deliver their own charging protocol. This is mainly done to get higher charging speeds. The main private protocols are:

- Apple Fast Charge
- Huawei's Fast Charge Protocol (FCP) and Super Charge Protocol
- VIVO Flash Charge
- Xiaomi Mi Turbo Charge
- Motorola Turbo power
- OnePlus Warp Charge and Oppo VOOC
- Realme DART

Table 2.1: USB and Fast Charging Specifications

Generation / Factor	Voltage [V]	Current [A]	Max Power [W]
USB Evolution, USB PD			
USB 1.0	5	0.5	2.5
USB 2.0	5	0.5	2.5
USB 3.0	5	0.5 / 0.9	4.5
USB Battery Charging (BC) 1.2	5	1.5	7.5
USB-C Current Mode (non-PD)	5	3	15
USB 3.1/3.2 (USB-C + USB-PD)	5/9/15/20/28/36/48	0.5/0.9/1.5/3/5	240
USB4 (USB-C + USB-PD)	5/9/15/20/28/36/48	0.5/0.9/1.5/3/5	240
Quick Charge Versions, QC			
Quick Charge 1.0	5	2	10
Quick Charge 2.0	5/9/12	1.67/2	18
Quick Charge 3.0	3.6 to 20 (200mV increments)	2.5/4.6	18
Quick Charge 4.0+	5/9 (USB-PD), 3.6 to 20V (200mV increments)	3 (USB-PD), 2.5/4.6	27 (USB-PD)
Quick Charge 5.0	5/9 (USB-PD), 3.3 to 20V (200mV increments)	3/5/>5	100+
Samsung Adaptive Fast Charging, AFC			
Adaptive Fast Charging	5/9	2	18
Super Fast Charging 1.0	11	2.25	25
Super Fast Charging 2.0	10	4.5	45

2.3.1 Implementation

Most USB-enabled devices use a separate IC to manage the USB charging protocol. This IC is responsible for regulating the voltage on the V-bus and controlling the current that powers the device. These charging ICs usually have a microcontroller and can include additional features such as over-voltage protection, temperature measurement, GPIO pins, and analog-to-digital converters. These chips commonly only go up to 100W.

In Figure 2.6 the block diagram of an implementation of all those features is shown. This is specific to the IP2368 chip but can be used as a standard representation of the block diagram that includes all necessary features such as I^2C connectivity, ADC (Analog-Digital Converter) for transferring measurements, and power control by a buck-boost. For this control, there needs to be a controller present on the chip.

These chips are sometimes integrated on breakout boards, this means that the supporting circuitry is already present and the chip can directly be tested.

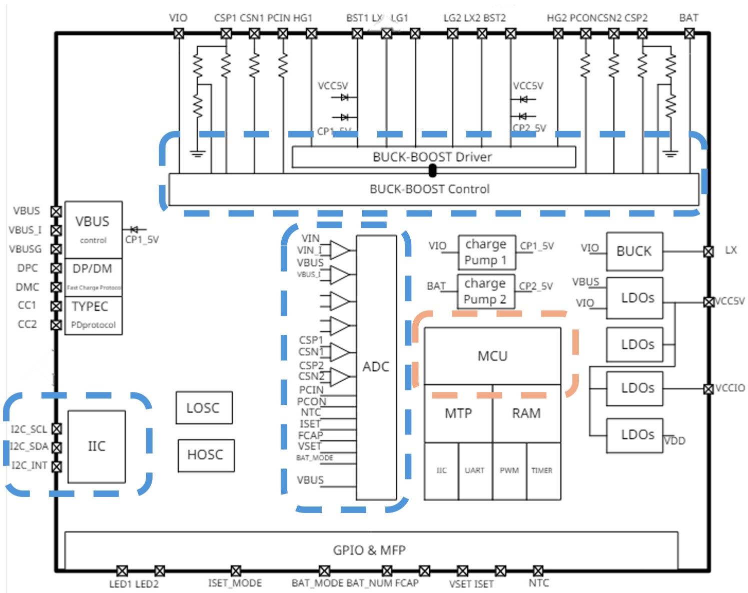


Figure 2.6: Block diagram of IP2368 [13]

2.4 USB Data

USB (Universal Serial Bus) is the most successful PC interface and can be found in almost all personal electronic devices, from computers, laptops, and telephones to cars and other devices. It connects external devices to the host device it is plugged into (as seen in Figure 2.7). These external devices could be keyboards, mice, audio/video devices, or storage options. The USB standard has evolved and is currently on USB 4 version 2.0.

As stated above, USB has multiple different versions with different data speeds for each version as shown in Table 2.2. Currently, most devices use USB 2.0, 3.0, or 3.1. USB 3.2 is still not commonly used. In addition, USB 4 is hardly applied to any devices yet.

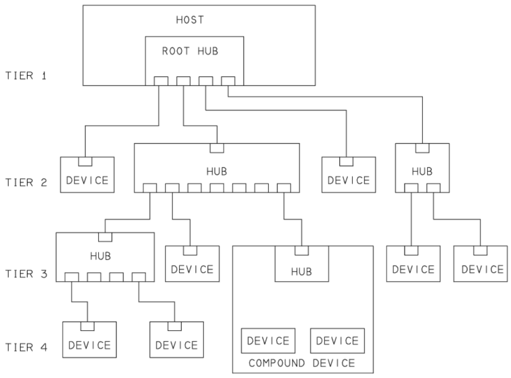


Figure 2.7: Example USB Topology [14]

Table 2.2: Speed of USB standards [15]

Standard	Max. data transfer speed
USB 1.0	1.5 Mbps
USB 1.1	12 Mbps
USB 2.0	480 Mbps
USB 3.0	5 Gbps
USB 3.1	10 Gbps
USB 3.2	20 Gbps
USB 4	40 Gbps
USB 4 version 2.0	80 Gbps

2.5 USB Multi-device Connectivity Hub

The data transfer should be possible on multiple USB ports. The most obvious solution would be to implement the data transfer as a **USB hub** as in Figure 2.8. This is a common application to link several data ports. the hub has a host system, like a laptop, connected to it that can communicate to the devices that are connected to the hub. The multiple devices attached to the hub are all available at the same time, but they share the bandwidth of the host system.

Another option is a **USB switch** as in Figure 2.9. This has the working of a MUX. The common port can only communicate with a selected port. This has the disadvantage that not all attached devices are available at the same time, but now the ports don't have to share the bandwidth, so the transfer speed can be a lot higher for the selected port.

USB hubs have one upstream port that connects to the host, and multiple downstream ports to connect devices. When data is received from the host through the upstream port, it is broadcast to all devices connected to the downstream ports [16]. Conversely, data from a downstream device is only forwarded to the upstream port and the host, not to other downstream devices.

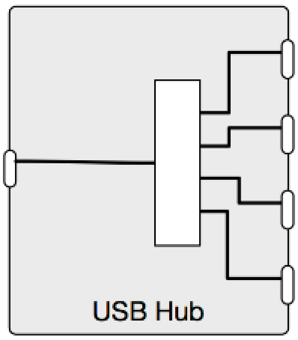


Figure 2.8: USB hub

With USB 3.0, a new Point-to-Point routing feature was introduced. This allows the host to send data directly to a specific downstream port instead of broadcasting to all ports, reducing congestion and power consumption for a USB hub.

USB hubs can be chained together, with each additional hub counting as a new "tier" in the USB topology, this can be seen in Figure 2.7. However, there is a limit of 7 tiers from the host to any device, including the hubs themselves [17]. This limitation is important to consider when designing complex USB systems.

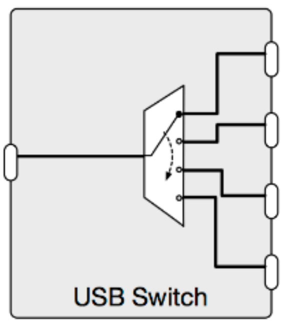


Figure 2.9: USB switch

2.5.1 Design of USB Hub

A USB hub consists mostly of one or multiple ICs. These ICs execute the functions for the USB hub. Typically, there can be 4 downstream ports with one upstream port. To implement a higher number of downstream ports, an IC with more downstream ports can be chosen, or multiple ICs can be linked together.

For a hub with up to 7 ports, an IC similar to the one shown in Figure 2.11 can be used. These ICs are a bit more unusual, but they operate in the same way as a 4-port hub. The USB5807c can be programmed via I^2C. However, this programming can only be done at startup (Figure 2.12).

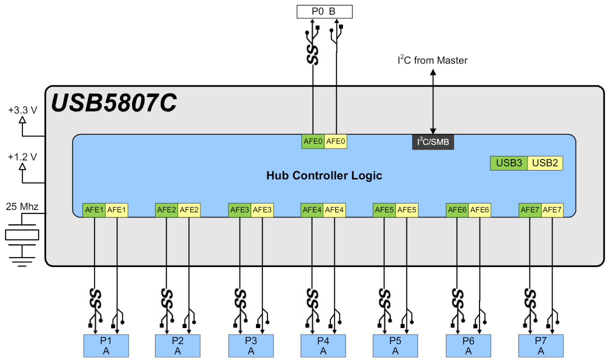
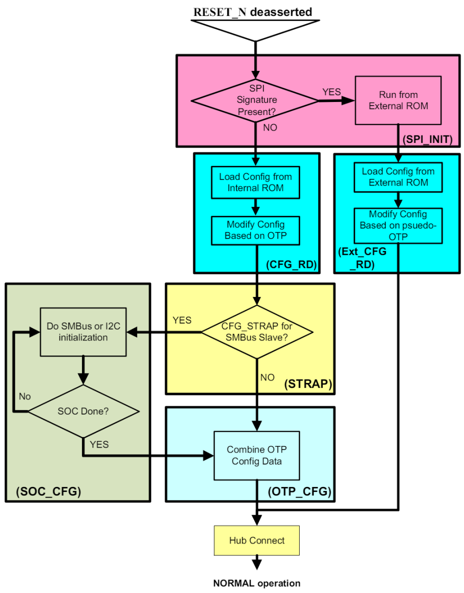


Figure 2.11: USB5807C block diagram [18]

Another way to connect multiple downstream ports to one upstream port is by linking multiple hubs. For example, This can be achieved by connecting one of the downstream ports of the first USB hub to an upstream port of the second USB hub as shown in Figure 2.10. Since the max speed of the upstream port is the same as the downstream port, this should not significantly decrease the speed of the second hub when the first hub is not already saturating the data upstream port.



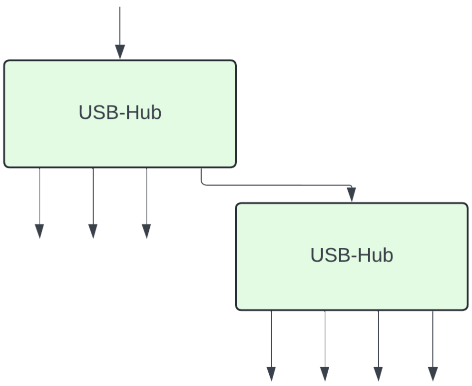


Figure 2.10: USB-Hub 7 port from 2 4-ports

Figure 2.12: Boot flowchart USB5807C [18]

The linking of multiple hubs is a common practice as can be seen with the teardown featured on elektroda.com [19], the inner workings of a TP-link USB 3.0 hub were examined. The teardown, seen in Figure 2.13, revealed the presence of two identical ICs, namely the RTS5411 [20], arranged as shown in Figure 2.10. This chip has the capability to switch the data connection on & off. Additionally, the *red-7 ports* from San Zang master, already owned by one of the group members, was found to be constructed in a similar manner as can be seen in Figure 2.14 where the top and bottom sides of the PCB are shown. This product utilized the GL3510 [21] for the USB-Hub IC. This is a chip similar to the one used in Figure 2.13 but it does not have the option to be configured via I^2C.

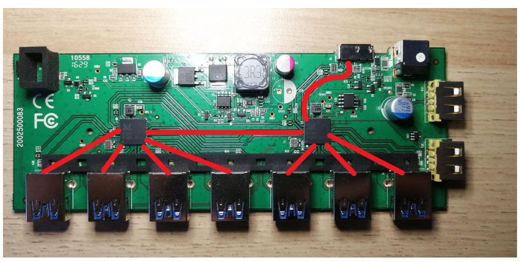


Figure 2.13: Tear down of TP-link USB 3.0 hub [19]

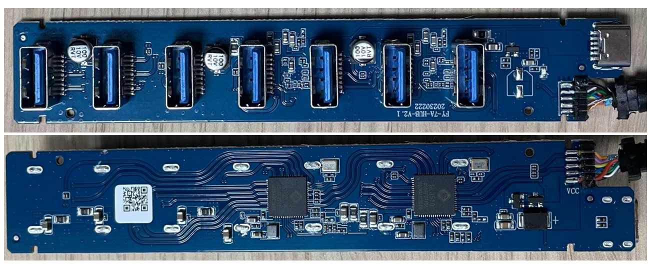


Figure 2.14: Tear down of red-7 port from San Zang master

2.6 Inter-chip Communication

Inter-chip communication refers to the methods used for data exchange between integrated circuits (ICs) within electronic systems. These methods are crucial in embedded systems, microcontrollers, and other digital electronics. Here we cover three widely used protocols: SPI (Serial Peripheral Interface), UART (Universal Asynchronous Receiver/Transmitter), and I^2C (Inter-Integrated Circuit).

