



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

VESTAVĚNÝ SYSTÉM PRO EVIDENCI SPOTŘEBY VODY

EMBEDDED SYSTEM FOR WATER CONSUMPTION REGISTRATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

TOMÁŠ HAJDÍK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. ZDENĚK VAŠÍČEK, Ph.D.

BRNO 2017

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačových systémů

Akademický rok 2016/2017

Zadání bakalářské práce

Řešitel: **Hajdík Tomáš**

Obor: Informační technologie

Téma: **Vestavěný systém pro evidenci spotřeby vody**

Embedded System for Water Consumption Registration

Kategorie: Vestavěné systémy

Pokyny:

1. Seznamte se s protokolem WMBUS a principem bezdrátového odečtu stavu vodoměru. Seznamte se s platformou ESP8266 umožňující realizaci vestavěného systému dostupného prostřednictvím WIFI a dále přístupovými terminály RFID.
2. Navrhněte systém na bázi ESP8266, který bude umět evidovat množství vody spotřebované jednotlivými uživateli, kteří se budou identifikovat prostřednictvím bezkontaktní RFID karty.
3. Zpracujte studii na výše uvedené téma.
4. Navržený systém implementujte formou prototypu. Správce nechť interaguje se systémem prostřednictvím mobilního telefonu a aplikace běžící na ESP8266.
5. Diskutujte parametry navrženého řešení.

Literatura:

- Dle pokynů vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Splnění bodů 1 a 2 zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

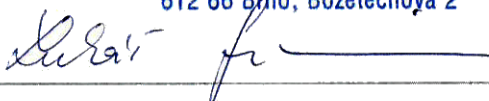
Vedoucí: **Vašíček Zdeněk, Ing., Ph.D., UPSY FIT VUT**

Datum zadání: 1. listopadu 2016

Datum odevzdání: 17. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta informačních technologií
Ústav počítačových systémů a sítí
602 00 Brno, Božetěchova 2



prof. Ing. Lukáš Sekanina, Ph.D.
vedoucí ústavu

Abstrakt

Bakalárska práca sa zaoberá problematikou evidencie stavu bezdrôtových vodomeroch. Cieľom je návrh a implementácia funkčného prototypu vstavaného systému na báze Wi-Fi modulu ESP8266, ktorý je schopný evidovať množstvo spotrebovanej vody u jednotlivých užívateľov prostredníctvom bezdrôtového odpočtu stavu vodomera. Užívatelia majú možnosť identifikácie bezkontaktnou RFID kartou. Vstavaný systém zabezpečuje svoju dostupnosť a sprístupňuje dáta administrátorovi pomocou siete Wi-Fi. Správca interaguje so systémom pomocou mobilného telefónu a aplikácie bežiacej na module ESP8266. Práca zahŕňa návrh hardwarovej konštrukcie a implementáciu softwaru pomocou jazyka C++. Výsledkom je funkčný prototyp systému, ktorý prináša nové možnosti do oblasti bezdrôtového odpočtu vodomeroch, nakoľko nevyžaduje cenovo ťažko dostupné a vysoko špecializované prístroje pre evidenciu spotreby, je prístupný z mobilného telefónu a tým rozširuje trh aj pre bežných spotrebiteľov.

Abstract

The bachelor thesis deals with the issue of the status of wireless water meters. The goal is to design and implement a function prototype of the embedded system based on ESP8266 Wi-Fi module that is capable of registering the amount of water, consumed by individual users, via the wireless meter readout. Users can be identified by a contactless RFID card. Embedded system ensures its availability and makes data available to the administrator via Wi-Fi network. The administrator interacts with the system by using a mobile phone and an application running on the ESP8266. The work includes hardware design and software implementation using C++ programming language. The result is a functional prototype of the system that brings new possibilities in the sphere of wireless meter readers because it does not require unaffordable and highly specialized appliances for recording consumption, is accessible from the mobile phone and thus extends the market to ordinary consumers as well.

Kľúčové slová

Evidencia spotreby vody, Vstavaný systém, ESP8266, Wi-Fi modul, RFID, Webový server, Bezdrôtový vodomera, RTC, DS3231, WMBUS, Framework 7

Keywords

Registration of water consumption, Embedded system, ESP8266, Wi-Fi module, RFID, Web server, Wireless water meter, RTC DS3231, WMBUS, Framework 7

Citácia

HAJDÍK, Tomáš. *Vstavěný systém pro evidenci spotřeby vody*. Brno, 2017. Bakalárska práca. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Vašíček Zdeněk.

Vestavěný systém pro evidenci spotřeby vody

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením vedúceho bakalárskej práce pána Ing. Zdeňka Vašíčka, Ph.D.. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Tomáš Hajdík
15. mája 2017

Podakovanie

Týmto by som chcel poďakovať predovšetkým vedúcemu práce Ing. Zdeňkovi Vašíčkovi, Ph.D. za jeho odborné vedenie, pomoc a cenné rady počas tvorby tejto bakalárskej práce. Zároveň by som chcel poďakovať za zapožičanie vývojových modulov.

Obsah

| | | |
|----------|---|-----------|
| 1 | Úvod | 5 |
| 2 | Vstavaný systém | 7 |
| 2.1 | Definícia vstavaného systému | 7 |
| 2.2 | Požiadavky na vstavaný systém a jeho obmedzenia | 7 |
| 3 | Platforma ESP8266 | 9 |
| 3.1 | Parametre čipu ESP8266 | 9 |
| 3.2 | Pamäť ESP8266 modulu | 10 |
| 3.3 | Flash pamäť | 10 |
| 3.4 | Rozhranie Wi-Fi | 11 |
| 3.5 | Rozhranie SPI | 12 |
| 3.5.1 | Rozhranie SPI v module ESP8266 | 12 |
| 3.6 | Rozhranie I2C | 13 |
| 3.6.1 | Rozhranie I2C v module ESP8266 | 14 |
| 3.7 | Rozhranie UART | 14 |
| 3.7.1 | Rozhranie Uart v module ESP8266 | 15 |
| 3.8 | NodeMCU | 15 |
| 4 | Prístupový terminál RFID | 16 |
| 4.1 | Princíp činnosti RFID technológie | 16 |
| 4.2 | MFRC522 modul | 17 |
| 5 | Možnosti užívateľského rozhrania | 19 |
| 5.1 | Alternatívy užívateľského rozhrania | 19 |
| 5.2 | Framework7 | 20 |
| 5.2.1 | Popis základných elementov | 20 |
| 6 | Meranie spotreby vody a technológia Wireless M-Bus | 22 |
| 6.1 | Zameranie | 22 |
| 6.2 | Režimy činnosti | 23 |
| 6.3 | Kódovanie prenášaných dát | 23 |
| 6.4 | Podpora na strane výrobcov | 24 |
| 7 | Návrh vstavaného systému | 25 |
| 7.1 | Špecifikácia navrhovaného prototypu | 26 |
| 7.2 | Návrh hardvéru vstavaného systému | 26 |
| 7.2.1 | Pripojenie RFID MFRC522 modulu k prototypu | 26 |

| | | |
|-----------|---|-----------|
| 7.2.2 | RTC modul | 27 |
| 7.2.3 | Pripojenie RTC DS3231 modulu k prototypu | 27 |
| 7.2.4 | Spojenie prototypu s bezdrôtovým vodomerom | 27 |
| 7.2.5 | Pripojenie čipu ESP8266 k sieti Wi-Fi | 28 |
| 7.3 | Návrh užívateľského rozhrania a dátových štruktúr | 28 |
| 7.3.1 | Návrh spôsobu uloženia dát o spotrebe vody | 29 |
| 7.3.2 | Návrh grafického užívateľského rozhrania | 31 |
| 8 | Implementácia funkčného prototypu | 33 |
| 8.1 | Implementácia softvéru pre vstavaný systém a modul ESP8266 | 33 |
| 8.2 | Implementácia dátových štruktúr slúžiacich na uchovanie evidenčných dát | 35 |
| 8.3 | Implementácia webového grafického užívateľského rozhrania | 36 |
| 8.4 | Princíp komunikácie webového rozhrania a navrhnutého prototypu | 40 |
| 9 | Parametre a optimalizácia navrhnutého prototypu | 42 |
| 10 | Finálna podoba prototypu a jeho využitie | 43 |
| 11 | Záver | 46 |
| | Literatúra | 47 |
| | Zoznam použitých skratiek | 49 |
| | Prílohy | 50 |
| A | Obsah priloženého pamäťového média | 51 |
| B | Výsledný prototyp | 52 |
| C | Zoznam použitých komponentov | 54 |
| D | Plagát | 55 |

Zoznam obrázkov

| | | |
|------|--|----|
| 2.1 | Blokový diagram všeobecného vstavaného systému [9]. | 7 |
| 3.1 | Diagram funkčných blokov modulu ESP8266 [12]. | 10 |
| 3.2 | Bloková schéma základného SPI rozhrania [10]. | 12 |
| 3.3 | Bloková schéma SPI prenosu [10]. | 13 |
| 3.4 | I2C paket [10]. | 13 |
| 3.5 | Asynchrónny sériový dátový rámec UART [10]. | 14 |
| 3.6 | Vývojový modul Node MCU Devkit 1.0 s osadeným čipom ESP8266 [2]. . . | 15 |
| 4.1 | Pasívny RFID transpondér [1]. | 16 |
| 4.2 | Pasívne RFID, ilustrácia odrazu signálu medzi čítačkou a transpondérom [22]. | 17 |
| 4.3 | RFID čítačka s mikrokontrolérom MFRC522 [4]. | 18 |
| 4.4 | Zjednodušená blokovaná schéma MFRC522 [18]. | 18 |
| 6.1 | Zameranie protokolu Wireless M-Bus [17]. | 22 |
| 7.1 | Blokový diagram vstavaného systému a jeho komponentov. | 25 |
| 7.2 | Modul DS3231 AT24C32 obsahujúci IIC sériové rozhranie [8]. | 27 |
| 7.3 | Schéma pripojenia modulov RTC a RFID k NodeMCU s čipom ESP8266. . . | 28 |
| 7.4 | Navrhnutý prototyp a komunikácia s okolím. | 29 |
| 7.5 | Formát jedného sektoru binárneho súboru slúžiaceho na uloženie údajov uživateľa. | 30 |
| 7.6 | Wireframe grafického užívateľského rozhrania správcu systému. | 31 |
| 8.1 | Znázornenie logickej štruktúry implementovaného programu a jeho rozdelenia na inicializačnú fázu a fázu behu programu. | 34 |
| 8.2 | Prihlasovacia obrazovka grafického užívateľského rozhrania. | 37 |
| 8.3 | Hlavná obrazovka grafického užívateľského rozhrania. Vľavo zoznam evidovaných kariet. Vpravo výber obdobia v kalendári. | 38 |
| 8.4 | Zobrazenie akčného zoznamu s nastavením usporiadania. | 39 |
| 8.5 | Detailný náhľad spotreby jedného užívateľa. | 40 |
| 10.1 | Konštrukcia výsledného prototypu so spusteným záložným napájaním. . . . | 43 |
| 10.2 | Vnútro prototypu zhora. Vľavo prototyp obsahujúci RFID modul. Vpravo zobrazenie modulov NodeMCU, RTC a prijímača CC1101. | 44 |
| 10.3 | Prototyp s otvoreným krytom. | 44 |
| 10.4 | Bočný pohľad na prototyp s umiestneným RFID modulom. | 45 |
| B.1 | Fotografia prototypu v puzdre s pripojenou anténou. | 52 |

| | | |
|-----|--|----|
| B.2 | Fotografia prototypu zobrazujúca uloženie komponentov bez RFID modulu pre lepšiu viditeľnosť | 53 |
| D.1 | Zhrňujúci plagát práce. | 56 |

Kapitola 1

Úvod

V súčasnosti nás obklopujú mnohé komplexné systémové celky, ktoré si vyžadujú presné riadenie, no zároveň od nich ľudia vyžadujú čo najväčšiu mieru jednoduchosti z pohľadu používania a ovládania. Takéto rozhranie zabezpečujú vstavané systémy, ktoré sú ako jednouchový prvok často vyrábané sériovo a sú úplne zabudované do konkrétneho zariadenia. Správny vstavaný systém je ten, ktorý používateľ ani nevníma a netuší, že s ním pracuje. Tak ľuďom uľahčuje inak zložité činnosti a zároveň šetrí ich drahocenný čas. Preto však vznikajú nároky na to, aby bol ľahko ovládateľný a intuitívny.

Prístroje používané pre bezdrôtovú evidenciu stavu vodomerov sú veľmi nákladné. Využívajú ich najčastejšie iba priamo vodárenské spoločnosti pre zjednodušenie činnosti odpočtu, či už v neprítomnosti klienta alebo z dôvodu obmedzeného prístupu priamo k meradlu.

