Bachelor Thesis

Czech Technical University in Prague

F3 Faculty of Electrical Engineering
Department of Microelectronics

Smart MPPT Solar Charger

David Rajchman

BACHELOR'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Rajchman David**   Personal ID number: **507258**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Microelectronics**

Study program: **Electronics and Communications**

II. Bachelor's thesis details

Bachelor's thesis title in English:

Smart MPPT Solar Charger

Bachelor's thesis title in Czech:

Smart MPPT solární nabíje ka

Guidelines:

1. Make market research for MPPT solar chargers available in the Czech Republic.
2. Design your own MPPT charger using a dedicated integrated circuit. Required framework parameters: remote monitoring, display for displaying operating parameters, compatibility with LFP, current protection, temperature monitoring, protection against battery undercharging.
3. Compare your proposal with the market research output.
4. Realize the charger in the form of a functional sample.
5. Test the operation and evaluate the achieved parameters.

Bibliography / sources:

1) Performance Analysis of MPPT, Gupta Ankur Kumar, ISBN 979-8889952794
, 2023
2) MPPT Design Techniques for Stand Alone PV System, Kumar Vipin, ISBN 979-8889952770, 2023

Name and workplace of bachelor's thesis supervisor:

Ing. Vladimír Janí ek, Ph.D.   Department of Microelectronics FEE

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **13.02.2024**   Deadline for bachelor thesis submission: **24.05.2024**

Assignment valid until: **21.09.2025**

_____   _____   _____
Ing. Vladimír Janí ek, Ph.D.   prof. Ing. Pavel Hazdra, CSc.   prof. Mgr. Petr Páta, Ph.D.
Supervisor's signature   Head of department's signature   Dean's signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

_____   _____
Date of assignment receipt   Student's signature

Acknowledgements

Rád bych poděkoval vedoucímu své bakalářské práce, Ing. Vladimíru Janíčkovi, Ph.D., za vynikající vedení, cenné rady a konstruktivní připomínky během konzultací. Také bych chtěl poděkovat své přítelkyni za trpělivost, kterou se mnou měla během psaní této práce, a své rodině, která mi byla vždy oporou.

Declaration

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Během vypracování této práce byly používány nástroje umělé inteligence v souladu s rámcovými pravidly používání UI na ČVUT k následujícím účelům: korekce anglicky psaného textu za použití aplikace Writefull; vyhledávání části informačních zdrojů k tématu za použití nástrojů Microsoft Copilot a ChatGPT, přičemž výstupy UI nebyly považovány za zdroj informací; a tvorba firmware za použití nástroje GitHub Copilot, konkrétně k asistenci s hledáním chyb v kódu a generováním repetitivních částí programu. Generace nových částí programu nebo vytváření algoritmů nástrojem Copilot nebylo využito.

V Praze, 24. května 2024

Abstract

This thesis focuses on the design and implementation of a smart solar charger utilizing the MPPT (Maximum Power Point Tracking) algorithm, aimed at facilitating the integration of a solar system into a smart home for users. The thesis begins with a market survey that presents various price categories of MPPT solar chargers. Based on this survey, a device concept is defined, which is then realized from start to finish.

For the implementation of the smart solar charger, a printed circuit board (PCB) was designed, integrating a microcontroller capable of WiFi communication, a charging circuit utilizing a synchronous buck converter, and a display with buttons for local device configuration. Furthermore, firmware was developed for the microcontroller to manage communication with the smart home, its own web interface, and other features. A case with connectors was designed for the entire device. The functionality of the prototype was verified at the end of the thesis.

Keywords: MPPT, ESP-IDF, Solar charger, ESP32-C6, ESP32, Home AssistantLTC4015, PCD8544

Supervisor: Ing.Vladimír Janíček, Ph.D.

Abstrakt

Tato práce se zaměřuje na návrh a realizaci chytré solární nabíječky využívající algoritmus MPPT (Maximum Power Point Tracking), která má uživatelům usnadnit integraci solárního systému do chytré domácnosti. Práce začíná průzkumem trhu, který představuje různé cenové kategorie MPPT solárních nabíječek. Na základě tohoto průzkumu je definován koncept zařízení, který je následně realizován od začátku do konce.

Pro realizaci chytré solární nabíječky byla navržena deska plošných spojů, která integruje mikrokontrolér s možností komunikace přes WiFi, nabíjecí obvod využívající synchronní buck převodník a displej s tlačítky pro lokální konfiguraci zařízení. Dále byl vyvinut firmware pro daný mikrokontrolér, který zajišťuje komunikaci s chytrou domácností, vlastní webové rozhraní a další funkcionality. Pro celé zařízení byla navržena krabička s konektory. Na závěr práce byla ověřena funkčnost prototypu.

Klíčová slova: MPPT, ESP-IDF, Solar charger, ESP32-C6, ESP32, Home Assistant, LTC4015, PCD8544

Překlad názvu: Smart MPPT solární nabíječka

Contents

Figures

Tables

Chapter 1

List of Terms and Abbreviations

- **AGM** [Absorbent Glass Mat], a lead acid battery technology where the acid is soaked into a layer made of glass fiber fabric

- **AP** [wireless Access Point], a name for a device that creates a WiFi network

- **ESP-IDF** [ESPressif Iot Development Framework], a framework based on C language compatible with C++ used for developing aplications for ESP32 microcontrolers

- **FPWM** [Forced Pulse Width Modulation]

- **freeRTOS** [free Real Time Operating System] [1], a open source operating system for microcontroles, serves as the basis of ESP-IDF

- **GPIO** [General Purpose Input Output], a term for a microcontroller pin that provides multiple functionalities, typically followed by an identifier for example GPIO1

- **GEL** Lead acid battery technology using silica gel as a medium for absorbing the acid. Silica gel

is a porous form of SiO2 that absorbs water or aqueous solutions

- **HAL** [Hardware Abstraction Layer], a software layer that interprets universal bus commands into microcontroller specific implementations

- **I2C** [Inter Integrated Circuit], a serial communication bus [2]

- **ISR** [Interrupt Service Routine], a non-blocking, highest priority function that is triggered by a hardware interrupt event

- **JTAG** [Joint Test Action Group], an industry standard for testing and integrated circuits, ESP-IDF offers an JTAG debuger based on OpenOCD, that comunicates with the JTAG interface on Espressifs microcontrolers

- **LFP** [Lithium Iron Phosphate], a battery technology also referred to as LiFePO4

- **MPPT** [Maximum Power Point Tracking], a technique used for efficient energy extraction from solar panels

1

- **MQTT** [Message Queuing Telemetry Transport], a communication protocol[3]

- **NMC** shorthand for lithium Nickel Manganese Cobalt oxide batteries, also referred to as Li-NMC, LNMC, or NCM

- **PCB** [Printed Circuit Board]

- **PFM** [Pulse Frequency Modulation]

- **SMBus** an extension of the I2C specification which is generally compatible [4] with I2C communication, but implements additional functionality such as packet error checking (PEC) [5]

- **S.M.S.C** Name of the device developed during this thesis, stands for Smart Mppt Solar Charger

- **SoC** [System on a Chip], integrated circuit that includes not just processing cores but also RAM, peripherals and other supporting circuits

- **SPI** [Serial Peripheral Interface],[6] a commonly used bus to connect peripherals to a microcontroler

- **SSID** [Service Set IDentifier], is the WiFi network name, can be up to 32 characters long

- **STA** [STAtion], a WiFi AP that is not hosted by the ESP32-C6, to which the ESP32-C6 can connect if switched to STA WiFi mode

- **THT** [Through Hole Technology mounting], is a type of electronic component mounting that relies on pins being soldered inside holes

- **VAT** [Value Added Tax]

Chapter 2

Introduction

Renewable energy sources are becoming increasingly popular, many people are investing into rooftop solar power plants to power their home and lower their energy bills. Due to the rising demand prices for solar panels and other componets are constantly decresaing. For this reason low voltage off grid solar systems with batteries are starting to become economically viable. They are well suited for providng power to recreational vehicles, camping sites, cottages, and many more locations where high power electronic devices are not required. Due to the remote nature of their deployment customers are trying to find solutions on how to monitor their solar system remotely. In this thesis, I aim to introduce the basic concepts related to MPPT charging. A survey will be conducted on the view of the currently available commercial solutions. Then I will design and develop a functional prototype of a smart MPPT solar charger. The work on this project will include a custom PCB design, software written using the ESP-IDF framework and an enclosure designed in Autodesk Fusion. The finished prototype will be compatible with the Home Assistant smart home software. The prototype will be tested to validate the design.

The motivation for this project were my personal interests in internet of things and smart home automation. I could not find a suitable solar MPPT charger, that would seamlessly integrate into a smart home solution. I also wanted to improve my programming skills and develop a complete application utilizing a more advanced microcontroller than 8 bit Arduino boards.

Chapter 3

Theoretical Analysis

■ 3.1 Solar Panel Technologies

As of 2023, three main solar panel technologies are present on the market, each with unique properties. While these technologies differ, all solar panels operate on the same basic principle. They are assembled from multiple solar cells of the same type, connected in series and parallel to achieve the desired voltage and maximum power.

The majority of new installations in 2023 consist of silicon solar panels [7]. These panels are based on the crystalline structure of silicon and can be divided into polycrystalline and monocrystalline. Polycrystalline solar panels, which are older and cheaper at the time of this writing, have an efficiency of about 20 % depending on the quality of production [8]. Monocrystalline solar panels, currently the most commonly installed, achieve higher efficiency around 25 % and can harvest the most energy per square meter of panel area [9].

Thin-film technology, due to its low efficiency of around 9 %, is used mainly in special applications. One of its advantages is the flexibility of TF solar panels, which can be bent to adhere to uneven surfaces, such as a car roof.

■ 3.2 Solar Battery Technologies

For installing self-sufficient solar systems with a capacity in kWh, there are 2 types of batteries that can be recommended for use.

The oldest but still used technology is lead-acid batteries. These batteries can be divided into several other types based on construction and maintenance methods. The most well-known are lead-acid batteries with a sulfuric acid solution. The construction of these batteries can be open, where it is necessary to replenish evaporated water, or hermetically sealed, where no further maintenance is required, although these batteries must not be tilted. There are also lead-acid batteries where the solution is absorbed into a solid medium. The two most utilized technologies are AGM batteries, and GEL batteries. Such a battery can then be placed in any position. These batteries also have better resistance to deep discharge, allowing their full capacity to be utilized

without the risk of damage.

A newer technology is lithium-ion batteries, which is a large group of various technologies. The largest market share of li-ion [10] [11] is occupied by NMC and LFP technologies. For energy storage applications, LFP batteries are more suitable, maintaining their capacity over a greater number of charging cycles and being safer in the event of failure, with a lower risk of fire and explosion, than NMC batteries. Another advantage is that they do not contain cobalt, whose production is very unecological and often employs child labor. Disadvantages of LFP include a higher price per Wh than NMC, as well as a lower volumetric and weight energy density compared to NMC. At the same time, LFP has a more complex evaluation of the current state of charge since their remaining energy dependence on voltage is not linear and has a very low slope in the 30 % - 80 % range. This causes a greater voltage drop under load than the voltage drop caused by the discharge of energy. Modern battery remaining energy calculation algorithms, however, successfully address this issue.

3.3 MPPT Algorithm for Maximizing Solar Panel Output

Solar panels have a nonlinear current-voltage dependency, and for all such sources, it applies that their performance changes depending on the voltage or current they operate with. Figure 3.1 plots 2 curves, the current-voltage dependency in green and the power-voltage dependency in blue. Important points on this graph are as follows:

- $\mathbf{U}_{oc}$ the voltage of the solar panel in open circuit

- **MPP** the maximum power point

- $\mathbf{U}_m$ the voltage at the maximum power point

- $\mathbf{P}_{max}$ the value of maximum power

To achieve the highest possible output from a solar panel, it is necessary to find the maximum power point and operate around it. This can be achieved using MPPT algorithms. Their basic principle is that they search for maxima on the power curve. On the power curve of an ideal solar cell, it is possible to find the global maximum power without issue, as there is only one local maximum across the entire curve. Real solar panels, and especially solar panel arrays have more complex curves, especially if part of the solar panel array is illuminated with different radiation intensity than the rest. In such cases, additional local maxima can emerge [12] at places with lower performance compared to the maximum possible (Figure 3.2). The algorithms use several basic techniques:

- **Full sweep** - a technique based on gradually increasing the working voltage, starting from open-circuit voltage to short-circuit, while measuring the power at each point. This method allows for the identification of

all maxima on the power curve and the global maximum. However, it is important to emphasize that during the execution of this technique, the solar panel is used inefficiently. Frequent application of this technique can lead to less efficient use of power than simply setting 70 % U_{oc}, the usual maximum power point location, as the working voltage of the converter

- **Oscillation around the working point** - this technique represents a mechanism that allows for precise adjustments to the working point with maximum power. It is based on quickly testing the vicinity of the working point in a range of a few percent of U_{oc}. If a direction is identified in which the power increases compared to the currently set point, the working point is gradually shifted in this direction. It is important to emphasize that this method is not capable of detecting whether the local maximum is also the global maximum. This presents a risk that the converter will gradually move to a place with low energy yield.

- **Partial sweep** - a method representing a variation of the mentioned full sweep technique, but characterized by limiting the width of the area it works in to approximately twenty percent of U_{oc} with the center at the current local maximum U_m, or at 70 % U_{oc}. This approach allows for the identification of most local maxima and subsequently detecting the global maximum among them. It is necessary to emphasize that there are situations when the global maximum is outside the interval of this sweep, in such a case it remains undetected.

Using these techniques, it is possible to construct an algorithm that exhibits a high probability of operating at the global maximum most of the time. At the same time, it minimizes the potential energy loss associated with searching for this maximum.
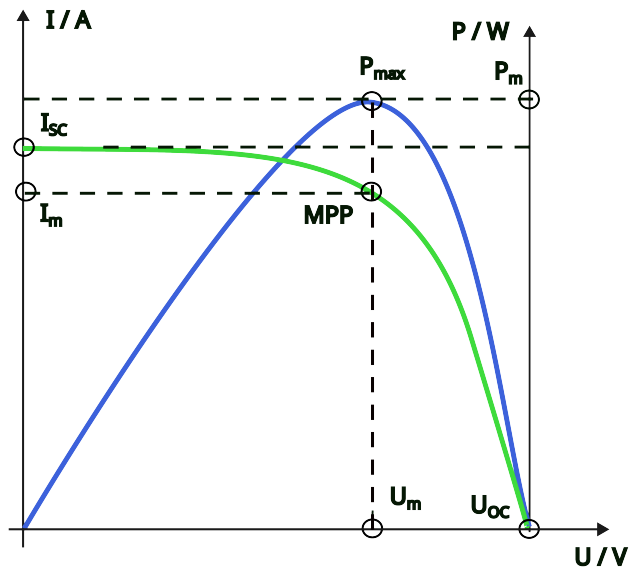
Figure 3.1: IV load characteristic of an ideal solar panel. Current dependency on voltage in green, power dependency on voltage in blue (an illustrative image based on [12])
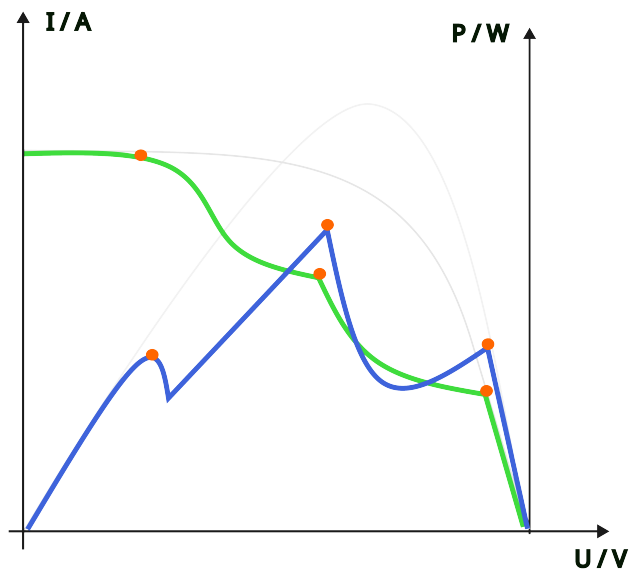

Figure 3.2: IV load characteristic of a partially shaded real solar panel. Points of local maxima marked in orange (an illustrative image based on [12])

Chapter 4

Market Study

4.1   Device Requirements

To prepare for the project, a brief market analysis of the present state in the (smart) MPPT battery charger industry is conducted to comprehend the capabilities and costs of the devices. The basic requirements that the selected device must meet, and the subsequent evaluation criteria, stem from the model situation of their use. This situation is defined as follows:

The device will be used to create a self-sufficient energy system at a location not connected to the power grid. This place can be stationary (e.g., a mountain cabin) or mobile (e.g., a recreational vehicle). One of the bonus requirements for the devices in the market study is the ability for remote monitoring of the situation. The entire system should operate on a low direct current output voltage (e.g., 12 V). The possibility of connecting 230 V appliances should be expandable by possibly connecting an inverter, for which an output should be provided. An important deciding factor would be the system's ability to work with solar panels and batteries, which are not proprietary to the manufacturer.