2.6.1 Serial Peripheral Interface (SPI)

SPI is a synchronous serial communication protocol used primarily for short-distance communication, typically within the same device.

Key Features

- **Master-Slave Architecture:** One master device controls one or more slave devices. (Figure 2.15)
- **Full-Duplex Communication:** Data can be sent and received simultaneously.
- **Clock Signal (SCK):** The master generates a clock signal to synchronize data transmission.
- **Data Lines:**
  - MOSI (Master Out Slave In)
  - MISO (Master In Slave Out)
- **Chip Select (CS):** Used by the master to select individual slave devices.
- **Speed:** Generally faster than I^2C and UART.

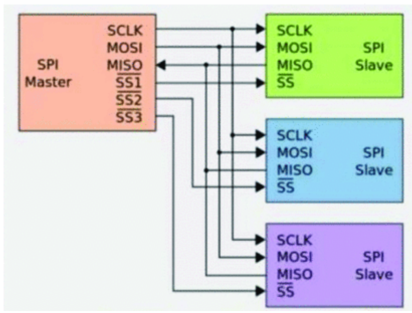


Figure 2.15: SPI working diagram [22]

Use Cases

- Sensors
- SD cards
- LCD displays

Advantages

- High speed
- Simple hardware implementation

Disadvantages

- More pins required compared to I^2C
- No error checking mechanism

2.6.2 Universal Asynchronous Receiver/Transmitter (UART)

UART is an asynchronous serial communication protocol widely used for communication between computers and peripherals.

Key Features

- **Asynchronous Communication:** No clock signal; both devices must agree on the baud rate.
- **Data Format:** Typically 8 data bits, 1 start bit, 1 stop bit, and no parity.
- **Full-Duplex Communication:** Data can be transmitted and received simultaneously. (Figure 2.16)
- **Baud Rate:** Must be set to the same value on both communicating devices.

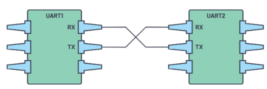


Figure 2.16: UART wiring [23]

Use Cases

- Serial ports
- GPS modules
- Bluetooth modules

Advantages

- Simple implementation
- No need for a clock signal

Disadvantages

- Slower compared to SPI and I^2C
- Requires precise timing

2.6.3 Inter-Integrated Circuit (I^2C)

I^2C is a synchronous, multi-master, multi-slave, packet-switched, single-ended, serial communication bus.

Key Features

- **Multi-Master/Slave Architecture:** Multiple masters and slaves can be connected on the same bus. (Figure 2.17)

- **Two-Wire Interface:**
  - SDA (Serial Data Line)
  - SCL (Serial Clock Line)
- **Addressing:** Each device on the bus has a unique address.
- **Speed Modes:**
  - Standard (100 kbps)
  - Fast (400 kbps)
  - High-speed (3.4 Mbps)

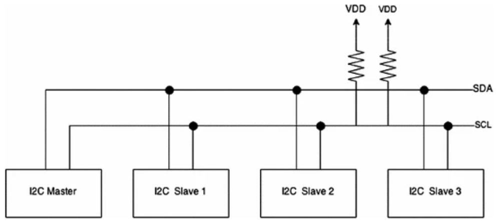


Figure 2.17: I^2C communication diagram [24]

Use Cases

- Sensors
- EEPROMs
- Real-Time Clocks

Advantages

- Only two wires required
- Supports multiple devices on the same bus

Disadvantages

- Slower than SPI
- More complex protocol

Protocol Message

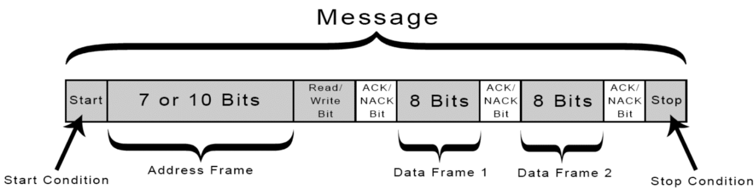


Figure 2.18: I^2C package [25]

The I^2C package is structured as seen in Figure 2.18. This figure shows a writing operation. First, the master sends the 7-bit address of the slave it wants to write to, followed by the read or write bit to form the standard 8-bit package (one byte). When the message is received, the slave transmits an acknowledgement bit to let the master know that it is listening to the following data. The data is sent in the same structure, a byte followed by an acknowledgement sent by the slave.

2.7 Voltage and Current Measurements

In some applications, the voltages and current going through the USB Type-C port need to be measured externally from the IC responsible for the power delivery protocol. This can be done with op-amp circuits or with external independent sensors.
For measuring the voltage which the USB Type-C port is delivering, a simple voltage divider with an analog-to-digital converter can be used. If more stability or a lower current usage is needed, a voltage follower can be added to create a stable readable voltage for the analog-to-digital converter (Figure 2.19).

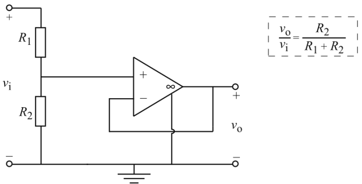


Figure 2.19: Voltage follower [26]

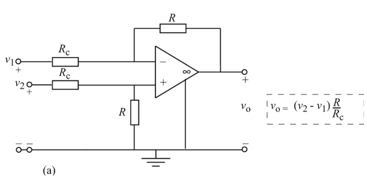


Figure 2.20: Differential voltage amplifier [26]

Figure 2.21: shunt resistor [27]

For the current measurement, a differential amplifier (Figure 2.20) and a shunt resister (Figure 2.21) can be used. This is a small but accurate register where the voltages over the resistor would be small but linear to the current flowing through the resistor. This small voltage can be amplified with the differential amplifier to a voltage level that an analogue-to-digital converter can measure.

Chapter 3

Design Description

In this chapter, the design choices made from the requirements will be described. First, it outlines the system requirements that roll out of the high-level requirements in section 3.1, followed by the subsystem specifications in section 3.2, including the USB implementation and USB hub sub-modules. It discusses the power delivery system options in section 3.3 and details the USB data hub sub-module implementation in section 3.4. Finally, it explains how these sub-modules are integrated into the overall system in section 3.5.

3.1 System Requirements

To get the USB part of the product right, it's important to figure out some specific requirements based on the overall project goals. These USB-specific requirements will help ensure that everything works smoothly together.
Here are the USB requirements derived from the main product requirements that are stated in Table 1.1:

Table 3.1: System Requirements

USB Ports and Protocols
The product must have six USB Type-C charging ports.
The product must support up to 100W of charging.
The product should support the modern charging standards.
Power Measurement and Control
The product must be able to measure current, voltage, and power per USB Type-C port.
The product must have the ability to control the power delivered through each USB Type-C port, where the control itself is done by the control-subgroup, to ensure: – The total power does not exceed 300W. – Each port does not exceed 100W.
USB Data Hub Capabilities
The product should be usable as a USB data hub.
The product should support the highest speed for USB data transfer.
The product should allow the data connection of an individual port to be turned off.

3.2 Subsystem Specifications and Requirements

Based on the system requirements, the system can be divided into 2 subsystems: the USB system and the Data system. There is a need for six USB Type-C ports, all with the same functions, including charging with specific protocols, measuring voltage and current, controlling power, and establishing a data connection. Considering that the USB Type-C ports can be operating at different charging voltages, it is logical to create six individual identical **USB implementation sub-modules**.

Another significant task not yet covered in the USB implementation sub-module is the USB data hub. This will be its own **USB hub sub-module** and could connect the USB data connection and the controller data connection to an external PC.

These subsystems have their own subsystem requirements that are extracted from the system requirements or from the way the system is divided.

3.2.1 USB Implementation Sub-module

The **USB implementation sub-module** plays a crucial role in controlling the functionalities of USB ports directly. These ports are responsible for charging devices according to specified standards. For instance, the USB PD protocol involves a handshake mechanism to determine the voltage and current the connected device requires. Typically, this handshake is facilitated by a specific IC. Additionally, USB Type-C charging involves multiple voltage levels, necessitating a DC-DC converter integrated into the charging IC or as a separate component—to generate these voltages, depending on the power source's output.

Another aspect of the sub-module involves measuring the electrical characteristics of the USB Type-C charging port. While some charging ICs handle this measurement internally, in certain scenarios, an external measuring unit may be required to gauge the outgoing power because the IC cannot export that information.

Lastly, the sub-module must facilitate the passage of USB data to the **USB hub sub-module**. Given that some data pins serve dual functions in certain charging protocols, a data multiplexer must be incorporated into the design. This allows the pins to be utilized for both the charging handshake and USB data transmission.

Table 3.2: High-level requirements of the new list

Mandatory requirements or constraints
The product must perform the communication for the charging protocols
The product must provide the voltages and current needed for the charging protocols
The product must measure the voltages and currents used while charging
The product must be controllable via an external controller
Trade-off requirements or objectives
The product should connect to the USB hub for data conductivity
The product should be at the highest USB speed (USB 4.0)

The structure depicted in Figure 3.1 follows from the subsystem requirements. In this design, the IC responsible for the USB charging handshake is positioned in the middle, flanked by the DC-DC converter and the measuring system on the left. The data MUX is situated on the right side.

In practice, the implementation of this sub-module can vary significantly. The chosen IC may perform a subset of the functions mentioned in the USB implementation sub-module requirements, while the remaining functions may need to be handled by an external circuit or IC.

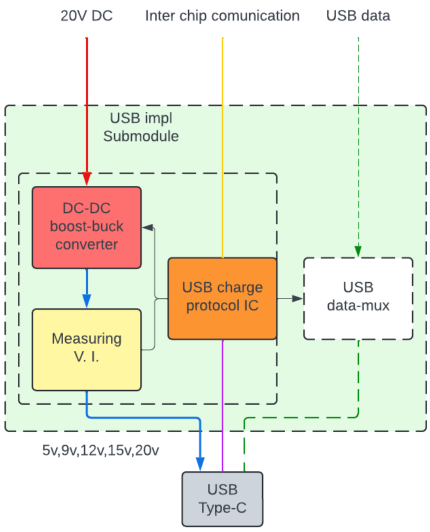


Figure 3.1: USB implementation sub module

3.2.2 USB Hub Sub-module

The USB hub sub-module serves as a central hub for managing USB data transmission within the system. It connects multiple data lines from components such as the USB implementation sub-module and the controller to a single main data line. This main data line is called the upstream port, while the connected lines are termed downstream ports.

This configuration is essential due to the USB protocol's structure, where there is one master device (the upstream port) that communicates with other devices downstream.

For safety reasons, such as preventing malicious activities, the USB ports should be controllable by the controller, allowing them to be turned on and off as needed. Like preventing a Rubber Ducky attack from succeeding, a cyberattack where a USB device emulates a USB keyboard to compromise a workstation.

As the whole USB Data feature is for the plus version, all requirements of this subsystem are defined as trade-off requirements. The whole implementation is shown in Figure 3.2.

Table 3.3: Subsystem Requirements

Subsystem Requirements
The sub-module should connect the different USB implementation sub-module with the external PC
The sub-module should control the data connection of the USB ports
The sub-module should be controlled by the controller
The sub-module should connect the controller to the external PC
The sub-module should be at the highest USB speed

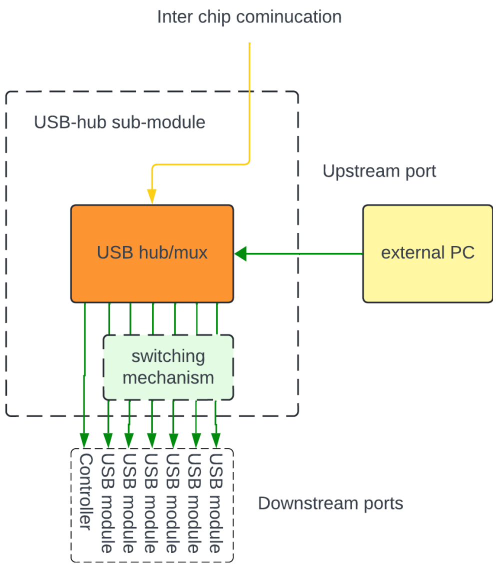


Figure 3.2: USB hub sub-module

3.2.3 Sub-module Integration

The sub-modules defined above must work together to create a total system that meets all system requirements. As mentioned, the system needs six USB Type-C ports, each with a **USB implementation sub-module** connected to it. Each sub-module needs power, a communication connection with the controller, and a connection to a USB Type-C port. These sub-modules also have an outgoing data connection that can connect to the **USB hub sub-module**. Additionally, the **USB hub sub-module** must communicate with the controller and have an external PC as an upstream port. For the controller, the control-subgroup has chosen to use an MCU which is why this choice is applied from now on. Based on these requirements, the design shown in Figure 3.3 was created.