Hlavným cieľom tejto práce je navrhnúť a implementovať prototyp vstavaného zariadenia a proces internej evidencie tak sprístupniť širokej verejnosti od správcov budov, ubytovacích zariadení, firemných priestorov až po poskytovateľov privátneho ubytovania či majiteľov inteligentných domácností. Tí tak budú môcť riešiť internú evidenciu vo vlastnej réžii s minimálnymi investičnými nákladmi a nákladmi spojenými s prevádzkou celého systému.

Evidenčný systém je schopný bezdrôtovo komunikovať s vodomerom a zaznamenávať spotrebované hodnoty. Globálna použiteľnosť si vyžaduje viacero užívateľov, čím však vzniká potreba ich jednoznačnej identifikácie. To je dôvodom využitia RFID terminálu.

Správca systému tak bude mať možnosť jednoduchej evidencie spotreby všetkých ľudí v systéme. Zvolený modul 32-bitový ESP8266 obsahuje Wi-Fi transceiver a zároveň je ľahko dostupný na súčasnom trhu. Tým sa stáva z neho vhodná platforma slúžiaca ako jadro celej aplikácie a zároveň umožňujúca jednoduché pripojenie mobilným telefónom. Modul taktiež obsahuje vnútornú pamäť s možnosťou využitia primárne ako úložného priestoru pre dáta, ktoré bude tento systém spracovávať, v prehľadnej forme zobrazovať a taktiež umožňovať jednoduchú filtráciu.

Čip ESP8266 bol pôvodne navrhnutý ako rozširujúce Wi-Fi SoC riešenie k rodine mikročipov Arduino a umožnil tak bezdrôtové internetové pripojenie. Práca však využíva nástroje, ktoré umožňujú pre tento modul navrhnúť vlastný firmware, čo pôvodne vôbec nebolo v zámere výrobcov tohto čipu.

Navrhnutý systém má za úlohu priniesť verejnosti nové riešenie pre evidenciu spotrebovanej vody. Kládne dôraz na jeho jednoduchú dostupnosť pomocou mobilnej platformy, užívateľsky prívetivé rozhranie, nízke finančné náklady a čo najefektívnejšie využitie zdro-

jov. Taktiež v neposlednom rade pri jeho návrhu zohľadňuje dĺžku prevádzky výsledného systému, keďže pri evidencii spotreby vody sa predpokladá dlhodobý chod systému.

Kapitola 2

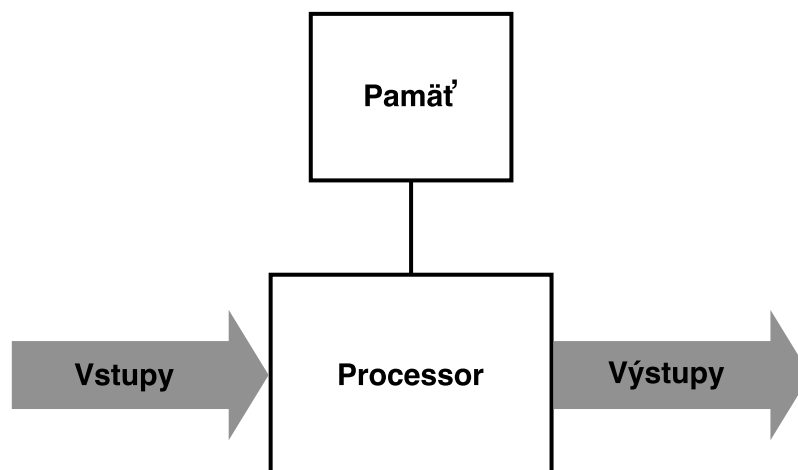
Vstavaný systém

2.1 Definícia vstavaného systému

Vstavaný systém je počítačový systém, ktorý je účelovo určený pre konkrétnu aplikáciu v systéme [20]. Taktiež môže byť chápaný ako kombinácia počítačového hardvéru a softvéru, prípadne ďalších mechanických alebo iných častí, určených na výkon osobitých funkcií. Najčastejšie je vstavaný systém súčasťou nejakého väčšieho systému. Ak je vstavaný systém dobre navrhnutý, existencia procesoru a softvéru môže pracovať bez zásahu používateľa.

2.2 Požiadavky na vstavaný systém a jeho obmedzenia

Všetky vstavané systémy pozostávajú z nejakého typu vstupov a výstupov. Vstupy do systému väčšinou zahŕňujú senzory, sondy, komunikačné signály alebo tlačidlá. Výstupy tvoria väčšinou displej, svetelné kontrolky, komunikačné signály alebo priamo zmeny fyzického sveta vid' obrázok č. 2.1. Každý systém je zvyčajne unikátny, pretože má úplne odlišne definované požiadavky, ktoré sú zapríčinené kompromismi pri jeho vývoji. Medzi hlavné obmedzenia pre vstavané systémy patria nízky výpočtový výkon a malé množstvo pamäte. Požiadavky na vstavaný systém sú najmä výrobná cena, počet vyrobených jednotiek, očakávaná doba životnosti, spotreba energie a spoľahlivosť [9].



Obr. 2.1: Blokový diagram všeobecného vstavaného systému [9].

V niektorých systémoch musí softvér nielen pracovať deterministicky (zakaždým presne rovnako), ale tiež v reálnom čase (vždy reagovať na udalosť dostatočne rýchlo). Kompilátory vstavaných systémov častokrát podporujú len jazyk C, alebo C a C++. Navyše veľa C++ kompilátorov pre vstavané systémy implementuje len malú podmnožinu jazyka, pričom obyčajne chýba viacnásobná dedičnosť, spracovanie a ošetrovanie výnimiek a taktiež šablóny [20].

Vstavaný systém môže byť celý implementovaný na jednom čipe len s malým množstvom podporných komponentov. Výhoda vstavaného systému je, že celková funkcionálnosť je definovaná pomocou softvéru a nie hardvéru. Softvér je najčastejšie uložený v systéme s nevolatilným typom pamäte. To zabezpečuje, že systém je odolný voči chybám a v prípade chyby je schopný sa obnoviť. Hlavnou výhodou pri nasadení vstavaných systémov v praxi je ľahká sériová výroba a jednoduchší vývoj ako pri komplikovaných elektrických obvodoch [10].

Kapitola 3

Platforma ESP8266

Táto kapitola sa venuje platforme ESP8266. Popisuje jej základné parametre vyrábaného čipu. Zaoberá sa možnosťami čipu a jeho pripojenia k sieti Wi-Fi. V kapitole sú spomenuté možné využiteľné komunikačné rozhrania, princíp ich fungovania a spôsob realizácie v rámci čipu ESP8266. V závere kapitola popisuje modul NodeMCU, ktorý je vybavený čipom ESP8266 a zjednodušuje tak vývoj aplikácií. Kapitola je zakončená možnosťami vývoja softvéru pre čip ESP8266 a spôsobom jeho programovania.

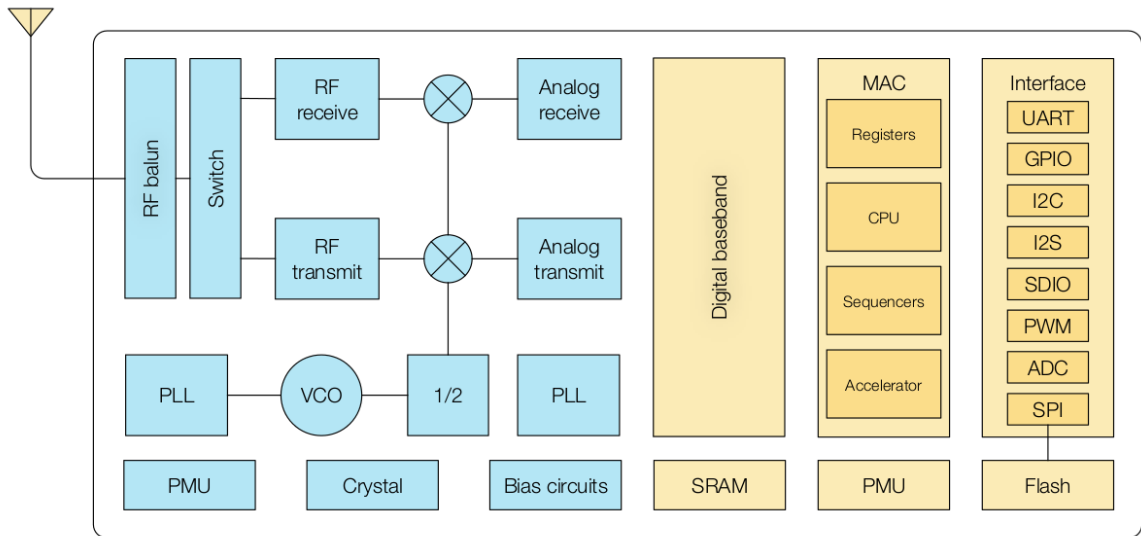
3.1 Parametre čipu ESP8266

Modul ESP8266 je vysoko integrované Wi-Fi SoC riešenie. Systém zahŕňa všetky elektronické súčiastky do jedného puzdra čipu. Tým prináša užívateľom efektívne využitie energie, kompaktný dizajn a spoľahlivý výkon. Kompaktný dizajn minimalizuje veľkosť PCB a taktiež minimalizuje potrebu externých súčiastok.

S kompletnou a sebestačnou schopnosťou Wi-Fi pripojenia môže ESP8266 pracovať buď ako samostatná aplikácia alebo ako pridružený modul k celej rade MCU. Ak ESP8266 pracuje ako samostatná aplikácia, umožňuje rýchly štart celého modulu. Integrovaná vysokorýchlostná cache pomáha zvýšiť výkon systému a pomáha optimalizovať systémovú pamäť. Základné parametre modulu sú:

- 2.4 GHz Wi-Fi transceiver
- Tensilica L106 32-bit CPU
- UART, SPI, I2C, GPIO, ADC, PWM
- 2.5 V - 3.6 V
- 32-pin (5 mm x 5 mm)

ESP8266 modul taktiež obsahuje okrem vstavaného Wi-Fi aj vylepšenú verziu 32-bitového procesoru Tensilica Diamond L106 viď obrázok č. 3.1. Frekvencia procesoru je 80Mhz. V základe iba 20% zásobníka má obsadená Wi-Fi, zvyšok môže byť použitý na užívateľské aplikácie. Mikroprocesor môže byť prepojený s externými senzormi a ďalšími zariadeniami cez GPIO, SPI, I2C, UART a ADC rozhrania. Prevádzkové napätie sa pohybuje medzi 2.5 V až 3.6 V, pričom priemerný prúd je 80 mA [12].



Obr. 3.1: Diagram funkčných blokov modulu ESP8266 [12].

3.2 Pamäť ESP8266 modulu

ESP8266 integruje radič pamäte a pamäťové jednotky vrátane SRAM a ROM. Ak ESP modul pracuje v režime stanice a je pripojený k routeru, veľkosť pamäte RAM je menšia ako 50kB. V SoC neexistuje žiadna programovateľná ROM. Užívateľský program musí byť uložený v externej pamäti prístupnej cez SPI rozhranie. ESP8266 používa externú SPI Flash ROM na uloženie používateľského programu. Modul podporuje 16MB maximálnu teoretickú kapacitu [11]. Rozloženie celej pamäte je zobrazené v tabuľke 3.1. Chránená oblasť je zvyčajne chránená proti čítaniu aj zápisu. Oblasť RAM je určená pre uloženie dát spustenej aplikácie a systému. Obsah tejto časti pamäte sa stratí pri vypnutí alebo režime spánku. Úsek ROM je určený iba na čítanie. Obsahuje inštrukcie, ktoré sú naprogramované v čipe ESP8266. Dôležitou časťou je úsek mapovania SPI Flash. Externá pamäť je mapovaná na tento adresový rozsah prístupný pomocou SPI rozhrania. Mapovanie je potrebné zapnúť v ROM. Oblasť mapovania I/O portov obsahuje rozhrania pre väčšiu časť periférnych zariadení vo vnútri čipu ESP8266 [13].