4.2   Available Solutions

To ensure a more objective comparison, this study will only evaluate solar battery chargers with the MPPT algorithm. Other system components (e.g., solar panel, or battery) will only be considered if the product requires proprietary components. Chargers capable of utilizing a solar panel of 400 W or more will be selected for comparison. If the manufacturer offers multiple models graded according to maximum power, the cheapest model that meets the minimum power requirement will be chosen. The charger must also be capable of charging lead-acid or lithium batteries. For this study, various solutions from different market categories have been selected. One representative was chosen for each category. Selected categories:

1. The cheapest possible device meeting the requirements of the model situation without remote monitoring.

2. Mid-range price category, the main requirement being the capability of remote monitoring.

3. Higher price category, possibility of remote monitoring without the need for own inventions, high efficiency.

4. Complete system including battery.

◼ 4.3 Selected Products for Comparison

◼ 4.3.1 Lowest Price Category

As a representative, a module from the manufacturer CARSPA was chosen, with the product name MPPT12/24-20-20A [13]. At the time of writing, it had a price of 1590 CZK including VAT on the Czech market [14]. Besides the basic functionality of an MPPT charger, this product does not offer any additional features. This is most evident in the inability to remotely analyze data, as this model does not have an external communication bus. Also, configuration via a button menu will be more challenging than with the help of a bluetooth application in more expensive solutions. Finally, this charger does not support lithium batteries, only all technologies of lead-acid batteries.

◼ 4.3.2 Mid-range Price Category

For this category, the primary specification was the requirement for the possibility of remote monitoring. Therefore, the EPEVER XTRA2210N unit [15] was chosen, which has an output through the RS485 bus with the Modbus protocol. There are several open-source projects [16] capable of communicating with this unit and transferring data to any cloud or Smart Home solution. The price of the unit alone is 1800 CZK [17] at the time of writing. The price of the RS485 adapter is in the order of lower hundreds of crowns.

◼ 4.3.3 Higher Price Category

The representative of this category was chosen to be the Dutch company Victron Energy and their product SmartSolar 75/15 [18] for a price of 2900 CZK [19], at the time of writing. Unlike the previous categories, Victron Energy is a recognized brand in the solar charging market. Their products are considered quality, and their specifications can be trusted. The SmartSolar 75/15 solar charger offers the possibility of connection via Bluetooth, which enables easy system configuration through applications and its potential monitoring. For remote monitoring of data, it is possible to use one of the open-source projects [20], which transfers data from the serial port to the device. Alternatively, an official 4G modem [21] can be purchased for approximately 5000 CZK including connection plan for 5 years, which allows remote monitoring even in places without internet connection. Another option is to

use the LoraWan communication module [22] for around 3500 CZK (does not include the cost of network connection). A bonus feature of this charger can be considered the algorithm for optimizing battery life, which activates in case the battery is not charged to its maximum value within one day.

4.3.4 Complete System Including Battery, but Without Solar Panels

As an interesting alternative to solar chargers, a complete system from EcoFlow (or similar solutions from other companies) can be chosen. The Delta 2 product [23] integrates a 1 kWh LFP battery, an MPPT charger with a maximum power of 500 Wp, a 230 V mains voltage inverter with a power of 1800 W and various outputs for charging electronic devices. It also supports Wifi connection and subsequent remote monitoring through an app, or integration into smart home systems. A great advantage of this product is that the entire system (except solar panels) is integrated and does not require additional components, or wiring. This makes the solution safer, as there is no situation where the user might miswire the circuit and thereby cause a fire, or explosion. If necessary, the battery capacity can be expanded by up to an additional 2 kWh by purchasing external battery modules. The main disadvantage of this system is its high price. The base Delta 2 unit costs 22 500 CZK [24], the expansion battery module 1 kWh costs 16500 CZK [25]. For comparison, 1 kWh of LFP battery cells costs approximately 4000 CZK [26]. Other disadvantages include low flexibility of the solution. If the user requires more than 3kWh of battery capacity, they must purchase a more expensive model, which also offers greater power for solar charging. The same applies in the opposite situation.

■ 4.4 Comparison

From the comparative table 1, significant differences in device prices are evident, so it is not possible to unambiguously recommend only one device for all customers looking for a solution for their cabin or recreational vehicle. If the customer does not require remote monitoring of the system state, the device from EPEVER seems like the best choice, which is only slightly more expensive than the representative of the cheapest category but offers additional configuration possibilities via bluetooth, or the ability to work with lithium batteries.

If the customer needs remote monitoring and is willing to invest their own time into implementing a data bus converter of the solar charger to the internet, the EPEVER solution still appears to be a better choice, as it is 1000 CZK cheaper than the SmartSolar system from Victron Energy. The motivation to invest in the Victron Energy system could be the fact that Victron is a more trustworthy company, and their system is easily expandable with additional modules in the future.

A customer who wants to purchase a solution requiring no complex implementation will be deciding between products from Victron and EcoFlow. If they have experience with wiring electronic devices, the solution from Victron Energy is more advantageous, as it has much greater flexibility, moreover, it allows for remote monitoring via LTE, which can be an advantage, for example, in a recreational vehicle, which often has as the only source of internet a mobile hotspot on the phones of the occupants.

The EcoFlow solution is most suitable for temporary installations. With the investment in portable solar panels, the entire system can be assembled within a few minutes. At the same time, it does not require any additional connected devices for the customer to start drawing energy from the batteries (inverter, or for example, a phone charger), which was necessary with the previous systems.

(a) : CARSPA MPPT12/24-20-20A

(b) : EPEVER XTRA2210N

(c) : Victron SmartSolar 75/15

(d) : EcoFlow Delta 2

Figure 4.1: Selected category representatives

Table 1: Comparative table

device	CARSPA	EPEVER	Victron	EcoFlow
price [CZK]	1590	1809 +	2900 ++	22 500
internet connection	✗	✓	✓	✓
maximum power [W]	480	520	440	500
maximum input voltage [V]	65	100	75	150
maximum charging current[A]	20	20	15	15
support for lead-acid batteries	✓	✓	✓	✗
support for lithium batteries	✗	✓	✓	✓
Battery life optimization	✗	✗	✓	✓
Bluetooth connectivity	✗	✓	✓	✗

13

Chapter 5

Device definition

Market research indicates that there is no affordable MPPT solar charger available, that can integrate with Smart Home servers or cloud services. Numerous third-party initiatives have been developed to connect MPPT chargers to the internet by translating data from the serial or RS485 bus of the devices. Victron Energy's solutions, priced above 6000 CZK, provide the necessary integrations, but they utilize mobile networks or LoRa rather than a local server on the same WiFi.

It is clear that all the leading manufacturers of MPPT chargers use efficient designs. Therefore, the outcome of this project should also meet this standard of efficiency to remain competitive. For this reason, an MPPT charging controller integrated circuit from a reputable manufacturer was chosen for its high power conversion efficiency and established designs, instead of using the microcontroller to also handle the MPPT charging. Additionally, the microcontroller included on the device must be capable of WiFi connectivity to facilitate integration with Smart Home systems. An on-device status display and user interface was considered a feature worth the additional cost, in order to provide a backup to the network-based solution. The price of the resulting solution should be less than what is needed to set up the Victron Energy network connectivity. The ease of use should be better than a non-smart MPPT charger with a third-party networking adapter.

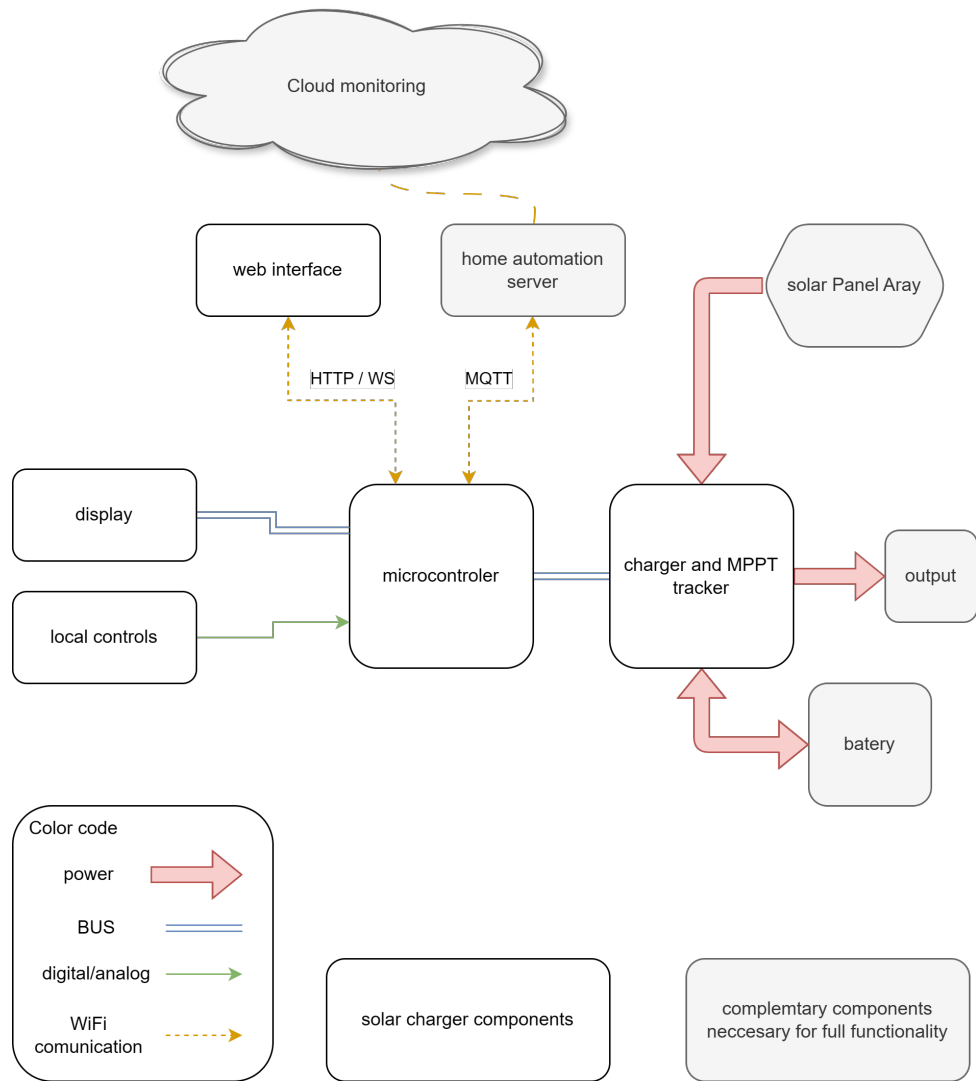The block diagram of the resulting device can be found on figure 5.1.

Figure 5.1: device block diagram

Chapter 6

Charge controller

Selecting the appropriate charge controller is critical for this project and is at the heart of the product. The extensive selection criteria, especially the requirement for compatibility with lead-acid chemistries that are often not included in newer products, limited the available options. Research revealed that the most viable option was the LTC4015 from Analog Devices. This controller supports LFP, Li-Ion, and Lead Acid batteries, incorporates the MPPT algorithm, and allows for monitoring and configuration via I2C compatible SMBus [4]. However, it has a limitation: a 35 V input voltage ceiling (40 V absolute maximum). This limitation impacts the types of solar panels that can be used with the finished charger, as the no-load voltage of solar panels in very bright conditions can be significantly higher than the maximum power point voltage under moderate lighting. As a result, this limits the battery voltage options for the system, since the maximum power point of the solar panel needs to be at a voltage higher than the maximum charge voltage of the battery to use the panel more efficiently.

In low-voltage grid power systems, this device typically operates with either a 12 V or 24 V nominal battery voltage. The 12V option is more prevalent because it enables the straightforward operation of appliances intended for personal vehicles. Employing a 12V battery would create a wide enough range of usable maximum power point voltages that would enable the most efficient extraction of energy from the solar panel during all conditions.

6.1 Complementary components selection

The LTC4015 datasheet [27] offers detailed instructions for selecting all necessary complementary components. Choosing the buck converter components and the ideal diodes MOSFET required particular focus. The goal of utilizing the entire spectrum of input voltages up to 35 V implies that the selected current handling capacity of the S.M.S.C. will determine the choice of components.

The LTC4015 controls the maximum current by monitoring the voltage across two current shunt resistors, one measuring the battery current and the second measuring the input current. The LTC4015 functions in such a manner that the voltage differential across the associated shunt resistors does not

surpass 32 mV. It should be noted that the LTC4015 is unable to control the output current draw directly from the voltage source. It attempts to decrease the input current by limiting the currents used for battery charging, but can only reduce this charging current to zero. Consequently, an additional current limiting device such as a fuse is necessary at the output. The LTC4015 has an adjustment resolution for the input current limit of 5 bits (64 steps) and 4 bits (32 steps) for the battery current limit.

For the input current measurement, a resistor value of 1mΩ has been chosen, and for the battery current measurement, a 2mΩ resistor is used. This configuration establishes current limits of 32 A and 16 A, respectively, giving each an adjustment resolution of 0.5 A.

With the current limits known, the rest of the power circuitry components can be selected. The LTC4015 utilizes a synchronous buck converter architecture. That generally achieves superior conversion efficiency with high loads than asynchronous buck converters [28]. Synchronous buck converters utilize 2 MOSFET transistors to switch the current. These two transistors should be placed as close to each other as possible to reduce the size of the high-frequency current loop. A dual transistor package was chosen for this purpose. Other key transistor parameters were considered when selecting the exact part. These parameters include:

- **On resistance at 5 V** The transistor will be driven by a 5V power suply, this means that a logic level transistor is required with the lowest on resistance possible at 5 V.

- **Maximum drain-source voltage** The maximum drain to source voltage needs to be higher than the input voltage due to the potential for voltage spikes during switching operation. A value of at least 60 V for the maximum drain-source voltage was chosen, which is 20 V higher than the input voltage limit of the LTC4015.

- **Maximum power** The maximum power dissipated by the package needs to be high enough to enable continuous operation at maximum charging current. The LTC4015 datasheet provides formulas (1)(2)(3) for calculating conductance losses and transition loses (transition losses occur only in the top transistor) which account for the majority of heat generation of the transistor package. (1)(2)(3).

$$P_{\text{COND(TOP)}} = \left(\frac{V_{\text{BAT}}}{V_{\text{SYS}}} \cdot I_{\text{CHG(MAX)}}^2 \right) \cdot R_{\text{DS}}(\text{ON}) \tag{1}$$

$$P_{\text{COND(BOT)}} = \left(1 - \frac{V_{\text{BAT}}}{V_{\text{SYS}}} \right) \cdot I_{\text{CHG(MAX)}}^2 \cdot R_{\text{DS}}(\text{ON}) \tag{2}$$

$$P_{\text{TRAN}} \approx 0.5 \cdot V_{\text{SYS}}^2 \cdot I_{\text{CHG(MAX)}} \cdot C_{\text{RSS}} \cdot f_{\text{SW}} \tag{3}$$

The selection of a fast-switching transistor is critical. The DMTH6010LPDQ package from Diodes Incorporated was chosen, which includes two N-channel MOSFETs with a very low on-resistance of $16\,\text{mΩ}$ and a maximum drain to source voltage of 60 V per transistor. When calculated using the maximum

18

power values the circuit would experience, the entire transistor package would produce about 2 W of heat. According to the datasheet, the package can dissipate 2.8 W when mounted on a 1-inch square patch of 2 oz copper. The PCB, however, will be produced with a 1 oz copper thickness and a slightly smaller mounting area. Nevertheless, by reducing the charging current to 12 A instead of 16 A, the generated power will be just slightly greater than 1 W, which should be manageable with the existing transistor mounting setup.

The charger designs also requires 2 aditional transistor that act as ideal diodes. MOSFET, one N-chanel and one P-chanel. The ideal diode transistors are not switched at high speed, so the only key parameters are the on resistance at logic voltages, and the total power dissipation limits for the package. For the P-chanel MOSFET, the DMP3010LPSQ [29] from diodes incorporated was selected and the NTMFS5C670NLT1G [30] from onsemi was selected as the N-channel MOSFET. both provide sufficient current handling capabilities and can resist the voltages used in the S.M.S.C.