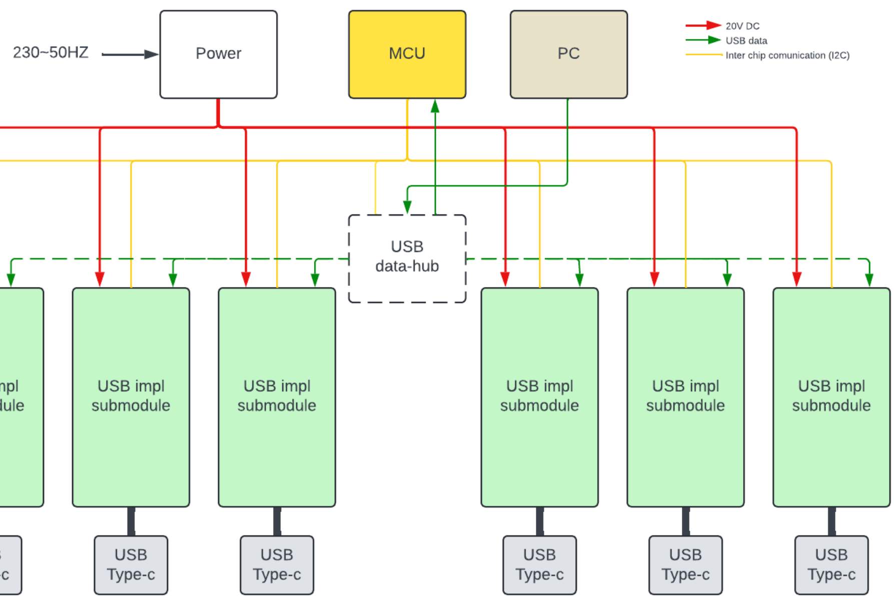


Figure 3.3: Overview USB implementation

The design in Figure 3.3 only shows the connections and modules relevant to the USB integration system. This means that the structure to power the MCU and the USB hub sub-module is missing, as well as the touchscreen and other HMI components. These structures are explained in the safety implementation and control implementation respectively.

3.3 Power Delivery System Choice

This subsection will describe the chosen methodology of the chip used in the design. First, 3 different sorts of systems that can be used to integrate the USB PD functionality will be discussed, and then they will be compared with the system requirements needed for the power delivery.

3.3.1 Power Delivery Design Options

When designing power delivery systems there are several options to consider. These options vary in complexity, integration, and customization potential, each suited to different stages of development and specific design requirements. The three primary choices are development boards, PD modules, and stand-alone chips. Each offers distinct advantages and challenges, from ease of testing and evaluation to the flexibility of final design integration. The different options are explained in detail by highlighting their unique features and considerations for creating a power delivery solution.

- **Development board**
  A development board is created by the manufacturer and can be used to test the chip's functionality. These boards are thus only created for testing and evaluating the chip's working. And can have a larger supporting circuitry surrounding the chip than would be needed for a final design. This is so all the features can be tested, not only the ones needed in the final design. This can create difficulties in converting the development board to a final compact design. The development board has a good connection interface to external MCU's.
- **PD module**
  This is a stand-alone module with the circuitry needed to provide the communication and the power needed for the USB type c sharing standards. They work out of the box and can thus be directly used for powering. Depending on the chip used in the design, the IC is a DC-DC converter or uses a stand-alone chip for the DC-DC conversion. The modules are not made to be connected to an external MCU and thus lack ready-to-use connections. But can have the needed inter-chip communication built into the chip. This makes the module function like a development board.
- **Stand-alone chip**
  A stand-alone chip can be used to create the USB charging module. Such a chip needs supporting circuitry to function. But can be customized to the user's specifications. The biggest drawback is that the function of the chip can only be determined after being sure that the supporting circuitry works as intended. Due to its admissibility and the freedom to choose any chip on the mark, the system can be designed to communicate with an MCU in mind.

3.3.2 Comparing Different Power Delivery Designs

In this subsection, the selection and implementation of the power system sub-module will be explained. First, the evaluation criteria will be presented, followed by the selection of a suitable system. The chosen system will then be thoroughly explained, including the reasons for its selection and how it can be used in our design.

A list of evaluation criteria was created (as listed in Table 3.4) based on the system requirements stated in Table 3.1. The criteria cover the functionality of the module for the final integration, as well as criteria for ease of production and testability. These aspects were particularly important before the scope change of the project. Due to the testability and production requirements, an ease of conversion on PCB requirement was added. This meant that the circuitry should be simple, consisting of a small number of components, and testable in a form that would represent what would be needed on the final PCB.

Table 3.4: evaluation criteria

- Charge capability
- Measure V, I, and P
- Dynamically change power
- Communicate with MCU
- Directly testable
- Ease of conversion on PCB
  - simple supporting circuitry
  - small number of components
  - testable in form that represents circuitry on final PCB

Table 3.5: Comparison of different power systems: Development boards, PD modules and Stand-alone chips

Criteria	Development board		PD modules		Stand-alone chip	
	MAX25430 [28]	TPS65982 [29]	IP2368 [13]	SW2303 [30]	MAX25430	TPS65982
Charge capability	✓	✓	✓	✓	✓	✓
Measure V, I and P	✓	✓	✓	×	✓	✓
Dynamically change power	✓	×	✓	✓	✓	×
Communicate with MCU	✓	✓	✓	✓	✓	✓
Directly testable	✓	×	✓	✓	×	×
Ease of conversion on PCB	×	×	✓	✓	×	×

3.3.3 IP2368 PD Module

From these criteria, the IP2368 PD-module was chosen as the core of the power system. This is a stand-alone bidirectional PD module typically used for a power bank implementation. The module is frequently used and there seemed to be support for connecting the module with an MCU via an Arduino library [13]. The module was chosen for its control and measurement features and its simple supporting circuit. In the data sheet of the main IC of the module, a typical use circuit (Figure 3.5) could be found to match the PD module.

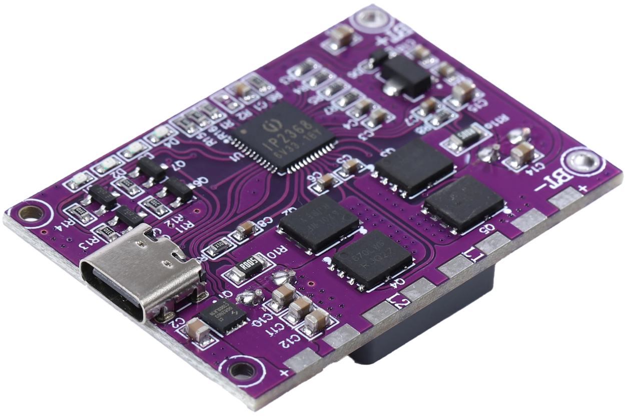


Figure 3.4: IP2368 module [13]

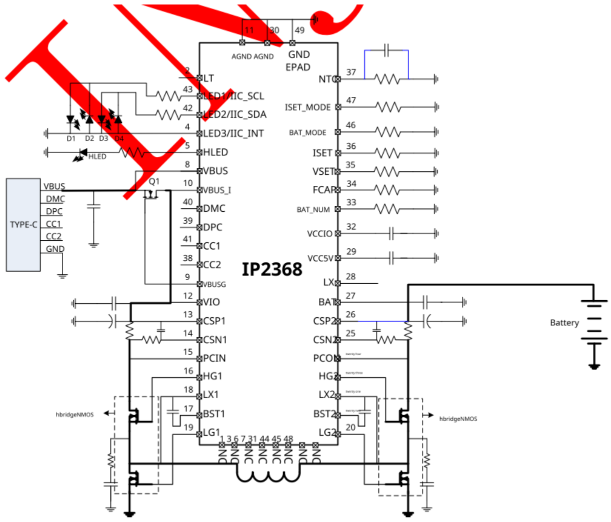


Figure 3.5: Typical implementation IP2368 Figure 3.4

From the datasheet of the IP2368 IC (ENG translation found in [13]), the following features are of interest for implementing the module as part of the **USB implementation sub-module**. These features include charging features (protocols and powers), measuring options (current, voltages and power), and the connection possibility. The features are shown in Table 3.6.

Table 3.6: Feature list of IP2368 [13]

Feature
Charging with up to 100W
Charge with the protocols: - AFC - FCP - SCP - PD2.0, PD3.0
Measuring outgoing voltages
Measuring outgoing and incoming currents
Controlling the power direction (only outgoing)
Setting the max power usage
I^2C interface for control

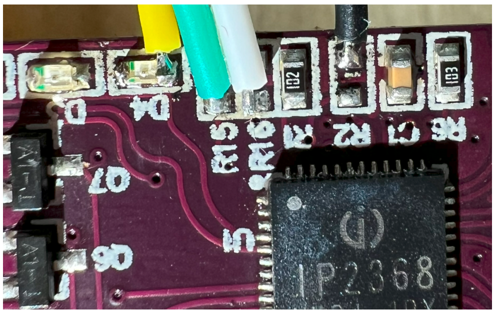


Figure 3.6: IP2368 module I^2C connection [13]

The datasheet indicates that the chip can be connected to an external MCU. This can be achieved as illustrated in Figure 3.7 and Figure 3.6. In this setup, the chip is connected via the standard SDA and SCL pins used for I^2C communication, along with a third pin for waking the IC when it is in sleep mode. The protocol mentioned in the datasheet is also available in a library specifically designed for the PD-Model. This Arduino library is created by Dmitriy Mitchenkov [31]. It should enable reading and controlling the IC. This library could be directly used in the project or for testing purposes. Apart from the library, the GitHub page also provides information on how to connect the module for this communication and translated datasheets.

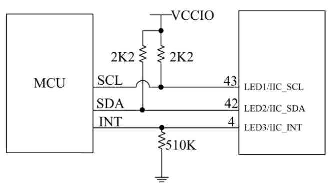


Figure 3.7: IP2368 I^2C wiring [13]

The IP2368 I^2C control functions as a register bank, where the registers can be both read and written. If a register needs to be changed, it must be read first. After that, the relevant part of the received byte can be modified and then written back into the register. In the context of the "USB implementation sub-module" function, there are several registers of interest which can control the power delivery functionality and take measurements of the outgoing power. In Table 3.7, these registers are listed, with their description and the reason of interest for the USB implementation. As seen in the table, the IP2368 can be configured as a source-only device. And the max outgoing power can be used to a number of specific values this would make dynamic power allocation possible by the MCU. The registers also make it possible to measure the outgoing power, voltage and current.

Table 3.7: Registers of interest for controlling and measuring charging port [13]

Register name	Description	Reason of interest
SYS_CTL0	Controls the sink functionality	Disable Sink functionality
SYS_CTL11	Controls the source functionality	Disable output for full shut off
SYS_CTL12	Controls the maximum power	Set max power 20-25-30-45-60-100 watt
TypeC_CTL17	Controls the Power Delivery (PD) functionality	Disable specific PD voltages if needed
BATVADC_DAT0 BATVADC_DAT1	Contains data for the input voltage measurement.	Measure input voltage (mV)
IVbus-Src-IADC-DAT0 IVbus-Src-IADC-DAT1	Contains data for the output current measurement	Measure output current (mA)
TypeC_STATE	Indicates the USB C charging state	Indicates if output is on
Vsys-POW-DAT0 Vsys-POW-DAT1 Vsys-POW-DAT2	Contains data for the output power measurement in milliwatts	Measuring output power (mW)

The functionality of the module was tested, and this will be covered in (section 4.2). This testing clarified that the IP2368 did not work as expected, so another solution must be chosen.

3.3.4 SW2303 PD Module

Due to the malfunction of plate number 1, as described in subsection 3.3.3 and explained in section 4.2, we also tested plate number 2 PD module. This module was already in possession of one of the group members, and the datasheet of the main IC on the module promised some desired features, which will be further explained in this section.

The SW2303 PD module, seen in Figure 3.8, is a power delivery module which is able to to charge with up to 100w at 20v. And should be able to control the power dynamic over I^2C. The interesting features of the module can be found in Table 3.8.

Table 3.8: Feature list of SW2303 [30]

Feature
Charging with up to 100W
Charge with the protocols: - AFC - FCP - SCP - PD3.0 - BC1.2
Controlling the power direction (only outgoing)
Setting the max power usage
I^2C interface for controlling outgoing power

Figure 3.8: SW2303 module

In the Table 3.8, it is evident that the module does not have the capability to measure outgoing power. However, because it can control outgoing power, it should be able to measure outgoing voltage and current. There may be a register, not mentioned in the datasheet, that stores this measurement. If this is not the case, but the other features do work, an external measuring circuit could be created (as explained in section 2.7).