3.3 Flash pamäť

Pamäť Flash je najnovší pokrok v technológii pamätí. Kombinuje všetky najlepšie vlastnosti pamäťových zariadení. Flash pamäte majú veľkú hustotu zápisu a nízku cenu. Taktiež sú nevolatilné, rýchle (hlavne pri čítaní) a elektricky prepisovateľné. Vlastnosti pamäte sú:

- nevolatilitnosť
- prepisovateľnosť
- sektor - najmenšia prepisovateľná jednotka
- limitovaný počet prepisovacích cyklov

| Adresa | Typ | Popis |
|-----------|--------|---|
| 00000000h | exc | Chránená oblasť. |
| 20000000h | žiadny | Nenamapované, opakované bajty 00 80 00 00. |
| 3FF00000h | I/O | Pamäťové mapovanie I/O portu Dport0. |
| 3FF10000h | žiadny | Nenamapované, opakované bity 00 00 00 00. |
| 3FFE8000h | RAM | Dáta RAM užívateľa. |
| 3FFFC000h | RAM | ETS systémové dáta RAM. |
| 40000000h | ROM | Vnútoraná ROM. |
| 40100000h | RAM | RAM inštrukcií. Používaná pre bootloader načítania SPI Flash. |
| 40200000h | flash | Mapovanie SPI Flash. |
| 60000000h | I/O | Pamäťové mapovanie I/O portov. |
| 60002000h | exc | Chránená oblasť. |
| 70000000h | žiadny | Nenamapované, opakované bajty 00 80 00 00. |

Tabuľka 3.1: Tabuľka zobrazujúca úseky rozloženia pamäťového priestoru čipu ESP8266 [13].

- rýchle čítanie, pomalý zápis
- priemerná cena

Zo softvérového hľadiska sú si Flash a EEPROM pamäte veľmi podobné. Hlavným rozdielom je, že pri Flash pamäti je možné zmeniť najmenšiu jednotku jeden sektor nie bajt. Predtým ako môže byť jeden sektor zapísaný, musí byť najskôr kompletne zmazaný. Typická veľkosť sektoru je v rozsahu 256 bajtov až 16 kilobajtov. Napriek tejto nevýhode je viac populárna Flash pamäť ako EEPROM a uplatňuje sa vo veľkom množstve pamätí ROM [9].

SPI Flash ROM v čipe ESP8266 pozostáva z viacerých úsekov. Prvým je úsek pamäťového priestoru pre uloženie užívateľskej aplikácie. Jednu oblasť tvorí aj časť pre SDK knižnicu. Ďalší úsek obsahuje prednastavenú konfiguráciu. Posledný úsek je prázdny, ale môže obsahovať Wi-Fi konfiguráciu. SPI Flash ROM je možné čítať ako dáta zarovnané po 4 bajtoch [13].

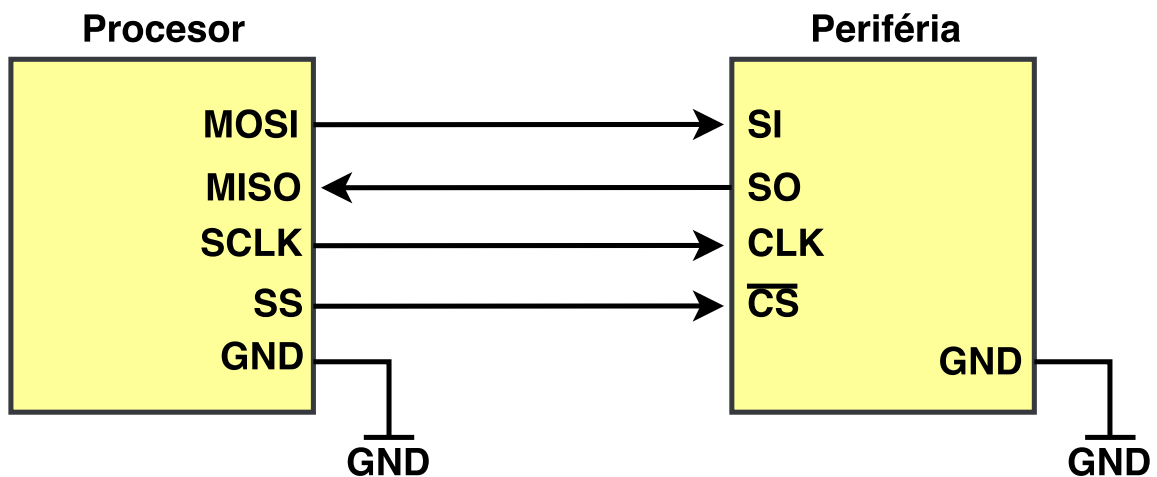
3.4 Rozhranie Wi-Fi

Modul obsahuje 2.4 GHz prijímač a vysielač. Vysielač poháňa anténu z vysokovýkonného CMOS výkonového zosilovača. ESP8266 implementuje Wi-Fi protokoly 802.11 b/g/n/e/i WLAN MAC a Wi-Fi Direct. Taktiež implementuje štandardné kanály podľa štandardu IEEE802.11 b/g/n. K dispozícii je zabezpečenie Wi-Fi pomocou WPA/WPA2 a šifrovanie WEP. Modul umožňuje používanie IPv4 protokolu na sieťovej vrstve. Transportnú vrstvu pokrývajú TCP a UDP protokoly. Podporované protokoly na aplikačnej vrstve sú HTTP a FTP [12].

3.5 Rozhranie SPI

SPI (Serial Peripheral Interface) bol vyvinutý firmou Motorola. Je to spôsob ako zabezpečiť výmenu informácií medzi dvoma zariadeniami. Ide o synchronný sériový protokol, umožňujúci zariadeniu typu master komunikovať so zariadením typu slave. Dáta sú vzorkované hodinovým signálom SCLK. Hodinový signál je ovládaný zariadením typu master. Kým nie je aktívne SCLK, nemôže dôjsť k prenosu dát. Signál SCLK tiež rozhoduje, kedy môžu byť dáta zmenené a kedy sú platné na čítanie. Polaritu hodín SCLK je možné nastaviť.

Rozhranie obsahuje piny SS, MOSI, MISO a SCLK. Slave select pin slúži pre adresáciu zariadenia na strane príjemcu. Signál je aktívny v úrovni logickej nuly a v úrovni logickej jednotky nie je vybrané žiadne slave zariadenie. Piny MISO (Master Input Slave Output) a MOSI (Master Output Slave Input) slúžia ako dve dátové linky smerujúce od mastera ku slave a naopak. Obrázok č. 3.2 zobrazuje rozhranie SPI medzi mikroprocesorom a perifériou [16].

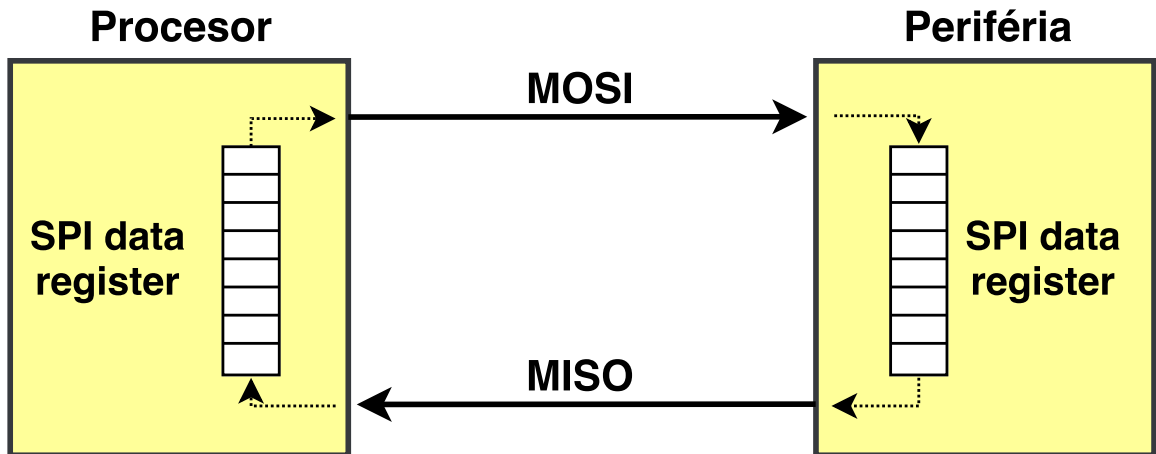


Obr. 3.2: Bloková schéma základného SPI rozhrania [10].

SPI je zároveň výmenný protokol, čo znamená, že jedny dáta sú odosielané a zároveň sú iné dáta prijímané. Obe strany master aj slave obsahujú posuvný register viď obrázok č. 3.3. Master začne prenos zapísaním bajtu do jeho SPI posuvného registra. Ako register prenáša bajt do zariadenia slave cez MOSI vodič, zariadenie slave odosiela obsah jeho posuvného registra späť zariadeniu typu master cez vodič MISO. Týmto spôsobom sa obsahy dvoch posuvných registrov vymenia. Obe operácie pre zápis aj čítanie sú vykonané simultánne. SPI preto patrí medzi veľmi efektívne protokoly [10].

3.5.1 Rozhranie SPI v module ESP8266

ESP8266 má tri SPI rozhrania. Prvé Master/Slave SPI, druhé Slave SDIO/SPI a tretie Master/Slave HSPI. Režim SPI môže byť implementovaný pomocou softwarového programu. Maximálna frekvencia SCLK je 80 MHz [12]. ESP8266 SPI modul má špeciálnu podporu pre flash pamäť v SPI rozhraní. Taktiež umožňuje špeciálne programovanie polaroty hodín a výberu medzi prvým MSB a LSB [11].



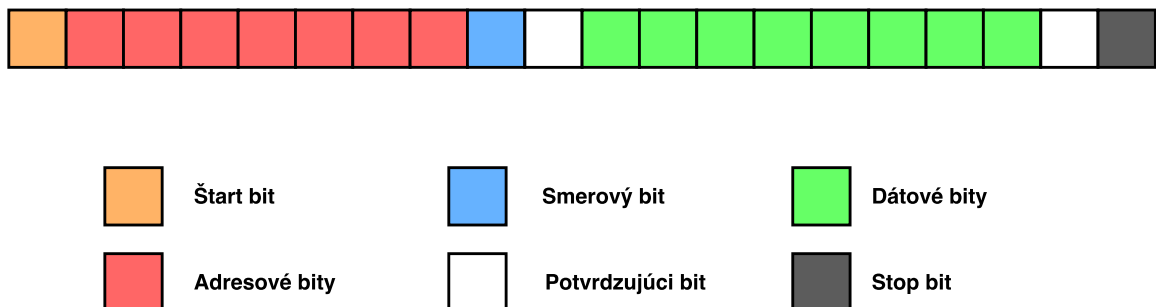
Obr. 3.3: Blokovaná schéma SPI prenosu [10].

3.6 Rozhranie I2C

I2C (Inter-Integrated Circuit) protokol, niekedy známy aj ako TWI (Two Wire Interface), bol vyvinutý firmou Philips Semiconductors, dnes známou ako NXP, v snahe vytvoriť jednoduchý obojsmerný protokol s použitím iba dvoch vodičov. Protokol využíva iba dva vodiče: sériovú dátovú linku SDA a sériovú linku pre prenos hodín SCL. I2C môže mať až 128 zariadení pripojených na rovnaké dva vodiče. Niektoré I2C zariadenia používajú 5 V a iné 3,3 V napätie, na čo si treba dať najväčší pozor pri ich používaní. Dôležité je pozorne si prečítať manuál a použiť správne napätie pred pripojením iného I2C zariadenia.

I2C protokol je podobný SPI protokolu v tom, že taktiež obsahuje master a slave zariadenia. Každé zariadenie má svoju I2C adresu. Každý dátový bit je odosielaný v každom takte hodinového signálu. Komunikácia začína, keď master odošle štart bit na zbernicu a je ukončená, keď master odošle stop bit [16].

Rozhranie I2C na rozdiel od SPI nevyužíva signál SS. Každé zariadenie má unikátnu adresu, pričom každý odosielaný paket začína s adresovými bitmi adresujúce dátové bity. Adresové bity sú nasledované smerovým bitom, ktorý podľa hodnoty 0 alebo 1 dáta buď odosiela alebo prijíma. Existuje však aj špeciálna adresa tvorená samými nulami, určená pre adresáciu všetkých zariadení. Príklad paketu prenášajúceho jeden byte dát je zobrazený na obrázku 3.4 [10].



Obr. 3.4: I2C paket [10].