Achieving optimal performance requires the correct selection of an inductor. The LTC4015 datasheet [27] provides a formula (4) to aid in determining the appropriate inductance. The datasheet recommends a switching frequency of 500 kHz as the optimal balance between the size of the inductor and the switching losses in MOSFETs, which has been adopted in this design. Although the prototype supports a charging current of 16 A, where the formula recommends a 4μH inductor for a 12 V battery voltage and 35 V input voltage, it will primarily be used with smaller currents during the experimental and development phase. Therefore, a 10μH inductor was chosen. The second key parameter for selecting an inductor is the saturation current, which is recommended to be at least 60% higher than $I_{\text{CHG(MAX)}}$. The PCB design aims to be as adaptable as possible, yet the inductor package footprints are not standardized. Consequently, a variety of inductors sharing the same footprint were chosen to cover a broad spectrum of parameters ($V_{\text{BAT}}, V_{\text{IN(MAX)}}, I_{\text{CHG(MAX)}}$). The XAL1510 series from Coilcraft [31] was selected, offering an inductance range from 4.7μH to 33μH, with saturation currents ranging from 16.7 A for the 33μH model to 39 A for the 4.7μH model, which should suffice for most configurations of this device. Additionally, an optimal inductance calculation spreadsheet was developed, available in the project's repository [32], to facilitate the selection of inductors.

$$L = \frac{V_{\text{BAT}} \cdot \left(1 - \frac{V_{\text{BAT}}}{V_{\text{IN(MAX)}}}\right)}{0.25 \cdot f \cdot I_{\text{CHG(MAX)}}} \tag{4}$$

6.2 Battery protection

LTC4015 provides the necessary safety features for a solar charger, which include basic battery protection against charging overcurrent, overvoltage, and overtemperature, if the battery is equipped with an NTC thermistor. It

does not however provide undervoltage protection and discharge overcurrent protection. These functions are usually handled by the battery management system, which is required for lithium batteries. However, for lead-acid batteries using a BMS is not a standard practice, and for this reason these functions must be implemented in the S.M.S.C. The battery output overcurrent protection and short circuit protection is provided by the output fuse which will quickly disconnect power output in case of a fault. Implementing battery undervoltage protection requires the ability to disconnect the output load, for this reason one of the GPIO pins of the ESP32-C6 is provided along with 3.3 V power lane and ground connection in a 3 pin header on the charger PCB. This header can be used in combination with a 3.3 V compatible MOS-FET module [33] to disable the output if the user requires this functionality. Undervoltage protection can be enabled in the ESP32-C6 source code, where the user can also adjust the output disconnect voltage levels for each type of battery chemistry. The under voltage protection algorithm will also disable all wireless communication if the voltage level drops 100 mV below the output disconnection level in order to reduce the S.M.S.C power consumption.

■ 6.3 Hardware design

To minimize the potential for design mistakes, the reference application schematic outlined in the datasheet [27] has been closely followed.

The LTC4015 comes with five configuration pins that are utilized to set the type of battery chemistry and the number of battery cells. Each pin can be adjusted to three possible states: connected to ground (LOW), connected to the internal 5 v power supply (HIGH), or left floating (high impedance state). To simplify configuration changes, each pin is equipped with a three-pin header with a spacing of 2.54 mm. The middle pin of each header is connected to the LTC4015, the lower pin to the ground, and the upper pin to the internal power supply of the LTC4015. In addition, a current-limiting

Figure 6.1: configuration headers, with jumpers

resistor has been incorporated in series with the 5 V pins to prevent power supply overloads during a short circuit. Typically, device configuration with headers involves male pins on the device and a female jumper to bridge two pins, which also protects the pins from short circuits caused by debris contact. However, the LTC4015 might require some pins to remain in a high impedance state, leaving all three pins exposed. As a result, the decision was made to use three-pin female headers (Figure 6.1) and employ short wire segments as jumpers, effectively preventing short circuits. The pin configuration reference table was printed on the silkscreen next to the connectors to assist in correctly configuring the pins.
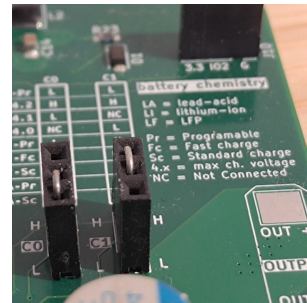
6.4 PCB design

A 4-layer printed circuit board was selected for this project it provides many benefits and the aditional manufacturing cost is usualy less then 50%. To minimize assembly costs, components were placed only on the top layer, this layer was used for routing connections whenever possible. The second layer served as a ground plane, the third as a power plane, and the fourth (bottom) layer functioned as an additional signal connec-

Figure 6.2: Kelvin connection

tion plane. Considerable care was taken to position the buck converter's components as closely together as possible (figure 6.3). Numerous vias were required to achieve the necessary current carrying capacity for the main circuits. Since a via cuts through all layers it passes through, it was crucial to ensure that the current paths on different layers were not obstructed by the via groups cutouts. In order to broaden the range of available parts and minimize the cost of components, 2 pad shunt resistor packages were used in the 2512 dimensions standard. To improve the accuracy of the current measurement, a Kelvin connection [34] to the resistor pads was utilized (figure 6.2)

6.5 Software Implementation

Analog Devices offers a universal C library and register definitions. For integration with the ESP-IDF, creating a hardware abstraction layer (HAL) was necessary `ltc4015_lib`. This layer is required to perform read or write operations on the I2C bus. The LTC4015 device uses a 7-bit I2C [2] slave address of 0x68 and features 16-bit wide registers for data storage, indexed by an 8-bit wide subaddress (command code). All communications must adhere to the specified protocol diagram (figure 6.4).

```c
//buffer for data | b[0] = low byte b[1] = high byte
//first byte recieved from LTC4015 is low byte
uint8_t buffer[2] = {0};
//initialize i2c message
i2c_cmd_handle_t cmd = i2c_cmd_link_create();
//send stat condition
i2c_master_start(cmd);
//send LTC4015 address
i2c_master_write_byte(cmd, (addr << 1) | I2C_MASTER_WRITE, true);
//send command code (register address)
i2c_master_write_byte(cmd, command_code, true);
//send start condition
i2c_master_start(cmd);
//send LTC4015 address
i2c_master_write_byte(cmd, (addr << 1) | I2C_MASTER_READ, true);
// Read 16-bit data | read 2 bytes and nack the last byte
i2c_master_read(cmd, buffer, 2, I2C_MASTER_LAST_NACK);
//send stop condition
i2c_master_stop(cmd);

//send the command (timeout after 1000ms)
esp_err_t ret = i2c_master_cmd_begin(i2c_num, cmd, 1000/portTICK_PERIOD_MS);
//clean up
i2c_cmd_link_delete(cmd);

//check if message was sent and acked by LTC4015
if (ret != ESP_OK) {
    //print the return reason
    ESP_LOGE(tag, "i2c_master_cmd_begin failed: %s", esp_err_to_name(ret));
    return -1;
}
```

Listing 1: part of a HAL function for reading LTC4015 data with I2C

The ESP32-C6 needs to access two types of data: configuration data, which remains constant during device operation such as the battery cell count, and runtime data, which needs regular updates. All measurable data, collected by the LTC4015 analog-digital converter, is organized into a C `ltc_runtime_data_t` struct. This struct is then used to distribute the data throughout the program. Additionally, several utility functions for handling runtime data are incorporated into the custom component. Below are some examples.

- `ltc_create_test_runtime_data_c` Creates random testing data using the true random generator in ESP-IDF, it can be configured to create data larger than expected output of the LTC4015, to test overflow protection of other helper functions.

- `ltc_create_csv_line_c` Creates a comma separated values (csv) line string with a constant length of exactly 48 bytes (including the newline

character), that uses hexadecimal encoding for higher density. There is also an inverse function which will decode the csv line and insert the data back to the struct.

- **mathematical operations** Functions have been developed to assist in calculating averages from multiple measurements, such as an addition function that totals the values for each element in the struct and stores it within one of the structs. The structs contain a unix timestamp [35] that cannot be combined, so the most recent timestamp is selected for the output data. The runtime data structure accommodates data in 32-bit integers, while the LTC4015 values are confined to 16 bits, guaranteeing that additions are performed safely without the danger of overflow.

The LTC4015 analog to digital converter is not enabled when a charging power supply is not connected, in order to conserve battery power. However, it can be forced to run by issuing a command to the appropriate register. For this purpose, a function named `ltc_force_meas` has been added, which will enable the measurement system. Once the system is enabled, it takes approximately 10 ms to perform the measurement.
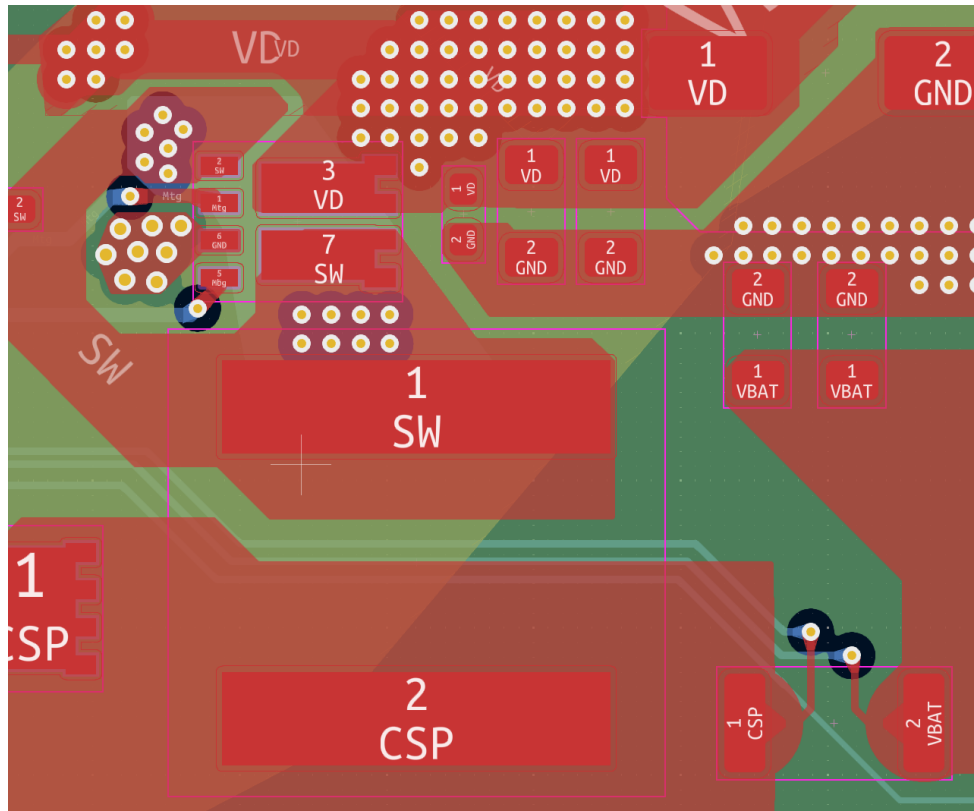
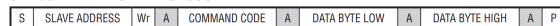**Figure 6.3:** buck converter components layout

Figure 6.4: LTC4015 I2C protocol

Chapter 7

Microcontroler

Selecting a proper microcontroler is crucial for this project. As it will turn the solar charger into a smart solar charger.

7.1 Selection

- **connectivity** The device needs to establish a WiFi connection for communication purposes and to set up a WiFi access point that hosts an initial configuration website. Additionally, the microcontroller must support I2C and SPI protocols to connect to the LTC4015 and the display driver, respectively.

- **power consumption** Increased energy consumption by the microcontroller reduces the device's overall efficiency. However, this issue diminishes as the energy output from the solar panel increases, given that the microcontroller's energy usage remains unchanged and the conversion losses scale with the power produced. In the Czech Republic, a typically positioned solar panel can produce daily between one to three [36] times its maximum power capacity in watt-hours (e.g., a 100 Wp panel might yield 100 Wh to 300 Wh each day).

- **development environment** Given that the project's outcome will be open source, it is advantageous for the development environment to be well-recognized within the IoT community. A popular development environment can simplify programming due to the availability of extensive resources.

Following a review of these factors, ESP32 family from Espressif has been selected the as the best option, primarily for its wireless features, affordability, and strong IoT development platform 'ESP-IDF' [37] based on a freeRTOS [1] (realtime operating system). Espressif offers two main ESP32 product lines. The S series employs a 32 bit Tensilica Xtensa 7 processor architecture with a maximum core clock speed of 240 MHz and is the successor to the original ESP32. Conversely, the C series adopts a 32 bit RISC-V architecture with maximum a clock speed of 160 MHz. The 'S' series is generally considered more powerful, featuring additional GPIO pins and PSRAM support. Given

25

the universal ESP-IDF, the specific processor architecture has minimal impact on the development of ESP32 applications. The decision was between the ESP32-S3 [38] and the ESP32-C6 [39] from these product lines. The ESP32-C6 offers superior connectivity features, including support for the WiFi 6 standard (limited to 2.4 GHz) and the IEEE 802.15.4 standard, essential for ZigBee and Thread protocols. Nonetheless, it features a single main core (alongside a secondary low-power core) and fewer pins than the ESP32-S3, which is equipped with dual cores that provide slightly superior performance per core and compatibility with WiFi 4. Dual core support is crucial in certain projects because it enables dedicating one core to the application and the other to managing WiFi communications and system operations. Ultimately, ESP32-C6 was selected, as it is possible for the application to operate on the same core as the WiFi tasks when the application timing is not critical and avoids long-duration interrupt routines. Furthermore, the ESP32-C6 provides sufficient pins to accommodate all the peripherals required for this project.

Like many Espressif microcontrollers, the ESP32-C6 is available either as a standalone SoC or as a component of a module. This module includes an external flash chip, a 2.4 GHz antenna, and other necessary components, and comes with CE and FCC certifications. The cost difference between the SoC and the module versions is minimal. The primary advantage of opting for the SoC over the module for low volume products is space savings; a custom-designed SoC circuit may fit in spaces too small for the module.

The smart solar charger would not benefit greatly from being slightly smaller; thus. The decision was made to use the ESP32-C6 module. Espressif offers 2 module types a "WROOM" or a "MINI", it is also possible to select both modules with an antenna connector instead of the integrated antenna.

The WROOM module with an integrated antenna has been selected. Unlike the MINI series of modules, the WROOM can be hand soldered if necessary and exposes 2 more GPIO pins than the MINI.

▉ 7.2 Hardware implementation

To minimize the likelihood of errors, adherence to the module peripheral reference design [40] was maintained. The module integrates nearly all necessary supporting circuits, requiring only a minimal number of additional components for operation. The power supply circuitry requires careful consideration, as the module may consume over 400 mA of peak current during WiFi transmission and when powering peripheral circuits at the same time. This requires a strategic placement of filtering capacitors near the module. Additionally, the design includes a USB interface, which needs to be routed as a differential pair signal; however, it does not require impedance matching circuitry [40].

The microcontroller utilizes certain GPIO pins as strapping pins to set the behavior of the ESP32-C6 during power-up. The strapping pin GPIO9 activates the flashing mode, which must be accessible to the user to initiate the flashing process as required. GPIO8 is responsible for managing the boot

option and needs to be in the HIGH state during flashing mode activation. Additionally, GPIO8 handles the printing of ROM code at startup, but this function can also be managed through eFuse bits, which means it can be continuously set to logic HIGH. GPIO15 determines the source of the JTAG signal. Since external JTAG debugters are not required, GPIO15 is configured to HIGH, enabling USB to be the JTAG source. Furthermore, the EN pin, which activates the processor when set to HIGH, functions as a reset button. The reference design from the ESP32-C6 datasheet is utilized here [40]. The full schematic of the ESP32-C6 support circuit is depicted in figure 7.1

7.3 Development enviroment

The primary development tool was the Espressif IoT development framework (ESP-IDF), which employs the C programming language and is built on freeRTOS for event-driven programming. This approach streamlines the workflow by eliminating the need for a complex interrupt handling system. The developed code is up-to-date with the latest version 5.2. For additional functionalities, such as custom flash memory partition tables, adjustments in the project configuration were required. The sdkconfig file from the project's repository [32] is essential for successful code compilation. The web user interface was created using Vite and React, incorporating npm packages such as Bootstrap and Chart.js. Detailed information on all packages and their versions required for the website's construction is available in the repository's readme.md file.

7.4 Flash storage

Ensuring data retention after a microcontroller reboot necessitates storing the data in the module's flash storage. For compact data types like WiFi credentials or display setting indices, ESP-IDF provides a non-volatile storage (nvs) library [41] that uses key-value pairs for data storage. However, this method is not suitable for large data sets, such as the web interface. For these purposes, ESP-IDF supports the FAT filesystem [42]. This requires setting up a custom FAT partition [43] in the flash memory. Two partitions have been established: a read-only partition for the web user interface and a writable partition for storing runtime data logs. The latter, being the final partition, can be expanded, allowing for the storage of many more days worth of data if a module with more than 4 MB of flash storage is used.