The datasheet only mentions one register (seen in Table 3.9), which should be able to control the maximum outgoing power. To set the IC to dynamic power configuration with I^2C, the *PSET* pin should not be connected. Additionally, the *SCK* and *SDA* pins need to be connected to a pull-up resistor and tho the MCU (*SCK*-*SCL* and *SDA*-*SDA*). Pin out can be seen in Figure 3.9.

Table 3.9: Register of interest for controlling charging port [30]

Register address	Description	Reason of interest
0x3C	Dynamic power allocation	Set max power functionality

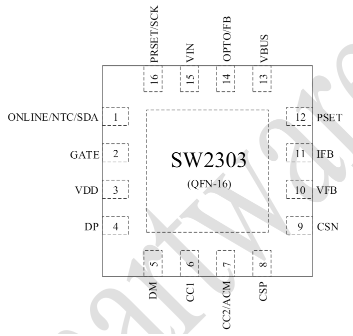


Figure 3.9: SW2303 pin out [30]

3.3.5 MAX25430 Development Board

Another option for the power system is the MAX25430 development board, seen in Figure 3.10; this board is created for evaluating the function of the MAX25430. Which is a chip capable of 100w PD charging with a built-in DC-DC converter.
It is a ready-to-test module, but due to it manny featuars hase a high number of sporting components placed around the core chip on the develement board. This is one of the reasons it was not tested as first as the project aimed at creating a final PCB at the end of the project.

The Dev board was chosen for the following features stated on it data sheet:

Table 3.10: Feature list

Feature
Charging up to 100W
Charging with protocol: - USB 2.0 legacy protocol incl: • BC1.2, Apple Carplay MFiR33, and OTG - USB PD2.0 and PD3.0
Measure the charging voltage
Control the outgoing power
Be controllable over I^2C

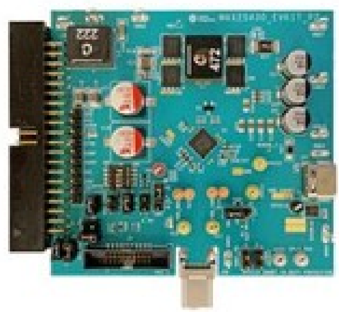


Figure 3.10: MAX25430 development board

The MAX25430 can be connected to an MCU via an I^2C bus using the standard SCL and SDA pins, as well as an alert pin (Figure 3.11). The alert pin is used to wake up the chip when it is in sleep mode.

The datasheet provides a comprehensive list of the registers available with the I^2C communication. The registers relevant to the USB implementation sub-module are presented in Table 3.11. Access to these registers allows the IC to monitor and regulate the outgoing charging capabilities. Testing of this functionality took place at the location referenced by (section 4.4), using a specific board. This board was equipped with an IC marked as MAX25430A, which has limited features compared to the capabilities of MAX25430. Consequently, it is not suitable for the USB implementation sub-module.

Table 3.11: Register of interest for controlling charging port [30]

Register address	Description	Reason of interest
0x1C	power_control	Turn off and on the output power
0x25	Device_capabilities_1	Set max power output
0x26	Device_capabilities_2	Set max power output
0x70	vbus_voltage_L	The measured voltage
0x71	vbus_voltage_H	
0x80	power output control	Set max voltage and current

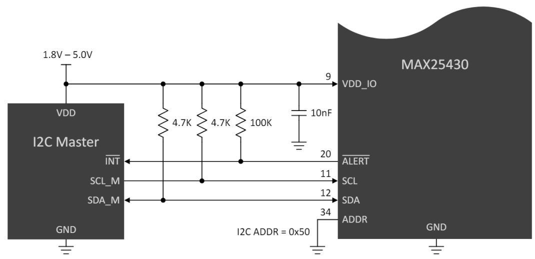


Figure 3.11: I^2C MAX25430

3.3.6 MAX25432B Development Board

As MAX25430A did not live up to the expectations set by the datasheet of MAX25430, MAX25432B came into the picture (Figure 3.12). This development board was not in stock when deciding to work with MAX25430A, which means that MAX25432B only came to light when MAX25430A did not meet the expectations.
MAX25432 is a comparable IC to the MAX25430 and shares most features, such as charging capability and the overall structure of the I^2C registers. The main difference is that both the A and B versions of the MAX25432 have an ADC readable with the I^2C connection and support full charging port shutdown as well as current limiting features.

The interesting core features:

Table 3.12: Feature list

Feature
Charging up to 100W
Charging with protocol: - USB 2.0 legacy protocol incl: • BC1.2, Apple Carplay MFiR33, and OTG - USB PD2.0 and PD3.0
Measure the charging voltage and current
Control the outgoing power
Be controllable over I^2C

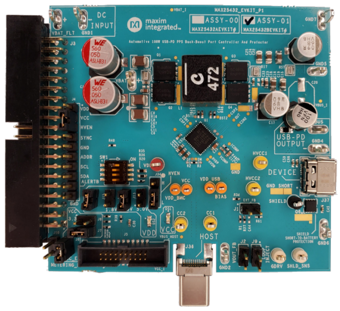


Figure 3.12: MAX25432B development board

Table 3.13: Interesting registers MAX25432B

Register address	Description	Reason of interest
0x1C	power_control	Turn off and on the output power
0x70 0x71	vbus_voltage_L vbus_voltage_H	The measured voltage
0x80	Vbus_current	The outgoing current
0x82	vbus_ilim_setup	Set the outgoing current limit

3.4 USB Data Hub Sub-module Implementation

Due to all of the features of this sub-module being characteristic of the features, this module will be developed so that the designed feature could be implemented. However, due to the time constraints of the project, the making of a custom PCB is unlikely to be possible, so an existing USB hub is implemented to work together with the USB implementation sub-module.

In order to meet the requirements of the USB hub sub-module, it is necessary to connect different USB implementation models to the upstream port that is linked to the computer. According to the literature review in section 2.5, there are two options for connecting multiple devices to one main device. A USB switch allows connection with one device at a time and can switch between different devices, while a USB hub can support communication with multiple devices simultaneously. Considering the requirements of the subsystems where multiple ports need to function at the same time, it is not feasible to use a switch. And thus the use of a USB hub is necessary for the design.

One of the requirements is to use the highest USB data speed when connecting the USB hub. Upon reviewing the available USB hub ICs, the USB version 4 hubs have a high cost and are complex to integrate. Additionally, hardly any device requires USB version 4. Hubs with USB 3.2 would also be unnecessary, since the data speed is quite high for this product's intended use, as the product will likely be used to connect phones and other personal devices that typically use USB 2.0 or 3.0. Therefore, for this implementation, USB 3.0 will be used. This decision is based on the availability of hardware for USB 3.0, and the same rationale applies to implementing a hub with USB 3.2.

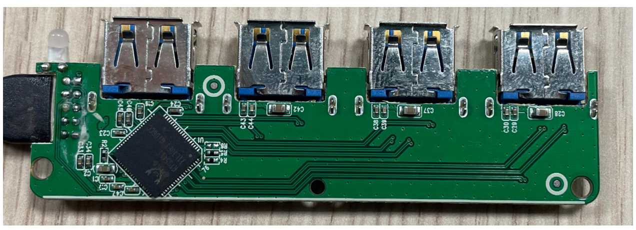


Figure 3.13: USB 3.0 hub from Anker

The final crucial requirement is that the USB ports must be able to be disconnected to convert a USB port to a charging port only. As mentioned in the technical review, this is partially achievable. Typically, the USB hub can only be configured at startup, so at that point, the decision has to be made as to which ports need to be active for data transfer and which ones do not.

For simplicity, the six USB ports will all be downstream ports. The upstream port will be a 7th port that will connect to an external PC. To transfer the logged data from the MCU to the external PC, an extra downstream connection is needed. So for a complete representation of the USB hub, one upstream connection and seven downstream connections are required.

A USB hub that was already in possession was torn down to inspect its working, this hub is shown in Figure 3.13. This teardown revealed that this hub used the same chip as the shown hub in Figure 2.13 from the Technical Review, the RTS5411 to be specific. This chip can turn on and off the data ports and be used with multiple chips to create a hub with more downstream connections. When accounting for cost it would also be better to take two smaller chips and link them, since a chip with more downstream ports is way more expensive.

So for showing the working of the USB data hub sub-module, the hub from Figure 3.13 can be used. Since it is in line with the requirements of the product. It realises data transfer with USB 3.0, it can switch data ports on and off, and can be used with a second hub to create a bigger hub.

3.5 System Integration

For the integration of the whole system, the choices made for the USB implementation sub-module and USB data hub sub-module, are combined in one system where it connects to the other subgroup designs seamlessly as shown in Figure 3.3.

When combining the six USB implementation sub-modules, errors may occur when trying to connect them to the MCU. This could be because the sub-module ICs have the same address. One way to solve this issue is by implementing a MUX in between, which ensures inter-chip communication between a port and the MCU. The used PD modules all had the same address and would require at least a 1-to-6 MUX (seen in Figure 3.14), while the used development boards had 4 different addresses, so a 1-to-2 MUX would be sufficient. For ease of conversion into a final product, the system can be designed to have 2 wings with each wing having 3 USB implementation sub-modules (as represented in Figure 3.15). Then each wing can be called from the 1-to-2 MUX.

The USB data hub sub-module will implement two identical chips to get the required amount of downstream ports with chips that only support 4 downstream ports. Each of those two chips can be integrated into one of the earlier-mentioned wings to avoid issues in inter-chip communication when the two data hub chips have the same address. The inter-chip communication is present for controlling the switching mechanism in the data hub sub-module that was mentioned in subsection 3.2.2.

The two wings are only connected via the data chips since the upstream of one of those chips is connected to a downstream of the other data hub chip. Furthermore, the two wings are both connected to the MCU with the 1-to-2 MUX for the inter-chip communication and the MCU is connected to one of the downstream ports of the data hub chips. The PC will be connected to the only available upstream connection since the other upstream is connected to a downstream of the chip to which the PC connects. Each wing is also connected to the power supply to get the power for the output of the USB ports.

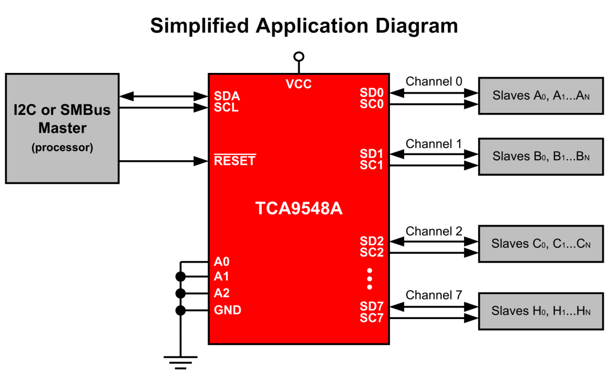


Figure 3.14: Simplified diagram TCA9548A I^2C mux [32]

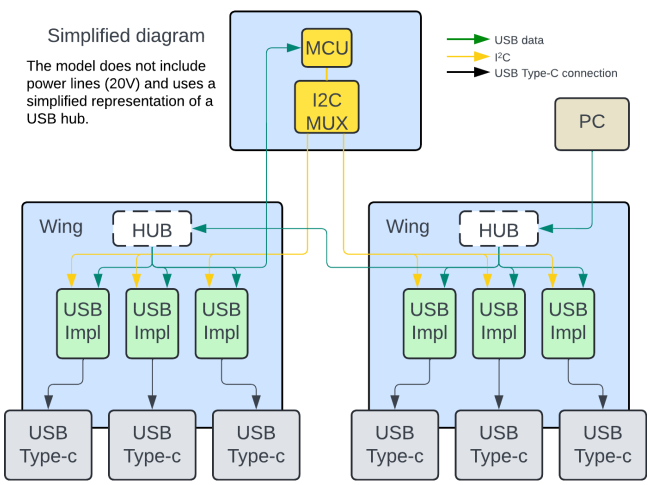


Figure 3.15: Simplified implementation diagram

Chapter 4

Validation and Evaluation

This chapter details the testing and evaluation of the system. It begins with the USB implementation sub-module setup in section 4.1 and the test procedure in subsection 4.1.2. The IP2368 PD Module is validated in section 4.2, followed by the SW2303 PD Module in section 4.3. The MAX25430A Development Board is reviewed in section 4.4, and the MAX25432B Development Board in section 4.5. Findings for the USB implementation are summarized in section 4.6, and the USB Hub Sub-module is discussed in section 4.7. Integration steps are in section 4.8, and next steps in section 4.9.

4.1 USB Implementation Sub-module test setup

The chosen options for the USB implementation will be tested and evaluated. Multiple solutions are tested due to not all of them working as expected. The solutions are evaluated based on the requirements set for the USB implementation, with priority given to the charging capabilities. This is the main feature of the **USB implementation sub-module**. The next important feature is measuring and controlling the output of the port. The USB pass-through capabilities will be tested if both tests yield the desired outcomes.

