

Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH



Instytut Informatyki

Praca dyplomowa inżynierska

na kierunku Informatyka
w specjalności Inżynieria Systemów Informatycznych

Uniwersalny system ładowania akumulatorów

Jakub Kowalczyk

Numer albumu 300238

promotor
mgr inż. Krzysztof Gracki

WARSZAWA 2022

Uniwersalny system ładowania akumulatorów

Streszczenie. Niniejsza praca opisuje system stworzony do ładowania różnych rodzajów akumulatorów. W jej wstępie uargumentowana została potrzeba istnienia takiego urządzenia oraz przedstawiono ogólny jego zarys. Następnie przybliżono pojęcie akumulatora elektrycznego, jego najpopularniejszych rodzajów oraz sposobów ich ładowania. Zasadnicza część pracy poświęcona jest projektowaniu samego systemu. W pierwszej kolejności zaprezentowana jest jego koncepcja. Następnie sformułowane zostają założenia funkcjonalne i systemowe. Mając je na uwadze, szczegółowo przedstawiane są kolejne elementy systemu. Są nimi: obwód elektryczny, mikrokontroler oraz serwer. Na koniec zaprezentowane są efekty pracy. Przyjrzeć się można charakterystyką baterii, które zostały uzyskane podczas ładowania przygotowanym systemem.

Podstawą działania zaproponowanego tutaj urządzenia są tak zwane profile ładowania. Za ich pomocą można przedstawić najpopularniejsze krzywe ładowania. Użytkownik tworzy takie profile przy użyciu aplikacji internetowej. Następnie za pośrednictwem serwera przekazuje je do mikrokontrolera, który to stara się je zaaplikować na umieszczonej w ładowarce baterii.

W ramach pracy powstał prototyp pozwalający ładować pojedyncze ogniwa o maksymalnym napięciu 9V. Podczas testów urządzenia ładowano akumulatory litowo-jonowe oraz niklowo-metalowo-wodorkowe. Osiągnięte rezultaty są zadowalające. Ładowarka poprawnie wykrywała koniec prowadzonego procesu oraz utrzymywała ustalone wcześniej wartości natężeń prądu i napięć.

System jest w pełni przystosowany do potencjalnych modyfikacji. Jego funkcjonalności można rozszerzać czy to poprzez zwiększanie precyzji komponentów ładujących, całkowite wymiany poszczególnych modułów, czy też przez rozwój strony internetowej pod kątem jakości użytkowej.

Słowa kluczowe: Ładowarka · Akumulator · Internet Rzeczy · Mikrokontroler

Universal battery charging system

Abstract. This work describes a system designed to charge various types of batteries. In its introduction, the need for such a device was discussed and its general outline was presented. Then concept of rechargeable batteries, their most popular types and algorithms of charging are introduced. The main part of this paper is devoted to the design of the system. First, general concept is presented. Then the functional and system assumptions are created. Bearing them in mind, the following elements of the system are presented in detail: electric circuit, microcontroller and server. Finally, the effects of work are presented in the form of battery characteristics obtained during charging with constructed system.

This charger uses so-called charging profiles to describe popular charging curves. Potential user can create such charging profiles by using the web application. Then they can be transferred to the microcontroller via the server. After being received, they are applied to the battery which is placed inside charging circuit.

During the work, a prototype was created, which allows to charge individual battery cells with a maximum voltage of 9V. During the tests, device charged lithium-ion and nickel-metal hydride batteries. The results achieved are satisfying. The charger correctly detected the end of the charge and was able to maintain previously set values of current and voltage.

The system is fully prepared to potential modifications. Its functionalities can be expanded either by increasing the precision of regulating components, complete replacement of individual modules, or by developing the website in terms of functional quality.

Keywords: Charger · Rechargeable battery · Internet of things · Microcontroller

Spis treści

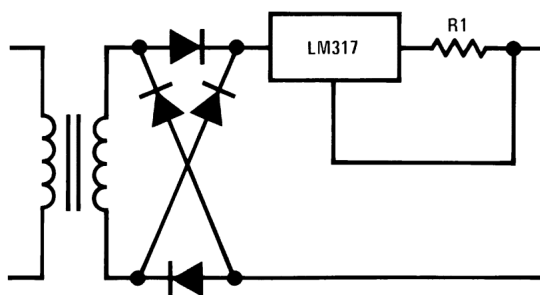
1. Wstęp	7
2. Akumulatory elektryczne i sposoby ich ładowania	10
2.1. Popularne rodzaje akumulatorów	10
2.2. Ładowanie akumulatorów	11
2.2.1. Ogniwa niklowo-kadmowe i niklowo-metalowo-wodorkowe	11
2.2.2. Ogniwa litowo-jonowe	14
2.2.3. Ogniwa kwasowo-ołowiowe	15
3. Projekt systemu	17
3.1. Założenia funkcjonalne systemu	18
3.2. Struktura profili ładowania	19
3.3. Obwód elektryczny	21
3.3.1. Regulator prądu	21
3.3.2. Sensory	25
3.3.3. Pętla rozładowująca	29
3.3.4. Rozprowadzenie ciepła	31
3.4. Oprogramowanie Mikrokontrolera	33
3.4.1. Struktura Klas	33
3.4.2. Kod programu mikrokontrolera	34
3.4.3. Komunikacja bezprzewodowa	38
3.5. Zewnętrzny serwer	40
3.5.1. Kontrola ładowarki za pomocą aplikacji internetowej	41
3.5.2. Oprogramowanie serwera	42
3.5.3. Struktura bazy danych	43
4. Efekty działania systemu	45
5. Wnioski	49
Bibliografia	53
Wykaz symboli i skrótów	54
Spis rysunków	55
6. Wyznaczania dokładnej wartości odniesienia 1.1V	56

1. Wstęp

Pierwsze próby magazynowania energii elektrycznej, podejmowane były przez człowieka długo przed powstaniem takich wynalazków, jak chociażby żarówka Edisona. Odkrycie ogniwa galwanicznego przypisuje się Alessandrovi Volcie, który dokonał tego już w 1800 roku[1]. Od tamtego czasu nieprzerwanie prowadzone są badania mające na celu udoskonalenie metod gromadzenia energii elektrycznej. Najpopularniejszym na małą skalę sposobem wciąż jednak są baterie ogni. Podzielić można je na dwa typy: pierwotne oraz wtórne [2] zwane inaczej akumulatorami. Pierwsze są nieładowne, drugie zaś są do tego przeznaczone. W dzisiejszych czasach szczególną uwagę przywiązuje się do ogniw wtórnych. Ma to związek przede wszystkim z ich rosnącym wykorzystaniem w urządzeniach przenośnych. Jeśli weźmie się pod uwagę większy przedział czasowy, akumulatory będą się cechować mniejszym kosztem oraz wpływem na środowisko w porównaniu do ich jednorazowych odpowiedników. Dla przykładu, najnowsze baterie niklowo-metalowo-wodorkowe typu AAA mogą być naładowane do 2100 razy[3], posiadając przy tym konkurencyjną pojemność w porównaniu do nieładownych ogniw alkalicznych[4]. Biorąc pod uwagę, że cena akumulatorów jest jedynie siedem razy większa, oszczędność w większym przedziale czasowym jest znaczna. Całkowita rezygnacja z baterii jednorazowych nie jest dobrym pomysłem, ponieważ również i one mają swoje zalety, o których warto wspomnieć. Ilość zgromadzonej energii w stosunku do masy przewyższa wartości spotykane przy ich ładowalnych odpowiednikach. Może mieć to znaczenie chociażby przy czujnikach dymu, dlatego też producenci odradzają używania w nich ogniw wtórnych.

Spodziewać się można, że udział akumulatorów w życiu codziennym człowieka z roku na rok będzie się jedynie zwiększał. Wystarczy spojrzeć chociażby na ciągły przyrost popularności urządzeń przenośnych. Po części przecież zawdzięczamy go właśnie współczesnym bateriom[5]. Ponieważ z reguły jest tak, że do każdego rodzaju akumulatora przypisać można oddzielną ładowarkę, pojedyncza osoba będzie w posiadaniu kilku z nich, mimo że będą się one różniły od siebie jedynie pojedynczymi parametrami. Przedstawiony w tej pracy system ma za zadanie ten problem rozwiązać. Umożliwiając ładowanie większości współczesnych ogniw, zwolniłby on potencjalnego użytkownika z konieczności posiadania różnego rodzaju ładowarek. Co więcej, poprzez ciągłe monitorowanie stanu baterii, zachowywałby jej żywotność, nie dopuszczając do przekroczenia pewnych granicznych wartości. Przeładowanie baterii, nadmierne rozładowanie czy też wysoka temperatura, wszystkie te czynniki są dla baterii szkodliwe[6], dlatego też należy na nie odpowiednio reagować.

W standardowych ładowarkach, sterowanie przepływającym przez obwód prądem odbywa się za pomocą regulatora prądu stałego. Otrzymać go można na różne sposoby, na przykład poprzez zastosowanie liniowego regulatora napięcia LM317[8], jak to przedstawione zostało na rysunku 1.1. Takie rozwiązanie dobrze spełnia swój cel, gdy mamy do



Rysunek 1.1. Ładowarka prądem stałym[7]

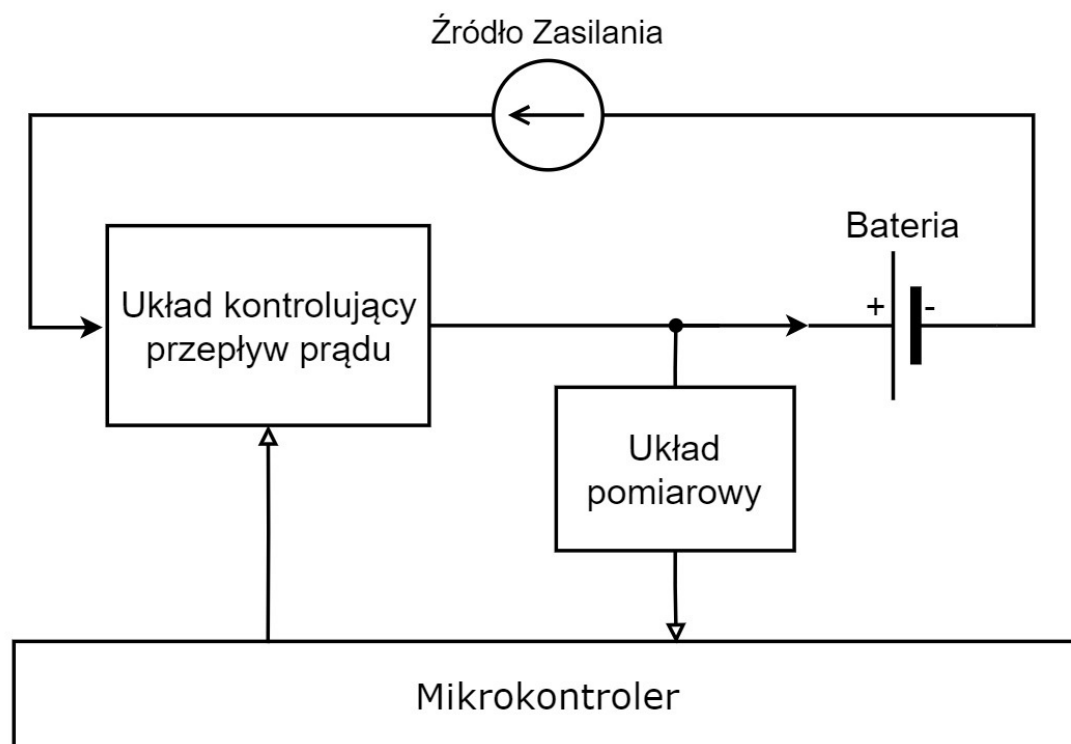
czynienia tylko z jednym konkretnym rodzajem baterii. Jednakże, aby uzyskać możliwość automatycznego sterowania prądem w większym zakresie, konieczne jest zastosowanie mikrokontrolera. Pozwala on bowiem precyzyjnie regulować pracę poszczególnych komponentów elektronicznych, na przykład poprzez użycie sygnałów PWM i co najważniejsze, jest programowalny. W przypadku ładowarki, działałby on na zasadzie pętli sprzężenia zwrotnego. Odczytując aktualny stan baterii i płynący w obwodzie prąd, mógłby decydować o nowym ustawieniu regulatora prądu w następnym obiegu pętli. Już w tym momencie jesteśmy w stanie nakreślić ogólny zarys tworzonego obwodu. Jego głównymi komponentami byłyby:

- regulator prądu;
- układ pomiarowy;
- mikrokontroler;

Wiedząc już, że mamy do dyspozycji układ mikroprocesorowy, możemy rozszerzyć nieco zdolności naszego urządzenia. Mianowicie wykorzystać koncepcję Internetu rzeczy¹ i umożliwić sterowanie nim w sposób bezprzewodowy. Komunikacja odbywałaby się za pośrednictwem zewnętrznego serwera w dwie strony. Użytkownik łącząc się z nim, mógłby przekazywać urządzeniu dane wejściowe, a ono przesyłałoby informacje o aktualnym stanie ogniw. Dzięki takiemu rozwiązaniu zyskujemy kilka istotnych rzeczy, mianowicie:

1. Pełną swobodę w tworzeniu interfejsu dla użytkownika — sposób w jaki będzie mu dane sterować ładowarką oraz w jaki przedstawiane mu będą zgromadzone dane zależy już tylko od programisty.
2. Przestajemy być ograniczeni przez z reguły niewielką moc obliczeniową mikrokontrolera — Możliwość gromadzenia większej ilości danych czy też szybszego ich przetwarzania.
3. Zwiększoną wygodę korzystania z systemu — Użytkownik mógłby korzystać z takich opcji, jak chociażby planowanie działania ładowarki w przyszłym czasie, monitorowanie żywotności baterii, przeglądanie statystyk.
4. Uproszczoną skalowalność systemu.

¹ koncepcja, według której jednoznacznie identyfikowalne obiekty mogą pośrednio lub bezpośrednio przysyłać, przetwarzać lub gromadzić dane za pośrednictwem sieci komputerowej.



Rysunek 1.2. Wstępny diagram układu

Istotne jest jednak aby działanie ładowarki nie było uwarunkowane posiadaniem połączenia z Internetem. Nie chcemy przecież, aby ładowanie baterii przerywane były przez chwilową utratę sygnału. Skorzystanie z manualnych przycisków czy diód informujących o poprawnym działaniu jest więc w tym przypadku koniecznością.

W kolejnym rozdziale zostaną omówione podstawy teorii ładowania współczesnych akumulatorów. Następnie sformułowane zostaną wymagania funkcjonalne projektowanego systemu. Mając je na uwadze, będzie można przystąpić do realizacji poszczególnych części prototypu, to jest układu elektrycznego, kodu mikrokontrolera i obsługi serwera. Na koniec przeprowadzone zostaną jego testy oraz wyciągnięte wnioski.

2. Akumulatory elektryczne i sposoby ich ładowania

Ogniwo, które ma zdolność wielokrotnego gromadzenia energii i jej oddawania nazywane jest akumulatorem elektrycznym. Są to chemiczne źródło prądu z uwagi na to, że energia jest magazynowana w postaci chemicznej. Przepuszczenie przez akumulator prądu elektrycznego z zewnętrznego źródła powoduje zajście odwracalnej reakcji chemicznej, tym samym zamieniając energię elektryczną w energię chemiczną[9].

2.1. Popularne rodzaje akumulatorów

Współcześnie używanych jest kilka rodzajów akumulatorów elektrycznych. Każdy z nich posiada pewne wady i zalety decydujące o ich zastosowaniu. Ograniczymy się tutaj do typów najczęściej spotykanych. Takich, które potencjalnie projektowana ładowarka miałaby za zadanie obsłużyć.

1. Akumulator Niklowo-kadmowy (NiCd): bateria starszego typu, powoli już znikająca z komercyjnego użycia. Dzieje się tak, ponieważ zawiera ona wysoce toksyczny kadm, a także jest wypierana przez pozostałe baterie o większej pojemności. Cechuje się większą odpornością na uszkodzenia w porównaniu do innych typów akumulatorów. Stosowana do pracy w cięższych warunkach z uwagi na dość wysokie zakresy maksymalnych temperatur[10]. Napięcia pojedynczego ogniwa rzędu 1.2V.
2. Akumulator Niklowo-metalowo-wodorkowy (NiMH): nowszy odpowiednik baterii NiCd. Najczęściej spotykany jako zastępniki nieładownych baterii alkalicznych, ale także swoje zastosowanie znalazł w samochodach hybrydowych. Jego rozwój w ostatnich latach sprawił, że wyparł on z rynku baterie NiCd, oferując większą pojemność, żywotność oraz gęstość energii[11]. Napięcie ogniwa takie samo jak w przypadku NiCd, to jest 1.2V.
3. Akumulator Litowo-jonowy (Li-Ion): Cechuje się wysokim napięciem pracy oraz dużą gęstością energii, jednakże dość kosztowny. Nie toleruje przeładowania, a w przypadku osiągnięcia zbyt dużej temperatury grozi pożarem lub wybuchem. Obecnie stosowany we wszelkiego rodzaju sprzęcie elektronicznym, głównie laptopach i telefonach komórkowych. Uznawany za najbardziej obiecujący typ akumulatorów. Nie występuje u niego tak zwany efekt pamięci², który można spotkać w bateriach NiCd i NiMH[9]. Na rynku można spotkać wiele typów baterii Li-Ion np. litowo-kobaltowe (LCO), litowo-manganowe (LMO) litowo-żelazowo-fosforanowe (Li-FePO₄). Baterie litowe, których elektrolit nie jest cieczą, nazywane są litowo-polimerowymi. Użycie polimerów pozwala na uzyskanie różnych kształtów przez baterię, na przykład formy cienkiego prostokąta spotykanego często w telefonach komórkowych. Napięcie nominalne ogniwa w przypadku baterii Li-Ion wynosi około 3.8V.