7.5 WiFi connectivity

The initial example of WiFi connectivity from the LearnESP32.com course served as the foundation for developing the WiFi features. Additional functionalities such as nvs data storage, reconnection timeout, and WiFi mode

```
# Name,    Type, SubType, Offset,   Size, Flags
nvs,       data, nvs,     ,         0x6000,
phy_init, data, phy,      ,         0x1000,
factory,   app,  factory, ,         1700K,
website,   data, fat,     ,         700K,
storage,   data, fat,     ,         1M,
```

Listing 2: custom partition table used with 4 MB modules

switching were incorporated subsequently. Presently, the WiFi operates according to the flow chart (figure 7.2). Initially, the ESP32-C6 lacks any stored credentials for a station in its flash memory, triggering the activation of AP mode where the default SSID and password are used if absent in flash. AP mode supports connection for up to 4 devices, and the ESP32-C6 runs the same web server as it does when connected to WiFi. This server enables users to configure STA a connection credentials, access point credentials, and MQTT broker connection details via HTTP requests. Upon receiving a request to update the STA password, the credentials are stored in nvs and the same operational loop that runs at initial boot is triggered. It attempts to connect to the station five times with brief intervals; if unsuccessful, it reactivates the AP mode but retains the credentials in nvs, allowing the connection process to be restarted via on-device controls. Successful connection to a station initiates the MQTT client. If the WiFi connection is lost, the system will attempt to reconnect five times before enabling the AP mode.
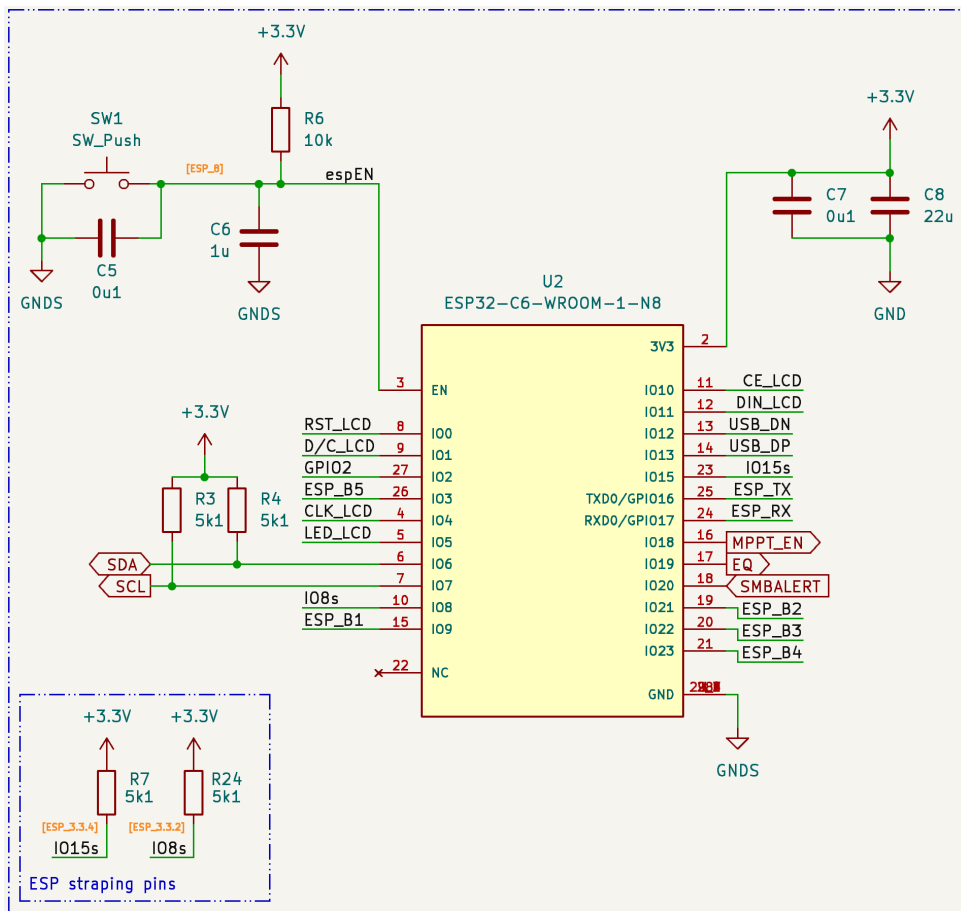
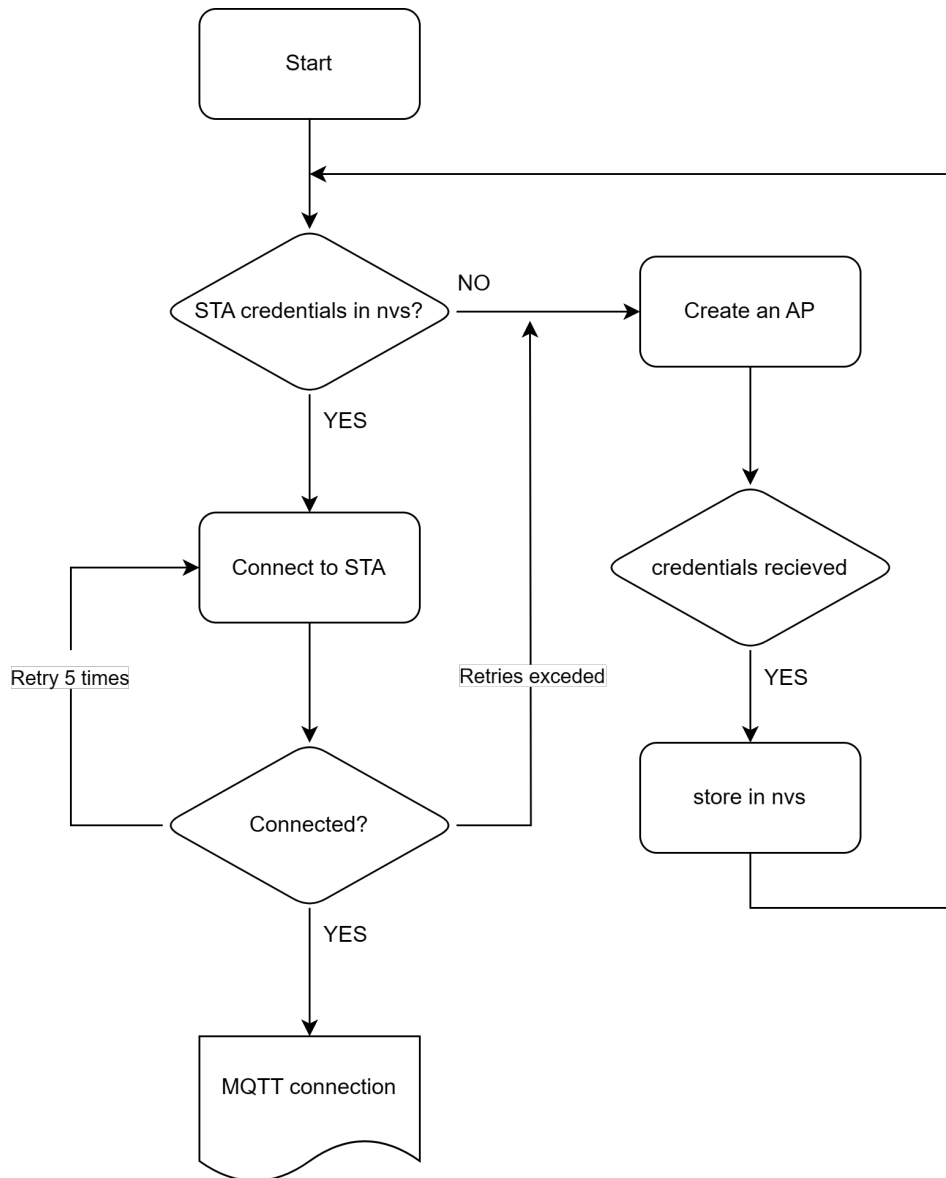**Figure 7.1:** ESP32-C6 and supporting circuits

Figure 7.2: Network connection logic sequence

Chapter 8

Display

The selection of an optimal display for the solar charger was guided by three principal objectives, ensuring that the device meets both functional and economic criteria:

- Readability: Essential for user interaction, the display must be legible under various lighting conditions. Although deployment in direct sunlight is not anticipated, it should maintain clarity in both dimly lit and brightly illuminated indoor settings.

- Power Consumption: The efficiency of the solar charger is paramount. Therefore, the selected display should minimally impact the device's overall power consumption and the standby discharge rate of the battery.

- Cost: To remain competitive, the project aims to balance quality with affordability. This approach aligns with market trends that favor cost-effective display solutions or eliminate them to reduce expenses.

Based on these criteria, three types of displays were considered as viable options:

- E-paper displays are distinguished by their paper-like appearance and readability in bright conditions, utilizing electrically charged particles suspended in liquid filled capsules to simulate ink on paper. This technology offers excellent viewing angles and high contrast, ideal for external lighting. The primary drawbacks include its lower performance in dim environments, requiring external light sources for visibility, and its cost, which can be significantly higher than alternative technologies. Despite e-paper displays' slow refresh rates, this limitation is inconsequential for the device due to the static nature of the displayed content.

- OLED monochrome screens, known for their energy efficiency compared to backlit LCD panels, illuminate individual pixels as needed, contributing to their near-infinite contrast ratios. These screens are cost-effective and provide sufficient viewing angles for readability. While their visibility in bright environments depends on the maximum brightness, they remain suitable for most indoor applications. However, their operational power consumption is higher than that of unilluminated display technologies, which poses a consideration for energy-sensitive projects.

■ Transflective and Reflective LCDs leverage ambient light for visibility, reducing power consumption. Transflective models utilize a semi-transparent layer to reflect external light or allow for backlighting in low-light conditions. Reflective LCDs, without backlighting, depend entirely on external lighting similar to E-paper displays. Both technologies offer advantages in terms of energy efficiency but have limitations in contrast and viewing angles compared to E-paper or OLED displays.

Following the evaluation of the selection criteria against the available options, the OLED display emerges as less appropriate for this application compared to the other displays. Interaction with the smart solar charger via the on-device controls will be minimal, hence the display's primary role will be to serve as a status indicator, offering essential information such as battery charge percentage at a quick glance. Consequently, a display that remains on continuously while using minimal power is preferable. The OLED screen, however, has the highest power consumption among the three candidates because it constantly illuminates its pixels.

Both of the final two candidates are characterized by their low energy consumption. However, E-paper displays tend to be more expensive compared to LCD screens. In addition, the availability of numerous supporting libraries for LCD driving integrated circuits simplifies their implementation in ESP-IDF.

8.1   Module selection

Directly interfacing LCD screens with ESP32-C6 is impractical. Consequently, a combined LCD and driver module is essential. My choice is the PCD8544 driven module that features a 1.6" 48x84 pixels LCD display. This module is cost-effective and widely available at approximately 100 CZK. It is commonly referred to as the "Nokia 5110 screen module" because it uses the same driver and a comparable display as the Nokia 5110 phone. Typically, the module includes an LED backlight, most commonly in blue or white colors.
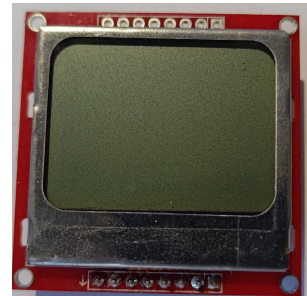
Figure 8.1: PCD8544 display module

8.2   Module implementation in design

The PCD8544 interface expands on the SPI bus architecture. This module requires 6 data pins 2 and operates on a 3. V power supply, identical to that of the ESP32-C6. Connection to the microcontroller can be made directly, with only the light pin (active LOW) needing a current limiting resistor. This resistor was chosen to protect the ESP32-C6 against potential overcurrent

Table 2: PCD8544 module pins

Module pin	ESP32-C6 pin
RST	GPIO0
SCE	GPIO11
D/C	GPIO1
SDIN	GPIO10
SCLK	GPIO4
LIGHT	GPIO5
VCC	3.3 V
GND	GND

faults in the LCD module caused by some manufacturers not using current-limiting resistors for the LED backlight in their module design. A resistor value of 100 Ω was chosen, which restricts the circuit's current to aproximately 13 mA under the worst-case condition of an unprotected red LED serving as the LCD backlight. The ESP32-C6 has a sink current limit of 26 mA [40]. The LCD module generally includes THT holes for a 1x8 pin header, spaced at 2.54 mm. This same hole pattern has been incorporated into the PCB design, which facilitates various assembly possibilities. Wires can be soldered directly to the THT holes for a fixed connection. Alternatively, a detachable connection can be established by soldering a pin header to the THT holes and connecting these headers using jumper wires.

8.3 Module software implementation

The ESP-IDF LCD library does not natively support the PCD8544 driver. However, third-party libraries are available. The U8g2 monochrome graphics library [44] was chosen, which provides hardware-independent support for many different display drivers, including the PCD8544. To use the library with ESP-IDF, a HAL is also required. After including the HAL [45], the U8g2 library provides easy access to many fonts and basic graphical features.

Chapter 9

User Interface

Choosing the right user interface, commonly abbreviated as UI, is crucial for any product. A poor UI design renders the device almost useless. Fortunately, this project is dedicated to developing a smart solar charger, implying that interactions will primarily occur via the internet. Consequently, the on-device interface is designed mainly for brief status checks and changing WiFi modes. The only display element will be the LCD module 8. The backlight on the display can also be used to alert the user by blinking. There are 2 input methods commonly used to interact with non-touchscreen displays.

- **Rotary encoder** Rotary encoder have 3 interactions, turning clockwise, turning anticlockwise and pressing on the encoder. This means that the user easily knows how to interact with this type of input. However, decoding the signals from rotary encoders requires additional steps and the encoder also protrudes from the device, which makes it thicker.

- **Buttons** Buttons can be implemented in various ways as input devices. A user interface based on buttons tends to be user-friendly, particularly when the buttons are clearly marked and avoid complex interactions such as simultaneously long-pressing two buttons to perform a function. Such complex interactions can be circumvented by adding additional buttons. A simple user interface might include four buttons, each assigned a single function: UP, DOWN, SELECT, and BACK. Implementing buttons on the hardware is simple, but it is crucial to effectively debounce signals.

The choice to implement a button-based user interface was mainly influenced by its straightforward implementation, but the reduced thickness of the final product was also a significant factor. A mix of software and hardware debouncing methods will be employed to ensure seamless functionality without any operational issues.

9.1 User interface design

The on-screen user interface will consist of 2 parts:

- **On-glance information** When the device is idle, it will show important information on the LCD screen with the backlight off. Given the limited

display size, only one large value will be shown alongside compact status indicators that reveal the connectivity status of the Wifi and MQTT. This display content will be refreshed periodically. Users have the option to choose which main value is displayed from the runtime measurement data.

▪ **Menu configuration** Upon pressing the SELECT button, a straightforward menu-driven interface emerges. This interface presents additional information and facilitates basic setting adjustments, primarily to initiate connectivity and modify the on-glance display settings. The custom-created software for the menu screen supports a three-tier menu structure with up to five options on a single menu. Each option can either lead to a lower tier or perform a specific function when the SELECT button is pressed. The diagram of how the menu is structured can be seen in figure 9.1

▍ 9.2   UI software implementation

The functions and algorithms managing the menu system were designed to be highly adaptable. Each menu screen is stored in an array of rows defined by `pcd_menu_item_t` structures, which contain three variables: the text shown on the screen, a pointer to a function that executes when the menu is selected, and a pointer to a child menu. These arrays can hold up to five objects, with `pcd_menu_item_t` filled with NULL marking the end of the menu array. The input handling task then triggers the menu drawing task to render the menu after processing the input. Global variables keep track of the current state of the menu, such as the current row index and the parent menu of the currently selected menu. When the select button is pressed, the function for the current row is executed unless a NULL pointer is present, in which case the child menu is selected and displayed. If a menu item activates functionality that requires continuous updates, such as charging state monitoring, a new task is created to manage updates and screen rendering. This task is terminated if any buttons are pressed, returning the user to the menu from which the task was initiated. Pressing the back button while the main menu is displayed starts the on-glance display task. To ensure effective data transfer between tasks and indicate its availability, freeRTOS queues and semaphores are utilized.

```c
// item of the menu struct
typedef struct pcd_menu_item_t {
    char* text;
    void (*action)();
    struct pcd_menu_item_t* submenu;
} pcd_menu_item_t;

//menu example
pcd_menu_item_t pcd_menu[] = {
    {"Charger Status", NULL, charger_status_menu},
    {"Network", NULL, network_menu},
    {"Data Storage", NULL, data_storage_menu},
    {"On-glance info", NULL, on_glance_control_menu},
    {NULL, NULL, NULL}, // End of menu
};
```

Listing 3: menu item definition and the main menu array

9.3 Button hardware implementation

The design of the button circuit ensures that they are in a LOW state when pressed and in a HIGH state when not engaged. This was accomplished using an external pull-up resistor. Additionally, a small ceramic capacitor was placed in parallel with the button connector to form an RC circuit with a time constant of around 1 ms. The device includes five control button circuits (figure 9.3), however, button 1 is linked to the GPIO9 pin on the ESP32-C6, which is also used for controlling the boot mode of the chip. If this pin is LOW at startup, the microcontroller enters a flashing mode to download the program. Consequently, button 1 is excluded from regular use, leaving four buttons for user interface control. When flashing the chip with ESP-IDF, it is not necessary to press the boot button since the flashing mode can be activated externally. The device features 2-pin 2.54 mm spacing male headers (figure 9.4) for button connections, a commonly used button connector, for example to connect desktop computer case buttons to the computers motherboard.