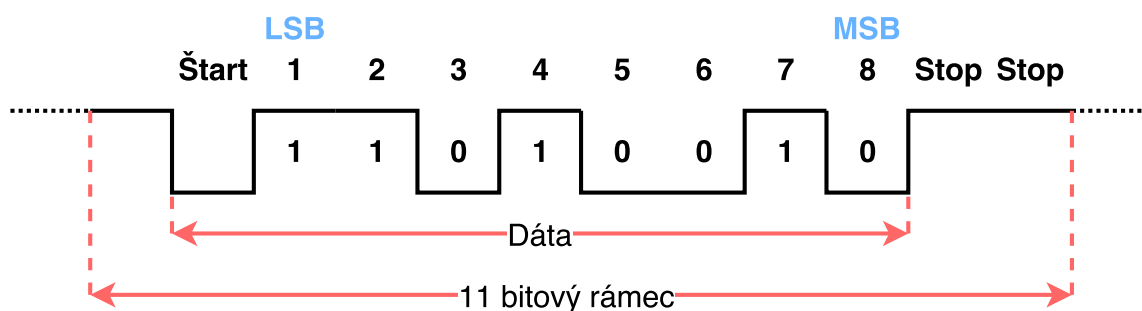
3.6.1 Rozhranie I2C v module ESP8266

ESP8266 má dve emulované I2C rozhrania používané na pripojenie k mikrokontroléru a iným periférnym zariadeniam a senzorom. ESP modul podporuje oba režimy I2C master aj I2C slave. Umožňuje tak čítanie aj zápis na rozdiel od iných I2C zariadení. Všetky GPIO piny môžu byť použité ako rozhranie pre I2C dáta alebo hodinový signál. Čip má vnútorné pull-up rezistory, ktoré môžu ochrániť aj vonkajšie pull-up rezistory. Funkčnosť I2C rozhrania môže byť taktiež realizovaná pomocou softvérového programu. Taktovacia frekvencia je maximálne 100 kHz. Dôležitým faktom je, že I2C hodinová frekvencia by mala byť vyššia než najpomalšia hodinová frekvencia zariadenia typu slave [11].

3.7 Rozhranie UART

Najjednoduchšia forma sériového rozhrania je UART (Universal Asynchronous Receiver Transmitter) niekedy označovaný aj ako ACIAs. Toto rozhranie je označované ako asynchrónne. UART pozostáva z dvoch častí a to z prijímača (označovaného Rx), ktorý konvertuje sériový bitový prúd na paralelné dáta pre mikroprocesor a z vysielача (označovaného Tx), ktorý konvertuje paralelné dáta z mikroprocesoru do sériovej podoby pre vysielanie. Rozhranie taktiež poskytuje informácie o stave ako napríklad, že prijímač je plný (dáta boli prijaté) alebo vysielач je prázdny (čakanie na prenos je dokončené). Pri prenose cez rozhranie UART má väčšinou každé zariadenie svoj lokálny hodinový signál. Zariadenia tak musia operovať na presne rovnakej frekvencii s rovnakou fázou.

Prenosový formát používa jeden začiatkový štart bit na začatie prenosu a jeden alebo dva stop bity na konci každého prenášaného znaku. Prijímač si zosynchronizuje jeho hodinový signál na prijatom štart bite a potom vzorkuje jednotlivé dátové bity. Po prijatí stop bitu(ov) prijímač predpokladá, že prenos bol úspešný a prijal platný znak. Konverzia z paralelných na sériové bity je väčšinou vykonávaná dedikovaným UART hardvérom. No v niektorých systémoch, ktoré obsahujú len paralelné vstupno/výstupné porty môže byť konverzia vykonávaná taktiež pomocou softvéru, ktorý prepína jeden pin paralelného rozhrania. Pre možnosť detekcie chýb vzniknutých pri prenose sa často pridáva paritný bit. UART spočíta paritu z prenášaného bajtu a po prenose vykoná kontrolu parity. Existuje niekoľko typov parity, pričom najzákladnejšie sú párna a nepárna parita [10].



Obr. 3.5: Asynchrónny sériový dátový rámeč UART [10].

3.7.1 Rozhranie Uart v module ESP8266

ESP8266 má dve UART rozhrania UART0 a UART1. Dátové prenosy z alebo do UART rozhrania môžu byť realizované pomocou hardvéru. Dátová prenosová rýchlosť cez UART rozhranie dosahuje 115 200 x 40, čo je 4,5 Mb za sekundu. UART0 môže byť použitý pre komunikáciu, pretože podporuje riadenie toku na rozdiel od UART1, ktorý plní len funkciu vysielania dát signálom Tx. Zvyčajne sa preto používa na odosielanie logovacích dát.

V predvolenom nastavení UART0 odosiela na výstup nejaké informácie pri zapnutí a bootovaní procesoru. Prenosová rýchlosť odosielaných informácií je úmerná frekvencii externého kryštálového oscilátora. ESP8266 podporuje prenosové rýchlosti v rozsahu od 300 Bd do 115 200 Bd. V prípade, ak je frekvencia oscilátora 40 MHz, prenosová rýchlosť pre výstup je 115 200 Bd. Ak je frekvencia kryštálového oscilátora iba 26 MHz, prenosová rýchlosť je 74 880 Bd [12].

3.8 NodeMCU

Modul Node MCU je vývojový kit navrhnutý firmou NodeMCU Inc. Ide o nízkonákladový vývojový open-source modul, ktorý je zameraný na poskytnutie jednoduchej konfigurácie a nastavení. Je veľmi ľahko dostupný za približne menej ako 2 doláre a tým sa z neho stáva veľmi užitočný komponent umožňujúci rýchle prototypovanie pri čo najnižších nákladoch [5]. Hardvérová platforma je založená na vývojovom module ESP8266 s Lua vývojovým prostredím pre aplikácie z oblasti IoT. Hardvérová výbava obsahuje možnosť pre pripojenie mikro USB kábla a umožňuje tým jednoduché nahrávanie vlastných programov rovnako ako pri vývojových doskách Arduino. Ide o otvorený hardvér s ESP8266-12E mikroprocesorom obsahujúci 32Mbit (tj. 4MB) pamäte flash [3].

Piata generácia vývojového kitu označovaná ako Devkit 1.0 využíva radič CP2102 ako rozhranie medzi sériovou linkou UART a USB konektorom. Ponúka tak možnosť jednoduchého nahratia firmvéru. Ten je možné nahráť pomocou nodemcu-flasher dostupného na GitHubu. Ponúka taktiež zabudované rozhrania ako napríklad: I2C, SPI, PWM, Wi-Fi, ADC, GPIO, UART. Zároveň ponúka využitie časovača, súborového systému a systémového api [3].



Obr. 3.6: Vývojový modul Node MCU Devkit 1.0 s osadeným čipom ESP8266 [2].

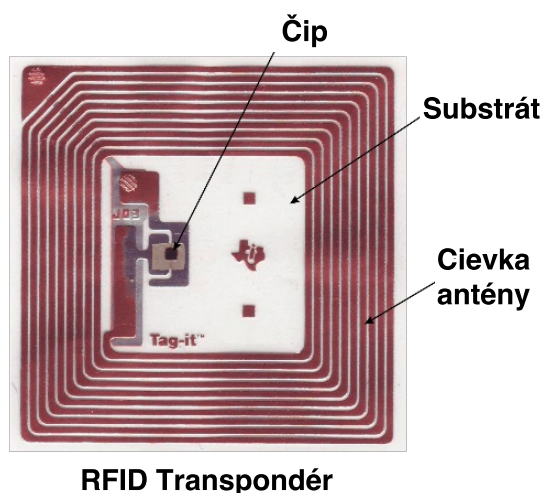
Kapitola 4

Prístupový terminál RFID

4.1 Princíp činnosti RFID technológie

RFID je skratka anglického pomenovania Radio Frequency Identification, čo v preklade znamená rádio frekvenčná identifikácia. Ide o technológiu, ktorá umožňuje identifikáciu objektu s využitím rádiového systému. Je to bezkontaktný, bezdrôtový dátový prenos na krátku vzdialenosť. RFID systémy sa rozdeľujú na základe využívaného rádiových frekvenčného pásma a potreby napájania. Podľa použitej frekvencie sú to nízkofrekvenčné, vysokofrekvenčné, ultravysokofrekvenčné a mikrovlnné systémy. Podľa potreby napájania ich delíme na pasívne, polo pasívne a aktívne [22].

RFID systémy sú vždy zložené z dvoch častí, z čítačky a transpondéru. Čítačka typicky obsahuje rádio frekvenčný modul pozostávajúci z prijímača a vysielača a riadiacu jednotku. V závislosti na technológii a dizajne, čítačka môže umožňovať buď len čítanie alebo čítanie aj zápis. Navyše veľa čítačiek je vybavených ďalším rozhraním, napríklad I2C alebo SPI, ktoré mu umožňujú preposielať dáta do ďalšieho systému. Príklad RFID čítačky s mikrokontrolérom MFRC522 je vidieť na obrázku č. 4.3 [14].

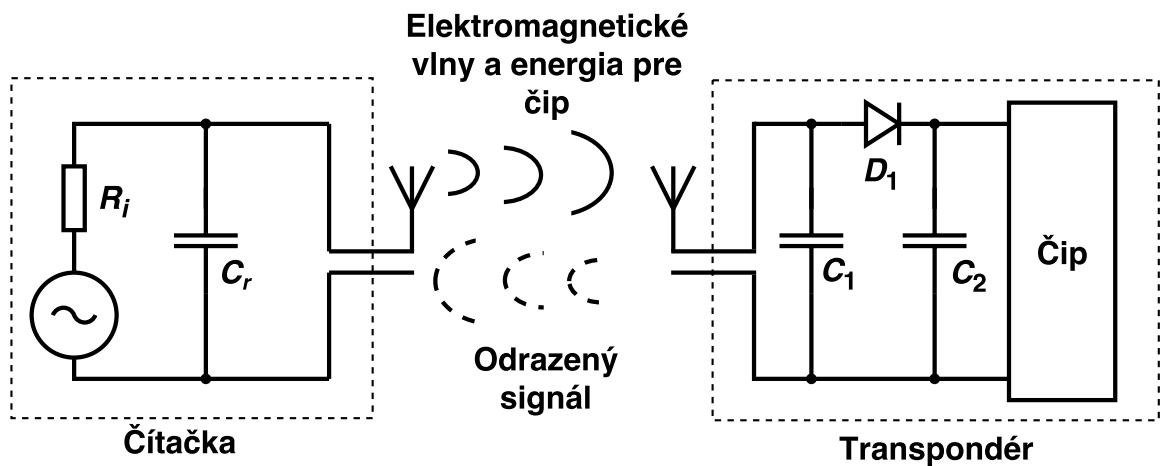


Obr. 4.1: Pasívny RFID transpondér [1].

Transpondér je zväčša umiestnený na objekte, ktorý má byť identifikovaný. Transpondér tvorí cievka, kondenzátor a obvod umožňujúci uloženie informácie. Identifikačné karty

obsahujúce transpondér umožňujú dátový prenos a uloženie dát bez použitia galvanického spojenia, len s využitím elektromagnetického poľa. Transpondér, ktorý neobsahuje žiadne svoje samostatné napájacie zariadenie, je úplne pasívny ak je umiestnený mimo aktívnej zóny čítačky. Transpondér je aktívny len vtedy, ak je v aktívnej zóne čítačky. Energia potrebná na napájanie transpondéra je dodávaná transpondéru cez bezkontaktnú cievku spolu so synchronizačnými pulzmi a dátami [14].

Pasívny RFID systém pracuje nasledujúcim spôsobom: čítačka vyšle rádiové vlny a tým nabije cievku v transpondéri. Akonáhle energia dosiahne prahovú hodnotu a nabije kondenzátor, súčiastky v transpondéri začnú pracovať a rádiový signál z čítačky je modulovaný identifikačnými dátami vo vnútri transpondéra a odrazený späť smerom k čítačke. Odrazené signály sa demodulujú v čítačke a tým sa získajú identifikačné informácie o transpondéri. Odras signálu je ilustrovaný na obrázku č. 4.2. Energia prijatá anténou transpondéra cez indukčnú cievku je taktiež závislá na vzájomnej vzdialenosti vo voľnom priestore. Kapacitná indukcia je využívaná len pre veľmi krátke vzdialenosti v rozmedzí 0.1 cm až 1 cm [22].

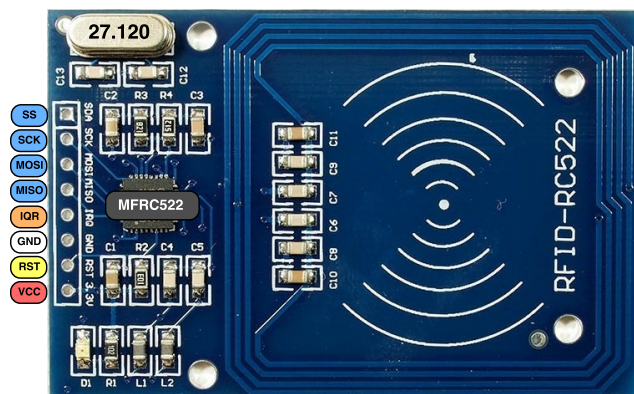


Obr. 4.2: Pasívne RFID, ilustrácia odrazu signálu medzi čítačkou a transpondérom [22].