4.1.1 Testing Methodology

To test the charging capabilities, the solution under testing is connected to a lab power supply set to the $20V$. A USB Type-C cable connects a power trigger to the outgoing USB Type-C port. The power trigger is a module that acts as a USB Type-C sink protocol and performs the handshake with the source to determine the voltages being used. The PD power trigger, in Figure 4.1, should be set to request 5V, 9V, 12V, and 20V. A load can be connected to the screw terminals, and power resistors with a resistance of 10 Ω are used for the load (Figure 4.2). These resistors are connected in parallel and series to create the desired load.

Figure 4.1: Power trigger[33]

To determine the control and measurement of the solution, we first tested the communication with an external MCU. For the IP2368 and the SW2303, we used an STM32 (Figure 4.3) connected via the I^2C protocol. Initially, the STM32 attempts to identify all connected devices by sending the address over the I^2C line and waiting for an acknowledgement signal (subsection 2.6.3). If the address is found, the register can be read and potentially written to determine the measurement and control capabilities.

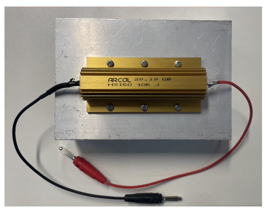


Figure 4.2: Power resistance 10 Ω [34]

Both development boards came with their own microcontroller and software. The microcontroller was responsible for the I^2C interface with the USB PD IC on the board, enabling direct access to read and write the registers and evaluate their functions.

To determine the data capabilities of a USB, the device is placed between a PC and a USB device such as a USB stick or a keyboard and mouse. This allows testing of the USB protocol to see if it works and the speed at which it operates. The testing can be done by transferring files from the computer to the USB stick and then back to the PC.

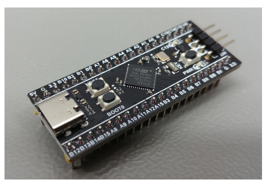


Figure 4.3: Black pill STM32 [35]

4.1.2 Test Procedure

The test that needs to be performed to validate the capabilities of the power delivery system are:

1. **Power delivery**
   - Can it support all required voltages?
   - Can it provide the specified power?
   - How stable is the power output (deviation from stable voltage)?
     - Stability with no load
     - Stability with load
2. **Measurement and Control Capabilities**
   - Is it possible to communicate with the MCU?
   - Can it provide accurate measurements?
   - Is it possible to control the power output?
3. **USB Data Capabilities**
   - Can it connect a downstream device?
   - At what speed is it possible to transfer data?

The first two tests can be performed using the test setup shown in Figure 4.4. In this setup, a 20V (5A max) power supply is connected to the power delivery system. The power delivery system is linked to a power trigger via a USB Type-C cable. This power trigger acts as a sink and can be set to request 5, 9, 12, 15, or 20 volts. The USB trigger is then connected to a load to assess the **power delivery** function of the power delivery system.

For testing the **Measurement and Control Capabilities**, the MCU (STM32 or feather board for MAX DEV kit) is connected to a PC. Both function as I^2C converter, where the I^2C functionality can be tested in combination with the functionality of the control and measure registers accessible via the I^2C communication.

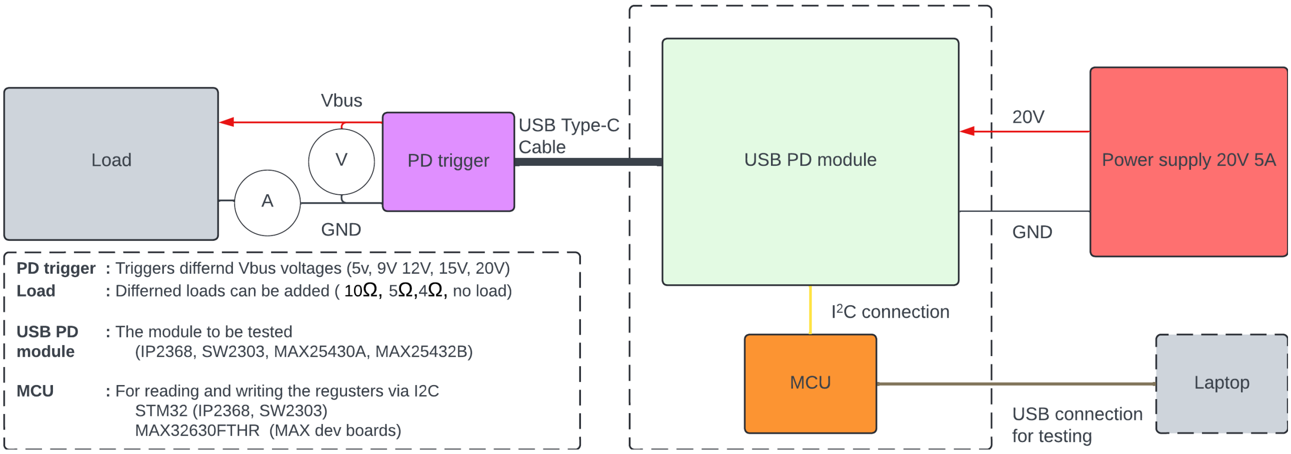


Figure 4.4: Test setup USB implementation

For the **power delivery**, the ingoing voltage and current are read off from the power supply, and the voltage and current delivered to the load are measured with 2 multimeters. These values ar measured for all the different voltages able to be triggered (5,9,12,15,20 V) and with 4 different loads (no load, 10Ω, 5Ω, 4Ω). With the measurements, voltage error (difference between the theoretical triggered voltage and the measured voltage), the delivered power and the efficiency can be calculated. With these measurements, the power delivery system can deliver the amount of power stated in the datasheet.

To test the **Measurement and Control Capabilities**, there are 2 categories of devices to test: the PD modules and the development board.
For the PD module, a STM32 is connected to the module via the I^2C lines and to a PC. This will first test the I^2C connection by running an address search. The Arduino IDE package for the STM32[36] comes with one, and therefore, that one is used. It works by sending the address it will test over the I^2C bus. If a device with that address is connected to the I^2C bus, it will respond by sending an acknowledgement bit back to the master. The STM32 will be able to read that and determine that a device with that address is connected.
For the MAX DEV kits, a feater board with a microcontroller is included, as well as software to connect to it (all can be found on [37] and [38]). And the checking the I^2C connection is done by the software.
After checking the connection to the PD model, the **measurement capability** can be checked; this is done by reading the register where the voltage, power and current values will be stored. And compare them with the values measured with the multimeters and connected power supply. This will be done with the different voltage triggers and loads connected to it.
The maximum current or power value in the register can be written by the MCU or feature board for **control**. By adjusting the load and the voltage of the PD trigger, it can be determined if this value is not exceeded.

To test the **USB data capabilities**, connect a downstream device to the USB Type-C port and a computer to the host connection port. Start by connecting a USB 2.0 device such as a mouse or keyboard to confirm that the data connection is functioning. If successful, you can then connect a USB 3.2 device, such as a USB storage stick, to test the data transfer speeds.

4.2 IP2368 PD Module

The IP2368 is the first solution tested, the datasheet mentioned that it would be able to charge with up to $100W$ and be able to measure the outgoing and ongoing voltage, current and power. And control the max amount of power delivered to the USB Type-C port.
First, the Charging capability will be checked and evaluated. After that, the measurement and controllable will be tested. As seen in subsection 4.1.2.

4.2.1 Power Delivery

Testing the charging capability was done by connecting the module to the power source, this one done as seen in Figure 4.5. The load is connected via a USB Type-C cable and the power trigger to the USB Type-C port also seen in Figure 4.5. When turning on the lab power supply, it is expected that the PD module will turn on and provide power to the load.

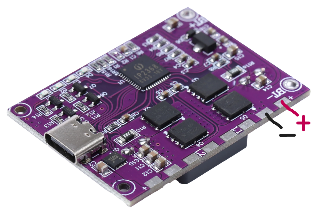


Figure 4.5: IP2368 module with power connected[13]

This was at first not the case; to operate the PD module, it first needed to be used as a sink to start working. This means that a power source, like a power bank or a USB power brick needs to be connect to the USB port to allow the module to work.
This is probably because the typical use of the IP2368 is for Power banks, where the module is permanently connected to a battery and, therefore, only needs to be initialized (connected to a power source).

Now that the PD module is functioning as a source, the power delivery functionality can be tested; this is done with the procedure explained in subsection 4.1.2. The result of this test can be found in Appendix A. Interesting observations of this test can be seen in Table 4.1.

Table 4.1: Observations of IP2368 Power Delivery

IP2368							
Trigger V	**No load dV**	**10 Ω**		**5 Ω**		**4 Ω**	
		dV	**eff (%)**	**dV**	**eff (%)**	**dV**	**eff (%)**
5	0,087	-0,11	70,96	-0,173	82,92	-0,212	83,72
9	0,16	-0,16	87,31	-0,28	89,46	-0,34	90,78
12	0,22	-0,25	90,83	-0,44	91,84	-0,54	91,93
15	0,27	-0,3	92,82	-0,53	92,27	-1,5	92,06
20	0,39	-0,4	93,57	-0,73	92,18	-2,93	86,94

Based on the measurements and observations, it is evident that the PD module can supply all the required voltages. However, under certain conditions, the voltage output may reach up to 1.5V without any power supply limitations. For instance, when the PD trigger is set to 20V and a 4 Ω load is connected, the power supply drops to 16.7V, possibly due to the power supply's limitation of 100W. Considering the 92.2% efficiency seen at other voltage levels, the power supply would need to deliver at least 108W to provide the required 100W at the USB Type-C port.
Additionally, although not explicitly tested, the efficiency of the PD module is relatively high, reaching up to 94% when delivering 100W.

4.2.2 Measuring and Controlling Capabilities

The next test to do is to test the controllability, the first step is to test if the PD module can be found on the I^2C bus. This will be done with a STM32, wich has a I^2C bus that works on 3.3V the same voltage as the IP2368. The guide found on GitHub by Dmitriy Mitchenkov [31] was followed for the connection to the MCU. After the connection was made the I^2C address searchers program was run, as mentioned in subsection 4.1.2. This gave no result.

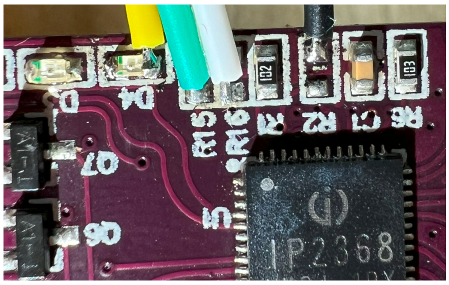


Figure 4.6: IP2368 I^2C connection [13]

To determine why the I^2C communication was not working, different options where tested:

1. Reduce the I^2C speed from the STM32 default of 1 Mbit/s to 400 Kbit/s.
2. Connect external pull-up resistors to the SCL, SDA, and INT lines as shown in Figure 4.6.
    - The VCCIO was first taken from the 3.3V bus of the STM32.
    - Later, it was tested with the VCCIO line of the IP2368 (not recommended due to the I^2C pins of the IP2368 being 3.3V tolerant and VCCIO being 5V).
3. Write while being used as a sink.

After testing the PD module with the above-mentioned options, the I^2C functionality was still not working. The voltage on the 3 busses was also measured and all 3 were around $0.2V$, pulled down by the IP2368 IC. Because the I^2C protocol is an active low protocol, this would explain why the MCU is not able to connect to the PD module.

Due to the I^2C communication not working, the manufacturer was contacted (INJONIC) [14]. This led to the release of a newer version of the datasheet (version 1.63). However, if the specific function from the IC was required, customization was necessary. The new datasheet did not provide any new insight into the working of the I^2C implementation.
Dmitriy Mitchenkov, the library creator for connecting the IP2368 to an Arduino, was also contacted. He encountered the same difficulty as us and mentioned that he was only able to read and write to the PD module while it was working as a sink. He also noted that not all PD modules work due to multiple versions in circulation. Some of them have the I^2C functionality, while others don't. Additionally, even when the I^2C functionality is available, it's not always stable and may work on some occasions but not on others.

Due to the aforementioned insights from Dimitry and the complications we encountered when testing the modules ourselves, it was determined that the PD module IP2368 is not suitable for use in the Power Delivery system.

4.3 SW2303 PD Module

The SW2303 is a PD module similar to the IP2368 where it can be used as a stand alone system. But instead of typically being used for power banks, the SW2303 can only be used as a PD source. Therefore, the module should work immediately when connecting it to a power source, unlike the IP2368 which needs to be used as a sink first to make the chip functional.