Push buttons

9.4 Button software implementation

Setting up buttons in ESP-IDF is straightforward. Initially, the pins need to be set as input, followed by assigning a freeRTOS ISR to these pins. It is crucial for the ISR to operate swiftly because the ESP32-C6 features only one high performance core, implying that the highest priority ISR might impede the execution of the WiFi or IP stack. To address this, a deferred interrupt architecture was adopted where a single ISR manages all button events. This

37

ISR merely forwards the pressed button's identifier to a queue. Subsequently, a separate low priority task was established, designed to retrieve and process items from the queue and execute actions.

The low priority task is equipped with a basic debouncing filter. It sets a debounce boolean to true upon receiving a button id from the queue, and initiates a freeRTOS timer that resets the debounce bit to false when it expires. While the debounce boolean remains true, the task disregards any further inputs. Through experimentation with different buttons, it was determined that a debounce duration of 100 ms effectively eliminates most bouncing errors. Additionally, it takes around 150 ms to refresh the LCD menu following any action, indicating that the debouncing process is faster than the menu refresh.

9.5   Web user interface

The device provides a web-based user interface. This interface is developed with the React library and the Bootstrap framework. It interacts with the device through HTTP [46] requests and a WebSocket [47] connection to transmit real-time data. Featuring a grid and card layout [48], the interface adapts well to various screen sizes, including mobile devices. It includes three distinct sections.

- Display Section: This section presents real-time data and allows users to activate graphs for each metric, which trace the metrics history from when the websocket connection began.

- Configuration Section: Users can adjust settings for the LTC4015 and network configuration, including MQTT server details, WiFi access credentials, and device access point SSID and password.

- History Section: This section provides a feature for downloading a log of operational data in csv [49] format. It employs a script to transform the compressed csv file sent from the device into a standard csv file, which is then ready for download.
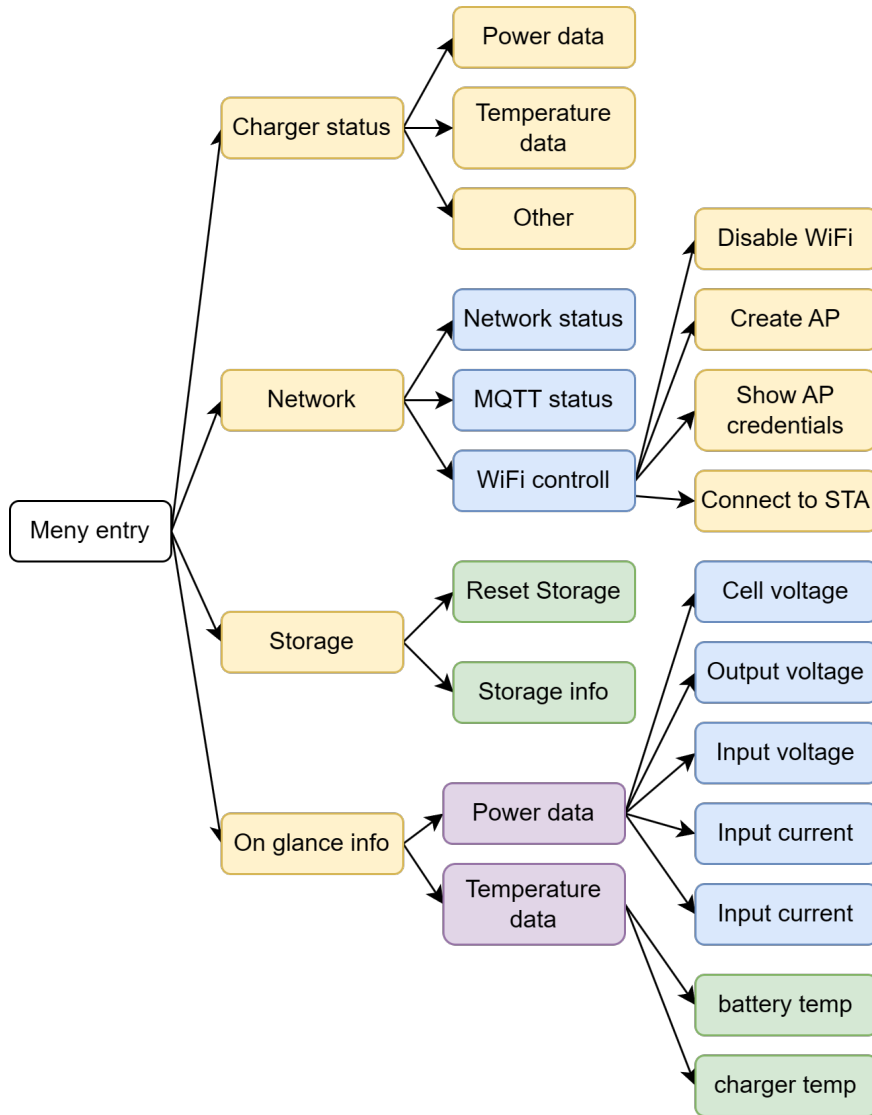
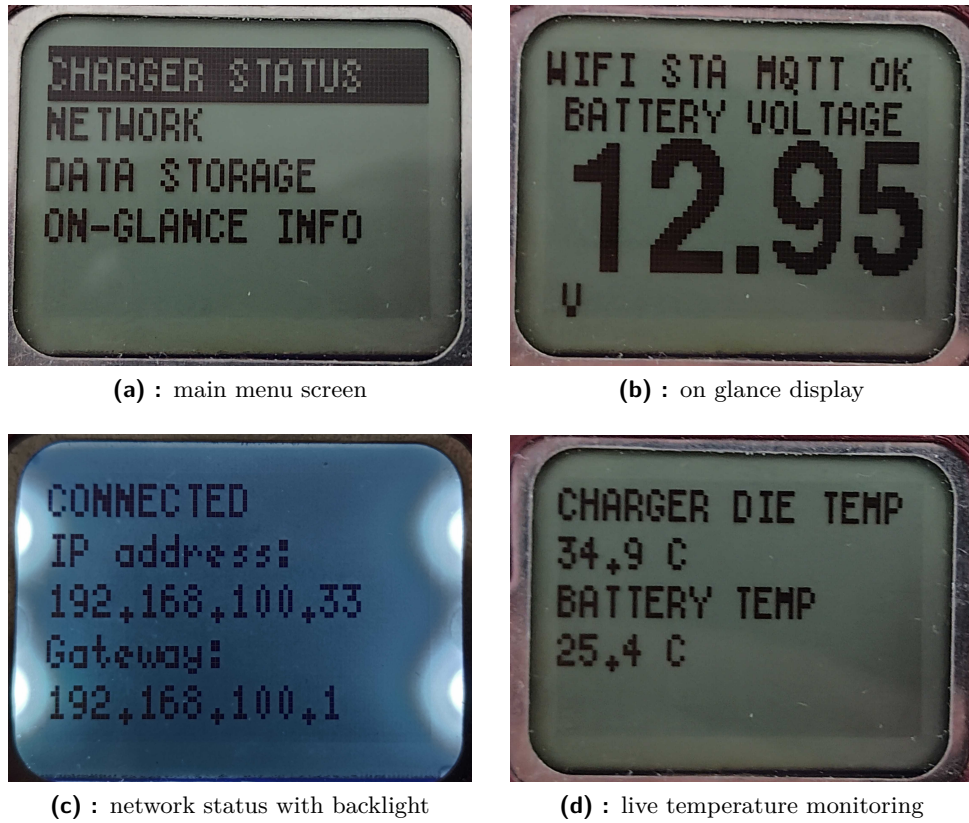**Figure 9.1:** user interface menu structure

(a) : main menu screen

(b) : on glance display

(c) : network status with backlight

(d) : live temperature monitoring

Figure 9.2: On screen user interface examples
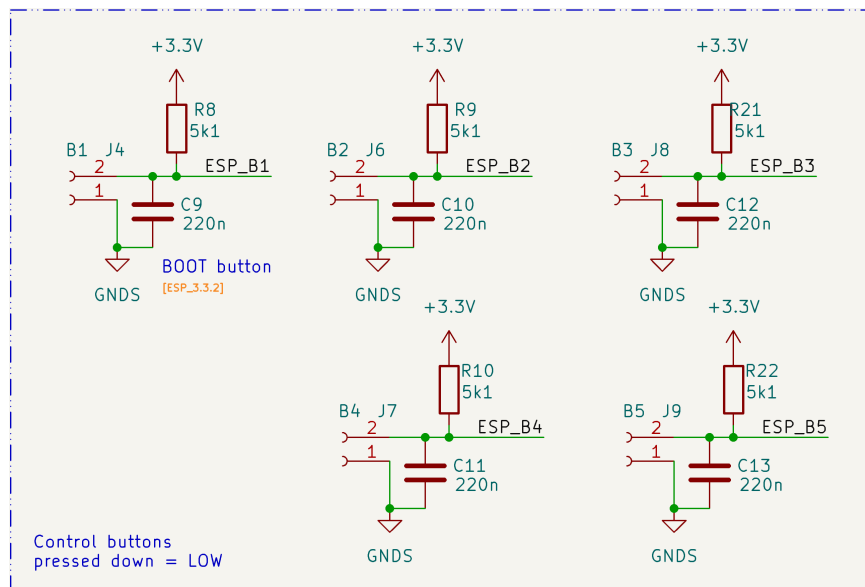

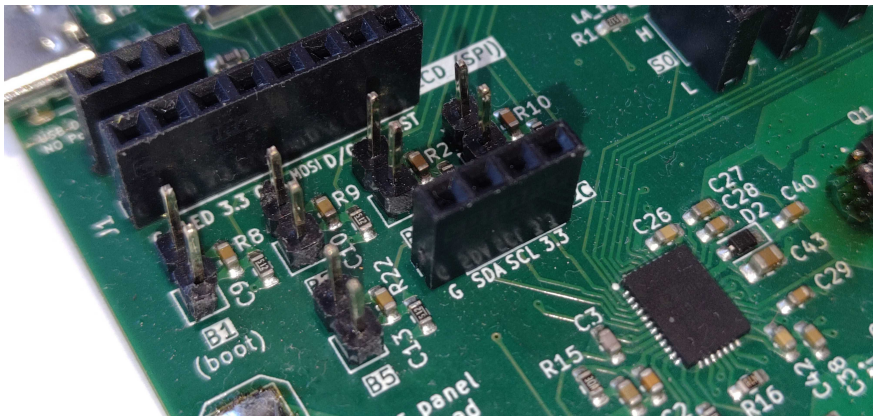**Figure 9.3:** button circuit schematic

40

Figure 9.4: 2.54 mm male connectors for connecting buttons
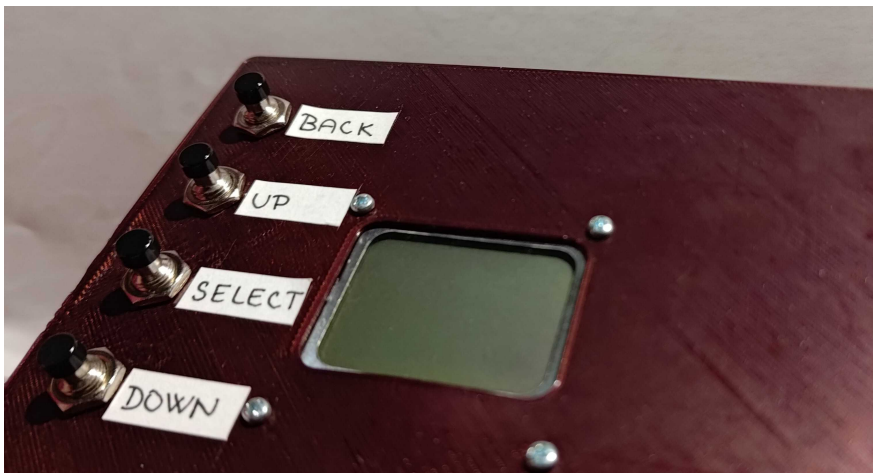
Figure 9.5: user interface buttons with labels
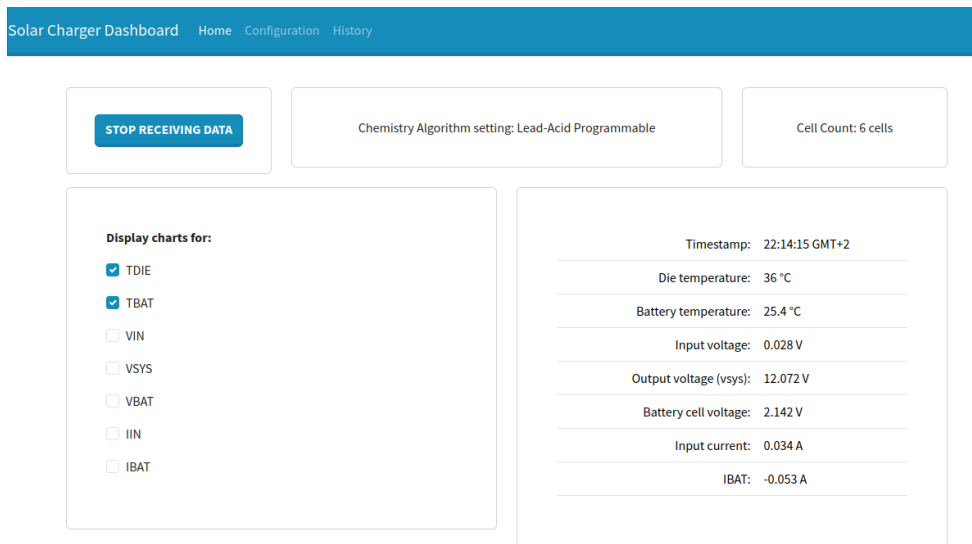
Figure 9.6: web UI | realtime data dashboard

41

Figure 9.7: web UI | configuration section

Figure 9.8: web UI | submitting AP configuration

Figure 9.9: web UI | historical data section before data is ready to be downloaded

Figure 9.10: web UI | historical data available for download

Chapter 10

Voltage Converter

The microcontroller selected for this project requires a stable operating voltage of 3.3 V. Therefore, it is necessary to step down the battery cell voltage to this level. Given that the battery pack voltage will be at least 12 V, only a switching converter is a viable option. A linear voltage regulator would exhibit a conversion loss of more than 50 %.
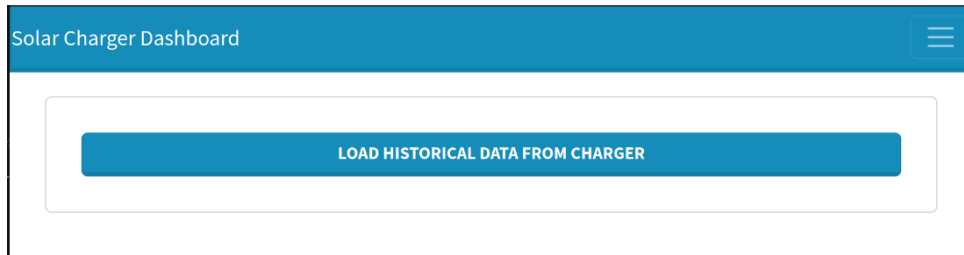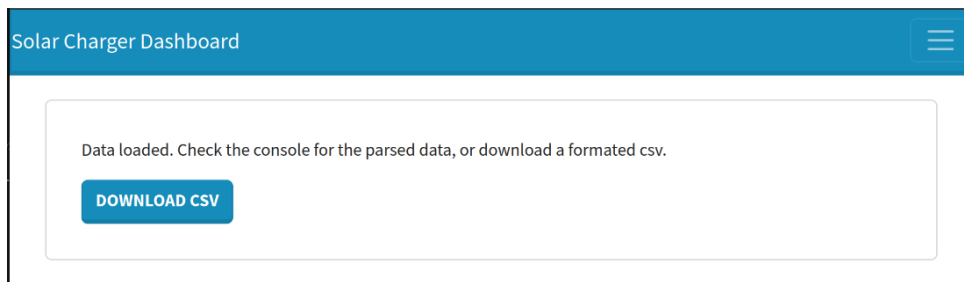
The implementation of a switching voltage regulator is not the primary focus of this project. A solution that is simple to implement was chosen. The parameters for selection are as follows:

- Operating Voltage: The output voltage of the regulator should be 3.3 V. The input voltage range should accommodate the universal use of this solar charger without limitations. The battery pack voltage is variable and can reach up to 24 V nominal voltage. Consequently, the regulator should have a maximum input voltage rating of at least 30 V.

- Current Capability: The recommended power supply current capacity for the ESP32-C6 is 500 mA, which includes a safety margin. According to the datasheet, the ESP32-C6 draws a peak current of approximately 400 mA while transmitting Wi-Fi packets and operating all peripheral components. With the Wi-Fi radio in receiving mode, the ESP32-C6's peak current should be less than 100 mA.

  The device interface screen will also draw power from the 3.3 V rail. With the backlight on, the current consumption should be less than 30 mA.