4.2 MFRC522 modul

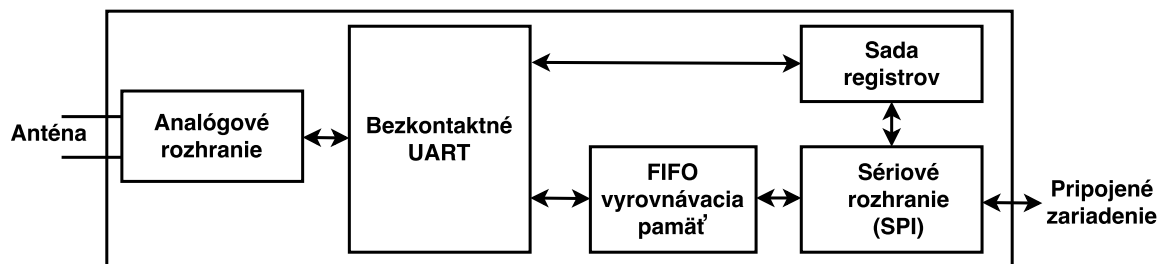
MFRC522 je bezkontaktný modul obsahujúci rovnomenný mikroprocesor MFRC522 vyrobený firmou NXP. Modul obsahuje komunikačné sériové rozhranie SPI. Je vybavený kryštálovým oscilátorom s frekvenciou 27,12 MHz. Vstupné napätie je podporované v rozsahu 2.5 V až 3.3 V. Spotreba modulu sa pritom pohybuje medzi 13 mA až 26 mA. MFRC522 podporuje všetky varianty protokolu MIFARE. Bezdrôtová komunikácia prebieha na frekvenčnom pásme 13.53 MHz. MFRC522 je schopný riadiť čítanie aj zápis. Prijímací modul okrem demodulácie a dekodovania signálov transpondéra umožňuje taktiež detekciu chýb pomocou parity. Modul pri bezkontaktnnej komunikácii využíva vyššie prenosové rýchlosti, až 848 kBd, v oboch smeroch. SPI rozhraním komunikuje rýchlosťou 10 Mbit za sekundu. V MFRC522 je použitých 8 x 64 bit vyrovnávacej pamäte typu FIFO.

Mikroprocesor MFRC522 pozostáva z viacerých komponentov. Analógové rozhranie spracováva demoduláciu analógových signálov. Bezkontaktné UART riadi požiadavky komunikačného protokolu a zabezpečuje tak spoluprácu s pripojeným hostiteľským systémom. Vyrovnávacia pamäť FIFO zaisťuje rýchle a plynulé dátové prenosy medzi vstupno výstup-



Obr. 4.3: RFID čítačka s mikrokontrolérom MFRC522 [4].

ným rozhraním, bezkontaktným UART a naopak. Mikroprocesor má možnosť implementácie rôznych rozhraní pre zvýšenie flexibility modulu. Schematický blokový diagram na obrázku 4.4 zobrazuje vzájomné prepojenie jednotlivých komponentov [18].



Obr. 4.4: Zjednodušená bloková schéma MFRC522 [18].

Adresa a dáta sú obojsmerne sériovo prenášané cez rozhranie I2C do pripojeného mikrokontroleru. Jednotka komparátoru porovnáva stav vstupného napätia a pri detekcii výpadku napájania automaticky prepne na záložné napájanie z batérie. Odchýlka udávaná výrobcom je ± 2 minúty za rok v rozmedzí $-40\text{ }^{\circ}\text{C}$ až $+85\text{ }^{\circ}\text{C}$. Obsah registra pre uloženie času a kalendára je uložený v BCD (binary-coded deciaml) formáte [15].

Kapitola 5

Možnosti užívateľského rozhrania

Táto kapitola sa venuje možným alternatívam vytvorenia užívateľského rozhrania vstavaného systému. V kapitole sú vysvetlené možnosti aplikovania frameworku pri tvorbe grafického užívateľského rozhrania. Popisuje základné možnosti frameworku a jeho elementov, ktoré je možné využiť pri návrhu.

5.1 Alternatívy užívateľského rozhrania

Prvou možnosťou je využitie hardvérového zobrazovacieho zariadenia. Na trhu sú dostupné mnohé moduly s integrovaným RGB maticovým displejom, ktoré by bolo možné použiť pri tvorbe užívateľského rozhrania¹. Tieto moduly sú však veľmi finančne nákladné, čo by zvyšovalo výrobné náklady spojené s tvorbou prototypu. Prototyp by okrem samotného maticového displeja musel taktiež obsahovať aj ovládacie tlačidlá. Zároveň by bol od užívateľa vyžadovaný priamy kontakt pri snahe o informatívne zobrazenie spotreby. Prototyp by tak neposkytoval všetky výhody spojené s ovládaním a správou spotreby vody.

Ďalšou možnosťou tvorby užívateľského rozhrania je vytvorenie aplikácie, ktorá by umožňovala správcovi komunikovať s prototypom po sieti a prenášať údaje o spotrebe. Nedostatkom tohto riešenia je potreba inštalácie aplikácie ešte pred samotnou komunikáciou s prototypom. Aplikácia by bola neprenosná v rámci mobilných platforiem a vyžadovala by tak viacero nezávislých implementácií. Správca systému je taktiež obmedzený používaním iba mobilného telefónu s inštalovanou aplikáciou.

Tretou najefektívnejšou možnosťou je implementovať grafické užívateľské rozhranie v podobe webovej stránky. Stránka by bola optimalizovaná práve pre mobilné zariadenie a tým by sa priblížila použitiu mobilnej aplikácie. Pozitívnym aspektom však zostáva nezávislosť na mobilnej platforme zariadenia a odstraňuje tak nedostatok spojený s doprednou inštaláciou. Nakoľko čip ESP8266 obsahuje integrovanú flash pamäť, je možné uložiť grafické užívateľské rozhranie priamo v prototypu a poskytnúť ho správcovi na vyžiadanie. Tvorbu webového rozhrania je možné optimalizovať využitím dostupného frameworku a jeho predpripravených komponentov.

¹Maticový RGB displej je dostupný na stránke <https://www.displaymodule.com/products/dm-oledc15-622>.

5.2 Framework7

Framework7 je zdarma volne dostupný mobilný HTML framework určený na vývoj hybridných mobilných alebo webových aplikácií s prirodzeným dizajnom Androidu alebo iOS. Hlavným prínosom Frameworku7 je tiež možnosť rýchleho vytvárania aplikácií s HTML, CSS a JavaScriptom. Framework7 nie je dizajnovo kompatibilný so všetkými platformami. Je zameraný iba na dizajn iOS a Google Material. Framework7 bol navrhnutý podľa oficiálneho dizajnu spoločnosti Google, tým prináša rovnaké vlastnosti rozhraní od vizuálnej stránky až po interakciu s prvkami vrátane efektov.

Framework7 ponúka širokú paletu elementov a widgetov ako napríklad:

- vyskakovacie okná
- prihlasovacie obrazovky
- zobrazenie zoznamov
- bočné panely (vľavo/ vpravo s možnosťou prechodov)
- indikátory aktivity
- formulárové prvky
- prehľad textových správ
- záložky
- tlačidlá

a mnoho ďalších. Veľkú mieru prispôbitelnosti umožňujú taktiež témy a farebné schémy. Tvorcovi aplikácie tým ponúka množstvo prvkov ako môže realizovať grafické rozhranie s využitím ľubovoľných prvkov, ktoré animáciami a dotykovou interakciou charakterizujú danú platformu. Výhodou je, že pri vytváraní aplikácie nie je potrebné písať obsah v JavaSkripte. Animácie využívajú hardvérovo akcelerované CSS prechody s cieľom dosiahnuť čo najlepší vizuálny výsledok [6].

5.2.1 Popis základných elementov

Framework7 prichádza s prihlasovacou obrazovkou pripravenou k použitiu. Je možné ju použiť ako vyskakovacie okno alebo ako samostatnú stranu. Aplikovanie prihlasovacej obrazovky vyžaduje použitie triedy "login-screen". V mnohých prípadoch je potrebné, aby sa prihlasovacia obrazovka zobrazila ako prvá pri otvorení užívateľského rozhrania. Pre dosiahnutie tohto efektu je potrebné pridať triedu "modal-in" do vrstvy prihlasovacej obrazovky.

Užívateľské rozhranie môže byť rozdelené na časti pomocou navigačnej lišty nachádzajúcej sa v hornej strane obrazovky. Je to pevná plocha, ktorá obsahuje názov stránky a navigačné prvky. Pozostáva z troch hlavných častí: ľavej, strednej a pravej. Každá časť môže obsahovať ľubovoľný HTML obsah. Tieto časti je možné využiť ako odkazy. Môžu zároveň obsahovať ľubovoľnú ikonu.

Možnosti Frameworku7 rozširuje zobrazenie posúvacích zoznamov. Tie umožňujú prehľadávanie zoznamu prvkov, pričom je možné definovať akciu pri kliknutí. Rozšírením na posuvný zoznam prináša definíciu novej akcie pri posunutí položky listu na stranu. Framework7 podporuje aj často používané funkcie bez potreby práce s Javascriptom. V prípade potreby zmazania položky zoznamu, je potrebné pridať triedu "swipeout-delete".

Vyhľadávací panel umožňuje užívateľovi vyhľadávať v posúvacom zozname prvkov. Najčastejšie býva umiestnený vo vnútri strany tesne nad alebo pod obsahom. Dôležité je vykonať aj inicializáciu vyhľadávacieho panela pomocou JavaScriptu.

Kalendár je dotykovo optimalizovaný komponent, ktorý poskytuje jednoduchý spôsob spracovania dát. Je možné ho využiť ako vložený komponent alebo ako prekrývaci element. Parametre kalendára umožňujú kalendár optimalizovať pre výber jednej hodnoty, viacerých hodnôt alebo pre výber celého rozsahu. Nastavenie kalendára je potrebné modifikovať v JavaScripte.

Akčný list je vyskakovacie okno prezentujúce užívateľovi súbor alternatív pre ďalší postup v danej úlohe. Rovnako môže byť použitý pre vyzvanie užívateľa na potvrdenie potenciálne nebezpečnej akcie. Akčný list obsahuje titulok a jedno alebo viac tlačidiel, z ktorých každé zodpovedá jednej akcii. Tlačidlá môžu byť organizované do skupín [6].

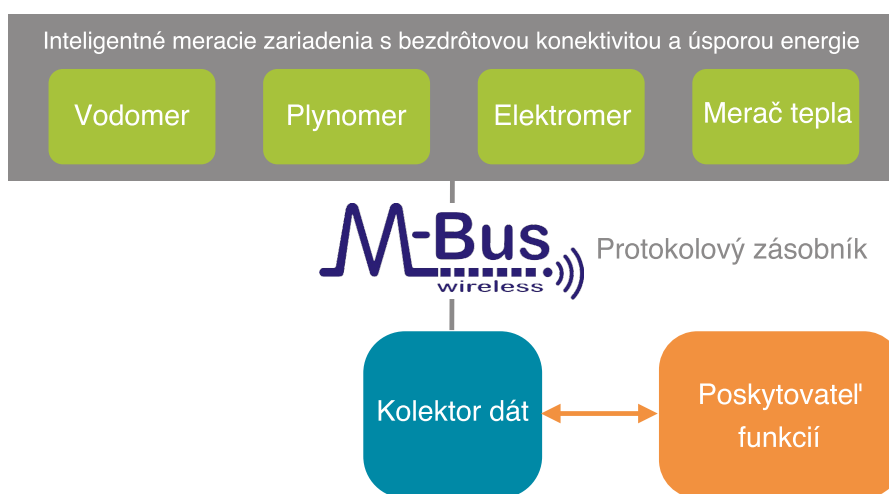
Kapitola 6

Meranie spotreby vody a technológia Wireless M-Bus

Termín internet vecí získal obrovskú popularitu pri rozvoji bezdrôtových senzorov ako sú inteligentné meradlá, domáce automatizačné zariadenia a nositeľná elektronika. V tomto prípade ide o bezdrôtové pripojenie pre inteligentné meracie systémy. Jeden z najužitočnejších bezdrôtových protokolov pre inteligentné meranie, ktorý sa objavil v posledných rokoch, je Wireless M-Bus. Je široko využívaný v meracích zariadeniach v celej Európe [17].