² zjawisko powodujące pozorną lub rzeczywistą utratę pojemności akumulatora. Obserwowane w przypadkach niepełnego rozładowania lub nadmiernego ładowania

4. Akumulator kwasowo-ołowiowy: Najstarszy typ akumulatora. Mimo tego wciąż popularny, z uwagi na niską cenę i możliwość dostarczania dużego prądu. Stosowany przede wszystkim w samochodach osobowych i przemyśle. Pojedyncze ogniwo tego typu posiada napięcie około 2V. Najczęściej spotykane są baterie złożone z sześciu ogniw, dające tym samym napięcie rzędu 12V.

Z wymienionych tutaj typów baterii najbardziej interesujące są dla nas ogniwa NiMH oraz Li-Ion. Nie tylko z powodu ich popularności na rynku czy dużej pojemności w stosunku do masy ale także dlatego, że spotykane są one w postaci pojedynczych ogniw, czego nie można raczej powiedzieć o akumulatorach kwasowo-ołowiowych.

2.2. Ładowanie akumulatorów

Wybór algorytmu ładowania powinien być w pierwszej kolejności ukierunkowany typem akumulatora. Zastosowanie sposobu, do którego bateria nie jest przystosowana, może w najlepszym przypadku skrócić jej żywotność, w najgorszym doprowadzić do zapłonu lub wybuchu.

Przed przejściem do omówienia poszczególnych algorytmów ładowania, dobrze jest wiedzieć czym jest i co oznacza wartość C. Podaje się ją często jako zamiennik natężenia prądu, gdy mowa jest o ładowaniu lub rozładowywaniu baterii. Jest ona równa stosunkowi przepływającego przez nią prądu, do teoretycznego prądu jaki dostarczyłaby w ciągu godziny, gdyby była całościowo rozładowana. Innymi słowy jest to iloraz prądu i pojemności. Na przykład dla baterii o pojemności 1Ah, 2C byłoby równe prądowi ładującemu o natężeniu 2A[6].

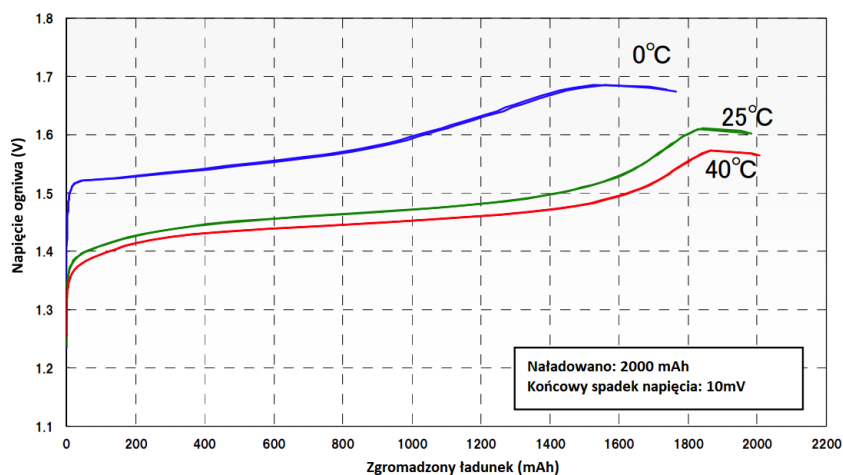
2.2.1. Ogniwa niklowo-kadmowe i niklowo-metalowo-wodorkowe

Zacznijmy od omówienia baterii NiCd i NiMH. Są one zazwyczaj ładowane stałym prądem. Ponieważ w miarę nabierania energii, ogniwa zmieniają swoje napięcie, w obwodzie ładującym nie obejdzie się bez pewnego rodzaju regulatora prądu³. W przypadku szybkiego ładowania, które producenci podają jako zalecaną metodę[3], natężenie prądu powinno wynosić około 1C i może być ono utrzymywane od początku aż do końca ładowania. Dodatkowym krokiem, jaki można podjąć po szybkim ładowaniu, jest zastosowanie prądu podtrzymania. Służy on zachowaniu pełnego naładowania ogniwa, po tym, jak główna część ładowania została zakończona. Prąd który jest wtedy stosowany, ma wartość od 0.03C do 0.05C. Jest on opcjonalny z uwagi na to, że może on doprowadzić baterię do przeładowania, lecz ogniwa niklowe, a przede wszystkim NiCd są na to dość odporne. Dzieje się tak za sprawą wytwarzanego we wnętrzu tlenu, który spowalnia procesy chemiczne zachodzące w baterii w przypadku przeładowania[6]. Oznacza to też, że w przypadku baterii niklowych nasza ładowarka nie musi zachowywać dużej precyzji

³ Bez regulatora prądu, natężenie w trakcie ładowania malałoby, utrudniając wykrycie warunku końcowego

ładowania. Podsumowując, proces ładowania baterii NiCd i NiMH przedstawia się dość prosto, jego fazami są:

1. prąd stały
2. prąd podtrzymania (opcjonalnie)



Rysunek 2.1. Charakterystyka ładowania baterii NiMH[12]

Sprawa się jednak komplikuje, jeżeli chodzi o wykrycie momentu zakończenia tego procesu. Zaczniemy od sposobu najprostszego, czyli ładowania przez stały okres czasu. Umieszczamy naszą baterię w ładowarce i wiemy dokładnie za ile minut do niej wrócić, aby zastać nasze baterie naładowane. Jak się można domyśleć, jest to dość naiwne podejście. Po pierwsze, bateria nie musi być w pełni rozładowana, gdy umieścimy ją w urządzeniu, przez co ryzykujemy jej przeładowaniem. Po drugie, proces ładowania może przebiegać inaczej w zależności od aktualnej temperatury otoczenia i ogniwa. Po trzecie, charakterystyki baterii niklowych, nawet tych o tej samej pojemności, nie są sobie równe. Zależą one od budowy danego akumulatora ale także stopnia jego zużycia. Może to zaskakiwać, ale najprostsze ładowarki działają właśnie na tej zasadzie[13]. Istnieje jednak warunek, który może częściowo pomóc zastosować metodę stałego czasu. Ponieważ baterie nigdy nie zatrzymują całego ładunku, który przez nie przepływa⁴, maksymalny czas można przyjąć jako odwrotność wartości C.

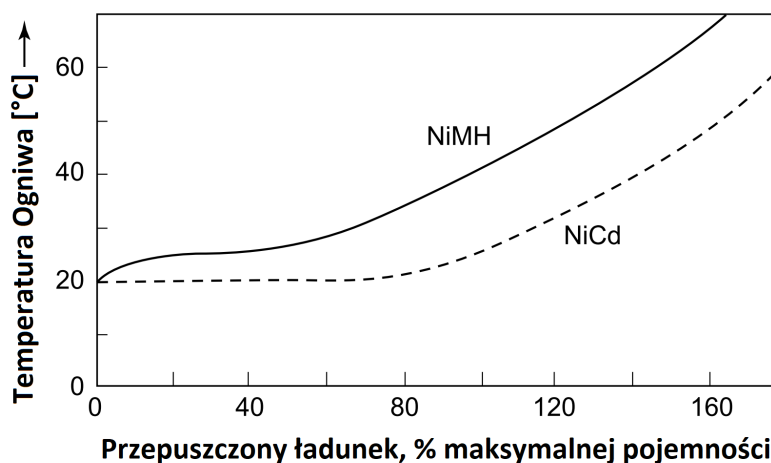
Drugim sposobem jaki można zastosować, jest ustalenie maksymalnego napięcia ogniwa. W momencie osiągnięcia go ładowanie jest przerywane. Przyjrzyjmy się jednak charakterystyce ładowania przedstawionej na rysunku 2.1, dostarczonej przez producenta akumulatorów NiMH. Jak można zauważyć, maksymalne osiągnięte napięcie różni się, w zależności od temperatury otoczenia. Jest to główny czynnik decydujący o niskiej skuteczności tej metody.

⁴ Dzieje się tak, chociażby z uwagi na stratę energii w postaci wytwarzanego ciepła

Lepszym pomysłem może okazać się liczenie zmian napięcia baterii w czasie. Jeżeli ich liczba osiągnęłaby wystarczające niski poziom, można byłoby uznać ładowanie za zakończone, gdyż jak widać na rysunku 2.1 napięcie ogniwa pod koniec ładowania stabilizuje się.

Kolejną i najczęściej wykorzystywaną w przypadku szybkiego ładowania metodą, jest wykrywanie końcowego spadku napięcia. Przy bateriach niklowych można zaobserwować zjawisko niewielkiego spadku napięcia, gdy zbliżają się one do osiągnięcia maksymalnego stopnia naładowania. Dla charakterystyki na diagramie 2.1 jest to zaledwie 10mV, w przypadku ładowania prądem 1C. Sposób ten jest jednak odradzany, gdy ładowanie odbywa się wartościami prądu mniejszymi niż 0.5C, ponieważ spadek napięcia jest wtedy prawie niezauważalny.

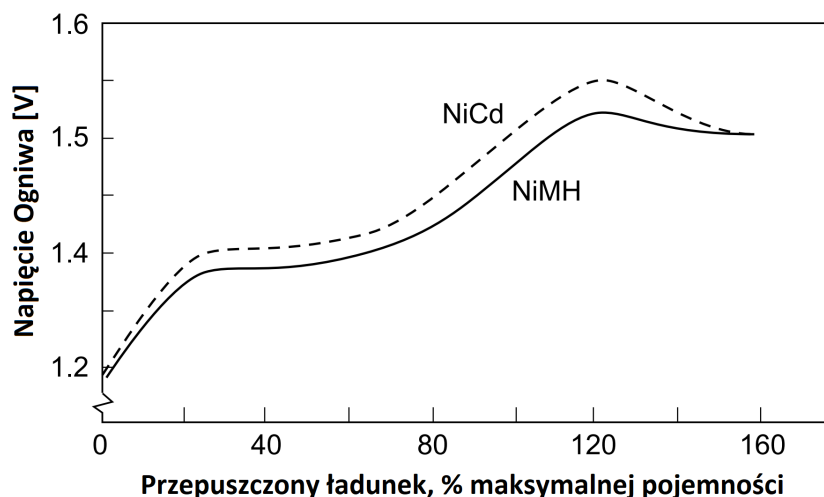
Następna metoda opiera się na pomiarze temperatury ogniwa. Analogicznie jak w przypadku napięcia, ładowanie może zostać zakończone w przypadku gdy temperatura osiągnie maksymalny poziom. Metodę tę stosować powinno się niezależnie od tego, czy koniec ładowania wykrywamy w inny sposób, jako że temperatura jest głównym czynnikiem uszkadzającym ogniwo.



Rysunek 2.2. Temperatura ogniw w zależności od przepuszczonego ładunku[6]

Ostatni omawiany tutaj sposób polega na mierzeniu temperatury w czasie. Im więcej ładunku skumulowanego zostanie w baterii, tym szybciej jej temperatura będzie rosła. Zależność tą zaobserwować można na diagramie 2.2. Jako rozsądną wartość kończącą przyjmuje się 1°C/min, aczkolwiek zależy ona przede wszystkim od tempa ładowania.

Metody ładowania baterii NiCd i NiMH są niemal identyczne. Czy znaczy to, że ta sama ładowarka jest w stanie obsłużyć obydwa te typy? Okazuje się, że niekoniecznie. Zwróćmy jeszcze raz uwagę na diagramy 2.2 oraz 2.3. Na pierwszym zauważyć można, że temperatura NiCd rosnąć zaczyna dopiero w późnym etapie ładowania. Sprawia to, że jest ona niższa i tym samym bardziej odporna na przeładowanie. Na drugim rysunku zaobserwować można kolejną znaczącą różnicę.



Rysunek 2.3. Porównanie charakterystyk ogniw NiCd i NiMH[6]

Spadek napięcia jest dużo bardziej widoczny przy ogniwie NiCd. Dochodzimy więc do wniosków, że baterie te są nie tylko prostsze do naładowania, ale również trudniejsze do przypadkowego przeładowania niż ich metalowo-wodorkowi odpowiednicy.

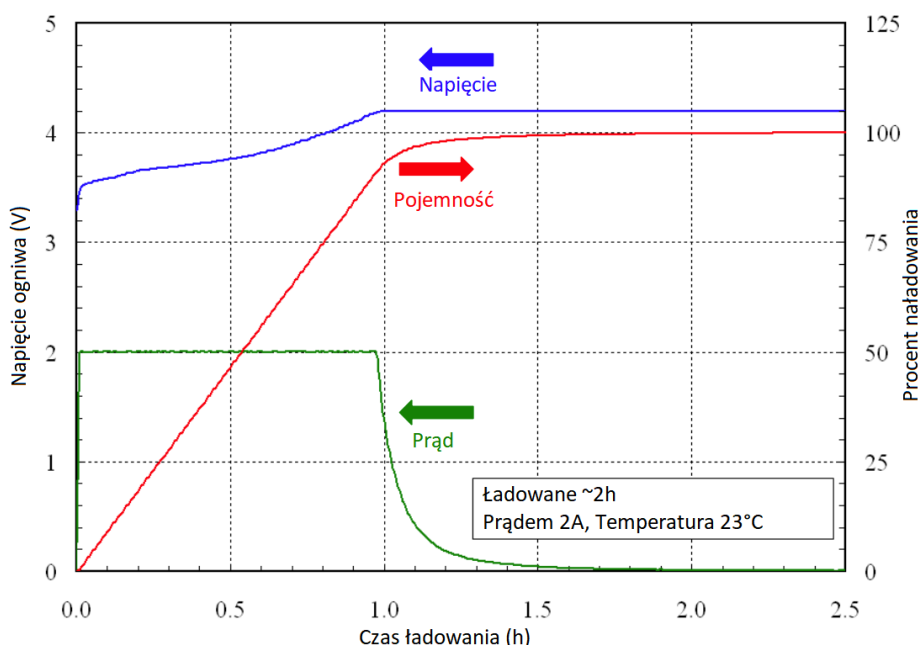
2.2.2. Ogniwa litowo-jonowe

Baterie Li-Ion wymagają innego traktowania jak akumulatory niklowe. Są one dużo wrażliwsze na potencjalne błędy. Cały proces ładowania jest ściśle ustalony i nie ma zbyt wielu miejsc w jakich można byłoby go urozmaicić. Odbywa się on następującymi krokami:

1. ładowanie wstępne (opcjonalne)
2. stały prąd
3. stałe napięcie

Ładowanie wstępne różni się od kroku drugiego jedynie natężeniem. Jest ono dużo niższe, z reguły wynosi 1/10C. Etap ten służy stopniowemu wprowadzeniu baterii w proces ładowania poprzez odnowienie aktywnych materiałów[14]. Jest ono opcjonalne, ale zalecane jest jego wprowadzenie, jeżeli ogniwo mogło być znacząco rozładowane, to jest poniżej 3V. Przy ponownym osiągnięciu tej granicy można przystąpić do etapu drugiego, czyli ładowania prądem stałym. W przypadku akumulatorów Li-Ion zalecane natężenie prądu to 0.5C do 1C. Trzeci i ostatni krok następuje po osiągnięciu przez ogniwo określonego przez producenta napięcia maksymalnego, z reguły jest to 4.2V lub 4.1V. W tym momencie ładowarka przechodzi w tryb stałego napięcia, dostarczając do baterii coraz to mniejszy prąd i kończąc ładowanie przy wartości 0.05C — 0.1C. Przykładowa charakterystyka widoczna jest na rysunku 2.4. Jak widać, w przypadku akumulatorów Li-Ion nie ma dużego wyboru, jeżeli chodzi o czynniki kończące. Zastosowanie ograniczenia czasowego oraz temperatury jest jednak zalecane jako warunek dodatkowy.

Mimo że cały proces jest dość liniowy, nie jest on pod żadnym pozorem prosty. W naszym przypadku może okazać się trudny do zrealizowania z uwagi na dość wąskie progi maksymalne. Ponieważ baterie Li-Ion są wysoce wrażliwe na przeładowanie[6], napięcie nie powinno przekraczać 1% wartości maksymalnej. Precyzja odczytów będzie tutaj odgrywała kluczową rolę.