  Therefore, the minimum current capability of the 3.3 V power supply was set at 600 mA. It is crucial to note that switching power supplies are most efficient when operating above 50 % load. Hence, the current capability of the selected power supply should not exceed 1 A, as it would not operate efficiently most of the time.

- Documentation and Design Guidelines: Comprehensive documentation, component selection guidelines, and board layout recommendations are crucial. The manufacturer of the selected converter must provide these resources, as the author of this work is not a professional switching power supply designer.

- Cost: The cost of the entire switching power supply circuit is an important consideration, as this component does not enhance the parameters of the solar charger in any significant way. Assembly cost must also be considered, meaning the circuit should require only a small number of readily available external components.

The LMR544XX, where XX = "06" for 0.6 A version or "10" for 1 A version, integrated circuit from Texas Instruments[50] has been selected. It is part of the "Simple Switcher" lineup, indicating that the documentation provided is very clear and comprehensive. All other requirements are met or exceeded. There are four variants of this chip; they differ by maximum output current (0.6 A or 1 A) and behavior under low load.

The 0.6 A current capability is deemed optimal. The version DBVR, which offers PFM mode during low-load periods, is better suited for this application as PFM mode allows for higher efficiency than the FPWM mode of the FDBVR version. Although PFM mode exhibits higher voltage ripple than FPWM mode [51], the ESP32-C6 can tolerate up to 0.3 V deviation from the operating voltage. The voltage ripple during PFM will be less than 100 mV peak to peak and can be further reduced by adding additional filtering capacitors.

■ 10.1 Complementary Component Selection

The basic circuit requires only five external components, connected according to the reference schematic shown in Figure 10.1. Starting with the output voltage adjustment, two resistors need to be selected, R_{FBT} the top resistor and R_{FBB} the bottom resistor. The division ratio of these two resistors determines the output voltage, while the series resistance balances current consumption and interference resistance. The recommended range for R_{FBT} is 10 kΩ to 100 kΩ, and its value is determined by the equation 5, where V_{REF} is defined as 0.8 V, and the value of R_{FBB} can be decided by the user. The desired values were calculated as $R_{\text{FBT}} = 47$ kΩ and $R_{\text{FBB}} = 15$ kΩ, which achieves the required voltage with just a 0.25 % error.

$$R_{\text{FBT}} = \frac{V_{\text{OUT}} - V_{\text{REF}}}{V_{\text{REF}}} \times R_{\text{FBB}} \qquad (5)$$

The minimum inductance necessary for the output inductor is determined by the equation 6. The maximum input voltage of the switching covnerter is 36 V, and the output current is 0.6 A. The converter switching frequency is set to 1.1 MHz. the $K_{textIND}$ value, which represents the ripple current of the inductor as a fraction of the maximum current, should be between 0.2 and 0.6 as suggested by the datasheet [50]. Which also advises selecting a smaller inductance value if possible. The chosen inductance of 10μH corresponds to a K_{IND} 0.45. Additionally, it is crucial that the inductor's current carrying capacity is adequate; the chosen component has a saturation current of 2.4 A and an RMS current of 1.5 A.

$$L_{\mathrm{MIN}} = \frac{V_{\mathrm{IN(MAX)}} - V_{\mathrm{OUT}}}{I_{\mathrm{OUT}} \cdot K_{\mathrm{IND}}} \cdot \frac{V_{\mathrm{OUT}}}{V_{\mathrm{IN(MAX)}} \cdot f_{\mathrm{SW}}} \tag{6}$$

In addition, a $22\mu\mathrm{F}$ capacitor was chosen for output filtering. This capacitor was placed close to the inductor, seccond 22 $22\mu\mathrm{F}$ capacitor was placed next to the 3.3 V power input of the ESP32-C6 module, providing more than adequate current spike handling ccapacity.

▋ 10.2   PCB layout

LMR54406 has strategically placed the 6 pins of its package, to allow single-layer design. Texas Instruments also provides a layout examples to show optimal component placement, in order to minimize the size of the high-speed switching current loops. The final design of the converter circuit can fit inside an area smaller than 15 mm by 15 mm (figure 10.3).
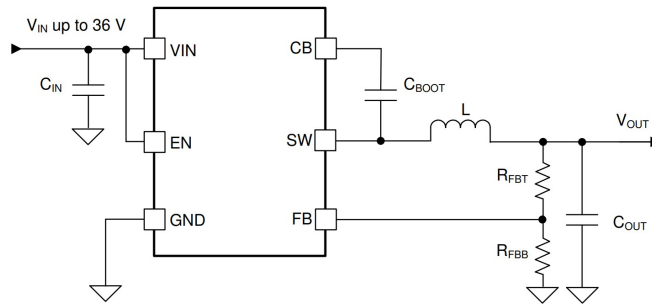
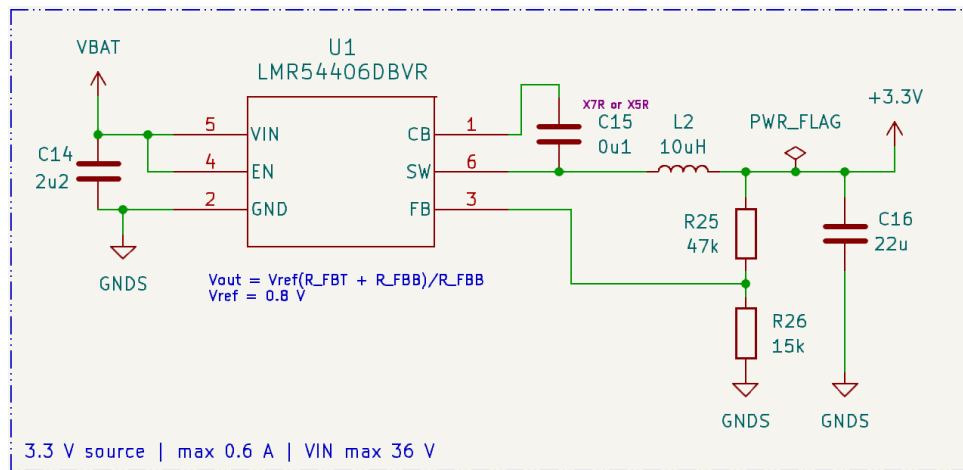**Figure 10.1:** LM54406 application schematic

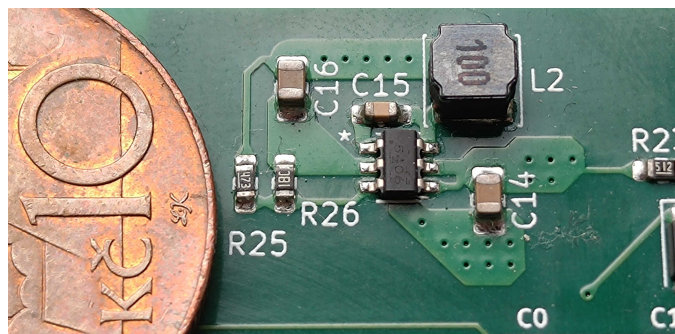**Figure 10.2:** LM54406 in design schematic

Figure 10.3: LM54406 assembled

Chapter 11

Home automation connectivity

A significant benefit of the Smart solar charger over conventional models is its seamless integration with the widely-used home automation system, Home Assistant. This system operates as a self-hosted platform that centralizes control over various devices from multiple manufacturers. The Home Assistant server can be efficiently run on single-board computers like the Raspberry Pi 4 or within a Docker container, without the need for any cloud-based services. Additionally, a recent update introduced a locally hosted voice assistant for managing the smart home network. These capabilities make Home Assistant particularly suitable for managing remote settings such as a wood cabin, camping trailer, or houseboat.

Home Assistant operates with a network of devices, where each device may be associated with several entities, and each entity represents a specific data type, such as temperature or voltage. Home Assistant automatically captures and records all data received from these entities. Devices are established via integrations, serving as connectors between Home Assistant and specific manufacturer protocols. A full description of Home Assistant's concepts is available on the official website [52]

One of many integrations available for Home Assistnat is the MQTT protocol [3] integrations, this serves as backbone for many other services like Zigbee2MQTT, which integrates Zigbee 3.0 devices into the ecosystem. The reason why many services use the MQTT interaction to connect to Home Assistant instead of a custom integration is the MQTT device discovery feature [53]], which automatically adds devices and entities to Home Assistant without the need for user interaction. The decision was made to also use the MQTT integration and its protocol to add the Smart solar charger device into Home Assistant.

11.1 Device discovery

The discovery process for the home attendant library mqtt requries a sepparate message for every entity of a device. These messages must be sent to a specific topic, with structure

```
<discovery_prefix>/<component>/[<node_id>/]<object_id>/config
```

- `discovery_prefix` seperates home assistant topics from other topics on the MQTT server, this prefix defaults to homeassistant.

- `component` is the type of mqtt component for example a sensor, binary sensor, switch and many others, A complete list can be found in the official documentation [53]

- `node_id` is an

The payload for the configuration topic must be in a JSON format [54]. and is specific for every component type. However, there are many configuration variables utilized for all component types. These are examples of the most important universal variables

- `device_class` is a parameter that automatically assigns frontend properties such as icons in Home Assistant

- `name` a user readable name for the entity in home assistant, if left as null, it will be aromatically generated from device_class

- `state_topic` a MQTT topic where the state of the entity will be published, it will default to the same topic as the dicovery message topic, but with /state instead of /config.

- `value_template` is a necessary property when multiple entities utilize the same state_topic. This demands that the state value be received as a JSON containing values for each entity subscribed to that state_topic. The value_template [55] specifies how to retrieve the appropriate value for each entity from the JSON message.

- `unique_id` is a identifier which must diferent for every Home Assistant entity

- `device` is a complex variable that requires a JSON map as input, within which the properties of the Home Assistant device to which the entity is linked can be defined. It is not required to replicate the entire map for each configuration message; if the device identifier values match those of other previously configured entities, the MQTT discovery system will automatically complete the remaining device description map.

Other configuration variables can be found on the official page for every type of component [53].

◼ 11.2 Mqtt implementation

ESP-IDF provides an MQTT client component [56]. Additional functionality has been implemented for sending configuration and state messages. The MQTT connection is established once the WiFi has successfully connected to an access point. Should the server configuration be in place, the device will

MQTT topic: homeassistant/sensor/SMART_SOLAR_CHARGER_TDIE

```json
{
  "name": "Die Temperature",
  "device_class": "temperature",
  "state_topic": "homeassistant/sensor/SMART_SOLAR_CHARGER/state",
  "unit_of_measurement": "°C",
  "value_template": "{{ value_json.TDIE }}",
  "unique_id": "SMART_SOLAR_CHARGER_TDIE",
  "device": {
    "identifiers": [
      "SMART_SOLAR_CHARGER"
    ],
    "manufacturer": "David Rajchman",
    "name": "Solar Charger"
  }
}
```

Listing 4: sample discovery message for the die temperature sensor

make five attempts to connect to the MQTT server. If these attempts fail, it will cease further attempts. The connection will be re-attempted only if a new server configuration is entered or if the WiFi connection is reestablished. Once the device has successfully established a connection with the MQTT broker, it will transmit all discovery messages and signal the MQTT connection status to the measurement loop, initiating the transmission of state data.

Chapter 12

Local data storage

When the device is linked to the Home Assistant smart home system, it archives each data point in its database for later analysis. Nonetheless, there might be times when the device is not connected to an mqtt server, necessitating a local data storage solution to allow user access to device logs. The only storage option is the internal flash chip of the ESP32-C6 WROOM module, available in sizes of 4, 8, or 16 megabytes. With the application and website consuming approximately 3 MB, only 1 MB remains for data storage if the least expansive module version is used. The desired storage duration is at least three days. A csv [49] file format was chosen as the base of the storage system. To maximize data compression while maintaining ease of decoding, the data was stored in hexadecimal format. The system's objective is to continuously replace the oldest data with new measurements, thus always fully utilizing the available 1 MB without any waste. A circular queue file system was implemented, requiring each row to maintain a fixed length to facilitate easy navigation to any row by calculating the exact starting character of the row. Negative values for battery current and temperature, which add an extra character in the row, led to the decision to compute and store the absolute values separately from their signs. The csv line generation function was incorporated within the `ltc4015_lib` 6.5, using four hexadecimal characters to record voltage and current measurements, corresponding to the full 16-bit register width. For temperature readings, only two characters were used, reducing the precision to 1°C, covering a range of -256 °C to 256 °C. This precision is sufficient for monitoring the battery's condition and checking if the LTC4015 chip is overheating. According to the datasheet, LTC4015 will cease charging operations if the die temperature is above 160 °C. Each data line consists of 48 characters, equivalent to 48 bytes, as the data employs extended ASCII format. In the ESP-IDF, a partition size of 1 M corresponds to $1024 \cdot 1024$ bytes, providing 1048576 bytes for the csv file storage. This capacity can accommodate over 21000 entries, sufficient for just over three days of data storage with a logging interval of 15 seconds. Data can be retrieved via an API endpoint. To enhance data usability, a script was developed for the web interface which retrives the file, converts hexadecimal data to decimal, correctly adjusts the signs of numbers, organizes the data from oldest to newest, eliminates blank lines, as the csv

file on the device keeps a constant number of lines, which may include lines filled with spaces if the file has not been completely populated., and generates a new downloadable csv file.

Chapter 13

Enclosure

Finalizing the device necessitates a physical enclosure. The case was fabricated using a 3D printer, which is the most cost-effective method to produce small quantities. The case design was developed using Autodesk Fusion CAD software [57]. The enclosure is designed to house the device, its display, buttons, connectors, and the output fuse.

For the electrical connections of the solar panel and battery, an XT60 connector was utilized. The XT60 [58], a universal 2-pin DC connector, was initially popularized in hobby aircraft modeling and FPV drone racing for connecting lithium batteries to aircraft. It is highly durable, offers reverse polarity protection, and supports up to 30 A of continuous current and 60 A instantaneous current. Owing to these attributes, the XT60 connector is increasingly being used for connecting solar panels and batteries to solar chargers. Panel-mounted versions of this connector were employed, and an automotive blade fuse of regular size was chosen for the output connector's fuse. The fuse will be placed inside a panel housing and will disconnect the positive output voltage if overloaded. 2.5 mm^2 wire is used to connect the charger PCB to the connectors and fuse.

Figure 13.1: XT60 male and female connectors

The enclosure's design is divided into two sections. The first section includes the bottom and one sidewall. The charger PCB is affixed to the bottom, and the power connectors and fuse are mounted on the right sidewall. This configuration allows the power components to be assembled as a single unit, which is then inserted into the second part of the enclosure, comprising the top cover and remaining sidewalls. The display and buttons, mounted on the second part, connect easily to the charging PCB when the two parts are

merged. Once assembled, the enclosure shields the USB-C port and the fuse slot from dust. Only the XT60 connectors and the user interface remain accessible when the enclosure is sealed.

Mounting hardware:

- **LCD** four M2 screws at least 8 mm long and nuts are used to mount the LCD.

- **XT60 panel connectors** six M2.5 screws at least 10 mm long are used, nuts are required for the output male connector

- **Charger PCB** four 2.5 mm diameter 6 mm long self tapping screws are used to mount the charger PCB, it is also possible to use M2 screws in combination with heat inserts, if the PCB is expected to be mounted and remounted many times.