The SW2303 was tested because it was already in possession of the group members and had the power control as stated in the datasheet. Although it didn't have any advertised measurement features, an assumption was made that if it can control the power, it should also be able to measure the voltage and current it supplies to the USB Type-C port. Therefore, it could still be in one of the registers not mentioned in the datasheet. If not, a measuring circuit could be placed around the module to create the measurement functionality.

4.3.1 Power Delivery

To test the SW2303 PD module, the power connectors (seen in the left top of the PCB in Figure 4.7) were connected to the power supply.

The power delivery functionality was tested as described in subsection 4.1.2. The measurements can be found in Appendix A. The relevant values are presented in Table 4.2. From this table, it can be seen that the voltage difference from the triggered voltage is smaller than what was measured from IP2368, with the highest offset being $0.45V$. An interesting observation can be made for the measurement of 4Ω with the 20V trigger - here, SW2303 could not deliver any power as it shut down each time it powered up.
The efficiency is approximately the same, reaching a maximum of 95%, as expected from a DC to DC converter.

Figure 4.7: SW2303 module

Table 4.2: Observations of SW2303 Power Delivery

SW2303							
Trigger	**No load**	**10 Ω**		**5 Ω**		**4 Ω**	
V	**dV**	**dV**	**eff (%)**	**dV**	**eff (%)**	**dV**	**eff (%)**
5	0,113	0,027	81,40	-0,078	85,83	-0,11	85,89
9	0,16	-0,02	91,53	-0,19	91,21	-0,27	90,47
12	0,19	-0,04	93,57	-0,26	92,20	-0,37	91,11
15	0,23	-0,04	94,91	-0,3	92,75	-0,45	91,47
20	0,29	-0,08	94,00	-0,41	91,05	20	0,00

4.3.2 Measuring and Controlling Capabilities

To test the measuring and controllability features of the PD module with plate number SW2303, it must be connected to an STM32 as briefly mentioned in section subsection 3.3.4 and in the datasheet [30]. The IC needs to be set into an I^2C power allocation state, which is achieved by connecting the *PSET* pin to VDD (3.3V).

However, when testing the I^2C functionality of the SW2303, a problem arose. The SDA pin followed the same voltage as the voltage at the Vbus (voltage of the USB Type-C connector). This voltage was too high for the STM32 and caused it to fail the test.
Following this discovery, all pins connected to the STM32 were tested to determine if the SW2303 was in the I^2C power allocation state. This was done with no success. It was therefore concluded that with the current PD module, it is not possible to implement power allocation over I^2C.
After attempting to contact the manufacturer (iSmartWare) with no success, the conclusion was made that the SW2303 PD module was not suitable for the Power system.

4.4 MAX25430A Development Board

The MAX2530A development kit is made to evaluate the MAX25430A and, therefore, comes with everything needed to test the functionality. Including a starting guide on how to set up the development board so it can be used as a source functionality (Power delivery).

4.4.1 Power Delivery

To test the power delivery functionality, the development board was connected as shown in the Quick Start (found on [38]) and to the measurement setup explained in subsection 4.1.2.
The different loads and voltages are tested as described in subsection 4.1.2; the results of these tests can be found in Appendix A. The interesting results can be found in Table 4.3. Here, it can be seen that the development board is not able to provide 12V. This is due to this voltage not being in the standard power delivery and legacy standards, therefore, it is not supported by default by the development board. Another interesting observation is that the divergence from the triggered value is much lower than those of the previously tested PD modules. In later tests, it was discovered that the still measured difference was due to the resistance of the USB cable.
Although described in the datasheet that the MAX25430A was not able to provide more than 75W as the development board started shutting down when connecting a 4Ω to the PD trigger set to 15V or higher.

Table 4.3: Observations of MAX25430A Power Delivery

MAX25430A							
Trigger	**No load**	**10 Ω**		**5 Ω**		**4 Ω**	
V	**dV**	**dV**	**eff**	**dV**	**eff**	**dV**	**eff**
5	0,158	0,077	62,7	0,002	73,6	-0,021	76,29
9	0	-0,14	81,1	-0,27	85,7	-0,31	86,52
12							
15	0,01	-0,22	90,6	-0,45	90,8	-15	0,00
20	0,01	-0,3	92	-0,58	90,8	-20	0,00

4.4.2 Measuring and Controlling Capabilities

To test the measuring and controlling capabilities, the included feater board (Figure 4.8) could be connected to the computer; this functions as a bridge between a PC and the MAX25430A, and with the software downloadable from the analogue devices site [37], the software can read and present all the readable registers. And write the ones writable. With the interesting registers found in subsection 3.3.5.

For the measurement, the *Vbus_voltage* can be read; this always returned 0 due to the chip on the development board being a version A and the ADC for measuring the voltage and currently only being available on version B of the IC. Some voltage measurements could be found in register *power output control*, where the set voltage output can be read and written.

For power control, the *power_control* register could be used; this register should be able to at least turn-off or on the output voltage of the USB Type-C port, but due to this feature only working on the B variant, it did not work with the current version.
The same holds for the *power_output_control* which is only available on the B variant.

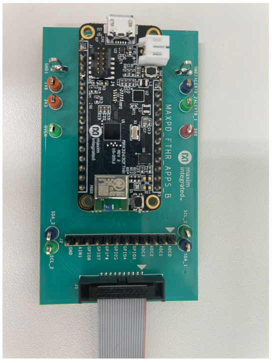


Figure 4.8: Feater board MAX25430 and MAX25432

In conclusion, the MAX25430A can provide charging capabilities of up to 75W. However, for measurements and controllability, the B version of the IC is required. It's worth noting that no mention of the B version can be found outside the datasheet of the general MAX25430 IC.

4.5 MAX25432B Development Board

The MAX25432B Development board is in the same family as the MAX25430A (section 4.4) and has comparable features and connection methods. It also comes with a feature board which is able to connect to the PC and serve's as a bridge to the I^2C registers in the IC.

4.5.1 Power Delivery

For testing the development board, the board is connected to power, as explained in the starter guide found on the site of the analog device [38]. The rest of the setup and test is done as mentioned in section subsection 4.1.2. The outcome of the test can be found in Appendix A.

The interesting observations can be seen in Table 4.4. Here, the same observation can be made as for the MAX25430A, where 12V is not supported, and the development board is not able to provide more than 75W of power.

Table 4.4: Observations of MAX25432B Power Delivery

MAX25432B							
Trigger	**No load**	**10 Ω**		**5 Ω**		**4 Ω**	
V	**dV**	**dV**	**eff (%)**	**dV**	**eff (%)**	**dV**	**eff (%)**
5	0,164	0,077	66,35	-0,003	75,60	-0,023	78,44
9	0,07	-0,08	82,85	-0,21	86,76	-0,25	87,37
12							
15	0,05	-0,2	91,14	-0,41	90,73	-15	0,00
20	0,09	-0,26	91,74	-0,53	89,60	-20	0,00

4.5.2 Measuring and Controlling Capabilities

To test the measuring and controlling capabilities, the Feater board (Figure 4.8) can be connected to the development board as explained in the starter's guide ([38]). This Feater board, in combination with the software designed specifically for this development board by Analog Devices, makes it possible to control the output voltage and current of the USB Type-C port and measure the current as well as the voltage.

The voltage can be controlled with the *VBUS Non-Default Target (low and high byte)*, allowing for use between 0 and 20V. For current control, the *VBus ilim setup byte* can be used to set a current limit. This current limit register was changed to multiple different values and the tests showed that the current never exceeded the set limit.
The voltage and current are measured by the ADC and stored in the *VBus voltage (low and high byte)* and the VBus current byte.

To determine the accuracy of the voltage control and the current and voltage measuring capability, a comparable test was conducted for the Power Delivery capabilities. In this test, the voltage is set by the MAX25432B, and the measured voltage and current are compared to the values measured by the internal ADC. The results of the test can be found in Table 4.5.

Table 4.5: Observations of MAX25432B Power control and measuring

trigger	**10 Ω**					
	Multimeter		**Register**		**Diff**	
Vset	**V**	**I (A)**	**V**	**I (A)**	**V**	**I (A)**
5	5	0,49	5	0,45	-0,072	0,035
9	8,89	0,88	9	0,85	-0,110	0,026
15	14,82	1,46	15	1,45	-0,180	0,010
20	19,71	1,94	20	1,95	-0,290	-0,008

trigger	**5 Ω**					
	Multimeter		**Register**		**Diff**	
Vset	**V**	**I (A)**	**V**	**I (A)**	**V**	**I (A)**
5	5	0,91	5	0,9	0,154	-0,008
9	8,74	1,65	9	1,65	0,260	-0,004
15	14,56	2,79	15	2,75	0,440	-0,037
20	19,37	3,74	20	3,75	0,630	0,006

trigger	**4 Ω**						
	Multimeter			**Register**		**Diff**	
Vset	**V**	**I (A)**	**V (IC)**	**V**	**I (A)**	**V (IC)**	**I (A)**
5	4,80	1,12	5,01	5,00	1,10	0,011	0,021
9	8,66	2,03	9,04	9,00	2,05	0,040	-0,016
15	14,43	3,43	15,07	15,00	3,45	0,070	-0,016
20	19,19	4,61	20,04	20,00	4,60	0,040	0,008

The current measurement is stored in 7 bits, where the least significant bit represents 0.05 A. Therefore, the interpolation error is 0.025 A. Upon examining the measurements, there are only 2 instances where the difference between the current measured by the multimeter and the ACD in the document MA25432B is more than ± 0.025 A, with only a 10 mA difference.

The voltage measurement is stored in 10 bits, where the least significant bit represents 0.025V. This means the interpolation error is 0.0125V. The measured voltage difference is much higher than the interpolation error, especially in the measurements with the load of 4 Ω and 5 Ω. This difference occurred because the voltage was measured at the load with a multimeter, not at the Development board. When the voltage was measured at the Development Board (*V(IC)*), the difference was much smaller, only 70mV, approaching the interpolation error.

The voltage difference measured at the load and at the development board can be explained by the USB C cable, wich has a resistance and therefore creates a voltage drop. This resistance can be calculated by taking the voltage at the development board, at the load and dividing it by the current running through it. As seen in Table 4.6, this resistance of the cable is around 0.19 Ω.

Table 4.6: Resistance measurement cable

V_diff	I (A)	R (Ω)
0,208	1,12	0,186
0,38	2,03	0,187
0,64	3,43	0,186
0,85	4,61	0,184

The measurements were done without the power handshake since changing the value of the power trigger had no impact on the output voltage, the power that was manually set on the MAX25432B was the amount of voltage that was outputted to the port, so the manual set power overwrites the handshake. But the handshake works, as shown earlier, where the triggered voltages were met by the MAX25432B when it wasn't manually overwritten by a PC.

A different observation can be made comparing the manual test results (see Table 4.5) with the results from the test involving the handshake (see Table 4.4). During the handshake test, the MAX25432B was unable to provide 15V and 20V when a load of 4Ω was connected. However, when the voltage was set manually, the MAX25432B was able to provide 20V and 15V with a 4Ω load. This difference may be due to the maximum voltage drop that the MAX25432B allows for during the handshake process. This parameter can be adjusted via the I^2C bus, although we have not been able to test this yet due to issues with getting the handshake to work with I^2C connectivity.

4.5.3 USB data Capabilities

To combine the charging with the data transfer capabilities, the USB implementation sub-module should pass through the data from the USB Type-C port. From the datasheet of the MAX25432B, the chip should already pass the data through. So the data lines from the USB hub can be connected to each USB implementation sub-module. Due to time constraints, this has not yet been tested and will be covered in the next steps section 4.9.

MAX25432B contains an interesting register that can be written over I^2C. This register seems like it could control the data connection, as the D+ and D- pins of the USB 2.0 connection are also used for the charging protocol. There is an option to connect these lines only with the chip for charging or to pass them through to the host connection (in register *GENERAL_SETUP*). This feature could give the IC the capability to control the data connection, but this needs to be tested to be sure.

4.6 Findings USB Implementation Sub-module

First, the IP2368 was tested due to its many features. However, some issues arose during testing. The PD module needed to be initialized by using it as a sink first. Also, the feature of being controllable over I^2C was not working as expected. The next PD module (SW2303) was also tested, but it had the same issue with I^2C communication, which made it uninteresting.

The third option was the MAX25430, but only the A version of the chip was available separately on a development board. This version lacked the controllability and measurement features needed for the power delivery system, which were available on the B chip.

Lastly, the development board for MAX25432B lived up to the promises made in the datasheet. It could charge USB Type-C devices and demonstrate the capability to measure the outgoing voltage and current. It also allowed control over the outgoing current and, consequently, the maximum power. The datasheet promised USB throughput capabilities, but this has not been tested due to time constraints.

4.7 USB Hub Sub-module

The **USB hub sub-module** will focus on connecting and switching off and on the data connection from the external PC to the USB implementation sub-modules. In section 3.4 the Anker USB 3.0 hub was chosen.
Due to the trade-offs between the requirements of the USB sub-module, and the extended time taken for USB implementation due to several non-working options, there has been less emphasis on testing this module. Since the Anker USB hub is an off-the-shelf product, some trust has been placed in its advertised performance.