Rysunek 2.4. Charakterystyka ładowania ogniw Li-Ion[15]

2.2.3. Ogniw kwasowo-ołowiowe

Proces ładowania akumulatorów kwasowo-ołowiowych można bardzo dobrze opisać w sposób następujący. Jest to ładowanie ogniw Li-Ion tylko z mniejszą wymaganą precyzją. W tym przypadku kolejnymi etapami byłyby:

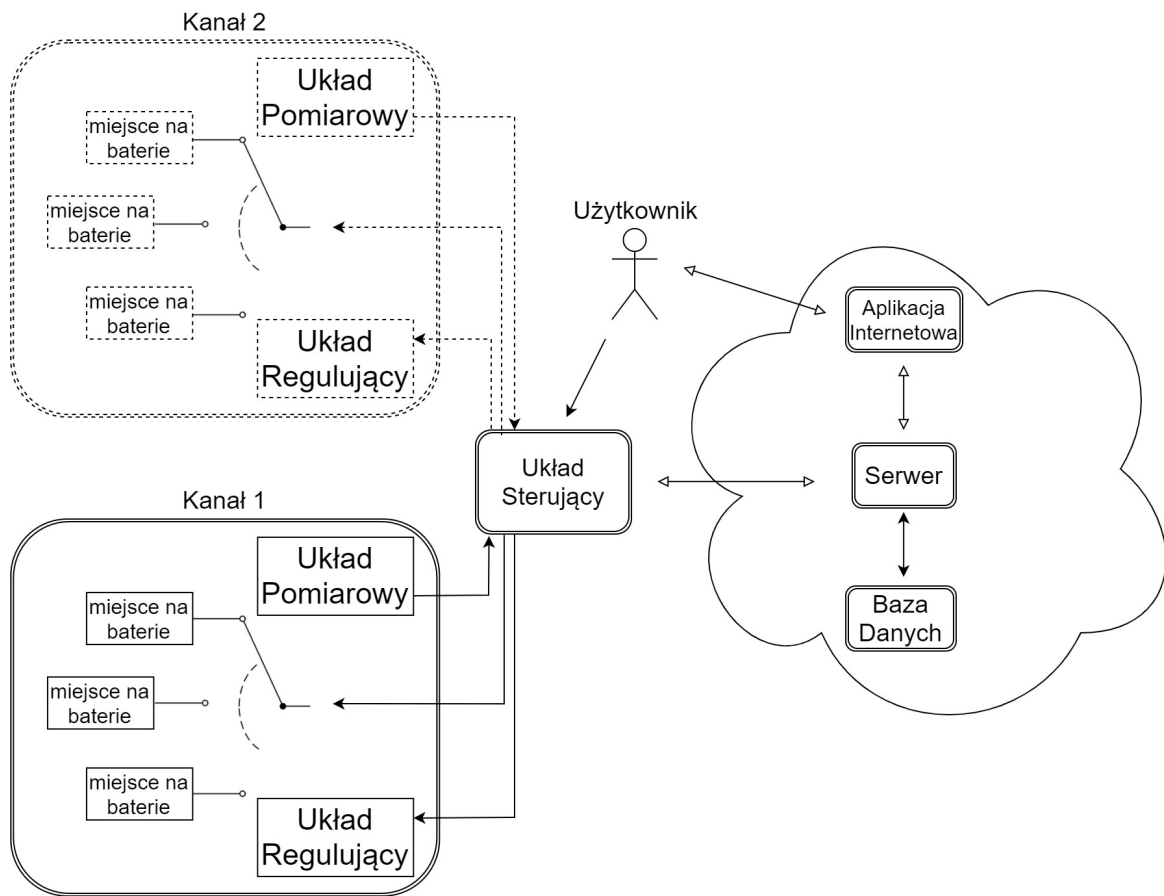
1. stały prąd
2. stałe napięcie
3. prąd podtrzymania

Typowe napięcie pojedynczego ogniwa, które chcemy utrzymywać w drugim etapie to 2.3 – 2.45V. Jak widać w odróżnieniu od akumulatorów Li-Ion tutaj zastosowanie prądu podtrzymania jest znów możliwe[16].

Podsumowując, samych metod ładowania baterii nie jest dużo. Można wręcz posuwać się do stwierdzenia, że wszystkie rodzaje "kroków" jakie do tej pory zostały omówione to lekko zmodyfikowane tryby stałego prądu lub stałego napięcia. Na przykład, ładowanie wstępne to ładowanie stałym prądem tylko o mniejszym natężeniu. Ładowanie prądem podtrzymania podobnie. Daje nam to do wiadomości, że budowana ładowarka musi

tylko obsługiwać te dwie konkretne metody. Tak naprawdę istnieją również inne sposoby ładowania baterii, o których nie wspomniano, chociażby ładowanie pulsacyjne, czy też ładowanie pakietów baterii z balansowaniem. Jednakże, aby zachować prostotę przygotowanego w przyszłych rozdziałach urządzenia, nie były one omawiane szczegółowo.

3. Projekt systemu



Rysunek 3.1. Schemat blokowy systemu

W projektowanym systemie wyróżnić można trzy główne komponenty, są to:

- kanał
- układ sterujący
- serwer

Kanał jest zbiorem elementów elektronicznych, pozwalających na ładowanie i rozładowywanie pojedynczych baterii. W jego skład wchodzi układ regulujący przepływem prądu oraz układ pomiarowy dostarczający dane do układu sterującego. W kanale występować może kilka miejsc na baterie. Ponieważ tylko jedna może być ładowana w tym samym momencie, obsługiwane są one na zasadzie kolejki. Jeżeli użytkownik potrzebuje możliwości ładowania kilku baterii jednocześnie, potrzebuje on do tego kilku kanałów.

Układ sterujący kontroluje działanie kanałów. Sposób w jaki to robi jest zdefiniowany przez użytkownika za pomocą aplikacji internetowej. Na przykład, zadaje mu on do naładowania baterie numer trzy z kanału drugiego sposobem stałego prądu. Możliwa jest też manualna kontrola układem sterującym jednakże jest ona ograniczona.

Serwer pośredniczy w wymianie danych pomiędzy układem sterującym a użytkownikiem. Gromadzi informacje o nim i jego preferencjach w bazie danych.

3.1. Założenia funkcjonalne systemu

W pierwszym rozdziale przedstawiony został ogólny zarys projektowanego systemu. Opierając się dodatkowo na poznanych sposobach ładowania akumulatorów, można przystąpić do szczegółowego sformułowania jego wymagań. Prezentują się one następująco:

1. **Możliwość ładowania różnych typów baterii** – urządzenie potrafi zastosować wszystkie wymienione w rozdziale drugim metody ładowania. Przedział dostarczanych natężeń i napięć powinien być na tyle szeroki, aby umożliwiał sprawną pracę z popularnymi rodzajami baterii.
2. **Możliwość rozładowywania akumulatorów** – dostępna jest oddzielna część obwodu służąca wytracaniu energii. Ogniwo nie powinno wymagać zmiany miejsca położenia w momencie przełączenia funkcjonalności.
3. **Monitorowanie stanu baterii** – wykonywanie pomiarów napięcia baterii, prądu przepływającego przez nią oraz jej temperatury.
4. **Wykrywanie końca ładowania** – wszystkie wymienione w poprzednim rozdziale metody na wykrycie końca ładowania, są dostępne. Wybór, które z nich zostaną użyte, należy do użytkownika.
5. **Obsługa wielu kanałów** – system jest w stanie obsługiwać równolegle więcej niż jedno ogniwo oraz ma możliwość kolejgowania zadań.
6. **Dodawanie kolejnych kanałów** – na swoje życzenie posiadacz ładowarki może dodawać kolejne miejsca na baterie oraz kanały.
7. **Bezprzewodowe sterowanie** – użytkownik łącząc się ze stroną internetową, jest w stanie za jej pośrednictwem wysłać polecenia do ładowarki.
8. **Przegląd statystyk** – użytkownik korzystając ze strony internetowej, ma możliwość przeglądania aktualnego stanu urządzenia i umieszczonych w nim ogniw.
9. **Możliwość zakładania kont użytkownika** – strona internetowa obsługuje system rejestracji i logowania.
10. **Działanie w trybie offline** – utrata połączenia z serwerem nie skutkuje przerwaniem działania reszty systemu. Dodatkowo, podstawowa obsługa ładowarki, jak zmiana trybu kanału, dostępna jest w sposób manualny.
11. **Wysoka dokładność** – pomiary mają być dokonywane z maksymalną możliwą w danym przypadku dokładnością. Ponieważ obsługa akumulatorów Li-Ion jest uwzględniona, dobrym punktem odniesienia jest 1% ich dopuszczalnego napięcia. (~40mV)
12. **Ciągłość działania systemu** – efektywność pracy nie powinna być zaburzana, chociażby przez takie czynniki jak temperatura.
13. **Zapewnienie długiej żywotności baterii** – sytuacje niekorzystnie wpływające na baterie powinny być unikane.

14. **Prostota w stosowaniu modyfikacji** – części składowe systemu powinny być od siebie niezależne. To samo tyczy się oprogramowania i jego struktur klas. Ma to na celu uproszczenie wprowadzania zmian oraz rozwijania przyszłego systemu.

3.2. Struktura profili ładowania

Działanie przygotowanego narzędzia opiera się na konstruowaniu, a następnie użyciu tak zwanych profili ładowania. Są to struktury danych, które odpowiadają jednemu etapowi, jaki wykonywany jest podczas ładowania akumulatora. Zawierają one następujące pola:

- metoda ładowania
- oczekiwane napięcie baterii
- oczekiwane natężenie prądu
- maksymalne napięcie baterii
- maksymalne natężenie prądu
- maksymalna temperatura baterii
- maksymalny czas ładowania
- odstęp czasowy pomiędzy liczeniem różnicy napięć
- minimalna różnica napięć w odstępie czasowym
- odstęp czasowy pomiędzy liczeniem różnicy temperatur
- maksymalna różnica temperatur w odstępie czasowym
- końcowy spadek napięcia

gdzie metoda ładowania przybiera wartość **stały prąd** lub **stałe napięcie**⁵.

Pierwsze trzy pola pozwalają manipulować charakterystyką ładowania. Pozostałe dziewięć odpowiada za parametry kończące. Przy użyciu kolejki takich struktur jest się w stanie opisać dowolną krzywą ładowania. Wszystkie poznane do tej pory czynniki kończące są również uwzględnione. Przykładowo, dla akumulatora niklowego o pojemności 1000mAh zastosować można następujące po sobie dwa profile:

profil 1:

- metoda ładowania: stały prąd
- oczekiwany prąd: 1000mA
- końcowy spadek napięcia: 10mV

profil 2:

- metoda ładowania: stały prąd
- oczekiwany prąd: 30mA
- maksymalny czas: 30min

⁵ Tak jak było już to wspomniane, większość stosowanych metod ładowania to zmodyfikowane wersje stałego prądu lub stałego napięcia. Nie ma więc potrzeby wyróżniać pozostałych sposobów, skoro jesteśmy w stanie opisać je za pomocą jednej z tych dwóch wartości.

Bateria ładowana byłaby początkowo stałym prądem o wartości jednego ampera. Gdy wykryty zostałby niewielki spadek napięcia, rozpocząłby się etap ładowania podtrzymania, który trwałby równe 30 minut. Nic nie stoi na przeszkodzie, aby wypełnić wszystkie dostępne pola profilu. Urządzenie zwracałoby wtedy uwagę na większą liczbę czynników kończących, tym samym zapewniając większą ochronę przed przeładowaniem. Przyjrzyjmy się jeszcze, jak mógłby wyglądać przykładowy zestaw profili, służący naładowaniu ogniwa Li-Ion o pojemności 2000mAh:

profil 1:

- metoda ładowania: stały prąd
- oczekiwany prąd: 200mA
- oczekiwane napięcie: 3000mV

profil 2:

- metoda ładowania: stały prąd
- oczekiwany prąd: 1500mA
- oczekiwane napięcie: 4200mV

profil 3:

- metoda ładowania: stałe napięcie
- oczekiwane napięcie: 4200mV
- oczekiwany prąd: 50mA

Jeżeli dana bateria będzie znacznie rozładowana, w pierwszej kolejności nastąpi ładowanie wstępne. Gdy ogniwo zostanie pobudzone i osiągnie napięcie 3V lub jeżeli z takim napięciem już zostanie w ładowarce umieszczone, ta przejdzie do drugiego etapu czyli, ładowania prądem stałym. Odczytanie po pewnym czasie na akumulatorze napięcia maksymalnego 4.2V, będzie skutkowało przełączeniem na ostatni profil i stopniowym zmniejszaniu natężenia prądu, aż do 50mA. Tutaj również użycie pozostałych pól jest wskazane, w szczególności tych związanych z temperaturą. Jednakże, zauważyć można, że niektóre parametry nie znajdują swojego zastosowania w przypadku baterii Li-Ion. Mowa tu o spadku napięcia oraz różnicy napięcia w czasie. Być może więc dostosowywanie profili do poszczególnych rodzajów akumulatorów byłoby lepszym pomysłem. Opisywany tutaj system rezygnuje jednak z posiadania typu baterii jako jednego z używanych parametrów. Zachowuje się więc on w stosunku do każdej równoważnie. Niesie to ze sobą pewne komplikacje, chociażby właśnie nadmierne struktury profili. Z drugiej jednak strony logika systemu staje się dużo prostsza i przejrzysta.

Za przygotowywanie profili odpowiada użytkownik. Dokonuje tego za pomocą aplikacji internetowej. Te następnie przesyłane są do serwera, skąd powiązana ładowarka może je pobrać i zaaplikować. Szczegółowy przegląd poszczególnych komponentów systemu rozpoczniemy od najniższego poziomu, to jest obwodu elektrycznego.

3.3. Obwód elektryczny

Na rynku spotkać można akumulatory o przeróżnych napięciach. Im większe napięcie zasilania zostanie zastosowane w ładowarce, tym większy wachlarz baterii będzie w niej możliwy do naładowania. Z drugiej jednak strony zbytne rozszerzenie zakresu napięć wiąże się z wymogiem dokładniejszej regulacji, a przede wszystkim z większą ilością ciepła potrzebnego do rozprowadzenia. W prototypie zdecydowano się użyć źródła napięcia 12V, dostarczającego maksymalnie 2A. Z takimi wartościami jesteśmy więc w stanie ładować wszystkie dostępne ogniwa oraz baterie do 9V. Poza zasięgiem zaś stają się potężniejsze akumulatory, jak chociażby większość kwasowo-ołowiowych, których to napięcie wynosi zazwyczaj 12V. Aby nie doprowadzać źródła zasilania do granic jego możliwości, urządzenie projektowano z myślą maksymalnego przepływającego prądu na poziomie 1.5A. Moc, którą należy rozłożyć w najgorszym przypadku, jest więc równa

$$P = 12V * 1.5A = 18W$$

Jest to wartość jeszcze nie stanowiąca dużych problemów. Oczywiście użycie tego konkretnego źródła zasilania nie jest obowiązkowe. W przyszłych iteracjach projektu mogłoby być ono zastąpione innym.

Moduł systemu odpowiedzialny za obieg prądu elektrycznego można podzielić na trzy grupy. Do pierwszej z nich należą elementy regulujące, które to określają natężenie prądu płynące przez obwód. Drugi, równie ważny zbiór, to sensory odczytujące informacje o stanie ogniw. Ostatnia grupa to obwód rozładowujący i przełączniki przełączające akumulatory pomiędzy nim a obwodem ładującym.

Jako mikrokontroler nadzorujący pracę całego urządzenia zastosowano Arduino Mega 2560. Taka decyzja została podjęta we wczesnej fazie projektu. Ukierunkowana była ona przede wszystkim istnieniem szerokich zasobów wiedzy dotyczących rodziny Arduino. Z biegiem czasu okazało się, że nie jest to optymalne rozwiązanie, ale o tym w późniejszej części tego rozdziału.

3.3.1. Regulator prądu

Jeżeli źródło zasilania dostarcza do obwodu stałe napięcie, jedynym sposobem na modyfikację natężenia prądu, zgodnie z prawem Ohma, jest zmiana rezystancji obwodu. W praktyce można zastosować jeszcze jeden sposób, mianowicie odłożenie stałego napięcia na elemencie układu. Ponieważ stosowany tutaj regulator prądu musi być przede wszystkim sterowalny za pomocą mikrokontrolera, do wyboru nie pozostaje dużo możliwości. Są nimi:

- potencjometr cyfrowy
- liniowy regulator napięcia
- tranzystor typu MOSFET

Niestety pierwszą opcję można od razu odrzucić z uwagi na to, że nie powstają potencjo-

metry cyfrowe o tak małych krokach, jakie by nas interesowały (mowa tutaj o 100 mΩ). Pozostałe dwie mają jednak szansę wpasować się w układ. Regulator napięcia jest świetnym wyborem z uwagi na liniową regulację. Działa on na zasadzie takiej, że utrzymuje na swoim wyjściu napięcie równe sumie napięcia na pinie odniesienia oraz stałej określonej wartości, w przypadku popularnego LM317 równej 1.25V. Zakładając, że będzie on sterowany 255 krokami⁶, i rezystancja układu wynosić będzie 6Ω, możliwość regulacji osiągnie poziom około:

$$\frac{12 - 1.25V}{6\Omega} * \frac{1}{255} = 7mA$$

Ma on jednak również swoje wady. Po pierwsze, nie uzyskuje się przy nim pełnego spektrum dostępnych natężeń z uwagi na minimalne napięcie wyjściowe. Dodatkowo jego parametry regulacji pogarszają się wraz ze wzrostem natężenia oraz zbliżaniem wartości napięcia wyjściowego do napięcia wejściowego. Specyfikacja LM317[8] jako granicę dobrej regulacji podaje wartość:

$$V_{in} > V_{out} + 3$$

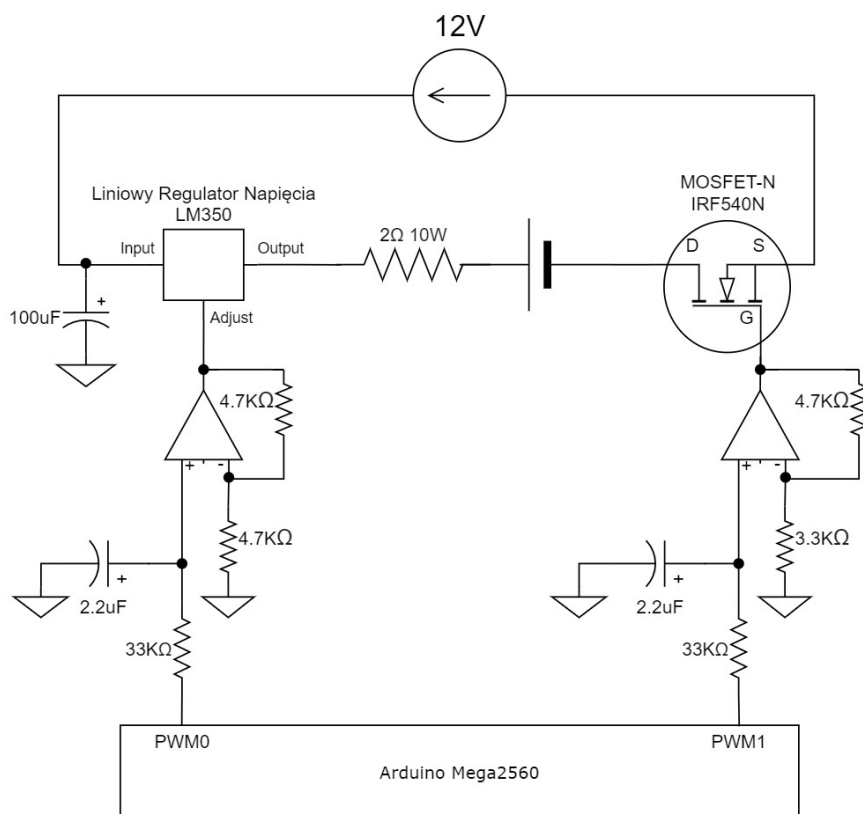
gdzie V_{in} to napięcie wejściowe, V_{out} napięcie wyjściowe.