6.1 Zameranie

Štandard WM-Bus (Wireless M-Bus) je špecifikácia rádiových komunikačného spojenia medzi meracím zariadením, ako je napríklad vodomer, plynomer, elektromer alebo jednotka merania tepla a zariadením na zber dát. Stáva sa široko akceptovaný v Európe pre inteligentné meracie aplikácie. Je definovaný európskou normou EN13757-4 z roku 2013 pre bezdrôtové meradlá v pásme SRD s typicky limitovaným výkonom 25-100 mW. Protokol Wireless M-Bus bol pôvodne navrhnutý pracovať iba v 868 MHz pásme, pričom jeho snahou je uľahčiť odpočet stavu meradla. Zvolené pásmo je výborným kompromisom medzi dosahom rádiových vln a veľkosťou antény [7].



Obr. 6.1: Zameranie protokolu Wireless M-Bus [17].

6.2 Režimy činnosti

Wireless M-Bus protokol špecifikuje komunikáciu medzi meracím zariadením a inými systémovými komponentami určenými k zberu dát. Protokol špecifikuje základné komunikačné módy viď tabuľka č. 6.1, pričom každý z nich obsahuje dodatočné sub-módy.

| Mód | Frekvencia (MHz) | Popis |
|----------------------|------------------|--|
| S (Stacionárny) | 868 | Merače odosielajú údaje párkrát za deň |
| T (Časté vysielanie) | 868 | Merače odosielajú údaje viackrát za deň |
| C (Kompaktný) | 868 | Vyššia verzia prenosu dát v režime T |
| N (Úzkopásmový) | 169 | Dlhý dosah, systém úzkeho pásma |
| R (Časté prijímanie) | 868 | Zariadenie číta viac meradiel na rôznych frekvenčných kanáloch |
| F (Časté Tx a RX) | 433 | Častá obojsmerná komunikácia |

Tabuľka 6.1: Tabuľka zobrazujúca frekvencie a módy protokolu Wireless M-Bus [17].

- **Režim S** - Stacionárny režim bol definovaný pre jednosmernú alebo obojsmernú komunikáciu medzi stacionárnym alebo mobilným zariadením.
- **Režim T** - V režime častého vysielania meracie zariadenie vysiela veľmi krátku správu (najčastejšie 2 - 5 ms) v určitom intervale (najčastejšie každých pár sekúnd), čím umožňuje vyčítanie dát pri prejazde okolo vysielača.
- **Režim R** - Režim častého prijímania dát, kedy meracie zariadenie v pravidelnom intervale kontroluje príjem budúcej správy od mobilnej rádiostanice. Po obdržaní správy sa zariadenie prepne do komunikačného režimu. V tomto režime je možné vyčítať dáta z viacerých meracích zariadení komunikujúcich prostredníctvom rôznych kanálov [21].

Bezdrôtová technológia M-Bus je mierne odlišná od tradičného ISO/OSI modelu a má menej požiadaviek na jednotlivé vrstvy. Tri hlavné vrstvy sú aplikačná vrstva, vrstva dátového spojenia a fyzická vrstva [17].

Prenášaný paket pozostáva z preambulovej sekvencie a synchronizačného slova, prvého bloku obsahujúceho 12 bajtov, druhého bloku s obvyklou dĺžkou 18 bajtov, ktorá však môže byť voliteľná, a z postambulovej sekvencie [19]. Paket je možné šifrovať najčastejšie pomocou AES-128. V prípade, ak sa vykonáva bezdrôtové spojenie len medzi dvomi Wireless M-Bus modemami, je povolený zjednodušený režim bez zasielania adresy a pridružených informácií o meracej jednotke. To výrazne skraca prenášaný paket [21].

6.3 Kódovanie prenášaných dát

Wireless M-Bus protokol využíva dva druhy kódovania prenášaných dát. Pre režim T v smere od zariadenia k meradlu a pre režimy S, R sa používa kódovanie Manchester. Bity s logickou hodnotou "0" sú kódované sekvenciou "10". Bity s logickou hodnotou "1" sú kódované

na postupnosť “01”. Pre režim T v smere od meradla k ostatným zariadeniam sa využíva kódovanie 3 zo 6, ktoré zároveň poskytuje vyššiu efektivitu ako kódovanie Manchester. V kódovaní 3 zo 6 je každej časti priradený unikátny kód. Každý 4-bitový kúsok dát je kódovaný ako 6-bitové slovo, ktoré musí mať rovnaký počet núl a jednotiek a musí obsahovať minimálne dva prechody [21].

6.4 Podpora na strane výrobcov

V súčasnosti je veľká podpora certifikovaných WM-Bus modulov s integrovaným WM-Bus softvérom aj na strane výrobcov meracích zariadení. Meracie zariadenia podporujúce WM-Bus protokol sú stále čoraz ľahšie dostupné na súčasnom trhu. Nedávno boli pridané do špecifikácie dve nové pásma s kmitočtom 169 MHz a 433 MHz. Vďaka kratšej vlnovej dĺžke tak poskytujú spojenie s oveľa dlhším dosahom.

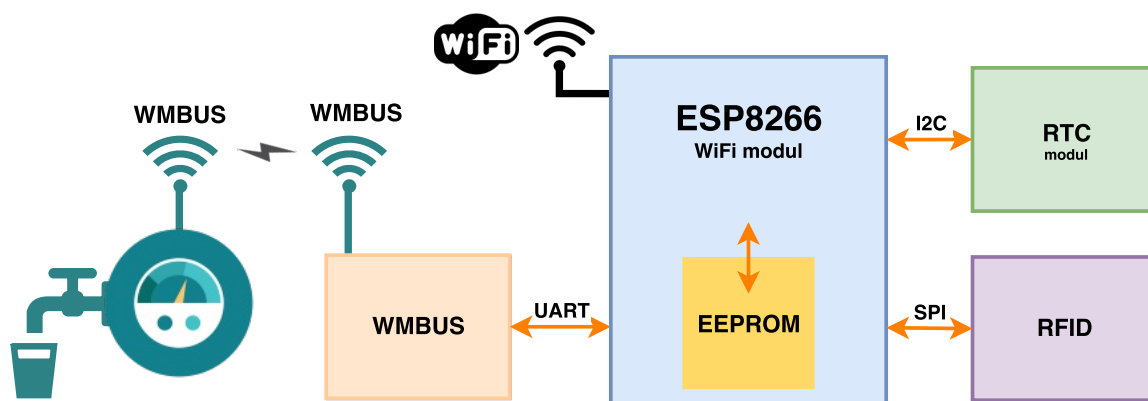
Pre nadviazanie spojenia a komunikáciu sa najčastejšie využívajú dvojčipové riešenia. Dvojčipové riešenie pozostáva z jedného čipu MCU s nízkou spotrebou a rádiových zariadení ako je napríklad CC110x, CC112x alebo CC120x pre obojsmernú komunikáciu. Existuje však aj jednočipové SoC riešenie CC430 vyrábané firmou Texas Instruments, ktoré integruje rádiové jadro CC1101 a 16 bitové MSP430 MCU [7].

Kapitola 7

Návrh vstavaného systému

Kapitola návrh vstavaného systému popisuje jednotlivé časti, z ktorých pozostáva výsledný prototyp a spôsob ich zostavenia do fungujúceho celku. Ďalej popisuje spôsob a možnosti vytvorenia softvérovej výbavy vstavaného systému. Samotný návrh pozostáva z dvoch častí - z hardvérovej časti a softvérovej časti.

- **Hardvérová časť** sa venuje výberu vhodných komponentov, návrhu vzájomného zapojenia, spôsobu komunikácie a využitia jednotlivých komunikačných protokolov v rámci predávania dát medzi zvolenými modulmi.
- **Softvérová časť** sa delí na dva nezávislé celky. Prvá oblasť sa zaoberá najmä návrhom grafického užívateľského rozhrania a jeho distribúcie prostredníctvom webového serveru a siete Wi-Fi správcovi systému priamo do jeho mobilného telefónu. Druhá oblasť sa venuje návrhu a spôsobu uloženia dátových štruktúr obsahujúcich informácie o evidencii a spotrebe vody. Taktiež sem patrí návrh softvéru pre mikrokontrolér, ktorý vykonáva samotnú komunikáciu medzi jednotlivými modulmi a udržiava vstavaný systém aktívny.



Obr. 7.1: Blokový diagram vstavaného systému a jeho komponentov.

7.1 Špecifikácia navrhovaného prototypu

Navrhnutý prototyp musí spĺňať základné požiadavky. Hlavnou požiadavkou je komunikácia s vodomermom a ukladanie informácií o spotrebe vody. Systém zároveň umožňuje evidenciu užívateľov s uložením dát spotreby vody pre každého užívateľa nezávisle. Užívateľom musí poskytovať jednoduchú identifikáciu pomocou bezkontaktného prístupového terminálu RFID a prístupovej karty.

Systém je prístupný a správcovi dáva možnosť interakcie pomocou siete Wi-Fi a jeho mobilného zariadenia. Aplikácia systému musí pracovať priamo v čipe ESP8266. Správca systému má možnosť prehľadného zobrazenia spotreby všetkých užívateľov. Systém správcovi poskytuje základnú sadu nástrojov a zobrazenie detailného pohľadu na konkrétne dáta spotreby vody jedného užívateľa. Systém musí byť schopný dlhodobej prevádzky a musí byť odolný voči výpadku napájania alebo strate bezdrôtového spojenia.

7.2 Návrh hardvéru vstavaného systému

Modul NodeMCU osadený čipom ESP8266 je vhodnou platformou, nakoľko ako SoC poskytuje pripojenie Wi-Fi a všetky komunikačné prvky sú vložené v rámci jedného čipu. Pri tvorbe vstavaného systému bude slúžiť ako centrálny kľúčový prvok a bude vykonávať všetky funkcie systému, od komunikácie s vodomermom po zasielanie dát o spotrebe. V režii modulu bude taktiež uchovávanie informácií a údajov o spotrebe jednotlivých užívateľov v internej flash pamäti a operácie pracujúce s týmito dátami.

Nakoľko čip ESP8266 disponuje pripojením Wi-Fi, nie je potrebné vybavovať prototyp akoukoľvek zobrazovacou technológiou v podobe LCD displeja pripojeného k prototypu, na ktorom by sa zobrazovala spotreba. Zobrazovanie hodnôt a údajov o spotrebe je realizované práve pomocou Wi-Fi, ktorým čip ESP8266 disponuje a pripojením mobilného zariadenia.

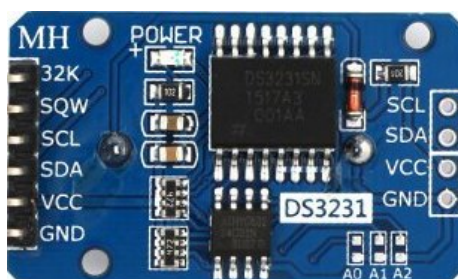
7.2.1 Pripojenie RFID MFRC522 modulu k prototypu

Potreba identifikácie užívateľov vyžaduje pripojenie RFID MFRC522 modulu, ktorý zabezpečuje čítanie bezkontaktných čipových kariet. Každý užívateľ systému bude disponovať jednou bezkontaktnou kartou. Kartou sa bude identifikovať v rámci vstavaného systému. Modul RFID disponuje sériovým komunikačným protokolom SPI (vysvetleným v kapitole 3.5). Ten umožňuje jeho pripojenie k NodeMCU modulu a tým aj k čipu ESP8266. Komunikačné vodiče MISO a MOSI MFRC522 modulu sú pripojené k pinom D6 a D7 modulu NodeMCU vid' obrázok č. 7.3. Nevyhnutné je taktiež pripojiť vodič CLK. Ten zabezpečuje prenos a synchronizáciu hodinového signálu. Komunikácia prebieha prostredníctvom pinu D5. Vodič CS na bezkontaktnom RFID module označený ako SDA, je pripojený k pinu D8. Napájacie napätie 3,3 V je dodávané z modulu NodeMCU. Napájanie vyžaduje pripojenie vodiča s nulovým potenciálom GND.

Priložením identifikačnej karty sa cez rozhranie SPI prenesú 4 bajty, ktoré budú programovo spracované a umožnia jednoznačnú identifikáciu daného užívateľa. Bajty je nutné programovo spracovať a vytvoriť z nich 32-bitovú unikátnu číselnú hodnotu typu unsigned int.