Tests can still be conducted to assess the configuration feature. The main IC of the Anker hub, RTS5411, has the capability to enable or disable ports on startup. This can be configured over I^2C and thereby be controlled by the MCU.

Another option was found in the USB implementations sub-module while exploring MAX25432B (section 4.5). The datasheet mentions a register that could be used to select the function of the D+ and D- (data pins) of the USB Type-C connector. This can be used for legacy charging protocols, USB throughput, or a combination of both.

4.8 Integration

For a working system, all submodules need to function together. To get the USB implementation working, six USB implementation submodules need to be connected to the MCU.

Due to the form of the powerstation and the possibility to split the modules into two sets, two almost identical clusters can be made, which will be named wings. On each wing, a USB hub IC is placed along with three USB implementation modules as seen in Figure 4.9.

The wings are connected to the main board, which houses the MCU and possibly an I^2C mux. The I^2C may be necessary because the MAX25432B can only be configured to 4 I^2C addresses. Therefore, to communicate with 6 modules, 2 isolated buses are required. This can be achieved using a mux, as explained in subsection 3.2.3, or a MCU with multiple I^2C buses. Additionally, the main board is where the connections to the screen, upstream PC, and power source are established.

To connect the wings, each wing has one upstream USB connection, a USB downstream connection, an I^2C bus, and a power connection to the main board. One of the wing's downstream ports is connected to the upstream port of the other, creating a 7-port USB hub structure. The remaining downstream port is connected to the MCU to connect the MCU to the PC for data logging or programming. The I^2C lines from each wing need to be connected to an isolated I^2C bus.

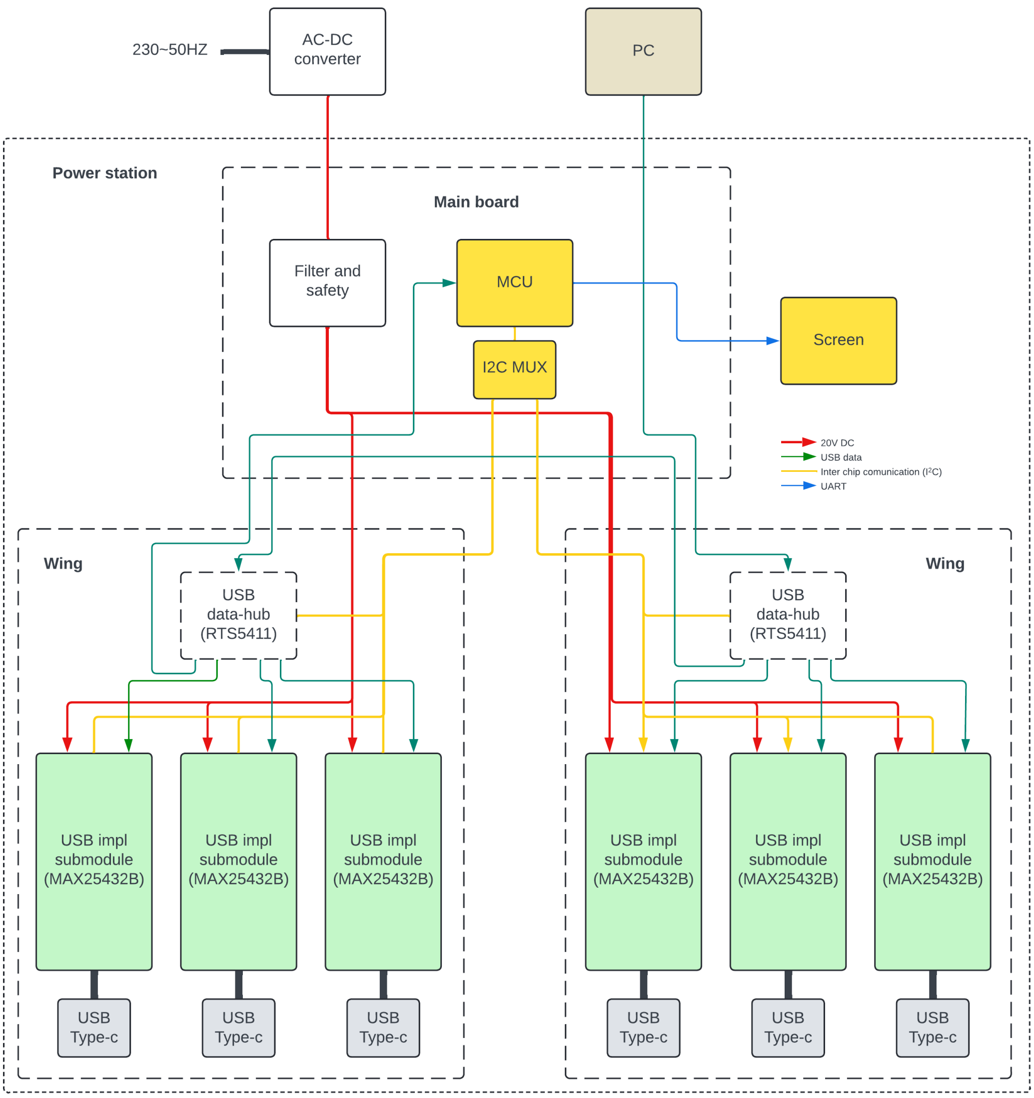


Figure 4.9: System integration diagram

In Figure 4.10, you can see the physical representation of the diagram in Figure 4.9. There are two identical wings where a volume representation of the core components is placed on a PCB-like structure. This demonstrates that it would be possible to fit all the core components in the powerstation casing, this casing can be seen in Figure 4.11. In the middle, the main board can be seen. This board connects to both wings to connect the data and power pins. It is populated with the STM32 (black pill) as well as the upstream port and power connector (both facing to the back side).

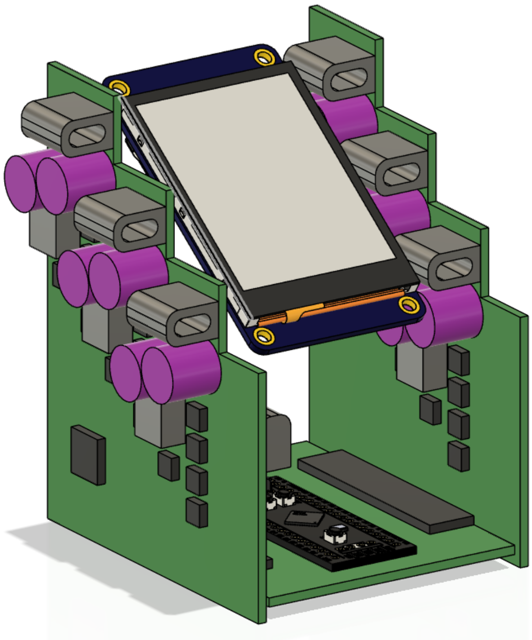


Figure 4.10: 3D representation powerstation electronics

Figure 4.11: Powerstation casing

4.9 Next Steps

The next phase of the project involves several critical steps to ensure the USB powerstation functions correctly and meets all requirements.

First, the feather board currently connected to the development board will be replaced with the chosen MCU. This step is essential to confirm that the MCU works seamlessly with other components. Key aspects such as the handshake protocol, power control, and measurement reading via the I^2C bus will be tested to ensure smooth operation.

Next, the data pass-through functionality of the MAX25432B will be validated. This involves checking if the data lines can be used alongside the power delivery function without any issues. This step is crucial to ensure that both power and data capabilities can work together effectively.

Integration tests with the other modules of the project will then be conducted. These tests aim to ensure that all modules work together harmoniously and that the powerstation meets all specified requirements.

Finally, real-world scenario testing will be performed to assess the system's performance under various conditions, like charging a phone or power bank. This testing will help identify any potential weaknesses or areas for improvement, ensuring the product's reliability and efficiency in practical applications.

By following these steps, the entire proof of concept for the USB powerstation will be validated, preparing it for future development and refinement.

Chapter 5

Conclusion

This chapter begins with a summary of the project in section 5.1, followed by key findings and recommendations in section 5.2. It concludes with an outline of potential future work in section 5.3.

5.1 Summary

The project aimed to design and develop a versatile USB powerstation as part of the Wrepair Powerstation Project. The key objectives were to support high-power delivery and data transfer through USB Type-C ports, integrate advanced charging protocols, and enable dynamic power allocation and real-time performance visualisation. The final design includes six USB Type-C ports, each supporting up to 100W, with a total power budget of 300W.

This project aimed to develop a versatile USB Type-C power station, integrating advanced charging protocols, real-time measurements, and power control capabilities. The goal was to create a modular system compatible with existing Wrepair stations, offering dynamic power allocation and performance visualisation.

The USB implementation subgroup approached this by focusing on selecting appropriate power delivery integrated circuits (ICs) and designing a system that could handle both power delivery and data transfer. This involved integrating advanced charging protocols, real-time measurements, and power control capabilities into a modular system that could seamlessly fit into existing Wrepair stations.

Several critical choices were made throughout the project. Initially, the IP2368 module was selected for its promising features. However, discrepancies in the datasheet and functionality led to reconsideration of this choice. Subsequently, the MAX25432B module was chosen for its robust power delivery capabilities, accurate voltage and current measurements, and reliable I^2C communication. Additionally, USB 3.0 was selected for the data hub sub-module due to its balance of speed and compatibility, ensuring efficient data transfer and power management.

The validation of these choices involved comprehensive testing and integration of the selected components. The MAX25432B module was thoroughly tested to confirm it met all design requirements, proving its suitability for the project. The integration of the USB implementation sub-modules and the data hub sub-module was also validated to ensure effective communication and functionality. The final design successfully supported both standalone operation and integration into the larger Wrepair station.

In conclusion, the project demonstrated a successful proof of concept for a high-power USB Type-C power station. The focused approach by the USB implementation subgroup, combined with strategic choices and thorough validation, ensured that the system met the demands of modern electronic devices and provided a solid foundation for future development and refinement.

5.2 Discussion and Recommendation

The project successfully achieved its objectives as described in section 1.4, providing a robust USB solution that integrates seamlessly with existing Wrepair stations. Initially, the IP2368 module was considered but replaced by the MAX25432B module due to performance issues of the IP2368. The MAX25432B met all design requirements, offering reliable power delivery, accurate measurements, and I^2C communication.

The MAX25432B module supported up to 100W per port, ensuring robust power delivery and control. A high-speed USB data hub was implemented using USB 3.0, facilitating efficient data transfer. Dynamic power control and real-time data visualization were effective, though full-scale testing with multiple ports was not completed.

Several tasks, including comprehensive full-scale testing, were not completed due to time constraints and scope adjustments. Initially aimed at delivering a final product, the project scope was shifted to focus on a proof of concept to manage the project within the available timeframe, prioritizing essential features, like power delivery, measuring the voltage and current and controlling the maximum allowed power per charging port.

Shifting to a proof of concept allowed for focused development and validation of essential components, ensuring robust core functionalities.

Future recommendations include enhancing inter-chip communication and conducting multi-port testing.

Further research needs to be conducted on the USB hub sub-module, where data isolation of single ports is required, but no proper solution has been designed yet.

In conclusion, the project successfully demonstrated a high-power USB Type-C powerstation concept, meeting modern electronic demands and laying a foundation for future enhancements.

5.3 Future Work

To enhance the project's outcomes, developing a comprehensive software library for the MAX25432B is essential. This will significantly improve control and usability by allowing for more precise and flexible management of the power delivery system. Enhancing the controllability of the USB hub through targeted modifications will ensure more reliable data and power management, addressing any potential issues with current configurations.

Creating a custom PCB design to integrate all components seamlessly will be a crucial step forward. This will not only increase the robustness and efficiency of the powerstation but also streamline the manufacturing process, making the product more viable for production.

Full integration testing of the entire product under real-world conditions is necessary to confirm that the power station meets all specified requirements and performs reliably. This step will validate the design and ensure it can handle various operational scenarios effectively.

Bibliography

[1] Wrepair, *Wrepair*, 2024. [Online]. Available: https://www.wrepair.com/ (visited on 05/01/2024).

[2] *Pinepower 120w desktop power supply (eu)*, 2024. [Online]. Available: https://eleshop.eu/pinepower-120w-desktop-power-supply-eu.html (visited on 06/13/2024).

[3] *Maant 8-port multifunctional charging station*, 2024. [Online]. Available: https://shop.rewa.tech/detail/maant-8-ports-multifunctional-chargering-station.html (visited on 06/13/2024).

[4] chargeasap, *280w zeus usb-c gan charger*, 2024. [Online]. Available: https://chargeasap.com/products/280w-zeus-charger (visited on 04/23/2024).