Odpowiednio dobrany tranzystor typu MOSFET może równie dobrze służyć jako regulator prądu. W tego rodzaju tranzystorach przepływające przez nich natężenie jest zależne od przyłożonego do bramki napięcia. W przypadku ładowarki najlepszym wyborem byłby MOSFET z kanałem wzbogaconym, który to w podstawowym stanie jest wyłączony. W odróżnieniu od regulatora napięcia, tranzystor daje możliwość regulacji w pełnym zakresie, przykładowo od 0 do 1.5A, ale z drugiej strony regulacja ta nie jest liniowa. Dodatkowo nie wszystkie 255 kroków mikrokontrolera będzie wykorzystanych do regulacji, gdyż w części z nich tranzystor będzie zamknięty lub już w pełni otwarty.

Regulator, jaki ostatecznie zastosowano w prototypie, składa się z regulatora napięcia LM350 oraz z tranzystora typu MOSFET-N IRF540N. Rozwiązanie takie pozwala wykorzystać zalety obydwu komponentów, które świetnie się uzupełniają. Dokładność regulacji znacząco wzrasta przy takim ustawieniu, ponieważ teoretycznie mikrokontroler ma do dyspozycji 255² kroków. Wadą takiego rozwiązania jest utrudnione odpowiednie zaprogramowania regulatora.

Co do wyboru konkretnego komponentu regulatora i tranzystora, LM350 został zastosowany z uwagi na jego maksymalne dopuszczalne natężenie równe 3A. W przypadku LM317 jest to tylko 1.5A. Byłaby to wartość niebezpiecznie bliska sztucznie ustalonej granicy natężenia w budowanym prototypie. Przemawiającymi za zastosowaniem IRF540 parametrami była jego bardzo niska minimalna rezystancja 40 mΩ oraz niska wartość napięcia bramka-źródło, która taką rezystancję ustala, to jest 10V[17]. Ponieważ zastosowany mosfet jest typu N, napięcie na jego źródle musi wynosić 0V. Oznacza to, że jego miejsce jest na samym końcu obwodu.

⁶ taką ilość kroków posiada wyjście PWM arduino

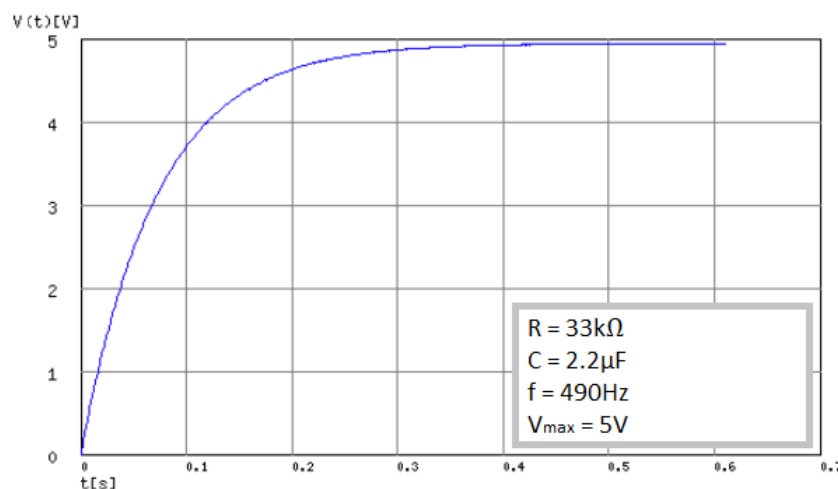


Rysunek 3.2. Zastosowany w prototypie układ regulatora prądu

Szczegółowy wygląd regulatora został przedstawiony na diagramie 3.2. Oprócz dwóch głównych komponentów, jakimi są LM350 oraz IRF540N można tam zaobserwować dwa prawie identyczne sposoby połączenia ich z mikrokontrolerem. W pierwszej kolejności stosowany jest filtr dolnoprzepustowy składający się z rezystora i kondensatora. Jego zadaniem jest przekształcenie sygnału PWM mikrokontrolera na sygnał analogowy. Gdyby element ten był pominięty, regulator napięcia miałby bardzo niestabilny punkt odniesienia, a pojemność bramki tranzystora wahałaby się. Innymi słowy, gwarantuje on stabilniejszą regulację kosztem jedynie rezystora i kondensatora. Ponieważ w projektowanej ładowarce stawiamy raczej na precyzję aniżeli czas odpowiedzi, zastosowane parametry filtra są dość wysokie. Dostarczany przez Arduino sygnał PWM ma częstotliwość 490Hz⁷, dokładając do tego opornik o rezystancji 33 kΩ i kondensator o pojemności 2.2 μF otrzymujemy charakterystykę skokową widoczną na diagramie 3.3. Odpowiedź czasowa jest na poziomie 0.3s, a szczytkowa okresowa zmiana napięcia jest ledwo zauważalna. Takie atrybuty są jak najbardziej akceptowalne.

Kolejny element zastosowany w obwodzie regulatora to wzmacniacz operacyjny, a dokładnie LM358. Jest to standardowy rodzaj wzmacniacza, więc bez problemu można go zastąpić innym podobnym.

⁷ tylko 2 piny mają możliwość dostarczania sygnału o częstotliwości 980Hz



Rysunek 3.3. Zasymulowana charakterystyka skokowa filtra dolnoprzepustowego[18]

Ponieważ maksymalne napięcie, jakie Arduino Mega jest w stanie do obwodu dostarczyć, wynosi 5V, bez wzmacniacza operacyjnego zakres pracy regulatora napięcia wynosiłby jedynie 1.25V-6.25V. Nie jest to oczywiście zadowalająca wartość. Po dwukrotnym wzmocnieniu sygnału przedział ten wynosiłby już 1.25-11.25 co jest dużo lepszym wynikiem. Taki efekt osiągnąć można poprzez zastosowanie wzmacniacza nieodwracającego z równymi wartościami rezystora w pętli sprzężenia zwrotnego i na wejściu odwracającym. Stosowanie większego wzmocnienia niż dwukrotne nie przynosi zbyt wielu rezultatów. Tak jak wcześniej wspomniano, regulator napięcia nie zachowuje obiecanych parametrów liniowej regulacji, gdy napięcie na wyjściu mało różni się od wejściowego i zawsze jakiś spadek napięcia na nim i tak wystąpi. Z drugiej jednak strony większe wzmocnienie przyda się w przypadku tranzystora IRF540N, gdyż im większe napięcie zastosujemy na jego bramce, tym mniejszą rezystancją będzie się on odznaczał. Zastosowany jest więc przy nim inny układ tranzystorów:

$$A = 1 + \frac{R1}{R2}$$

$$A = 1 + \frac{4.7k\Omega}{3.3k\Omega} = 2.42$$

$$V_{out} = A * V_{in}$$

$$V_{out} = 2.42 * 5V = 12.12V \quad (1)$$

Gdzie: A – wzmocnienie

R1 – rezystor w pętli sprzężenia zwrotnego

R2 – rezystor na wejściu odwracającym

Vin/Vout – napięcie na wejściu/wyjściu

Ponieważ wzmacniacze zasilane są przy użyciu tego samego źródła co główna pętla obwodu, ich maksymalne napięcie wyjściowe jest równe napięciu zasilania, to jest 12V. Wzmocnienie uzyskane przy użyciu rezystorów 4.7 k Ω i 3.3 k Ω daje więc możliwość osiągnięcia takiej wartości, jak to można zauważyć w równaniu 1.

Oprócz filtrów dolnoprzepustowych i wzmacniaczy, w regulatorze prądu zastosowane są jeszcze dwa elementy bierne. Kondensator o pojemności 100 μ F, który dołączony jest przed wejściem regulatora, służy przefiltrowaniu przepływającego przez niego prądu. Dodatkowo w przypadku gdy układ straci chwilowo zasilanie, blokuje przepływ prądu z baterii w nieodpowiednim kierunku, to jest przez wyjście regulatora, co mogłoby go poważnie uszkodzić[8]. Robi to na tyle długo, że tranzystor zdąży się już zamknąć z uwagi na brak zasilania wzmacniacza operacyjnego.

Ostatnim elementem regulatora prądu jest opornik umiejscowiony pomiędzy regulatorem napięcia a baterią. W prototypie zastosowane są szeregowo połączone dwa rezystory o oporze 1 Ω i mocy 5 W. Wysoka maksymalna moc jest w tym przypadku kluczowa z uwagi na to, że będą na nich odkładać się znaczne ilości energii. Zastosowanie w tym miejscu opornika daje nam dwie rzeczy. Pierwszą, jest oczywiście ograniczenie przepływającego prądu. Sterowanie nim za pomocą samego tranzystora byłoby bardziej niebezpieczne. Druga sprawa to praca jako punkt odniesienia dla sensorów.

3.3.2. Sensory

Aby mikrokontroler mógł poprawnie prowadzić zachodzące w urządzeniu procesy, musi mieć on odniesienie do ich aktualnego stanu. Jeżeli chodzi o samo ładowanie, informacje, które są mu potrzebne to:

- płynący w obwodzie prąd
- napięcie baterii
- temperatura baterii

Wartość natężenia prądu w najprostszy sposób można policzyć, korzystając z umiejscowionego w głównej pętli rezystora. Zgodnie z prawem Ohma natężenie równe jest:

$$I = \frac{U}{R}$$

gdzie R już znamy i wynosi ono 2 Ω . Pozostaje więc do wyznaczenia spadek napięcia na tym rezystorze. Aby go otrzymać, wystarczy oczywiście zmierzyć napięcie na obydwu jego końcach i wyciągnąć różnicę. Posłużymy się w tym celu wbudowanym w Arduino konwerterem analogowo-cyfrowym. Posiada on rozdzielczość 10 bitów, czyli dostępnych jest 1024 poziomów kwantyzacji. Najwyższe napięcie jakie spotkamy w obwodzie może być równe 12V. Daje nam to do wiadomości, że uzyskamy dokładność odczytów na poziomie:

$$\frac{12V}{1024} = 11.72mV$$

Aby odczytać napięcie w danej części obwodu, należy połączyć ją z dowolnym pinem analogowym⁸. Jednakże bezpośredni pomiar nie jest w tym przypadku możliwy, z uwagi na maksymalne dopuszczalne napięcie na pinach Arduino równe 0.5V + napięcie zasilania(5V). Co więcej, w prototypie używane będzie wbudowane w mikrokontroler odniesienie 1.1V, które gwarantuje większą dokładność pomiarów[19]. Dzieje się tak, gdyż w trybie podstawowym Arduino jako odniesienie wybiera napięcie zasilania. Jak się można domyśleć, nie należy ono do najstabilniejszych. Niestety wbudowane odniesienie ma też swoją wadę, jego prawdziwa wartość rzadko jest równa 1.1V. Jak podaje producent, przybiera ona liczbę z zakresu od 1.0V do 1.2V[20]. Istnieje jednak sposób na jej odczytanie. Instrukcja jak wyznaczyć dokładną miarę jest umieszczona na końcu tego dokumentu. W przypadku mikrokontrolera użytego w prototypie:

$$V_{ref} = 1.0765V.$$

To co więc pozostaje do osiągnięcia to obniżenie napięcia odczytywanego w danym punkcie do wartości mniejszej lub równej V_{ref} . Dokonać tego można za pomocą dzielników napięcia. Aby nie tracić rozdzielczości pomiaru, użyte w dzielnikach rezystory powinny być dokładnie dobrane. W pierwszej kolejności należy więc wyznaczyć maksymalne napięcie, jakie może w odczytywanym punkcie się pojawić. Poprzez zamknięcie tranzystora oraz całkowite „otwarcie” regulatora napięcia wywnioskowano, że wokół interesującego nas rezystora, można odczytać najwięcej 11.1V. Wartość tą przekształcić należy teraz na 1.0765V.

$$\frac{11.1V}{1.0765V} = 10.31$$

Otrzymana wartość to stosunek, jaki napięcie wejściowe musi mieć do napięcia wyjściowego. Równanie wyznaczające taki stosunek w dzielniku napięcia przedstawia się następująco:

$$\frac{R1 + R2}{R2}$$

Dokładne wartości rezystorów nie są ważne, o ile spełniają obliczony stosunek oraz są wystarczająco duże, aby nie ingerowały w przepływ prądu w głównym oczku. W prototypie zastosowano dzielniki przed rezystorem:

- $R1 = 96.4 \text{ k}\Omega$
- $R2 = 10362 \Omega$

oraz za rezystorem:

- $R1 = 97.3 \text{ k}\Omega$
- $R2 = 10398 \Omega$

Wartości takie osiągnięto poprzez mierzenie multimetrem oporów poszczególnych rezystorów. Ponieważ miały one tolerancję 5%, ich wartości oporu znacząco różniły się od tych

⁸ Należy pamiętać też o połączeniu masy mikrokontrolera z masą układu

nominalnych. Na przykład, na wartość 96.4 kΩ składał się jeden rezystor 100 kΩ. Stosując oporniki wyższej klasy, pomiary byłyby jeszcze dokładniejsze.

W opisany sposób odczytać można spadek napięcia na oporniku. To czego jeszcze potrzebujemy to napięcie ładowanego ogniwa. Sytuacja jest analogiczna, z jedną tylko różnicą, napięcie za baterią będzie co najwyżej równe około 11.1V - 1.2V, gdyż 1.2V to najmniejsze nominalne napięcie akumulatora. Otrzymujemy więc stosunek:

$$\frac{9.9V}{1.0765V} = 9.20$$

Zastosowane w dzielniku oporniki to:

- R1 = 98.3 kΩ
- R2 = 11 970 Ω

Napięcie przed baterią jest równe napięciu za rezystorem. Ostatni parametr, który wymagany jest do przeprowadzenia poprawnego procesu ładowania to temperatura baterii. Do jej uzyskania posłużyć musimy się zewnętrznym modulem. W przypadku prototypu zadanie to pełni LM35DZ, operujący w zakresie temperatur od 0 do 100°C. Zasilany jest on z głównego, 12V źródła. Zasada jego działania jest bardzo prosta. Na pinie wyjściowym podaje on napięcie równe:

$$V_{out} = T * 0.01$$

Odczytując więc je i dzieląc przez 0.01, otrzymujemy aktualną temperaturę.

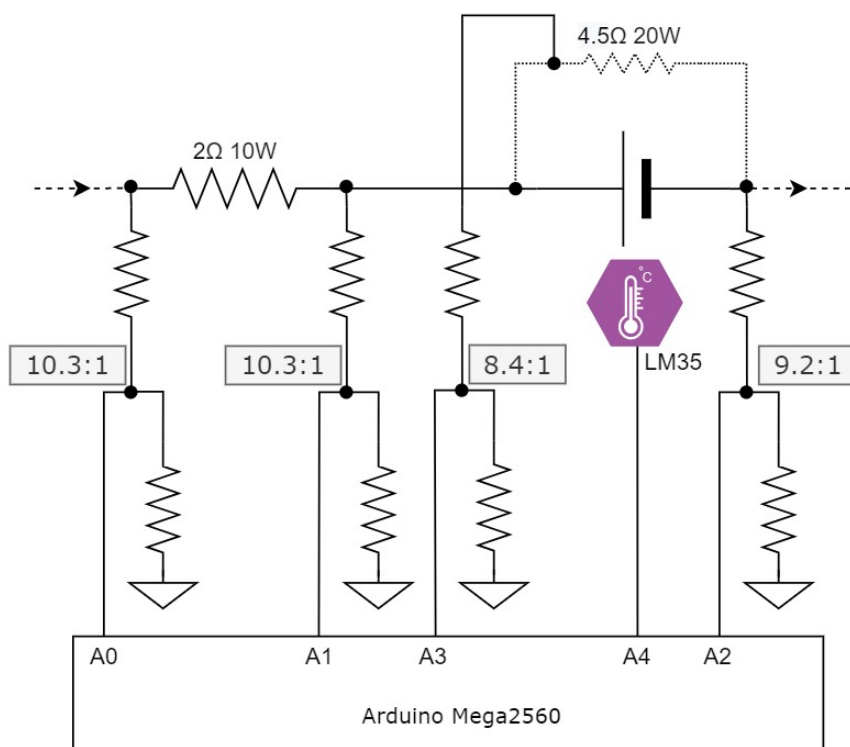
Parametry ładowania zostały już zebrane wszystkie. Ostatnim pomiarem, który będzie w ładowarce wykonywany, jest pobranie napięcia rozładowującej się baterii. Jest ono potrzebne, aby nie dopuścić do nadmiernego rozładowania. Pomiaru dokonuje się przed rezystorem w pętli rozładowania. Maksymalne napięcie, które tam można spotkać to 9V. Stosunek dzielnika napięcia w tym przypadku to:

$$\frac{9V}{1.0765V} = 8.36$$

użyte rezystory:

- R1 = 67.1 kΩ
- R2 = 9130 Ω

Na rysunku 3.4 można zaobserwować wszystkie zastosowane dzielniki wraz z ich stosunkami.