- **Enclosure** three 2.5 mm diameter 6 mm long self tapping screws are used to hold the two halves of the enclosure together.
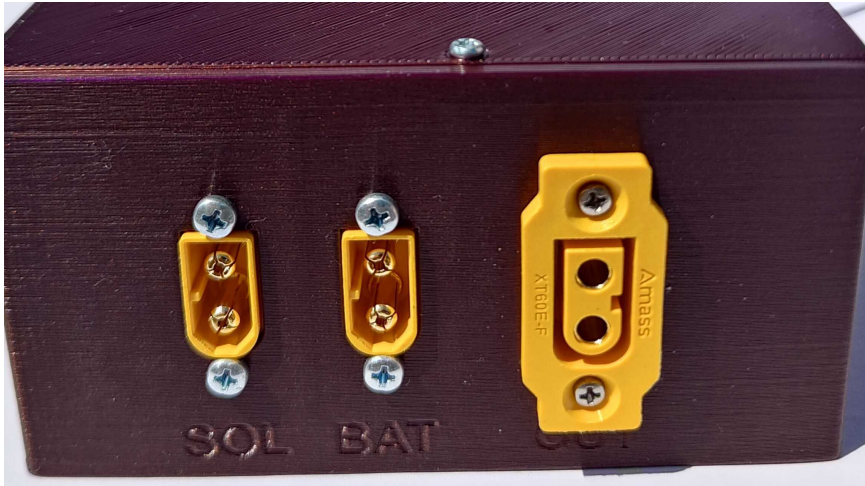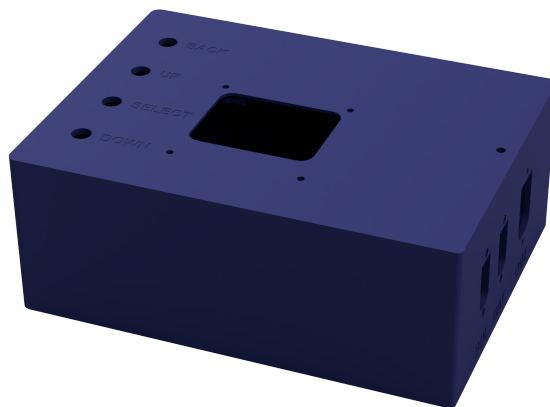
Figure 13.2: XT60 connectors on the S.M.S.C.
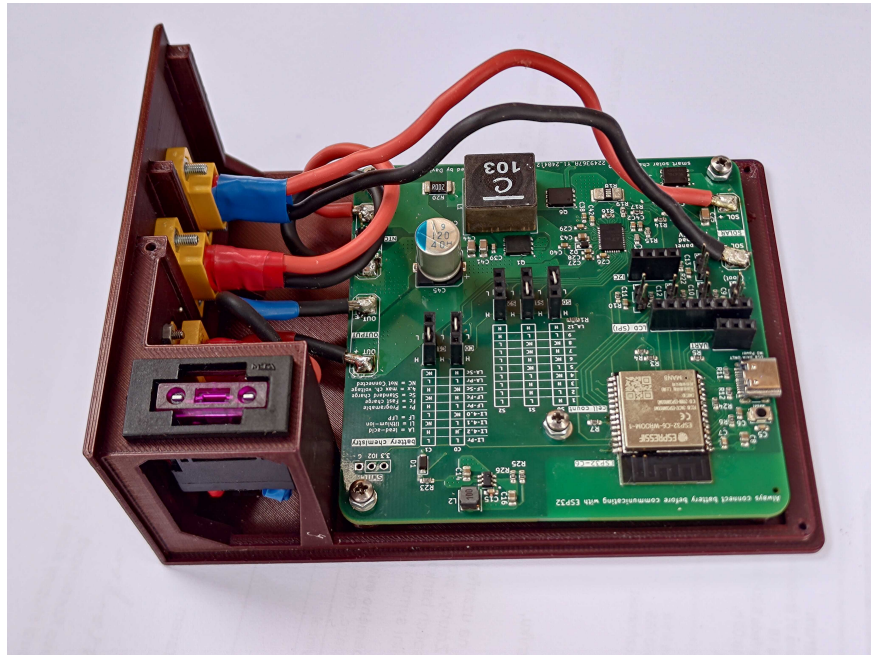

Figure 13.3: rendering of the enclosure

57

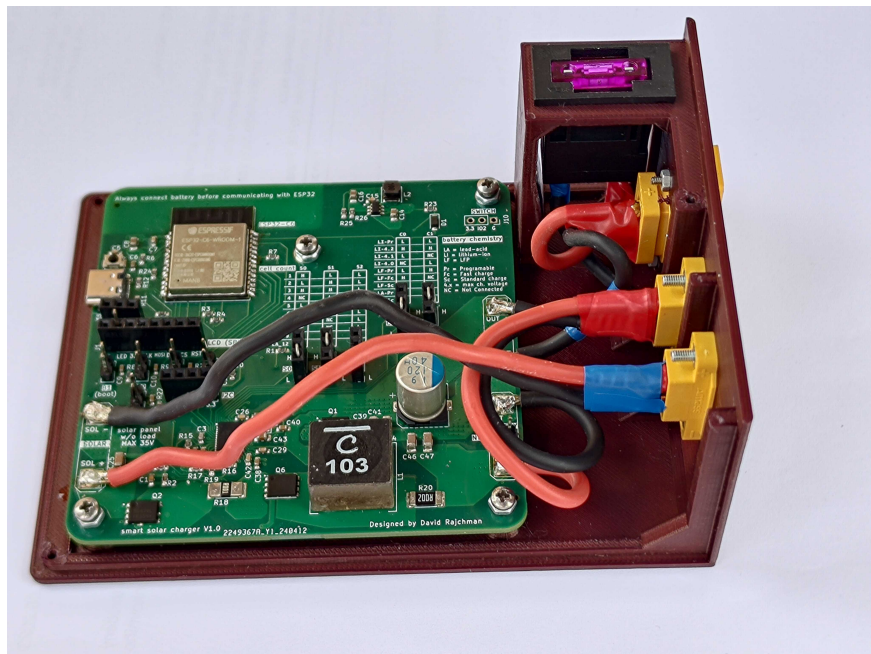**Figure 13.4:** bottom part of the enclosure assembled

Figure 13.5: bottom part of the enclosure assembled

Chapter 14

Device testing

In order to evaluate the if the complete charger works correctly, an operational test has been conducted. The goal was to test the ability of the charger to complete the charging cycle. Moreover, to test the ability of the MPPT algorithm to follow the maximum power point of the solar panel. Software stability testing was also conducted.

14.1   Testing setup

12 V 12 Ah AGM battery was utilized, throughout the whole testing procces. Due to the LTC4015 being a comercial product from a well established manufacturer, it can be assumed that if the charging circuit operates correctly with one type of battery chemistry it will also operate correctly with a different type of battery.

A 10 W 12 V solar panel was utilized to test maximum power tracking capabilities, datasheet for the solar panel model was not available; however, approximately 25 V open circuit voltage was measured on the panel was measured during a sunny day before the start of MPPT testing. This is slightly higher than other 12 V 10 W solar panels which normally have an open circuit voltage of 20 V. However, the 12 V branding of the solar panel only means that it is suitable for use with 12 V battery systems even under conditions with limited sunlight, which means that this type of solar panel could have been designed to provide higher voltages and lower currents and still use the 12 V 10 W branding.

Because the solar panel does not have enough power to fully charge the battery during the testing period, a laboratory power supply was used instead for the battery charging test.

sets of high power resistors were used to test the charger behavior during loaded output

14.2   Test results

During the MPPT algorithm test, the device was observed following a voltage level that was approximately 75 % of the open drain voltage of the solar

panel, which indicates the correct behavior. The test happened during periods of intermittent sun and cloud cover, it was observed that the solar charger correctly decreased the solar panel operating voltage during states of lower illumination and increased it when illumination increased. During this test, the device was connected to the MQTT broker of the Home Assistant that correctly captured the test results (Figure 14.3).

In the battery charging experiment, the charging was halted when the battery voltage reached a stable 13.2 V, corresponding to 2.2 V per cell (figure **??**), which matches the target voltage specified by the lead-acid battery charging protocol of the LTC4015. The application of a load to the output terminal was also evaluated. Using resistor combinations as the load, a current of 0.5 A was drawn from the output. Throughout the periods when the load was applied, the charger effectively redirected the power from charging the battery to powering the load. The connection to the Home Assistant server was unavailable during this test; therefore, monitoring was conducted through the web user interface, and the data was recorded in a csv file.

Over a four-day testing period, the ESP32-C6 firmware demonstrated stability without system crashes, and memory usage remained stable, suggesting the absence of memory leaks.
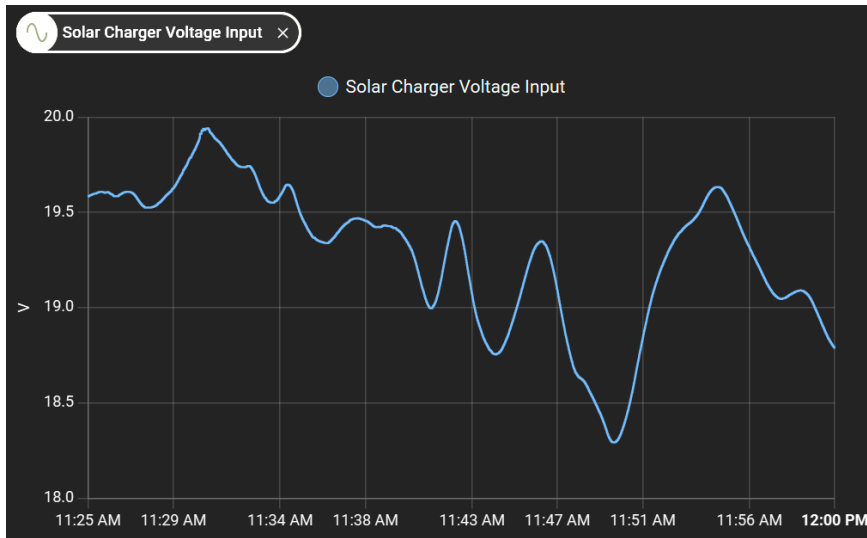
Figure 14.1: Input voltage during MPPT test in Home Assistant
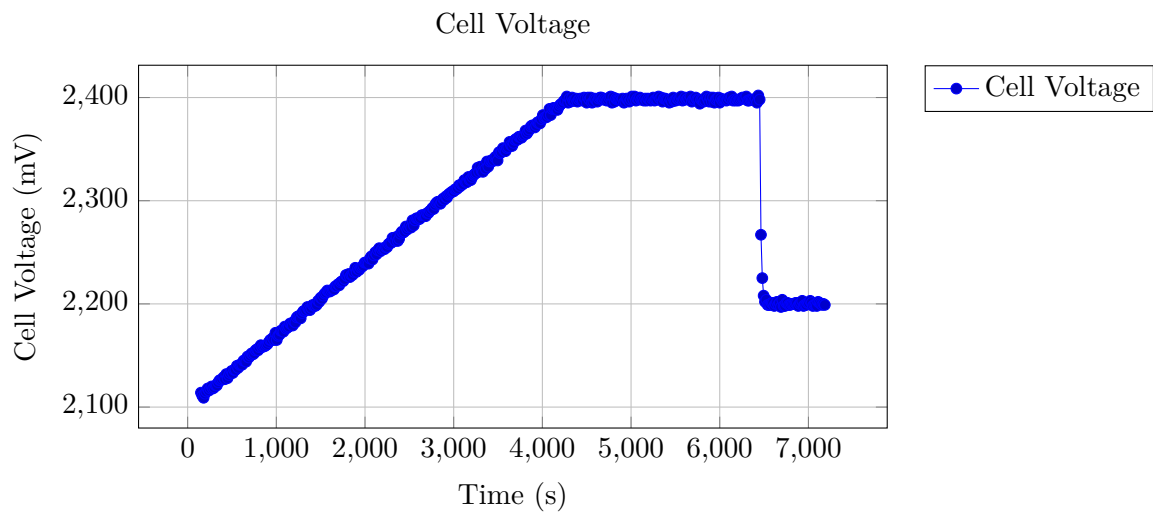

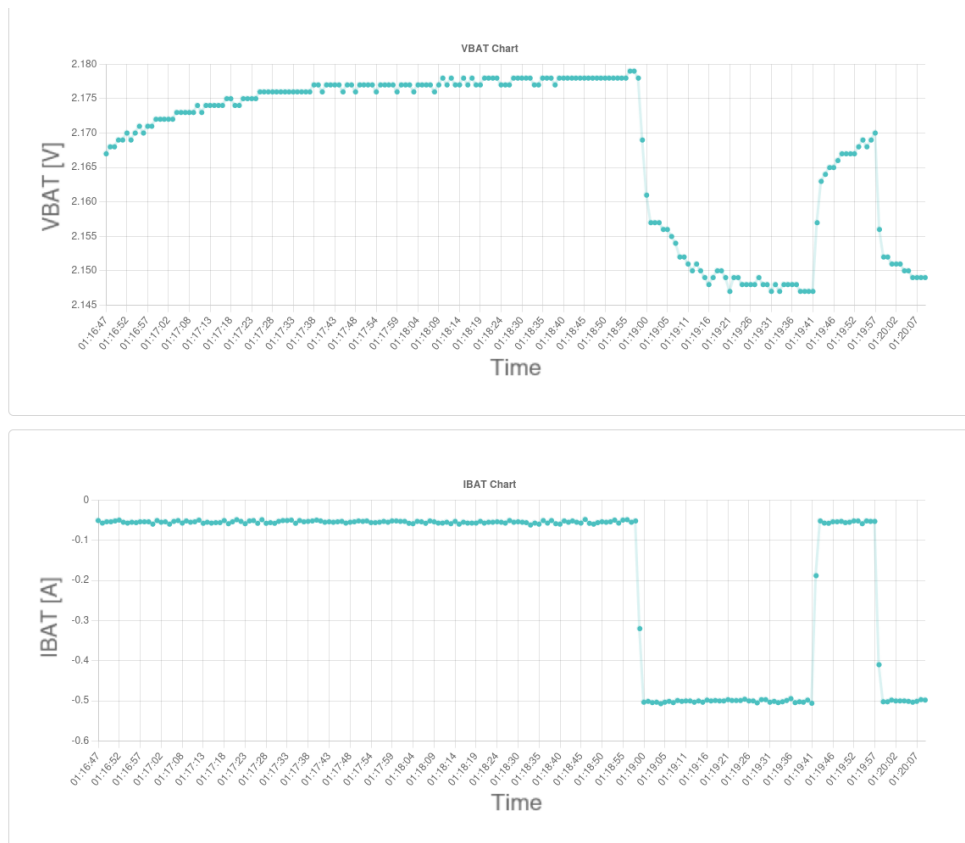**Figure 14.2:** Cell Voltage data from the charging test

61

Figure 14.3: Power drawing from output test, screenshot of the results from the web interface. (during the last load connection, voltage was supplied into solar input)

Chapter 15

Comparison to commercial devices

15.1   Cost of the smart solar charger

To fully contextualize the comparative analysis, it is essential to estimate the cost of the smart solar charger. This cost is categorized into fixed costs, such as the price of components, and variable costs, like manufacturing expenses. The determination of variable costs necessitates setting a manufacturing quantity. Producing at least 5 units per batch is considered ideal due to the cost structure of PCB manufacturing and assembly. Consequently, a batch size of 5 has been chosen for these calculations. The labor costs for assembling the components into a complete device are not included, as the intended users are hobbyists from the maker community who typically undertake assembly in their spare time. The pricing for the components, PCB production, and assembly derives from the initial order placed with JLCPCB. The expenditure for the 3D printing of the enclosure is based on the prevailing market rates, which range from approximately 50-150 CZK per hour of printing [59]. This variation is due to the choice of material and the complexity of 3D printing. Because the design of the case is straightforward and does not require specialized materials, 50 CZK was utilized for cost estimation. The time it takes to print the case is around 9 hours, assuming a layer height of 0.2 mm. Except for PCB orders, the cost of shipping was not accounted for.

Table 3: Costs per 1 device manufactured in batches of 5

Part of the device	Price in CZK
PCB	50
charger Components	550
PCB assembly	250
PCB order shipping and taxes	600
LCD module	150
additional components	300
3D printed case	450
Total cost	2350

■ **15.2   Capabilities comparison**

In terms of power handling capability, the device is comparable to the commercial solutions mentioned in the market study, with the exception of the maximum solar panel voltage allowed on the input. This restricts the usage possibilities and also limits the maximum charging power. Commercial solar chargers determine the maximum output power based on the highest voltage of the battery they can charge. If we limit the chargers to 12 V batteries, their maximum charging power will be comparable to that of the Smart MPPT charger. In terms of ease of use, the device offers an intuitive web-based user interface and offers integration into Home Assistant with just a few steps, which is a feature not offered in the commercially available MPTT chargers. The device also supports every commonly used battery chemistry type.

Table 4: Table comparing comerical solution to the the S.M.S.C.

device	CARSPA	EPEVER	Victron	EcoFlow	S.M.S.C
price [CZK]	1590	1809 +	2900 ++	22 500	2350 ++
internet connection	✗	✓	✓	✓	✓
remote configuration	✗	✓	✓	✓	✓
max. power [W]	480	520	440	500	190
max. power at 12 V [W]	240	240	180	N/A	190
max. input voltage [V]	65	100	75	150	35
max. charge current [A]	20	20	15	15	16
support for lead-acid b.	✓	✓	✓	✗	✓
support for lithium b.	✗	✓	✓	✓	✓

Conclusion

The aim of this project was to design a smart MPPT solar charger, focusing on ease of use, remote monitoring capability, compatibility with LFP batteries, and sufficient battery protection features. Initially, I presented a theoretical overview of the technologies used in solar charging, followed by a market analysis of MPPT chargers. From the analysis results, I established the requirements that the project outcomes should meet. I have developed a smart MPPT-enabled solar charger and built a functional prototype. This solar charger seamlessly integrates with the Home Assistant smart home system and features a web interface for monitoring and configuring charging parameters and network settings. It supports charging LFP, Li-Ion, and Lead Acid batteries, with battery types being adjustable post-manufacturing via jumper wires, allowing users to switch to different chemistries. The charging PCB design permits customization of the charging current; users can modify the maximum charging current by changing shunt resistors, and the web interface also allows for reducing the maximum currents. An Excel spreadsheet is provided to calculate the optimal inductor for the buck converter based on operating voltages and selected maximum current, from a range of widely available parts. Proper inductor selection can achieve charging efficiencies on par with commercial MPPT chargers. I have verified that the prototype can determine the maximum power point of a solar panel and effectively charge and monitor a lead-acid battery. Additionally, I tested the ESP32-C6 software for stability, confirming it operated continuously for 4 days without crashing, with consistent average memory usage throughout. The final device is cost competitive with commercial solutions when produced in small batches. The entire project is released under the GPLv3 open-source license and is accessible on a public GitHub repository [32].