[5] ugreen, *Ugreen nexode 300w usb c gan oplader met 5 poorten*, 2024. [Online]. Available: https://nl.ugreen.com/products/ugreen-nexode-usb-c-oplader-300w-5-poorts-gan-oplader?variant=43580137865415&currency=EUR&utm_medium=product_sync&utm_source=google&utm_content=sag_organic&utm_campaign=sag_organic&gad_source=1&gclid=Cj0KCQjwlN6wBhCcARIsAKZvD5isdEgeeuWVtfz_S_7gTWsYRSooNW7PNeStd9qjf9EThx_gqTCMMzEaAralEA wcB (visited on 04/23/2024).

[6] T. E. of Encyclopaedia, *Usb technology*, 2024. [Online]. Available: https://www.britannica.com/technology/USB (visited on 05/01/2024).

[7] usbmakers, *History of the usb flash drive*, 2019. [Online]. Available: https://www.usbmakers.com/history-of-the-usb-flash-drive.

[8] U. Trader, *The history of usb - universal serial bus*, Feb. 2013. [Online]. Available: https://www.usbtrader.com/blog/article/the-history-of-usb-universal-serial-bus.

[9] J. Melo, *The history of usb flash drives*, Oct. 2023. [Online]. Available: https://www.usbmemorydirect.com/blog/history-of-usb-flash-drives/.

[10] J. Axelson, *USB complete: the developer's guide*. Lakeview research LLC, 2015.

[11] A. Media, *What is usb type-c? everything you need to know*, 2024. [Online]. Available: https://www.androidauthority.com/what-is-usb-type-c-594575/ (visited on 05/13/2024).

[12] embeddedcomputing, *Designing more flexible power banks with usb type-c pd - embedded computing design*, 2024. [Online]. Available: https://embeddedcomputing.com/technology/analog-and-power/designing-more-flexible-power-banks-with-usb-type-c-pd (visited on 05/13/2024).

[13] D. Mitchenkov, *D-314/ip2368-arduino-library: An arduino library for interfacing with the ip2368 ic. provides functionalities for reading and managing charging states, battery percentage, and more.* 2024. [Online]. Available: https://github.com/D-314/IP2368-Arduino-Library (visited on 05/14/2024).

[14] *Injoinic technology*, 2024. [Online]. Available: http://www.injoinic.com/?lang=en-us (visited on 05/14/2024).

[15] *Usb-c en usb 3.1, usb 3.2, usb4 en usb4 v2 uitgelegd*, Sep. 2023. [Online]. Available: https://www.onlogic.com/nl/blog/usb-c-en-usb-3-1-usb-3-2-usb4-en-usb4-v2-uitgelegd/.

[16] *What is a usb hub: How does a usb hub work*, 2024. [Online]. Available: https://www.electronics-notes.com/articles/connectivity/usb-universal-serial-bus/usb-hubs.php (visited on 05/14/2024).

[17] *Usb tiers and system design*, 2024. [Online]. Available: https://avonic.com/usb-tiers-and-system-design/ (visited on 05/14/2024).

[18] M. T. Inc., *Usb5807c hub controller datasheet*, Accessed: 2024-05-31, 2019. [Online]. Available: https://ww1.microchip.com/downloads/aemDocuments/documents/UNG/ProductDocuments/DataSheets/00003182A.pdf.

[19] *Hub tp-link uh720 usb 3.0 - teardown, presentation, test*, 2024. [Online]. Available: https://www.elektroda.com/rtvforum/topic3698422.html (visited on 05/31/2024).

[20] Realtek, *Rts5411s - realtek*, 2024. [Online]. Available: https://www.realtek.com/Product/Index?id=4144&cate_id=195 (visited on 05/31/2024).

[21] Genesys Logic, *Gl3510 usb 3.1 gen 1 hub controller datasheet*, Accessed: 2024-05-31, Jul. 2018. [Online]. Available: https://www.semiee.com/file/GENESYS/GENESYS-GL3510.pdf.

[22] J. Chen and S. Huang, "Analysis and comparison of uart, spi and i2c," in *2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA)*, 2023, pp. 272–276. DOI: 10.1109/EEBDA56825.2023.10090677.

[23] *Uart: A hardware communication protocol understanding universal asynchronous receiver/transmitter | analog devices*, 2024. [Online]. Available: https://www.analog.com/en/resources/analog-dialogue/articles/uart-a-hardware-communication-protocol.html (visited on 06/10/2024).

[24] D. V. Gadre, S. Gupta, D. V. Gadre, and S. Gupta, "Serial communication: Spi and i2c," *Getting Started with Tiva ARM Cortex M4 Microcontrollers: A Lab Manual for Tiva LaunchPad Evaluation Kit*, pp. 211–238, 2018.

[25] C. Basics, *Basics of the i2c communication protocol*, Accessed: 2024-06-14, n.d. [Online]. Available: https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/.

[26] J. Hoekstra, *Versterkerschakelingen*, DRAFT-version, Dec. 2014.

[27] M. Nederland, *Lvm25fvr050e-tr ohmite | mouser nederland*, 2024. [Online]. Available: https://nl.mouser.com/ProductDetail/Ohmite/LVM25FVR050E-TR?qs=sE1HjTVAc2hOpytwwFq6jA%3D%3D&mgh=1&utm_id=20331598300&gad_source=1&gclid=CjwKCAjwgdayBhBQEiwAXhMxtgk57sqoyrXKUtPCfW 7goqbrl_hL6DwFE4Ytl8WxhfRoCpQcQAvD_BwE (visited on 05/28/2024).

[28] AnalogDevices, *Max25430 datasheet and product info | analog devices*, 2024. [Online]. Available: https://www.analog.com/en/products/max25430.html (visited on 05/14/2024).

[29] T. Instruments, *Tps65982 datasheet*, 2024. [Online]. Available: https://www.ti.com/product/TPS65982?utm_source=google&utm_medium=cpc&utm_campaign=app-null-null-gpn_en-cpc-pf-google-wwe&utm_content=tps65982&ds_k=TPS65982&dcm=yes&gad_source=1&gclid=Cj0KCQjw6uWyBhD1ARIsAIMcADqtAresHH_IrsUAu_FNpI_9Rtj35v77gnfVkleXBN79mPOirqaT14AaAld8EA wcB&gclsrc=aw.ds (visited on 05/14/2024).

[30] iSmartWare, *Sw2303 datasheet*, 2024. [Online]. Available: https://doc.diytrade.com/docdvr/2968899/51546274/1708928899.pdf (visited on 05/14/2024).

[31] Dmitriy, *Dmitriy's github*, Accessed: 2024-06-03, 2024. [Online]. Available: https://github.com/D-314.

[32] Texas Instruments, *Tca9548a low-voltage 8-channel i2c switch with reset function*, Accessed: 2024-06-14, 2016. [Online]. Available: https://www.ti.com/lit/ds/symlink/tca9548a.pdf.

[33] *Pd snel opladen testbord 5-20v verstelbare pd trigger board module usb type-c 100w spanning connetor voeding accessoires - aliexpress*, 2024. [Online]. Available: https://nl.aliexpress.com/item/1005006585829730.html?spm=a2g0o.productlist.main.1.19581N861N864G&algo_pvid=35cf9e05-e446-4702-92bc-fbeb83e5a531&algo_exp_id=35cf9e05-e446-4702-92bc-fbeb83e5a531-0&pdp_npi=4%40dis%21ILS%2126.77%218.80%21%21%2150.69%2116.66%21%402103893b171801159040002 2112000037736787125%21sea%21IL%211716954946%21&curPageLogUid=eAXsiNBUEWep&utparam-url=scene%3Asearch%7Cquery_from%3A (visited on 06/10/2024).

[34] *Arc hs10 10r j: Draadweerstand, axiaal, 10 w, 10 ohm, 5% bei reichelt elektronik*, 2024. [Online]. Available: https://www.reichelt.nl/nl/nl/draadweerstand-axiaal-10-w-10-ohm-5--arc-hs10-10r-j-p233363.html (visited on 06/10/2024).

[35] *Stm32f411ceu6 - weact black pill v2.0 | stm32-base project*, 2024. [Online]. Available: https://stm32-base.org/boards/STM32F411CEU6-WeAct-Black-Pill-V2.0.html (visited on 06/10/2024).

[36] STMicroelectronics, *Stm32 arduino core*, Accessed: 2024-06-12, 2024. [Online]. Available: https://github.com/stm32duino/Arduino_Core_STM32.

[37] Analog Devices, Inc., *Max25430a evaluation kit*, Accessed: 2024-06-12, 2024. [Online]. Available: https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/max25430aevkit.html#eb-overview.

[38] Analog Devices, Inc., *Max25432b evaluation kit*, Accessed: 2024-06-12, 2024. [Online]. Available: https://www.analog.com/en/resources/evaluation-hardware-and-software/evaluation-boards-kits/max25432bevkit.html#eb-overview.

Appendix A

Measurements Power delivery

SW2303		(Power supply: 20V with max 5A)														
with power trigger	No load	10 Ω load					5 Ω load (2 × 10 Ω parallel)					4 Ω load (2 × 10 Ω and 1 × 20 Ω in parallel)				
V (triggerd)	V (load)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)
5	5,113	5,027	0,502	2,52	0,155	81,40	4,922	0,98	4,82	0,281	85,83	4,89	1,212	5,93	0,345	85,89
9	9,16	8,98	0,899	8,07	0,441	91,53	8,81	1,758	15,49	0,849	91,21	8,73	2,172	18,96	1,048	90,47
12	12,19	11,96	1,197	14,32	0,765	93,57	11,74	2,345	27,53	1,493	92,20	11,63	2,897	33,69	1,849	91,11
15	15,23	14,96	1,496	22,38	1,179	94,91	14,7	2,934	43,13	2,325	92,75	14,55	3,625	52,74	2,883	91,47
20	20,29	19,92	2	39,76	2,115	94,00	19,59	3,918	76,75	4,215	91,05	tripping: no stable power connection				

ip2368		(Power supply: 20V with max 5A)														
with power trigger	No load	10 Ω load					5 Ω load (2 × 10 Ω parallel)					4 Ω load (2 × 10 Ω and 1 × 20 Ω in parallel)				
V (triggerd)	V (load)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)
5	5,087	4,895	0,49	2,40	0,169	70,96	4,827	0,962	4,64	0,28	82,92	4,788	1,189	5,69	0,34	83,72
9	9,16	8,84	0,883	7,81	0,447	87,31	8,72	1,738	15,16	0,847	89,46	8,66	2,151	18,63	1,026	90,78
12	12,22	11,75	1,175	13,81	0,76	90,83	11,56	2,304	26,63	1,45	91,84	11,46	2,846	32,62	1,774	91,93
15	15,27	14,7	1,47	21,61	1,164	92,82	14,47	2,886	41,76	2,263	92,27	13,5	3,355	45,29	2,46	92,06
20	20,39	19,6	2	38,49	2,057	93,57	19,27	3,846	74,11	4,02	92,18	17,07	4,246	72,48	4,992	86,94
												supply dropped to 16,7V for 20V triggered				

MAX25430A		(Power supply: 20V with max 5A)														
with power trigger	No load	10 Ω load					5 Ω load (2 × 10 Ω parallel)					4 Ω load (2 × 10 Ω and 1 × 20 Ω in parallel)				
V (triggerd)	V (load)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)
5	5,158	5,077	0,508	2,58	0,2057	62,69	5,002	0,997	4,99	0,3388	73,60	4,979	1,238	6,16	0,404	76,29
9	9	8,86	0,886	7,85	0,484	81,09	8,73	1,738	15,17	0,885	85,72	8,69	2,16	18,77	1,0847	86,52
12	There is no 12V output from the MAX25430A															
15	15,01	14,78	1,478	21,84	1,205	90,64	14,55	2,902	42,22	2,324	90,84	tripping: no stable power connection				
20	20,01	19,7	2	38,87	2,113	91,97	19,42	3,88	75,35	4,15	90,78	tripping: no stable power connection				

MAX25432B		(Power supply: 20V with max 5A)														
with power trigger	No load	10 Ω load					5 Ω load (2 × 10 Ω parallel)					4 Ω load (2 × 10 Ω and 1 × 20 Ω in parallel)				
V (triggerd)	V (load)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)	V (load)	A (load)	P (load)	A (power supply)	Eficency (%)
5	5,164	5,077	0,494	2,51	0,189	66,35	4,997	0,941	4,70	0,311	75,60	4,977	1,182	5,88	0,375	78,44
9	9,07	8,92	0,869	7,75	0,4678	82,85	8,79	1,668	14,66	0,845	86,76	8,75	2,079	18,19	1,041	87,37
12	There is no 12V output from the MAX25432B															
15	15,05	14,8	1,446	21,40	1,174	91,14	14,59	2,797	40,81	2,249	90,73	tripping: no stable power connection				
20	20,09	19,74	2	38,28	2,086	91,74	19,47	3,771	73,42	4,097	89,60	tripping: no stable power connection				