7.2.2 RTC modul

Modul reálneho času s čipom DS3231 je ľahko dostupný, nízko nákladový a extrémne presný modul reálneho času vid' obrázok č. 7.2. Obsahuje integrovaný kryštálový oscilátor TCXO s teplotnou korekciou. Zariadenie disponuje vstupom pre batériu, ktorá udržuje hodiny aktívne aj keď je prerušené hlavné napájanie. Integrácia kryštálového oscilátora zvyšuje dlhodobú presnosť. RTC modul udržuje informácie o sekundách, minútach, hodinách, dni, dátume, mesiaci a roku. Dátum sa na konci mesiaca automaticky nastaví na ďalší mesiac, pričom RTC modul berie do úvahy mesiace, ktoré majú menej ako 31 dní a taktiež zahŕňa korekciu priestupného roku. Hodiny môžu pracovať buď v 24 alebo 12 hodinovom cykle, pričom ponúkajú indikáciu AM alebo PM. Modul umožňuje nastavenie dvoch denných alarmov.



Obr. 7.2: Modul DS3231 AT24C32 obsahujúci IIC sériové rozhranie [8].

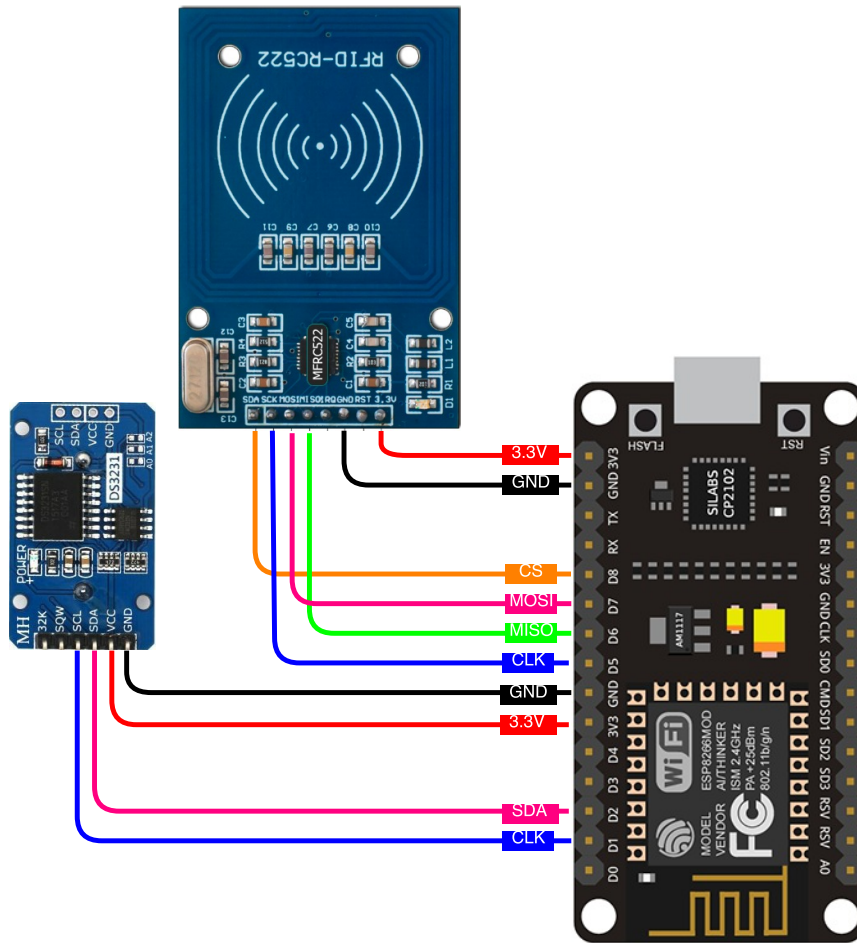
7.2.3 Pripojenie RTC DS3231 modulu k prototypu

Čip ESP8266 síce obsahuje integrovaný generátor hodín, no ten je z hľadiska merania reálneho času nepresný a spôsobuje odchýlku až 5 minút za mesiac (vid' str. 8 v dokumente [12]), čo je pri dlhodobej prevádzke vstavaného systému neprípustné. Pri systéme evidujúcom spotrebu vody je dôležité zaistiť presný čas a tým aj správne priradenie spotreby ku konkrétnemu dňu uloženému v pamäti. Preto bol zvolený modul reálneho času DS3231 poskytujúci vysokú presnosť. Ten je k čipu pripojený rozhraním I2C (vysvetleným v kapitole 3.6) a poskytuje mu presnú informáciu o aktuálnom dni.

Hodinový signál CLK v module reálneho času označený ako SCL, je pripojený k pinu D1 modulu NodeMCU. Pin SDA v module DS3231 vyhradený pre sériovú dátovú linku je pripojený vodičom k pinu D2 modulu NodeMCU vid' obrázok č. 7.3. Tým čip ESP8266 získa možnosť dáta o spotrebe vody priradiť k aktuálnemu dňu a prihlásenému užívateľovi a uložiť ich do internej flash pamäte. Modul tak zabezpečuje dlhodobú presnosť a synchronizáciu s reálnym časom, nakoľko sa predpokladá dlhodobý chod systému.

7.2.4 Spojenie prototypu s bezdrôtovým vodomermom

Údaje od vodomera modul ESP8266 získava cez sériové rozhranie UART (vysvetlené v kapitole 3.7). Pripojenie je realizované prostredníctvom pinov označených ako RX a GND. Tým je pripojený k rádiový frekvenčnému zariadeniu ako je napríklad CC1101 podporujúci bezdrôtovú komunikáciu protokolom W-MBus. Vodomer údaje zasiela automaticky v pravidelných časových intervaloch. Preto je postačujúce využitie pinu RX pre príjem sériových dát o spotrebe. Dáta budú prijímané cez sériové rozhranie po znakoch v číselnej podobe a ukončené znakom konca riadku.



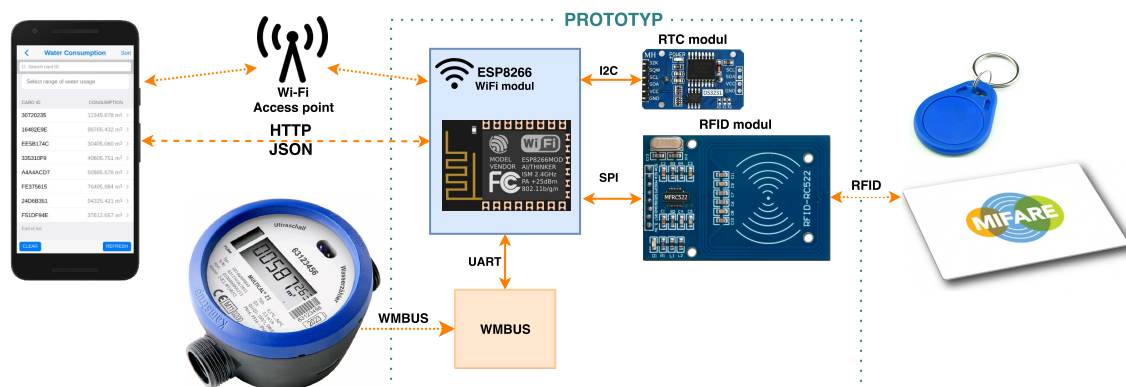
Obr. 7.3: Schéma pripojenia modulov RTC a RFID k NodeMCU s čipom ESP8266.

7.2.5 Pripojenie čipu ESP8266 k sieti Wi-Fi

Modul ESP8266 musí byť nakonfigurovaný v režime "device". To umožní vstavanému systému pripojiť sa do lokálnej siete prostredníctvom dostupného Wi-Fi prístupového bodu. Po pripojení k dostupnému prístupovému bodu s využitím HTTP protokolu na aplikačnej vrstve, modul umožňuje zdieľanie obsahu uloženého vo flash pamäti po lokálnej sieti vid obrázok č. 7.4. Obsahom je predovšetkým grafické užívateľské rozhranie a dáta o spotrebe užívateľov. Modul komunikuje s pomocou IPv4 protokolu na sieťovej vrstve a umožňuje šifrovanie WPA/WPA2. Tým sú zabezpečené do istej miery aj dáta o spotrebe prenášané do mobilného zariadenia správcu vstavaného systému.

7.3 Návrh užívateľského rozhrania a dátových štruktúr

Softvér zaisťuje v module všetku komunikáciu s okolitými modulmi, preto je pri jeho návrhu kladený dôraz na spoľahlivosť aj pri dlhodobej prevádzke. Softvér umožňuje pripojenie k predvolenej Wi-Fi sieti, distribúciu súborov cez webový server, správu dát o spotrebe vody, komunikáciu s čítačkou bezkontaktných kariet, udržiavanie aktuálnej informácie o dátume a samotnú komunikáciu s vodomerom.



Obr. 7.4: Navrhnutý prototyp a komunikácia s okolím.

Vývoj softvéru je možné uskutočniť vo vývojovom prostredí Arduino IDE. Vývojové prostredie je určené najmä pre vývoj softvéru pre čipy z rodiny Arduino. Prostredie umožňuje importovanie ďalších rozširujúcich knižníc a tým umožňuje vývoj softvéru pre čip ESP8266 a komunikáciu s okolitými komponentami ako sú MFRC522 modul alebo RTC modul s čipom DS3231. Knižnice sú dostupné na GitHub. Knižnice umožňujú modulu komunikáciu cez Wi-Fi využitím protokolov TCP a UDP na transportnej vrstve ISO OSI modelu. Ďalej umožňujú nastavenie webového HTTP serveru a DNS serveru pracujúcich na aplikačnej vrstve ISO OSI modelu. Poskytujú možnosť pre vytvorenie súborového systému v internej flash pamäti, prácu s SD kartou, ale aj funkcie pre komunikáciu cez rozhranie SPI a I2C.

7.3.1 Návrh spôsobu uloženia dát o spotrebe vody

Pri práci s pamäťou umiestnenou v module NodeMCU je možné využiť rozdielne spôsoby prístupu. Jedným je možnosť zapisovať a čítať dáta priamo v pamäti pomocou operácie pre zápis "write" a čítanie "read". Ďalšou možnosťou je vytvoriť interný súborový systém priamo vo flash pamäti. Vytvorenie, správu, prístup k súborom a operácie so súborami zabezpečuje knižnica FS.h. Nakoľko čip taktiež musí mať uložené súbory potrebné pre vykreslenie webového rozhrania, vyhovujúca je druhá varianta s použitím interného súborového systému.

Požiadavka ukladať dáta o spotrebe a súbory webovej stránky vyžaduje vytvorenie súborového systému v internej pamäti flash. Súborový systém bude slúžiť najmä na uloženie grafického užívateľského rozhrania. Nakoľko pamäť má veľkosť 4 MB, viď kapitola 3.8, zvyšné miesto je k dispozícii pre uchovanie dát o spotrebe. Dáta o spotrebe jednotlivých užívateľov budú uložené v jednom binárnom súbore.

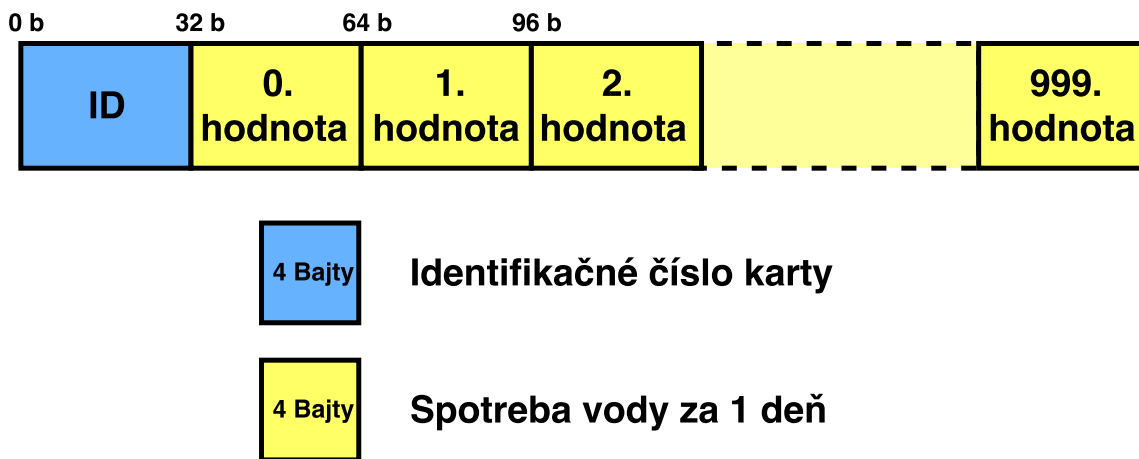
Pri návrhu vstavaného systému je zámer dosiahnuť čo najväčší počet evidovaných užívateľov spolu s čo najväčším evidenčným obdobím. Vzniká tým však rozpor s konštantnou veľkosťou dostupnej pamäte. Pamäť je v prípade potreby možné rozšíriť pridaním SD karty, čo by však malo za následok ďalšie zvýšenie výslednej ceny celého prototypu. Pre dosiahnutie čo najväčšieho evidenčného obdobia je dôležité správne zvoliť parametre, ktorými sú počet užívateľov, dĺžka evidovaného obdobia a najmenší rozlišovací interval evidovanej spotreby.