Rysunek 3.4. Część pomiarowa układu

Zakładając, że piny analogowe będą wykorzystane przez mikrokontroler tak jak na rysunku 3.4, możemy zastosować następujące równania:

$$I = \frac{A1 - A0}{2\Omega} \quad (\text{prąd})$$

$$V_{bat} = A2 - A1 \quad (\text{napięcie baterii ładowanej})$$

$$V_{roz} = A3 \quad (\text{napięcie baterii rozładowanej})$$

$$V_{reg} = 12V - A0 \quad (\text{spadek napięcia na regulatorze})$$

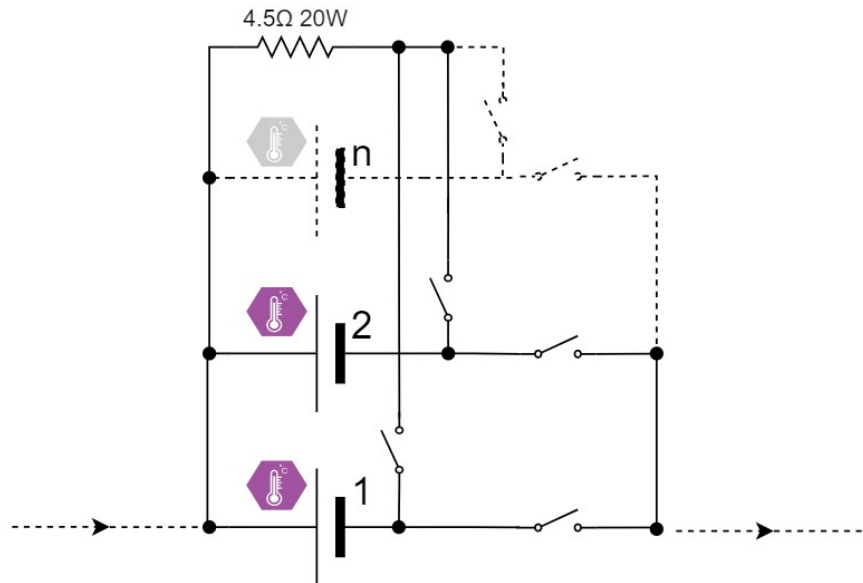
$$V_{trans} = A2 \quad (\text{spadek napięcia na tranzystorze})$$

$$T = \frac{A4}{0.01} \quad (\text{temperatura baterii})$$

Jak można zauważyć, wyznaczono dwa dodatkowe parametry. Korzystając z wcześniej ustalonych pomiarów, jest się w stanie policzyć spadki napięć, jakie występują na regulatorze napięcia oraz tranzystorze. Może być to użyteczne w celu obliczenia odkładanej na tych elementach energii cieplnej.

3.3.3. Pętla rozładowująca

Ostatnią częścią układu elektrycznego jest oczko służące rozładowywaniu akumulatorów. Można mu się bliżej przyjrzeć na rysunku 3.5.



Rysunek 3.5. Pętla rozładowująca i dodatkowe miejsca na ogniwa

Zastosowano w nim rezystor $4.5\ \Omega$ $20\ W$ zgodnie z założeniami, że maksymalne napięcie baterii stosowanej w ładowarce to $9\ V$. Jeżeli właśnie taką zaczniemy rozładowywać na danym rezystorze, moc, która się na nim odłoży, wynosić będzie:

$$P = \frac{U^2}{R} = \frac{81\ V^2}{4.5\ \Omega} = 18\ W$$

Na rysunku 3.5 ukazano również, w jaki sposób odbywa się przełączanie pomiędzy funkcjami ładowarki. Udział w tym biorą przekaźniki. W zależności od tego, który z nich jest zamknięty, bateria jest ładowana lub rozładowywana. W przypadku gdy oba są otwarte, jest w stanie oczekiwania. W aktualnej wersji urządzenie ma możliwość obsługi dwóch baterii jednocześnie, przy czym tylko jedna może być ładowana i tylko jedna rozładowywana. Jednakże, jeżeli ustalimy, że wszystkie umieszczone baterie mają zostać naładowane, odbywać się to będzie na zasadzie kolejki. W momencie, gdy zakończy się ładowanie jednej z nich, ładowarka od razu zabierze się za kolejną. Dodanie kolejnych miejsc na baterię wiąże się jedynie z kosztem 2 przekaźników, czujnika temperatury oraz komponentów do manualnego sterowania kanałem. Jeżeli jednak użytkownik nie chciałby ładować baterii po kolei, a jednocześnie, należałoby powielić cały obwód elektryczny.

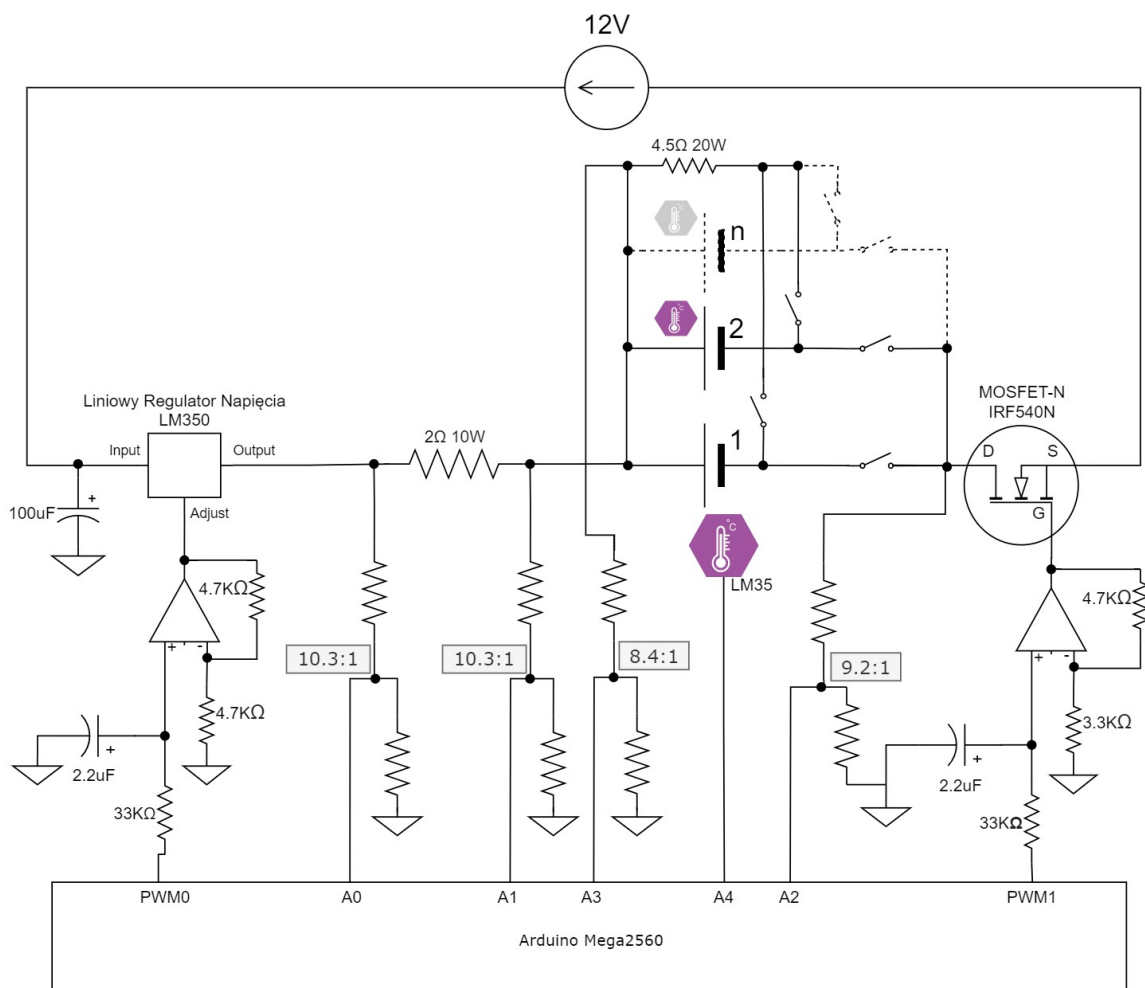
W przypadku dodawania miejsc na baterię, jak i powielania obwodu, w pewnym momencie zabrakłoby pinów w mikrokontrolerze. W takim wypadku należałoby skorzystać z pomocy multiplekserów. Świetną propozycją okazać się może CD4051. Jeżeli zaczyna brakować pinów cyfrowych, gdyż na przykład za dużo przekaźników zostało podpiętych,

3. Projekt systemu

użyć można go jako demultiplekser. Sprawę ułatwia założenie, że maksymalnie dwa przekaźniki na cały obwód mogą być zamknięte, więc dwa CD4051 zapewniają sprawne operowanie szesnastoma przekaźnikami, jedynie za pomocą ośmiu pinów cyfrowych. W przypadku braku pinów analogowych sytuacja jest analogiczna, tylko tym razem pełnił będzie on rolę multipleksera. Przełączając kanały jeden po drugim, zyskuje się możliwość odczytywania ośmiu sygnałów analogowych przy użyciu trzech pinów cyfrowych oraz pojedynczego analogowego.

Wszystkie poszczególne elementy układu elektrycznego zostały omówione. Jego pełny diagram można zaobserwować na rysunku 3.6. Dla większej przejrzystości zostało pominięte na nim:

- komponenty do manualnej obsługi kanału
- zasilanie wzmacniaczy operacyjnych
- zasilanie czujników temperatur
- zasilanie przekaźników i ich połączenia z pinami cyfrowymi



Rysunek 3.6. Pełny diagram obwodu

3.3.4. Rozprowadzenie ciepła

W przypadku pętli rozładowującej przeprowadzone zostały obliczenia sprawdzające, jaką moc musi ona umieć rozproszyć. Nie zweryfikowano jeszcze za to pętli ładującej.

Występują na niej trzy elementy, na których to odkładać się będzie najwięcej energii. Są nimi rezystor, regulator napięcia, oraz tranzystor. Zbadanie odporności układu na wytwarzane ciepło należy zacząć od obliczenia, ile tak naprawdę trzeba go na wymienionych elementach rozłożyć.

$$P_{max} = V_{max} * I_{max}$$

$$P_{max} = (12V - 1.2V) * 1.5A = 16.2W$$

Zakładając, że na ładowanym ogniwie najmniejszy osiągalny spadek napięcia to 1.2V, otrzymana wartość mocy wynosi 16.2W. Pierwszy skontrolowany niech będzie rezystor. Maksymalna moc odłoży się na nim wtedy, gdy w obwodzie płynął będzie największy prąd.

$$P = I^2 * R = (1.5A)^2 * 2\Omega = 4.5W$$

Jeżeli chodzi o rezystor 2 Ω 10 W, będzie on bez większego problemu spełniał wymogi. Wychodzi na to, że na regulatorze i tranzystorze odkładać się będzie w najgorszym wypadku:

$$16.2W - 4.5W = 11.7W$$

Zgodnie ze specyfikacją[21], rezystancja termiczna LM350 z otoczeniem wynosi 50 $\frac{^{\circ}C}{W}$, a graniczna temperatura operacyjna 125 $^{\circ}C$. Zakładając, że temperatura otoczenia utrzymuje się na poziomie 25 $^{\circ}C$, otrzymujemy maksymalną możliwą moc regulatora równą:

$$P = \frac{T_{max} - T_{amb}}{R_t} = \frac{125^{\circ}C - 25^{\circ}C}{50 \frac{^{\circ}C}{W}} = 2W$$

Analogiczne obliczenia dla IRF540N [17]:

$$P = \frac{175^{\circ}C - 25^{\circ}C}{62.5 \frac{^{\circ}C}{W}} = 2.4W$$

Otrzymane wartości są zasadniczo zbyt niskie. O ile regulator ma wbudowane zabezpieczenie przed wysokimi temperaturami, tak tranzystor najprawdopodobniej przy najwyższym obciążeniu uległby spaleni. Obniżanie maksymalnego natężenia nie wchodzi w grę, więc jedyne co można z zaistniałym problemem zrobić to zastosować jakiegoś rodzaju rozpraszacz ciepła. Większy radiator powinien spełnić wymogi. Wymagając, aby regulator i tranzystor miały możliwość rozpraszania co najmniej 12W, można policzyć, jaka byłaby

minimalna rezystancja termiczna stosowanego radiatora:

$$P = \frac{T_{max} - T_{amb}}{R_{tc} + R_{tr}}$$
$$R_{tr} = \frac{T_{max} - T_{amb}}{P} - R_{tc}$$

gdzie:

R_{tr} – rezystancja termiczna radiatora

R_{tc} – rezystancja termiczna komponentu do obudowy

W przypadku regulatora:

$$R_{tr} = \frac{125^{\circ}\text{C} - 25^{\circ}\text{C}}{12\text{W}} - 3.5 \frac{^{\circ}\text{C}}{\text{W}} = 4.83 \frac{^{\circ}\text{C}}{\text{W}}$$

W przypadku tranzystora:

$$R_{tr} = \frac{175^{\circ}\text{C} - 25^{\circ}\text{C}}{12\text{W}} - 1.25 \frac{^{\circ}\text{C}}{\text{W}} = 11.25 \frac{^{\circ}\text{C}}{\text{W}}$$

Stosując radiatory o takich lub niższych wartościach rezystancji, temperatura teoretycznie już nie sprawiałaby problemów. W prototypie użyto dwóch 6298B Aavid Thermalloy o rezystancji termicznej równej $3.9 \frac{^{\circ}\text{C}}{\text{W}}$ [22].

3.4. Oprogramowanie Mikrokontrolera

Arduino, tak jak większość innych podobnych mu systemów mikroprocesorów, programować można w języku C/C++. Z uwagi jednak na brak systemu operacyjnego, zarządzanie pamięcią nie jest wspomagane. Wiąże się to z dużym ryzykiem fragmentacji w przypadku dynamicznego alokowania danych. Mimo że wszystkie podstawowe operacje języka są dla programisty dostępne, ich wynik, może więc różnić się od tego znanego z komputerów osobistych. Niewielka ilość całkowitej pamięci oraz utrudnione zarządzanie nią, często też dyskwalifikuje z użycia znane biblioteki.

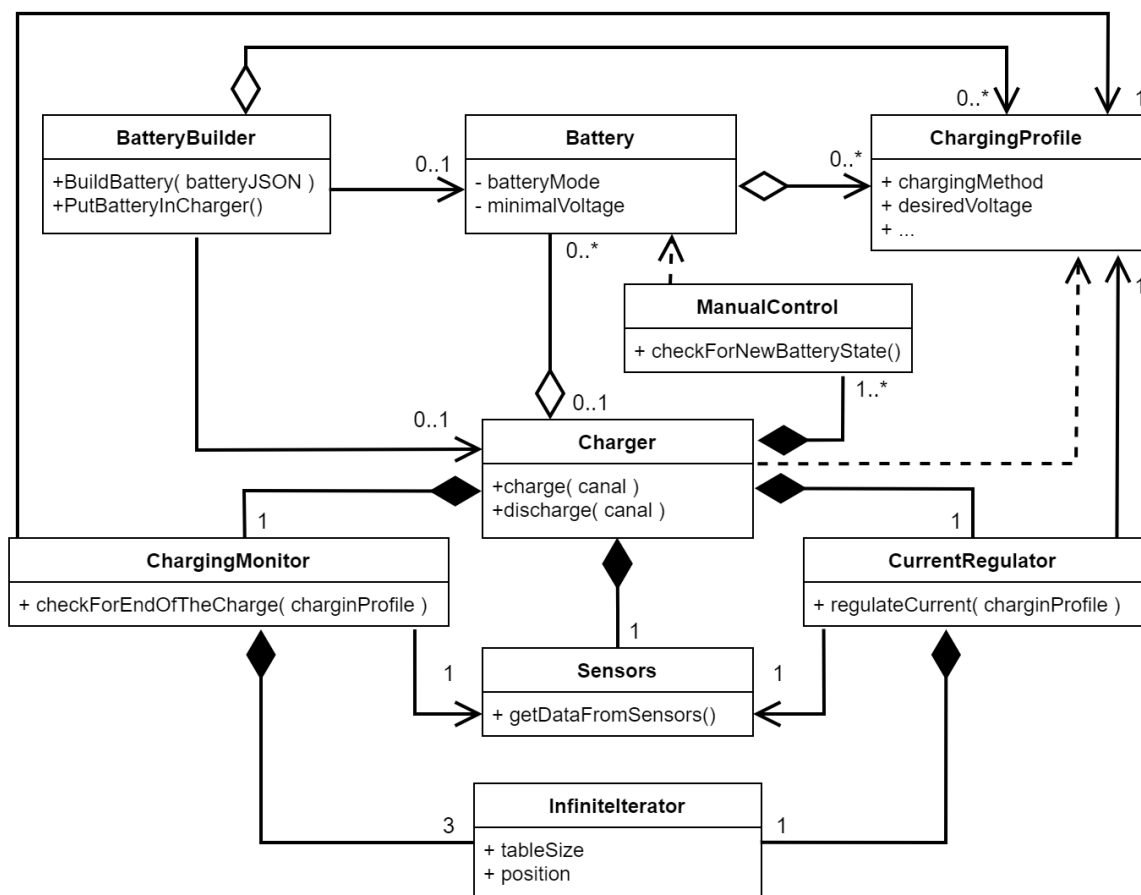
Kod prototypu ładowarki pisano w edytorze tekstu Visual Studio Code. Posiada on wysokiej jakości wsparcie dla języka C/C++ co znacznie ułatwia pracę. Kompilacja odbywała się przy pomocy środowiska Arduino IDE, umożliwiającego w prosty sposób zaprogramowanie mikrokontrolera, jak i komunikacje z nim przy pomocy interfejsu szeregowego UART.