I consider the outcome to be successful, but there are a few changes I would implement in the next iteration: Integrating the undervoltage protection disconection circtuits directly onto the PCB would enhance the product's seamlessness and efficiency, as the 3.3 V logic signal available to the MOSFET module does not provide sufficient voltage to fully activate most transistors, leading to heat losses and limiting the maximum current. Adopting an I2C interfaced screen would free up the SPI bus for connecting SD cards, which could be used to store extensive log data over many years. Additionally, I

would incorporate some form of connector or screw terminal directly on the PCB for power outputs, allowing the final product to be assembled without the need for soldering.

Chapter 16

Appendices List

- **README.md** Github repository readme file

- **Enclosure.zip** enclousure source files

- **firmware.zip** firmware source files

- **PCBdesign.zip** pcb source files

Bibliography

[1] "FreeRTOS - Market leading RTOS (Real Time Operating System) for embedded systems with Internet of Things extensions", FreeRTOS. (), [Online]. Available: `https://www.freertos.org/index.html` (visited on 05/19/2024).

[2] NXP, *I2C-bus specification and user manual UM10204*, 2021. [Online]. Available: `https://www.nxp.com/docs/en/user-guide/UM10204.pdf`.

[3] "MQTT - The Standard for IoT Messaging". (), [Online]. Available: `https://mqtt.org/` (visited on 05/13/2024).

[4] "SMbus". (), [Online]. Available: `http://smbus.org/` (visited on 05/16/2024).

[5] "SMBus Specifications". (), [Online]. Available: `http://www.smbus.org/specs/` (visited on 05/16/2024).

[6] "Introduction to SPI Interface | Analog Devices". (), [Online]. Available: `https://www.analog.com/en/resources/analog-dialogue/articles/introduction-to-spi-interface.html` (visited on 05/23/2024).

[7] S. Philipps and W. Warmuth, "Photovoltaics Report - Fraunhofer ISE", 2020, https://www.ise.fraunhofer.de/conte%0Ant/dam/ise/d.

[8] M. Green, E. Dunlop, J. Hohl-Ebinger, M. Yoshita, N. Kopidakis, and X. Hao, "Solar cell efficiency tables (version 57)", *Progress in Photovoltaics: Research and Applications*, vol. 29, no. 1, pp. 3–15, Jan. 1, 2021, ISSN: 1099-159X. DOI: `10.1002/PIP.3371`. [Online]. Available: `https://onlinelibrary.wiley.com/doi/full/10.1002/pip.3371` (visited on 12/11/2023).

[9] O. Ayadi, R. Shadid, A. Bani-Abdullah, M. Alrbai, M. Abu-Mualla, and N. A. Balah, "Experimental comparison between Monocrystalline, Polycrystalline, and Thin-film solar systems under sunny climatic conditions", *Energy Reports*, vol. 8, pp. 218–230, Nov. 1, 2022, ISSN: 2352-4847. DOI: `10.1016/J.EGYR.2022.06.121`.

[10]   Inés Rosellón Inclán. "Analysis of global battery production: Production locations and quantities of cells with LFP and NMC/NCA cathode material - Fraunhofer ISI". (2023), [Online]. Available: `https://www.isi.fraunhofer.de/en/blog/themen/batterie-update/globale-batterieproduktion-analyse-standorte-mengen-zellen-lfp-nmc-nca-kathoden.html` (visited on 12/13/2023).

[11]   "Trends in batteries – Global EV Outlook 2023 – Analysis - IEA". (2023), [Online]. Available: `https://www.iea.org/reports/global-ev-outlook-2023/trends-in-batteries` (visited on 12/13/2023).

[12]   J. Ahmed and Z. Salam, "An Improved Method to Predict the Position of Maximum Power Point During Partial Shading for PV Arrays", *IEEE Transactions on Industrial Informatics*, vol. 11, pp. 1378–1387, Dec. 1, 2015. DOI: `10.1109/TII.2015.2489579`.

[13]   "MPPT12/24-20-20A". (), [Online]. Available: `https://www.carspa.cc/mppt-mppt-series-12v24v-10a-80a/mppt1224-20-20a-12-24v-auto-mppt-solar-charge-controller` (visited on 12/14/2023).

[14]   "Solární regulátor MPPT CARSPA 20A - 12/24V | TIPA.EU". (), [Online]. Available: `https://www.tipa.eu/cz/solarni-regulator-mppt-carspa-20a-12-24v/d-250162/?gad_source=1&gclid=Cj0KCQiA7OqrBhD9ARIsAK3UXh2STcCBlyyvLP73dmiSBn3OULjLfrxLl6AklB_7MpOB_gdNhRPiHIAaArc8EALw_wcB` (visited on 12/14/2023).

[15]   "EPEVER XTRA2210N". (), [Online]. Available: `https://www.epever.com/product/xtra-10-40a-mppt-solar-charge-controller/` (visited on 12/14/2023).

[16]   Colin Hickey. "RS485-WiFi-EPEver". (), [Online]. Available: `https://github.com/chickey/RS485-WiFi-EPEver` (visited on 12/14/2023).

[17]   "EPEVER XTRA2210N price". (), [Online]. Available: `https://www.digitor.cz/cs/211222-epever-xtra2210n-solarni-mppt-regulator-12-24-v-xtra-20a-vstup-100v?gclid=Cj0KCQiA7OqrBhD9ARIsAK3UXh2YCZtCwjl8XPVI2eQoSrv9Fe29Ks2idZq2Ve1vUaAvO_EALw_wcB#tab-popis-produktu` (visited on 12/14/2023).

[18]   "SmartSolar MPPT 75/10, 75/15, 100/15 & 100/20 - Victron Energy". (), [Online]. Available: `https://www.victronenergy.com/solar-charge-controllers/smartsolar-mppt-75-10-75-15-100-15-100-20` (visited on 12/14/2023).

[19]   "MPPT regulátor nabíjení Victron Energy SmartSolar 75V 15A | ECO-produkt". (), [Online]. Available: `https://ecoprodukt.cz/p/74644-mppt-regulator-nabijeni-victron-energy-smartsolar-75v-15a-s-bluetooth-15250?gad_source=1&gclid=Cj0KCQiA7OqrBhD9ARIsAK3UXh0Iq8oZA5hoh60hPy99g_7idjYKRd2W9vIIe7kvxpkXLhRoaAorkEALw_wcB` (visited on 12/14/2023).

[20] Jiří Bětuňák. "GitHub - KinDR007/VictronMPPT-ESPHOME: Victron Mppt charger ve.direct to esphome node". (), [Online]. Available: `https://github.com/KinDR007/VictronMPPT-ESPHOME` (visited on 12/14/2023).

[21] "Victron Energy GlobalLink 520 pro monitorování a řízení | ECOprodukt". (), [Online]. Available: `https://ecoprodukt.cz/p/88164-victron-energy-globallink-520-na-monitorovanie-a-riadenie-systemu-41378?gad_source=1&gclid=Cj0KCQiA7OqrBhD9ARIsAK3UXh1SrSAg1xyLa2N-uRKd2GJEFT4SmcERZyeSUsywD4neFta6MZzvu1caAhENEALw_wcB` (visited on 12/14/2023).

[22] "Victron Energy dálkové ovládání VE.Direct LoRaWAN EU863-870". (), [Online]. Available: `https://www.conrad.cz/cs/p/victron-energy-dalkove-ovladani-ve-direct-lorawan-eu863-870-module-ass030538010-vhodne-pro-model-stridac-napeti-v-1924510.html?gad_source=1&gclid=Cj0KCQiA7OqrBhD9ARIsAK3UXh0PDe0fD-R6sCctz9MSLNq7knkY33_Pnu1TJTnTm2Vklf_5RgDskUAaAu` (visited on 12/14/2023).

[23] "EcoFlow DELTA 2". (), [Online]. Available: `https://us.ecoflow.com/products/delta-2-portable-power-station?variant=40569176326217` (visited on 12/14/2023).

[24] "EcoFlow DELTA 2". (), [Online]. Available: `https://ecoflow.cz/products/ecoflow-delta-2` (visited on 12/14/2023).

[25] "EcoFlow DELTA 2 Smart Extra Baterie". (), [Online]. Available: `https://ecoflow.cz/products/ecoflow-delta-2-smart-extra-baterie` (visited on 12/14/2023).

[26] "ČLÁNEK LIFEPO4 134AH 3,2V SVOLT". (), [Online]. Available: `https://www.mivvyenergy.cz/cs/lithiove-clanky/clanek-lifepo4-134ah-32v-svolt.html#/158-inicializacni_nabiti-ne` (visited on 12/14/2023).

[27] Analog Devices. "LTC4015 Datasheet and Product Info | Analog Devices". (), [Online]. Available: `https://www.analog.com/en/products/ltc4015.html` (visited on 05/16/2024).

[28] H. Eraydin and A. F. Bakan, "Efficiency Comparison of Asynchronous and Synchronous Buck Converter", in *2020 6th International Conference on Electric Power and Energy Conversion Systems (EPECS)*, Oct. 2020, pp. 30–33. DOI: 10.1109/EPECS48981.2020.9304966. [Online]. Available: `https://ieeexplore.ieee.org/document/9304966` (visited on 05/16/2024).

[29] "DMP3010LPSQ-13 Diodes Incorporated | Mouser", Mouser Electronics. (), [Online]. Available: `https://cz.mouser.com/ProductDetail/Diodes-Incorporated/DMP3010LPSQ-13?qs=M%2FOdCRO8QQ37bx%252BRJ7NXlA%3D%3D` (visited on 05/24/2024).

[30] "NTMFS5C670NLT1G onsemi | Mouser", Mouser Electronics. (), [Online]. Available: `https://cz.mouser.com/ProductDetail/onsemi/NTMFS5C670NLT1G?qs=85c0ukccmxF7%2F53ox6Y1og%3D%3D` (visited on 05/24/2024).

[31] "XAL1510 Series High Current, High L Shielded Power Inductors | Coilcraft". (), [Online]. Available: `https://www.coilcraft.com/en-us/products/power/shielded-inductors/molded-inductor/xal/xal1510/` (visited on 05/16/2024).

[32] "DavidRajchman/SmartMPPTSolarCharger", GitHub. (), [Online]. Available: `https://github.com/DavidRajchman/SmartMPPTSolarCharger` (visited on 05/24/2024).

[33] "High-Power Dual MOSFET Switch Module", ProtoSupplies. (), [Online]. Available: `https://protosupplies.com/product/high-power-dual-mosfet-switch-module/` (visited on 05/24/2024).

[34] T. Regan, J. Munson, G. Zimmer, and M. Stokowski, "Chapter 40 - Current sense circuit collection: Making sense of current", in *Analog Circuit Design*, B. Dobkin and J. Williams, Eds., Oxford: Newnes, Jan. 1, 2013, pp. 1111–1215, ISBN: 978-0-12-397888-2. DOI: 10.1016/B978-0-12-397888-2.00040-7. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/B9780123978882000407` (visited on 05/18/2024).

[35] "Unix Time Stamp - Epoch Converter". (), [Online]. Available: `https://www.unixtimestamp.com/` (visited on 05/16/2024).

[36] Jiří Zilvar. "Kolik elektřiny reálně vyrobí fotovoltaická elektrárna? Co ovlivní výkon elektrárny?", ESTAV.cz. (), [Online]. Available: `https://www.estav.cz/cz/10774.kolik-elektriny-realne-vyrobi-fotovoltaicka-elektrarna-co-ovlivni-vykon-vasi-elektrarny` (visited on 05/19/2024).

[37] "ESP-IDF Getting Started | Espressif Systems". (), [Online]. Available: `https://idf.espressif.com/` (visited on 05/19/2024).

[38] "ESP32-S3 Wi-Fi & BLE 5 SoC | Espressif Systems". (), [Online]. Available: `https://www.espressif.com/en/products/socs/esp32-s3` (visited on 05/19/2024).

[39] "ESP32-C6 Wi-Fi 6 & BLE 5 & Thread/Zigbee SoC | Espressif Systems". (), [Online]. Available: `https://www.espressif.com/en/products/socs/esp32-c6` (visited on 05/19/2024).

[40] Espressif, *Esp32-c6_datasheet_en*, Part ESP32-C6 Datasheet, Oct. 29, 2023. [Online]. Available: `https://www.espressif.com/sites/default/files/documentation/esp32-c6_datasheet_en.pdf` (visited on 04/29/2024).

[41] "Non-Volatile Storage Library - ESP32 - — ESP-IDF Programming Guide v5.2.1 documentation". (), [Online]. Available: `https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/storage/nvs_flash.html` (visited on 05/19/2024).

[42] "FAT Filesystem Support - ESP32 - — ESP-IDF Programming Guide v5.2.1 documentation". (), [Online]. Available: `https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/storage/fatfs.html?highlight=fat` (visited on 05/19/2024).

[43] "Partition Tables - ESP32 - — ESP-IDF Programming Guide v5.2.1 documentation". (), [Online]. Available: `https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-guides/partition-tables.html?highlight=partition` (visited on 05/19/2024).

[44] "U8g2", GitHub. (), [Online]. Available: `https://github.com/olikraus/u8g2/wiki/Home` (visited on 04/29/2024).

[45] M. Frey, *Mkfrey/u8g2-hal-esp-idf*, Apr. 18, 2024. [Online]. Available: `https://github.com/mkfrey/u8g2-hal-esp-idf` (visited on 01/30/2024).

[46] M. Bishop, "HTTP/3", Internet Engineering Task Force, Request for Comments RFC 9114, Jun. 2022, 57 pp. DOI: `10.17487/RFC9114`. [Online]. Available: `https://datatracker.ietf.org/doc/rfc9114` (visited on 05/20/2024).

[47] A. Melnikov and I. Fette, "The WebSocket Protocol", Internet Engineering Task Force, Request for Comments RFC 6455, Dec. 2011, 71 pp. DOI: `10.17487/RFC6455`. [Online]. Available: `https://datatracker.ietf.org/doc/rfc6455` (visited on 05/20/2024).

[48] "CSS grid layout - CSS: Cascading Style Sheets | MDN". (Mar. 25, 2024), [Online]. Available: `https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_grid_layout` (visited on 05/20/2024).

[49] "CSV, Comma Separated Values (RFC 4180)". (May 7, 2024), [Online]. Available: `https://www.loc.gov/preservation/digital/formats/fdd/fdd000323.shtml` (visited on 05/17/2024).

[50] Texas Instruments. "LMR54406 data sheet, product information and support | TI.com". (), [Online]. Available: `https://www.ti.com/product/LMR54406` (visited on 05/18/2024).

[51] Steven Keeping. "The Advantages of Pulse Frequency Modulation for DC/DC Switching Voltage Converters". (Mar. 25, 2014), [Online]. Available: `https://www.digikey.com/en/articles/the-advantages-of-pulse-frequency-modulation-for-dc-dc-switching-voltage-converters` (visited on 03/24/2024).

[52] Home Assistant. "Concepts and terminology", Home Assistant. (), [Online]. Available: `https://www.home-assistant.io/getting-started/concepts-terminology/` (visited on 05/13/2024).

[53] Home Assistant. "MQTT", Home Assistant. (), [Online]. Available: `https://www.home-assistant.io/integrations/mqtt/#mqtt-discovery` (visited on 05/13/2024).

[54] "JSON". (), [Online]. Available: `https://www.json.org/json-en.html` (visited on 05/13/2024).

[55] Home Assistant. "Templating", Home Assistant. (), [Online]. Available: `https://www.home-assistant.io/docs/configuration/templating/#using-templates-with-the-mqtt-integration` (visited on 05/13/2024).

[56] "ESP-MQTT - ESP32-C6 - — ESP-IDF Programming Guide v5.2.1 documentation". (), [Online]. Available: `https://docs.espressif.com/projects/esp-idf/en/v5.2.1/esp32c6/api-reference/protocols/mqtt.html` (visited on 05/13/2024).

[57] "Autodesk Fusion | 3D CAD, CAM, CAE, & PCB Cloud-Based Software | Autodesk". (), [Online]. Available: `https://www.autodesk.com/products/fusion-360/overview` (visited on 05/20/2024).

[58] "XT60-M AMASS - Plug | DC supply; XT60; male; PIN: 2; for cable; soldering; yellow | TME - Electronic components". (), [Online]. Available: `https://www.tme.eu/en/details/xt60-m/dc-power-connectors/amass/` (visited on 05/20/2024).

[59] "Cena 3D tisku | Přehledná ukázka ceny 3D tisku". (), [Online]. Available: `https://3dtisk.pro/cena-3d-tisku/` (visited on 05/23/2024).