3.4.1. Struktura Klas

Aby zapewnić odpowiednią przejrzystość kodu, ale także ułatwić jego ewentualne modyfikacje, pisany był on zgodnie z paradygmatem programowania obiektowego. Wyróżnić możemy w nim następujące klasy:

- **ChargingProfile** – odpowiednik poznanej wcześniej struktury profilu ładowania. Zawiera pola jednoznacznie charakteryzujące krzywą ładowania oraz parametry czynników kończących proces ładowania. Wykorzystywana jest głównie przez klasy **ChargingMonitor**, **CurrentRegulator** i **BatteryBuilder**.
- **Charger** – główna klasa. Reprezentuje cały układ elektryczny. Jej zadaniem jest kolejowanie procesów ładowania/rozładowywania.
- **ChargingMonitor** – analizuje aktualnie stosowany profil ładowania. Zbiera informacje o ładowanym ogniwie. W przypadku wykrycia końca ładowania informuje o tym klasę **Charger**.
- **CurrentRegulator** – odpowiada za nastawianie regulatora prądu. Robi to poprzez zmianę wypełnienia odpowiedniego sygnału PWM. Decyzje podejmuje na podstawie przekazanego mu profilu ładowania oraz aktualnych danych pomiarowych.
- **Sensors** – zajmuje się pobieraniem i przetwarzaniem aktualnych danych z układu pomiarowego.
- **BatteryBuilder** – zadaniem tej klasy jest przygotowywanie nowych profili ładowania. Dokonuje tego na podstawie dostarczonych ciągów tekstowych w formacie JSON. Gdy utworzy odpowiednią ilość profili, pakuje je w klasę **Battery** i przekazuje do **Charger**.
- **Battery** – klasa odpowiadająca jednej baterii. Zawiera stałą informację o sposobie jej ładowania i rozładowania.

- `ManualControl` – odpowiada za obsługę komponentów służących do manualnego sterowania ładowarką. W przypadku prototypu jest to pojedynczy przycisk oraz dwie diody LED.
- `InfiniteIterator` – pomocnicza klasa stworzona na potrzeby nieskończonego iterowania po tablicy.



Rysunek 3.7. Diagram klas z uwzględnieniem najważniejszych atrybutów

3.4.2. Kod programu mikrokontrolera

Mikrokontrolery po zaprogramowaniu działają najczęściej w nieskończonej pętli. Listing 1 obrazuje, jak wygląda ona w przypadku ładowarki. Ponieważ w przygotowanym podczas pracy prototypie występuje tylko jeden kanał, pojedyncza instancja klasy `Charger` jest użyta.

Listing 1. Główna pętla mikrokontrolera

```

19 void loop()
20 {
21     charger.getSensorData();
22     charger.adjustElectricalComponents();
23
24     if( builder.isNewBatteryReady())
25     {
26         builder.putBatteryInLinkedCharger();
27     }
28 }

```

Jest to swego rodzaju pętla sprzężenia zwrotnego. Najpierw odczytywane są dane pomiarowe. Na ich podstawie można zdecydować, jaki będzie następny stan ładowarki. W przypadku gdy do budowniczego trafiły nowe dane, zapisywane są one w głównej klasie. Jeżeli użytkownik dysponowałby więcej niż jednym układem ładującym, funkcja ta wyglądałaby trochę inaczej. Zamiast pojedynczej instancji klasy Charger, używałaby jej tablicy. Iterując po poszczególnych elementach, dawałaby wrażenie obsługi kilka kanałów jednocześnie.

Nie raz wspomniane było, że dokładne pomiary napięcia są dla tej pracy bardzo ważne. Dlatego też funkcja odpowiedzialna za ich odczyt, wyznacza ostateczny wynik na podstawie średniej z kilkudziesięciu próbek. Dodatkowo odczyty zbyt odstające od pozostałych są eliminowane.

Listing 2. Obsługa pomiaru napięcia na pinach analogowych

```

64 double Sensors::measureVoltage(int analogPin, double dividerMultiply)
65 {
66     const int NUMBER_OF_SAMPLES = 60;
67     const int RESOLUTION = 1023;
68     double splitedMean = 0;
69     int badSamples = 0;
70     for(int i = 0; i < NUMBER_OF_SAMPLES; ++i)
71     {
72         int readValue = analogRead(analogPin);
73         double result = readValue * 1000; //mV
74         result = result / RESOLUTION;
75         result = result * VOLTAGE_REFERENCE * dividerMultiply;
76
77         double currentMean = splitedMean * NUMBER_OF_SAMPLES / i;
78         if( i > 4 && (result > 1.15*currentMean || result < 0.85*currentMean))
79         {
80             result = currentMean;
81             ++badSamples;
82             if(badSamples > NUMBER_OF_SAMPLES / 10)
83             {
84                 return 0;
85             }
86         }
87         splitedMean += result/NUMBER_OF_SAMPLES;
88         delay(1);
89     }
90     return splitedMean;
91 }

```

Po pobraniu wszystkich napięć stosowane są równania przedstawione pod koniec rozdziału 3.3.2 do obliczania parametrów, takich jak płynący prąd czy spadki napięć. Będą z nich korzystać klasy ChargingMonitor oraz CurrentRegulator.

Głównym zadaniem klasy Charger jest kontrolowanie dwóch kolejek typu pierwsze weszło, pierwsze wyszło. Jedna z nich odpowiada bateriom oczekującym na ładowanie, druga tym czekającym na rozładowanie. Na pierwszych pozycjach obu tych kolejek stosowany jest odpowiadający im proces. Ponieważ nie chcemy, aby ta sama bateria mogła znajdować się w obu tych kolejkach jednocześnie, najlepszym na to sposobem jest powiązanie stanu baterii z kolejką, w której się ona znajduje. Baterie mogą więc przyjmować trzy stany, ładowania, rozładowywania oraz czekania. W momencie zmiany stanu automatycznie sprawdzane są kolejki pod kątem danej baterii i z nich usuwane/dodawane. W przypadku stanu „czekanie” bateria nie znajduje się w żadnej kolejce. Na podstawie stanów baterii oraz kolejek, klasa Charger decyduje, które przełączniki powinny być otwarte, a które zamknięte.

Rozważmy przykład ładowarki w której, dwie baterie mają stan „ładowanie”. Można być pewnym, że obie znajdują się w kolejce ładowania. W momencie, gdy wszystkie profile ładowania zostaną w przypadku ładującej się baterii zakończone, jej stan zostanie ustawiony na „czekanie”. Usuwana jest więc ona z kolejki, a jej miejsce zajmuje bateria druga. Kod 3 przedstawia zasadę obsługi kolejki ładowania.

Listing 3. Obsługa kolejki ładowania

```
54 void Charger::checkChargeQueue()
55 {
56     if(chargeQueue[0] == -1)
57         return;
58
59     ChargingProfile& profile = batteries[chargeQueue[0]].
60                                     getOngoingChargingProfile();
61
62     regulator.applyProfile( profile );
63     int result = monitor.checkForEndOfTheCharge( profile );
64     if( result == 1 ){
65         monitor.profileChargingEnded();
66         batteries[chargeQueue[0]].moveToNextStep();
67         if( batteries[chargeQueue[0]].isChargingCompleted() ){
68             setBatteryMode(chargeQueue[0], BatteryMode::Wait);
69         }
70     }
71     else if( result == -1 ){
72         setBatteryMode(chargeQueue[0], BatteryMode::Wait);
73     }
74 }
```

Klasa CurrentRegulator musi być odpowiednio dopasowana do zastosowanego w układzie elektrycznym regulatora prądu. W przypadku prototypu wykorzystuje ona na

przykład fakt, że taka sama zmiana wypełnienia sygnału PWM dla tranzystora, dużo bardziej wpływa na płynący prąd, niż gdyby była zastosowana dla regulatora napięcia. Najpierw więc dobierana jest wartość dla MOSFET'u. Gdy zacznie się ona wahać, przechodzi się w regulację dokładniejszą, czyli za pomocą LM350. W ten sposób uzyskuje bardzo dokładny, na poziomie 0.5%, prąd. Dodatkowo klasa CurrentRegulator monitoruje odkładaną na elementach regulatora prądu moc. Gdy przeważa ona za bardzo na jednym z nich, próbuje ją wyrównać.

ChargingMonitor, jak jego nazwa wskazuje, obserwuje aktualnie stosowany profil ładowania. W przypadku gdy uzna on, że ładowanie powinno zostać przerwane, zwraca odpowiedni komunikat do klasy wywołującej. Może on informować albo o poprawnym zakończeniu ładowania, albo o błędnym, jeżeli przekroczona została jedna z wartości maksymalnych. W przypadku wykrycia niepoprawnego włożenia baterii lub jej braku, przerwie on natychmiastowo ładowanie. Stosowanych jest wiele technik mających na celu zapewnienie prawidłowego identyfikowanie finiszu. Na przykład, nie wystarczy pojedynczy odczyt zbyt wysokiego napięcia do podjęcia decyzji o jego przekroczeniu. Potrzebne są do tego aż trzy seryjne. Jedną z funkcji monitora jest obliczanie stopnia naładowania baterii. Estymuje on przekazany jej ładunek i na podstawie stosunku z maksymalną pojemnością wylicza poziom naładowania. Niestety nie został zaimplementowany algorytm pozwalający oszacować początkową ilość energii więc ukazywane przez ładowarkę wartości mogą znacznie odbiegać od rzeczywistych.

Klasa ta posiada tabele historyczne, potrzebne do identyfikowania czy przekroczone zostały wartości związane ze zmianami w czasie. Do nawigowania po tych tabelach stworzono oddzielny iterator. Mowa tutaj o klasie InfiniteIterator. Działa ona na zasadzie takiej, że jeżeli spróbuje się za jej pomocą odnieść do indeksu wychodzącego poza ramy danej tabeli, ta odpowiednio go do niej dopasuje, jakoby był w nieskończonej pętli. Najlepiej można to zaobserwować na przykładzie 4.

Listing 4. Zasada działania nieskończonego iteratora

```
1  int main()
2  {
3      InfiniteIterator itr(5);
4      int x[5] = {0,1,2,3,4};
5      int result = itr.at(); //result == 0
6      result = x[itr+6]; //result == 1
7  }
```

Ostatnią większą klasą w programie mikrokontrolera jest BatteryBuilder. Jej powstanie uwarunkowane było potrzebą komunikacji urządzenia przez Internet. Dane przesyłane są pomiędzy serwerem a ładowarką w postaci ciągów JSON. Nie są to bardzo skomplikowane struktury, więc teoretycznie można byłoby ich deserializację przeprowadzać prostym wyszukiwaniem, a następnie wyłuskiwaniem znalezionych fragmentów. Jednakże lepsze efekty uzyskać można poprzez zastosowanie biblioteki ArduinoJSON[23]. Pozwala ona

w bardzo prosty sposób deserializować oraz serializować ciągi JSON. Listing 5 przedstawia, na jakiej zasadzie budowane są profile.

Listing 5. Budowanie profili z ciągu JSON

```
59 void BatteryBuilder::buildBatteryProfile(const String& profileJSON){
60     ChargingProfile profile;
61     deserializeJson(doc, profileJSON);
62     String method = doc["method"];
63     if(method == "stale_napiecie")
64         profile.method = ChargingMethod::constantVoltage;
65     else
66         profile.method = ChargingMethod::constantCurrent;
67
68     profile.maxVoltage = doc["maxVoltage"];
69     profile.maxCurrent = doc["maxCurrent"];
70     profile.maxTemperature = doc["maxTemperature"];
71     ...
72 }
```

3.4.3. Komunikacja bezprzewodowa

Ponieważ Arduino Mega nie posiada żadnego modułu umożliwiającego komunikację bezprzewodową, należało w tym celu skorzystać z pomocy zewnętrznego mikroprocesora. Wybór padł na ESP8266, a dokładnie na mikrokontroler Wemos D1 Uno R3. Posiada on wbudowany interfejs TCP/IP i umożliwia komunikację przy użyciu sieci Wi-Fi. Plan działania zakłada wysyłanie żądań HTTP do serwera producenta ładowarki, odebranie parametrów jednej baterii w postaci ciągu JSON, a następnie przesłanie ich za pomocą interfejsu I²C do Arduino, gdzie uległby deserializacji. Wszystko wydaje się łączyć w logiczną całość, jednakże występują po drodze pewne problemy, których uniknąć dało się, wybierając inny mikrokontroler do zarządzania elementami układu elektrycznego. Komplikacją jest tutaj komunikacja pomiędzy dwoma mikrokontrolerami. Wspomniany interfejs I²C wspiera przesył jedynie 32 bajtów w przypadku płyt Arduino. Ponieważ zwracane przez serwer ciągi JSON mogą zajmować ponad 1.5KB pamięci, trzeba użyć mechanizmu ich dzielenia. Ten zastosowany w prototypie przedstawiony jest w listingu 6. Oprócz tego aby, nie buforować zbyt dużej ilości pamięci, cały proces wysłania ciągu JSON jest podzielony na etapy:

1. przesłanie kolejno profili ładowania
2. przesłanie numeru kanału, dla którego dane zostały odebrane
3. przesłanie parametrów baterii

Zadaniem klasy BatteryBuilder po stronie Arduino jest odpowiednie złączenie wszystkich tych fragmentów.

Wszystkich tych utrudnień dało się uniknąć, poprzez zastąpienie Arduino Mega następcą ESP8266, czyli ESP32. Jest to mikrokontroler posiadający wszystkie charaktery-

styczne cechy jego poprzednika, jak i również przewyższający pod względem pamięci i szybkości Arduino Mega. Jedyne co przemawia za zastosowaniem dotychczasowego mikrokontrolera, jest duża liczba jego wejść i wyjść, ale tak jak już wspomniane było, ten atrybut można sztucznie nadrobić, poprzez użycie multiplekserów. Budowę prototypu dokończono jednak w mniej optymalnym układzie.

Listing 6. Przesył ciągu znaków za pomocą I²C

```

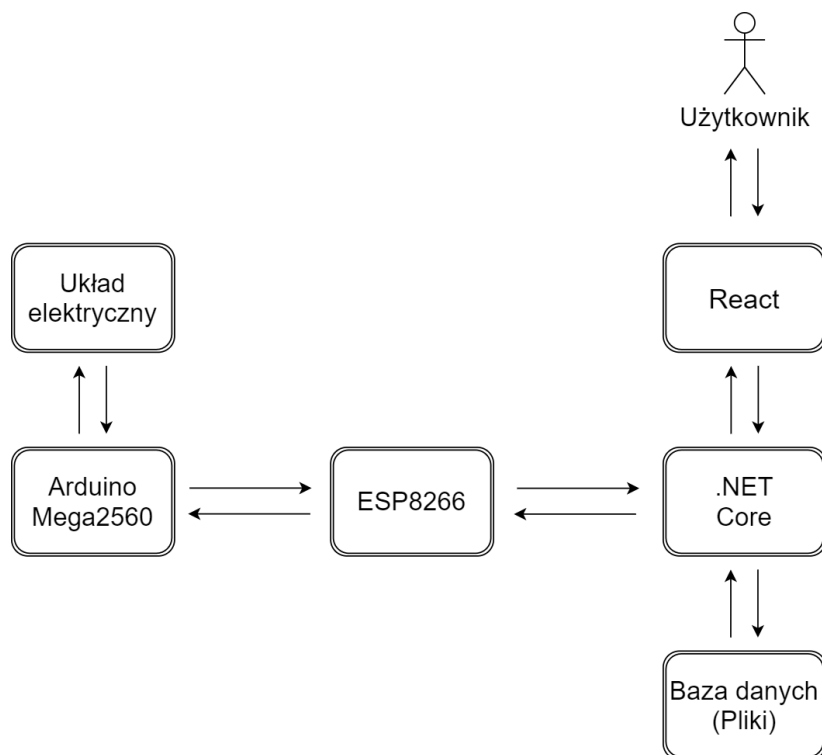
93 void sendStringToMicro(const String& data){
94     const int MAX_BYTES = 32;
95     String part;
96     int i = 0;
97     int diff = data.length()+1 - MAX_BYTES; //+1 z powodu '\0'
98     while( i < diff)
99     {
100         part = data.substring(i, i+MAX_BYTES);
101         Wire.beginTransmission(WIRE_SLAVE_ID);
102         Wire.write(part.c_str());
103         Serial.println(part);
104         Wire.endTransmission();
105         delay(1);
106         i += MAX_BYTES;
107     }
108     part = data.substring(i);
109     Wire.beginTransmission(WIRE_SLAVE_ID);
110     Wire.write(part.c_str());
111     Serial.println(part);
112     Wire.write('\0');
113     Wire.endTransmission();
114 }
```

Komunikacje pomiędzy urządzeniami wykorzystującymi interfejs I²C polega na nawiązaniu połączenia typu pan-niewolnik. W przypadku ładowarki panem jest ESP8266, niewolnikiem zaś Arduino Mega. W momencie rozpoczęcia komunikacji niewolnik odbiera przerwanie i zatrzymując aktualnie wykonywaną instrukcję, przechodzi od razu do jego obsługi. W celu przesłania jednego profilu ładowania, obsłużonych zostanie około 10 przerwań.

3.5. Zewnętrzny serwer

Kontrola uniwersalnej ładowarki akumulatorów odbywa się w głównej mierze w sposób bezprzewodowy. O ile urządzenie potrafi przechowywać przekazane mu raz struktury baterii i nie wymaga do działania stałego połączenia z Internetem, tak na początku użytkownika należy mu owe struktury dostarczyć. Konstruowanie i przesył nowych schematów, odbywa się poprzez aplikację internetową.

Interfejs użytkownika strony internetowej został przygotowany z użyciem biblioteki języka JavaScript – React[24], kod po stronie serwera wykorzystuje zaś framework .net Core, który to pisany jest w języku C#. React jest aktualnie bardzo popularnym narzędziem, dlatego pozyskać można o nim wiedzę z wielu użytecznych źródeł. Jest on także względnie prosty do opanowania, w przeciwieństwie do chociażby Angulara. Core wybrany został przede wszystkim za jego wieloplatformowość. Dobór takich narzędzi pozwala na uzyskanie aplikacji, do obsługi której użytkownik nie musi instalować żadnych komponentów, przy założeniu, że ma on dostęp do jednej z popularnych przeglądarek internetowych.



Rysunek 3.8. Komunikacja komponentów w systemie

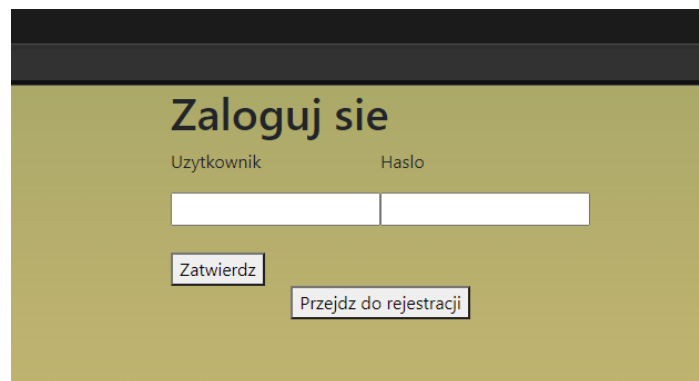
Serwer w obecnym stanie nie wykorzystuje tradycyjnej bazy danych. Zamiast tego informacje przechowuje w plikach. Rozwiązanie takie jest korzystne w przypadku gdy typów danych jest niewiele. Ładowarka przechowuje jedynie:

- dane uwierzytniające
- liste zalogowanych użytkowników
- dane użytkownika na które składają się, kanały, baterie, profile

W miarę rozrostu aplikacji, zmiana sposobu gromadzenia danych jest jak najbardziej wskazana, gdyż przechowywanie ich w plikach nie jest najoptymalniejszym rozwiązaniem.

3.5.1. Kontrola ładowarki za pomocą aplikacji internetowej

Aby dostać się do panelu sterownia, użytkownik najpierw musi przejść przez proces weryfikacji. Jeżeli nie posiada on jeszcze założonego konta, udać powinien się do menu rejestracji. Przy poprawnym wypełnieniu formularza zakładania konta, użytkownik poinformowany zostanie odpowiednim komunikatem, a w bazie danych serwera pojawi się nowe pole, zawierające ID, login i hasło. W tym momencie można wrócić do okna logowania.

The image shows a login form titled "Zaloguj sie" (Log in) on a light olive green background. Below the title, there are two labels: "Uzytkownik" (User) and "Haslo" (Password). Under each label is a white rectangular input field. Below the input fields, there are two buttons: "Zatwierdz" (Confirm) on the left and "Przejdź do rejestracji" (Go to registration) on the right. The buttons have a thin black border and a light grey background.

Rysunek 3.9. Panel Logowania

Zalogowany użytkownik ma dostęp do panelu kontrolnego ładowarki. Po wybraniu kanału⁹, a następnie baterii ukazują się ostatnie zgromadzone o niej informacje, to jest procent naładowania, napięcie oraz płynący przez nią prąd. Zauważyć też można listę profili przypisanych do danej baterii oraz jej atrybuty, to jest stan, pojemność i minimalne napięcie. Po naciśnięciu na dowolny profil przechodzi się do formularza, zawierającego wszystkie pola ze znanej już struktury profilu ładowania. Po odpowiedniej edycji formularza, pozostaje już tylko użyć przycisku zatwierdzającego, aby przesłać wprowadzone dane do bazy danych. Po upływie kilku sekund mikrokontroler wyśle kolejne żądanie HTTP do serwera, na które tym razem odpowie on nowo utworzoną strukturą baterii.

⁹ W przypadku strony internetowej kanał to pojedynczy układ elektryczny, odpowiednik klasy Charger w mikrokontrolerze.

kanal 0

Dodaj nowy kanal Usun zaznaczony

☒bateria1 ☐bateria2

Pojemnosc Minimalne napiecie

2100 1000

laduj

profil 0

profil 1

Zapisz zmiany Dodaj Profil

Procent Naładowania Napiecie Baterii Płynący Prąd

0.00% 0.00mV 0.00mA

Rysunek 3.10. Panel ustawień i monitorowania baterii

profil 0

Wroc Usun profil

Metoda

staly prad

Maks Napiecie

4280

Maks Prad

1550

Maks Temperatura

45

Maks Czas

3600000

Zamierzony Napiecie

4200

Zamierzony Prad

900

Czas probkowania napiecia

60000

Minimalna roznica sprobkowanych napiec

-999

Czas probkowania temperatury

60000

Maksymalna roznica sprobkowanych temperatur

Rysunek 3.11. Formularz edycji profilu ładowania

3.5.2. Oprogramowanie serwera

Informacje przychodzące do serwera zawsze przyjmują postać żądań HTTP. Każde z nich obsługiwane jest przez kontrolery, a dokładnie ich funkcje. Zostały zaimplementowane następujące kontrolery:

- LoginController – odpowiada za weryfikację logujących się użytkowników. Jeżeli zakończy się ona pomyślnie, nadaje użytkownikowi token, pozwalający mu na zachowywanie aktualnej sesji, mimo na przykład zamknięcia okna przeglądarki. Z to-

kenem powiązana jest wygenerowana pseudolosowo 16 bajtowa liczba, chroniąca przed próbą sfałszowania tokenu.

- LogoutController – usuwa użytkownika z listy zalogowanych oraz czyści jego token
- RegisterController – odpowiada za rejestrację nowych użytkowników. Waliduje przekazane mu atrybuty. W przypadku ich zatwierdzenia nadaje użytkownikowi ID i umieszcza jego rekord w pliku CSV.
- UserDataController – jego zadaniem jest tworzenie, edytowanie i odczyt plików XML, które to zawierają:
 - listę dostępnych kanałów
 - parametry baterii
 - informacje o stanie baterii
 - listy profili ładowania

Używany jedynie od strony React'a

- MicrocontrollerController – również ma dostęp do plików XML, lecz obsługiwany jest tylko od strony mikrokontrolera.

Aby zapobiec ciągłemu pobieraniu tych samych danych przez mikrokontroler, dodana została flaga, informująca czy nowe zostały udostępnione.

3.5.3. Struktura bazy danych

Dane poszczególnych użytkowników zapisywane są w oddzielnych plikach w formacie XML. Strukturę przykładowego pliku można zaobserwować w listingu 7. Pozostałe informacje, takie jak dane uwierzytelniające czy zalogowani użytkownicy, przechowywane są w plikach w formacie CSV.

Listing 7. Struktura plików bazodanowych XML

```
1 <User>
2   <charger>
3     <battery>
4       <mode>laduj</mode>
5       <capacity>850</capacity>
6       <minVoltage>1100</minVoltage>
7       <chargingProfiles>
8         <profile>
9           <method>staly prąd</method>
10          <maxVoltage>1500</maxVoltage>
11          <maxCurrent>1000</maxCurrent>
12          <maxTemperature>35</maxTemperature>
13          <maxTime>432000</maxTime>
14          <desiredVoltage>1500</desiredVoltage>
15          <desiredCurrent>850</desiredCurrent>
16          <voltageDeltaInterval>60000</voltageDeltaInterval>
17          <minVoltageDelta>0</minVoltageDelta>
18          <temperatureDeltaInterval>60000</temperatureDeltaInterval>
19          <maxTemperatureDelta>2</maxTemperatureDelta>
20          <endingVoltageDrop>10</endingVoltageDrop>
21        </profile>
22      </chargingProfiles>
23      <completion>10</completion>
24      <cellVoltage>1354.25</cellVoltage>
25      <currentFlowing>851.2</currentFlowing>
26      <isNewData>0</isNewData>
27    </battery>
28  </charger>
29 </User>
```

4. Efekty działania systemu

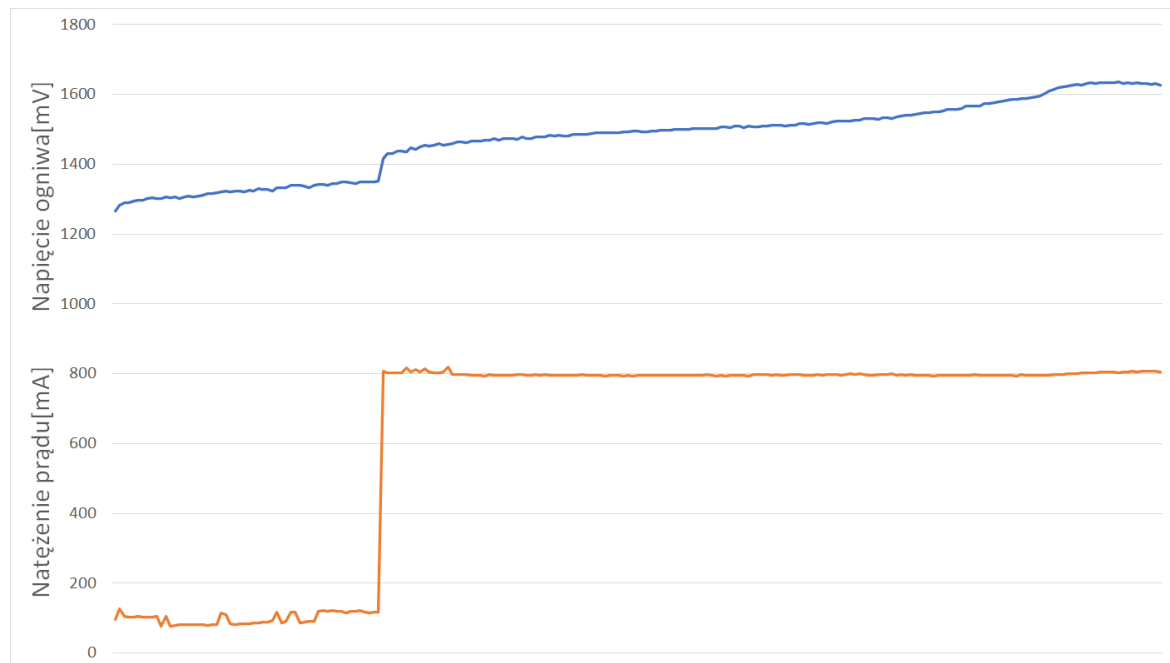
W ramach prototypu przygotowane zostało urządzenie o jednym kanale i dwóch miejscach na baterię. Rezultaty ładowania nim przykładowych akumulatorów zostały przedstawione w tej sekcji.

Testy jakości ładowania zostały przeprowadzone na bateriach:

- NiMH: BK-4MCC [12]
- Li-Ion: US18650VTC4 [15]

Polegały one na pełnym ich rozładowaniu a następnie naładowaniu przy pomocy przygotowanego urządzenia. Co 20 sekund do serwera przesyłane były informacje o napięciu ogniwa, jego temperaturze oraz przepływającym przez niego prądzie. Na ich podstawie można było nakreślić charakterystykę ładowania i porównać z tą przedstawioną przez producenta w specyfikacji. Testy przeprowadzone były w temperaturze pokojowej to jest około 25°C.

Pierwsza próba polegała na dwu etapowym naładowaniu akumulatora NiMH, o napięciu znamionowym 1.2V oraz pojemności 800mAh. Najpierw zastosowano ładowanie wstępne prądem 100mA. Pierwszy profil miał zakończyć się w momencie osiągnięcia na ogniwie napięcia 1350mV. Drugi profil polegał już na standardowym ładowaniu prądem stałym o natężeniu 1C. Jego koniec miał być wykryty w momencie odczytania przez ładowarkę niewielkiego spadku napięcia.



Rysunek 4.1. Uzyskana charakterystyka ładowania akumulatora NiMH

Jak można zauważyć po charakterystyce 4.1, profile zakończyły się zgodnie z założeniami. Na krzywej ładowania można zauważyć charakterystyczny dla akumulatorów

niklowych szybszy przyrost napięcia na ogniwie przed jego całkowitym ustabilizowaniem, a także wspomniany już końcowy spadek napięcia. Rezultat ładowania jest zadowalający, ładowarka poprawnie wykryła wszystkie wymagane od niej zakończenia profili, a także dość dobrze regulowała prąd, w szczególności w drugim etapie. Niedokładność regulacji w pierwszym wynikała ze zbyt dużej zmiany natężenia prądu w ramach pojedynczego kroku tranzystora. Gdy ładowarka wykrywała wahanie się ustawień tranzystora, przechodziła do sterowania regulatorem napięcia, jednakże różnica pomiędzy wartością docelową a aktualną była zbyt duża i wracała ponownie do regulacji tranzystorem. Czas trwania pierwszego profilu to 19 minut, drugi zaś zajął 56 minut. Wartym uwagi jest zaobserwowany skok napięcia na ogniwie, w momencie przełączenia profili. Pokazuje on, że napięcie odkładające się na baterii nie jest niezależne od natężenia przepływającego przez nią prądu. Uzasadnione tym samym jest stosowanie metody ładowania stałym prądem. W przypadku, gdyby nie był on wyregulowany, napięcie również zmieniałoby się, nie pozwalając na skorzystanie z takich technik jak chociażby wykrycie końcowego spadku napięcia.

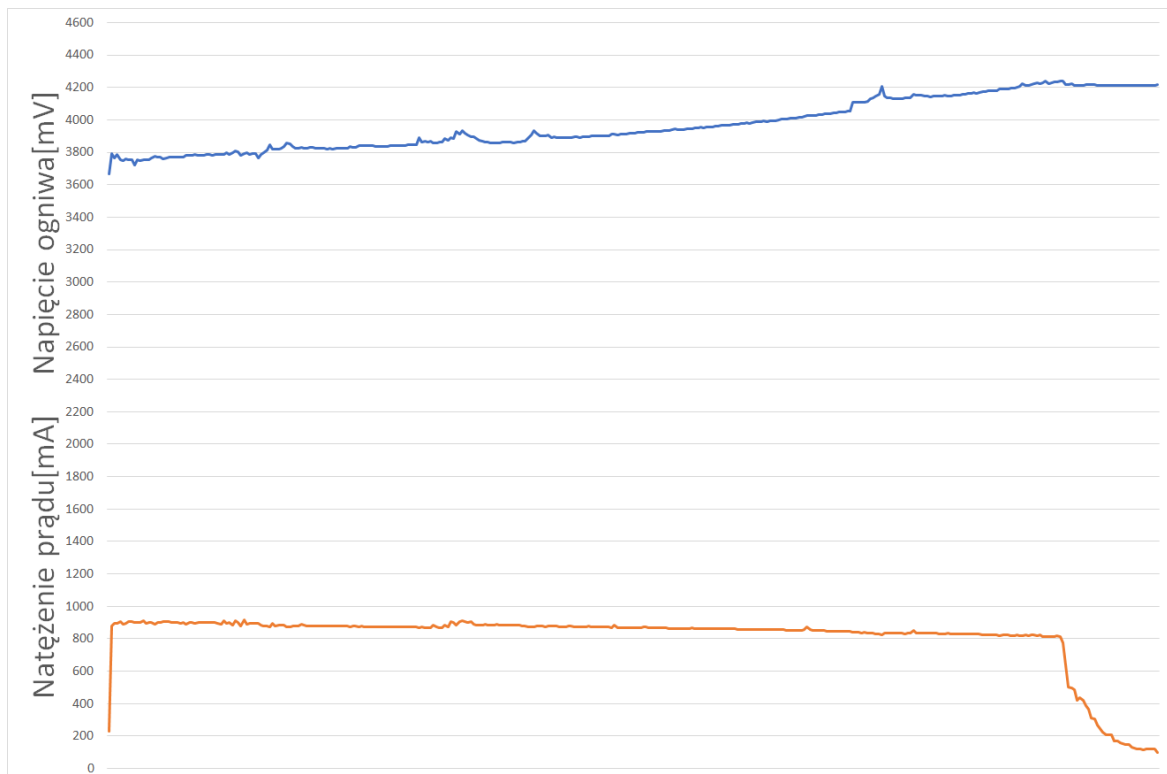


Rysunek 4.2. Zmiana temperatury podczas ładowania akumulatora NiMH

Rysunek 4.2 przedstawia zmiany temperatury podczas drugiego etapu ładowania. Znow można zauważyć wszystkie charakterystyczne cechy dla akumulatora NiMH, to jest,

powolny wzrost temperatury w całym okresie ładowania oraz dużo większy w końcowej fazie.¹⁰

W przypadku baterii NiMH ładowanie przebiegło niemalże bezproblemowo, przyjrzmy się teraz charakterystyce ogniwa Li-Ion, o napięciu znamionowym 4.2V oraz pojemności 2100mAh. Jest ona widoczna na rysunku 4.3. Tym razem zastosowano standardowy dla tego typu akumulatorów plan ładowania. Najpierw ładowanie powinno odbyć się stałym prądem o natężeniu 1A, po czym przy osiągnięciu napięcia na ogniwie równego 4.2V ładowarka powinna przejść do trybu stałego napięcia. Ten zaś miał ulec zakończeniu przy natężeniu 100mA. W przypadku ogniwa Li-Ion sprawa ma się troszeczkę gorzej. Okazuje się, że spadki napięć na obwodach układu elektrycznego nie pozwalają nawet osiągnąć docelowego natężenia 1A. Główną tego przyczyną jest to, że układ zbudowany jest na płytkach prototypowych, a do większości połączeń użyto kabli 20-30cm. Przyglądając się wykresowi natężenia, można zauważyć, że w miarę wzrostu napięcia baterii, regulatorowi prądu coraz trudniej było dostarczać wymaganą od niego wartość natężenia. Niemniej jednak, ładowarka ostatecznie poradziła sobie z naładowaniem akumulatorem Li-Ion, choć sam pierwszy etap zajął jej przy tym ponad 2 godziny. Śledząc przebieg napięcia na ogniwie, można zauważyć także niedokładności pomiaru. W pewnych momentach odczytywane wartości były mocno zawyżone.



Rysunek 4.3. Uzyskana charakterystyka ładowania akumulatora Li-Ion

¹⁰ Podczas pierwszego etapu temperatura ogniwa malała dlatego nie została uwzględniona, było to najprawdopodobniej związane z tym, że bateria nie zdążyła się ostudzić po szybkim rozładowaniu.

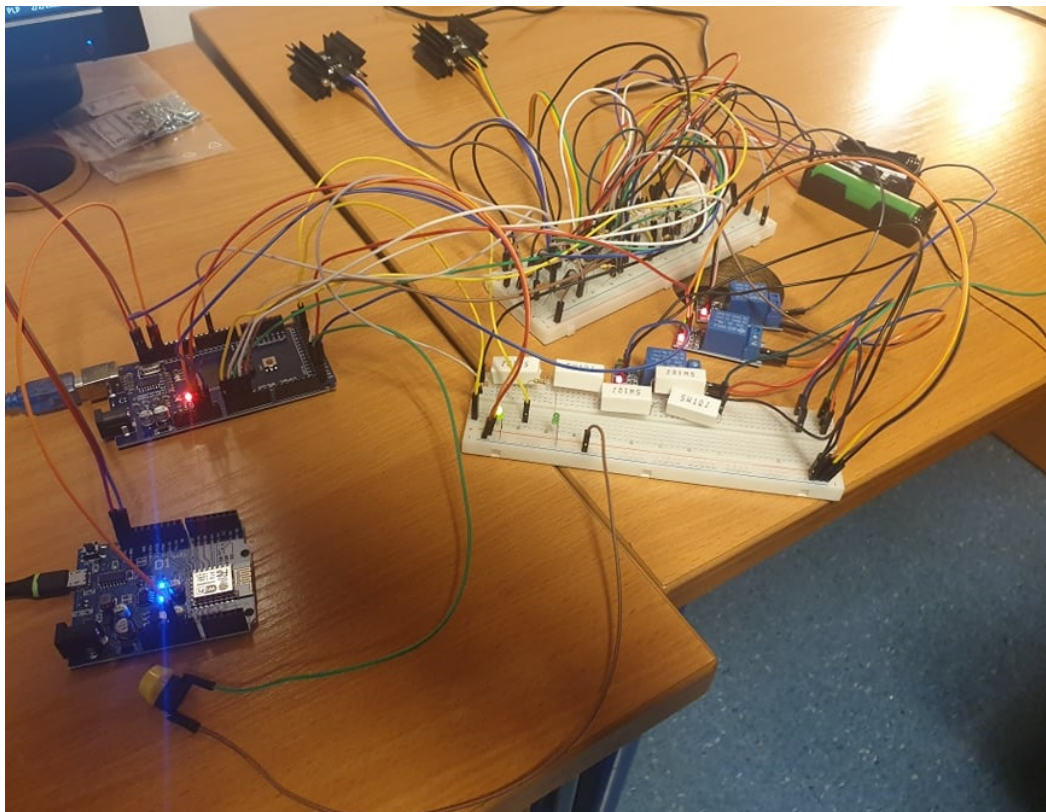
Ładowarka poradziła sobie z bezpiecznym naładowaniem najważniejszych typów akumulatorów, jednakże można dojrzeć w układach regulujących i pomiarowych pewne niedoskonałości. Wymiana tranzystora na taki o większym zakresie napięć, w których nie jest on ani otwarty ani zamknięty mogłaby poprawić jakość regulacji. Podobny efekt dałoby zastosowanie innego mikrokontrolera, na przykład ESP32, którego rozdzielczość wypełniania sygnału PWM jest większa. W przypadku pomiarów lepsze rezultaty byłyby do osiągnięcia poprzez zastosowanie wyspecjalizowanych układów pomiarowych lub rezystorów o mniejszej tolerancji. W przyszłych wersjach prototypu problem regulacji byłyby z pewnością do naprawienia.

5. Wnioski

Celem niniejszej pracy inżynierskiej było przygotowanie systemu zdolnego do ładowania i rozładowywania dowolnego typu akumulatora. Żeby taki system zaprojektować, należało najpierw poznać podstawowe typy baterii oraz metody, jakimi są one ładowane. Po wnikliwym rozeznaniu tego tematu sformułowana została jego koncepcja oraz założenia. Zgodnie z nimi system można podzielić na trzy główne komponenty. Pierwszym są kanały, czyli układy elektryczne pozwalające na ładowanie oraz rozładowywanie baterii o napięciu do 9V. Drugi to układ sterujący składający się z dwóch mikrokontrolerów. Ich cel to nadzorowanie nad poszczególnymi kanałami. Ostatnim komponentem jest serwer pośredniczący w komunikacji pomiędzy użytkownikiem a układem sterującym. Cechą charakterystyczną przygotowanego urządzenia jest sposób w jaki definiowane są krzywe ładowania dla baterii. Wykorzystywane są do tego struktury o dwunastu polach, za pomocą których uzyskać można i parametryzować poszczególne etapy najpopularniejszych algorytmów ładowania. Użytkownik systemu ma pełną kontrolę nad tym w jaki sposób jego baterie zostaną potraktowane. Może więc decydować o szybkości ładowania, jego stopniu, a tym samym o jego bezpieczeństwie i wpływie na żywotność baterii.

W pracy poruszono różnorodne tematy. Przedstawione zostały zasady działania podstawowych komponentów elektronicznych oraz ich zastosowania. Przeanalizowano negatywny wpływ ciepła oraz sposób, w jaki można z nim sobie poradzić. Pokazano, na jakiej zasadzie może wyglądać komunikacja pomiędzy urządzeniami elektrycznymi, czy to w sposób przewodowy interfejsem I²C, czy bezprzewodowy za pomocą Internetu i żądań HTTP. Poruszono także zagadnienia automatyki takie jak regulacja, sterowanie i wykonywanie pomiarów z wykorzystaniem mikrokontrolera. Ostatnia kwestia związana zaś była z tworzeniem aplikacji internetowej. Poszczególne wątki nie były sprawami zaawansowanymi, wynikało to z tego, że ich duże zróżnicowanie wymagało często od autora rozpoznawania problematyki od samego początku. Efektem tego jest niniejsza praca. W jej przygotowaniu niezwykle użyteczne okazały się takie źródła jak portal EEVblog[13] oraz książka „Battery Management Systems: Design by Modeling” [6]. Na uwagę jak najbardziej zasługuje również biblioteka arduinoJSON[23] czy framework React[24].

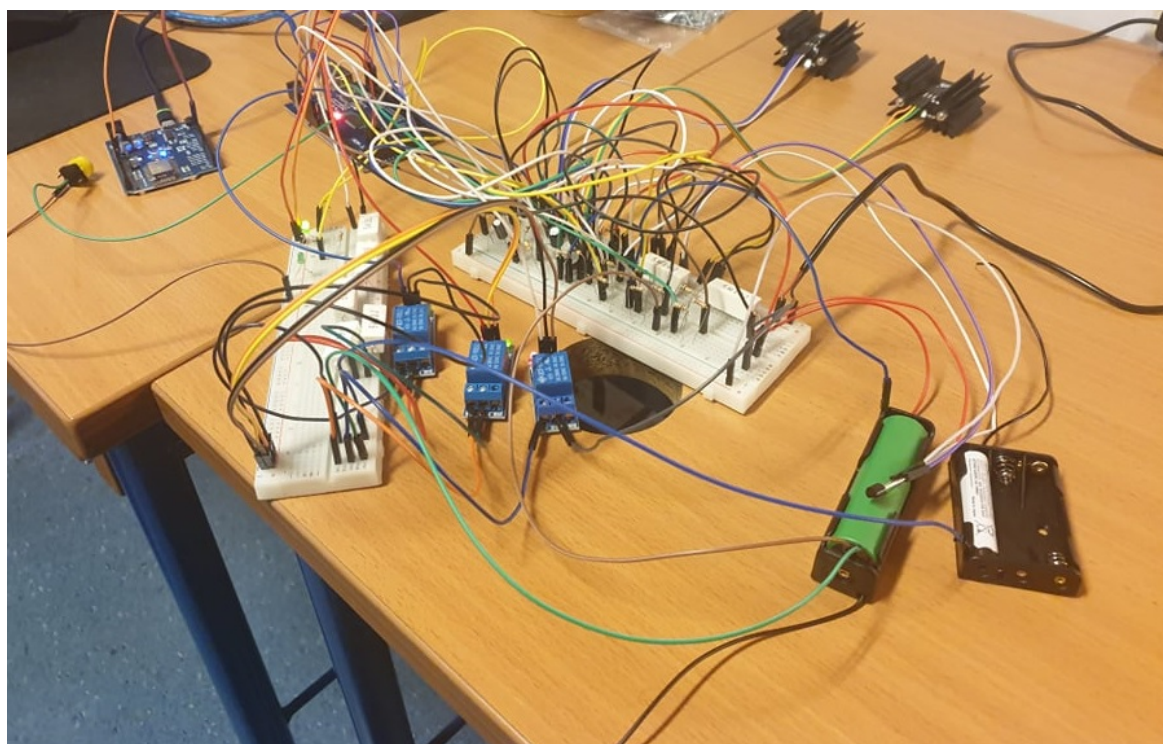
Skonstruowana uniwersalna ładowarka ma swoje wady i zalety. Wygoda jaką oferuje, jest opłacona złożonością. Wiele komponentów musi być ze sobą perfekcyjnie zgranych, aby taki system sprawnie działał. W przypadku przygotowanego prototypu nie zawsze taki efekt osiągnięto. Przykładem tego jest chociażby regulator prądu, któremu czasami spełnienie zadania przychodziło z trudnością. Znajdą się na szczęście aspekty, które lepiej wypadły na tle innych. Takim przykładem jest między innymi komunikacja pomiędzy serwerem, a mikrokontrolerem. Mimo że nie jest ona złożona, rzetelnie spełnia swój główny cel.



Rysunek 5.1. Prototyp

Przygotowany prototyp jest w pełni gotowy na potencjalne modyfikacje. Jego struktura pozwala stosować je niemalże w każdej jego części. Aplikacja internetowa i serwer mogłyby być rozwijane pod kątem większej liczby funkcjonalności. Nic nie stoi na przeszkodzie, aby dodać tam historię ładowanych baterii, statystyki użytkowników, planowanie nadsyłania profili do ładowarki, inteligentne tworzenie profili na podstawie preferencji użytkownika i wiele innych. Przy układzie elektrycznym można skupić się na zwiększeniu precyzji regulacji, ale i pomiarów. Dodanie dotykowego wyświetlacza byłoby świetnym pomysłem. W końcu kontrolowanie całego urządzenia za pomocą jednego przycisku nie należy do najbardziej intuicyjnych rozwiązań. Możliwości są niemalże nieograniczone.

Uniwersalnym ładowarkom wciąż brakuje pewnych aspektów, aby w pełni wyparły one tradycyjne systemy ładujące. Urządzenia wyspecjalizowane pod kątem jednej baterii są w końcu tańsze i sprawniejsze, jeżeli chodzi o wykorzystywaną energię. Być może jednak praca ta będzie pierwszym krokiem na drodze ich popularyzacji.



Rysunek 5.2. Prototyp

Bibliografia

- [1] M. S. Whittingham, "History, Evolution, and Future Status of Energy Storage", *Proceedings of the IEEE*, t. 100, s. 1518–1534, 2012.
- [2] Katedra Chemii Fizycznej Politechnika Gdańska, *Ogniwa Galwaniczne*, https://web.archive.org/web/20141027105523/http://fizyczna.chem.pg.gda.pl/files/2012/10/PTL_004.pdf, 2012.
- [3] *BK-4MCCA8BA NiMH Battery Panasonic*, Dostęp zdalny (23.01.2022): https://www.panasonicbatteryproducts.com/eneloop_rechargeable_batteries/eneloop-rechargeable-batteries-aaa-8-pack/, 2017.
- [4] *ENERGIZER E92 datasheet*, 2017.
- [5] S. Fletcher, *Bottled Lightning: Superbatteries, Electric Cars, and the New Lithium Economy*. New York: Hill i Wang, 2011.
- [6] H. Bergveld, W. Kruijt, P. Notten i P. H. L. Notten, *Battery Management Systems: Design by Modelling*. Boston : Kluwer Academic: Dordrecht, 2002.
- [7] T. Instruments, *Adjustable 3-Terminal Regulator for Low-Cost Battery Charging Systems*, https://www.ti.com/lit/an/snva581/snva581.pdf?ts=1642975924822&ref_url=https%253A%252F%252Fwww.google.com%252F, 2011.
- [8] *LM317 NationalSemiconductor datasheet*, 2007.
- [9] M. Bakierska i A. Chojnacka, "Akumulatory Litowe jako współczesne systemy magazynowania energii", *Wiadomości Chemiczne*, t. 68, s. 858–870, 2014.
- [10] D. Linden i T. B. Reddy, "Handbook of Batteries", w New York: R.R. Donnelley Sons Company, 2002, Chapter 28.
- [11] —, "Handbook of Batteries", w Chapter 29.
- [12] *Cell Type BK-4MCC Panasonic*.
- [13] D. L. Jones, *NiMH and NiCd Battery Charging*, <https://www.eevblog.com/batteries-charging/>, 2009.
- [14] D. L. Electronics, "Reasons for precharging lithium batteries when charging", Dostęp zdalny (26.01.2022): <https://www.large.net/news/6cu43qd.html>, 2019.
- [15] *US18650VTC4 Sony Energy Devices Corporation*, 2012.
- [16] P. Sonic, "How to charge a lead acid battery", Dostęp zdalny (26.01.2022): <https://www.power-sonic.com/blog/how-to-charge-a-lead-acid-battery/>.
- [17] *IRF540N FairchildSemiconductor datasheet*, 2002.
- [18] OKAWA Electric Design, *RC Low-pass Filter Design for PWM*, <http://sim.okawa-denshi.jp/en/PWMtool.php>.
- [19] Best-Microcontroller-Projects, *Arduino Voltage Reference*, <https://www.best-microcontroller-projects.com/arduino-voltage-reference.html>.
- [20] *ATmega640/V-1280/V-1281/V-2560/V-2561/V datasheet*.
- [21] *LM150/LM350A/LM350 3-Amp Adjustable Regulators*, 2013.
- [22] *Aavid Thermalloy 6298B-2 Datasheet*, 2001.
- [23] B. Blanchon, *A JSON library for embedded C++*. <https://arduinojson.org/>, 2021.

[24] Meta Platforms, *React*, <https://pl.reactjs.org/>, 2021.

Wykaz symboli i skrótów

FIFO – kolejka typu kto pierwszy przyszedł, pierwszy wyjdzie

HTTP – protokół internetowy, podstawowy sposób wymiany informacji

I²C – Magistrala do przesyłania danych pomiędzy różnymi urządzeniami elektrycznymi

JSON – format tekstowy do przesyłu danych

Li-Ion – akumulator litowo-jonowy

MOSFET – rodzaj tranzystora u którego przepływ prądu zależny jest od przyłożonego do bramki napięcia

NiCD – akumulator niklowo-kadmowy

NiMH – akumulator niklowo-metalowo-wodorkowy

PWM – metoda dostarczania sygnału w której ma on stałą amplitudę i częstotliwość, a sterowanie nim odbywa się poprzez zmianę jego wypełnienia

Spis rysunków

1.1 Ładowarka prądem stałym[7]	8
1.2 Wstępny diagram układu	9
2.1 Charakterystyka ładowania baterii NiMH[12]	12
2.2 Temperatura ogniw w zależności od przepuszczonego ładunku[6]	13
2.3 Porównanie charakterystyk ogniw NiCd i NiMH[6]	14
2.4 Charakterystyka ładowania ogniwa Li-Ion[15]	15
3.1 Schemat blokowy systemu	17
3.2 Zastosowany w prototypie układ regulatora prądu	23
3.3 Zasymulowana charakterystyka skokowa filtra dolnoprzepustowego[18]	24
3.4 Część pomiarowa układu	28
3.5 Pętla rozładowująca i dodatkowe miejsca na ogniwa	29
3.6 Pełny diagram obwodu	30
3.7 Diagram klas z uwzględnieniem najważniejszych atrybutów	34
3.8 Komunikacja komponentów w systemie	40
3.9 Panel Logowania	41
3.10 Panel ustawień i monitorowania baterii	42
3.11 Formularz edycji profili ładowania	42
4.1 Uzyskana charakterystyka ładowania akumulatora NiMH	45
4.2 Zmiana temperatury podczas ładowania akumulatora NiMH	46
4.3 Uzyskana charakterystyka ładowania akumulatora Li-Ion	47

5.1 Prototyp	50
5.2 Prototyp	51

6. Wyznaczania dokładnej wartości odniesienia 1.1V

Instrukcja przeznaczona dla mikrokontrolerów z rodziny Arduino. Jej celem jest wyznaczenie dokładnej wartości wbudowanego w układ odniesienia 1.1V, które zgodnie ze specyfikacją może wahać się w przedziale od 1.0V do 1.2V. Wymagane narzędzia to woltomierz oraz potencjalnie 2 kable i kondensator 100 μ F.

1. Upewnij się, że do pinu Vref nie jest podłączone żadne napięcie.
2. Wgraj do mikrokontrolera poniższy kod

```
1      void setup () {
2          analogReference (INTERNAL); // dla Mega INTERNAL1V1
3          pinMode (A0, INPUT);
4      }
5
6      void loop () {
7          analogRead (A0);
8      }
```
3. Odczytaj za pomocą woltomierza wartość napięcia na pinie Vref. „+” woltomierza powinien stykać się z pinem Vref, zaś „-” z pinem GND.
4. Dla lepszych rezultatów można podłączyć do pinu Vref kondensator 100 μ F.
5. Odczytaną wartość zanotować