Správne zvolenie počtu evidovaných užívateľov v systéme závisí vždy od konkrétneho nasadenia v praxi. Predpoklad je, že systém nájde uplatnenie najmä v ubytovacích zariadeniach, menších firemných priestoroch alebo pri správe bytových jednotiek, kde by počet ľudí nemal presiahnuť hodnotu 100. Najmenší rozlišovací interval pre uložené dáta o spotrebe je

taktiež voliteľný. Pri zvolení evidencie po jednotlivých týždňoch sa výrazne predĺži možnosť evidovaného obdobia, ale zmenší sa interval pre zobrazenie spotreby. Naopak, pri evidencii dát o spotrebe za každú hodinu je možné presné zobrazenie spotreby vody v konkrétny čas, no výrazne sa tým skráti celková doba evidencie spotreby vody z dôvodu obmedzenej kapacity pamäte.

Preto najvhodnejším riešením je ukladať spotrebu vody vždy pre konkrétny deň. Dáta prijaté z vodomera budú priradené a ukladané v rámci jedného dňa, čo je však s ohľadom na dĺžku prevádzky celého systému primerane jemný časový úsek. Pre dosiahnutie čo najväčšieho množstva evidovaných kariet s čo najväčšou dĺžkou evidovaného obdobia, bude súbor rozdelený na jednotlivé sektory. Každý sektor bude vyčlenený pre evidenciu spotreby vody jednej identifikačnej karty a tým jednému užívateľovi v systéme. Veľkosť jedného sektoru bude závislá na počte evidovaných dní, počte evidovaných užívateľov a veľkosti samotných dát udávajúcich spotrebu.

Jednotlivé sektory obsahujú menšie bloky. Každý z týchto blokov má veľkosť v závislosti na uložených dátach. Prvý blok v rámci jedného sektoru v súbore je vyčlenený pre uloženie identifikačného čísla karty. Veľkosť 4 Bajtov pre identifikačný blok je zvolená z dôvodu veľkosti identifikačného čísla bezkontaktnéj RFID karty. Identifikačné číslo karty obsahuje 32 bitov. Ak identifikačný blok obsahuje nulovú hodnotu, celý sektor sa pokladá za prázdny aj v prípade, ak ďalšie bloky uchovávajúce spotrebu obsahujú nenulovú hodnotu. Zvyšný počet blokov v jednom sektore za identifikačným blokom je možné ľubovoľne zvoliť podľa veľkosti dostupnej pamäte a počtu užívateľov. Tieto bloky obsahujú číselný údaj o spotrebe vody jedného užívateľa za jeden deň. Bloky uchovávajúce spotrebu majú tiež veľkosť 4 Bajtov, čo vychádza z maximálnej hodnoty, ktorú môže dosiahnuť počítadlo na vodomeri. Počítadlo obsahuje 5 celých a 3 desatinné miesta a preto rozsah, ktorý je možný uložiť na 4 B, je postačujúci.



Obr. 7.5: Formát jedného sektoru binárneho súboru slúžiaceho na uloženie údajov užívateľa.

Index aktuálneho dňa v sektore je hodnota vypočítaná na základe informácií získaných z RTC DS3231 modulu. RTC DS3231 modul uchováva číselnú hodnotu počtu dní od 1.1.1970. Hodnota určujúca deň je vydelená operáciou modulu 1000 a výsledkom je konkrétny index bloku. Po zaplnení posledného bloku tým však dochádza k cyklickému prepisovaniu starých hodnôt začiatočných blokov. Operácia modulo zaisťuje kruhový prístup k jednotlivým blokom obsahujúcim hodnotu spotreby vody. To znamená, že ak index

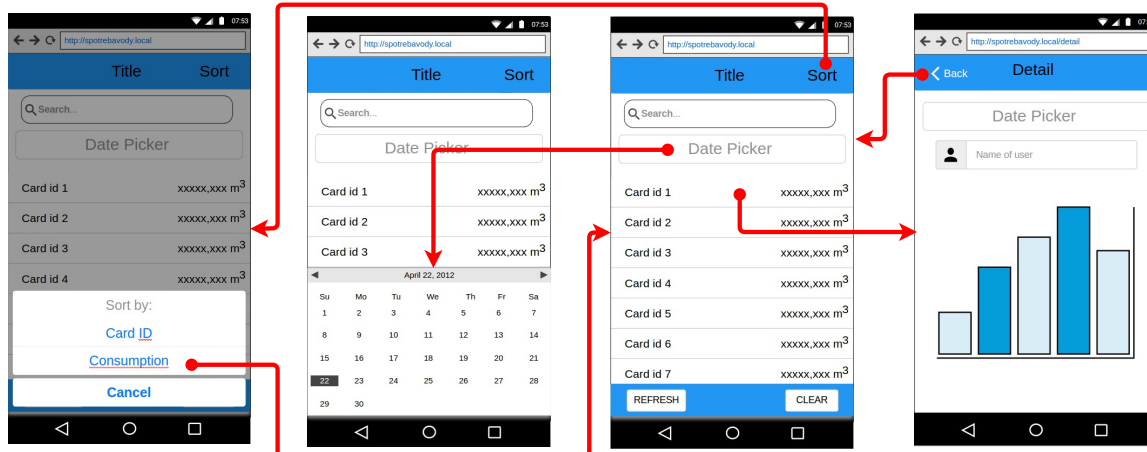
ukazuje na konkrétny blok v strede sektoru, bloky s menším indexom uchovávajú hodnotu predošlých dní až po prvý blok sektoru. Následne pokračujú od posledného bloku až po blok s indexom o jedno väčším ako aktuálny index.

Kruhovosť blokov však musia brať do úvahy taktiež podprogramy operujúce nad týmto zoznamom. Podprogramy budú navrhnuté tak, aby pracovali s lineárnou výpočtovou zložitostou a všetky informácie boli schopné spracovať pri jednom priechode binárnym súborom. Z dôvodu minimalizácie počtu zápisových cyklov, budú hodnoty spotreby aktívnej karty vždy načítané po priložení danej karty a aktualizované v dočasnej štruktúre **Record**.

7.3.2 Návrh grafického užívateľského rozhrania

Pri návrhu grafického užívateľského rozhrania je dôležité, aby len správca systému mal možnosť prístupu k týmto dátam. Užívateľské rozhranie tak musí obsahovať prihlasovaciu obrazovku, kde je potrebné prihlásenie správcu pred jeho interakciou so systémom. Rozhranie bude navrhnuté tak, aby sa po prihlásení zobrazila hlavná obrazovka. Na hlavnej obrazovke budú v zozname zobrazené všetky identifikačné karty, ktoré systém obsahuje, spolu so spotrebou vid' obrázok č. 7.6.

Hlavná obrazovka má poskytovať správcovi jednoduché možnosti zoradenia kariet, napríklad podľa spotreby alebo identifikačného čísla, či už vzostupne alebo zostupne. Pri zobrazení spotreby sa bude na hlavnej obrazovke nachádzať aj kalendár, v ktorom je možné zvoliť požadované obdobie, za ktoré sa má spotreba prepočítať. Zobrazovaná hodnota bude zobrazovať súčet hodnôt spotreby konkrétnych dní za zvolené obdobie. Grafické rozhranie má správcovi umožniť vyhľadať konkrétnu kartu podľa názvu vo vyhľadávacom poli. Správca po kliknutí do vyhľadávacieho poľa môže zadať identifikačné číslo karty, ktorá sa v zozname vyhľadá. Ak zoznam žiadnu kartu nenašiel, upozorní na to správcu systému.



Obr. 7.6: Wireframe grafického užívateľského rozhrania správcu systému.

Po kliknutí na kartu bude mať správca možnosť detailného náhľadu na spotrebu vody jedného užívateľa systému vid' obrázok č. 7.6. Detailný náhľad bude zobrazovať dáta o spotrebe v prehľadnom grafe, pričom bude zároveň obsahovať kalendár, v ktorom bude možné si zvoliť požadované obdobie. Po zvolení rozsahu dní v kalendári, sa graf automaticky aktualizuje. Graf sa musí automaticky prispôbiť zvolenému obdobiu a hodnotám spotreby. Súčasťou detailného náhľadu karty bude taktiež meno držiteľa identifikačnej karty. Celé

grafické užívateľské rozhranie bude prispôsobené a optimalizované pre zobrazenie na mobilnom zariadení, čo však z dôvodu orientácie obrazovky na výšku prináša isté obmedzenia, napríklad pri zobrazení rozsiahlych grafov.

Kapitola 8

Implementácia funkčného prototypu

Kapitola implementácia funkčného prototypu nadväzuje na kapitolu návrhu a popisuje implementáciu softvéru pre navrhnutý prototyp. Implementovaný softvér sa rozdeľuje do dvoch samostatných častí. Prvou je implementácia softvéru pre vstavaný systém. Druhou časťou je implementácia grafického užívateľského rozhrania. Ďalej sa zaoberá ich vzájomným prepojením do jedného kooperujúceho celku.

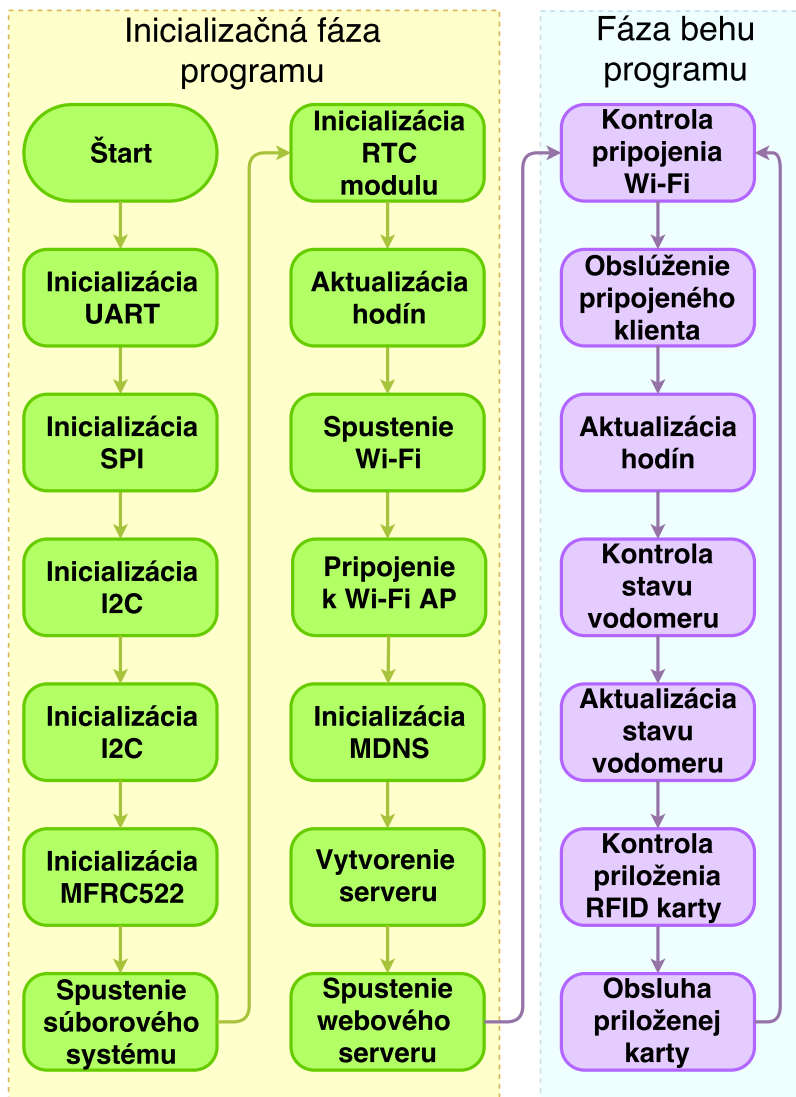
- **Softvér pre vstavaný systém** je časť venujúca sa programovaniu modulu ESP8266 a jeho podporných komponentov a spôsobu komunikácie s bezdrôtovým vodomerom i s grafickým užívateľským rozhraním. Zaoberá sa taktiež implementáciou ukladania dátových štruktúr, ktoré obsahujú údaje o evidencii a spotrebe vody v pamäti. Zároveň zahŕňa spôsob odosielania grafického užívateľského rozhrania po lokálnej sieti, nakoľko aj túto úlohu pokrýva SoC modul ESP8266.
- **Grafické užívateľské rozhranie** je implementované takmer nezávisle na vstavanom systéme. Pri jeho implementácii sú využívané rovnaké technológie ako pri tvorbe webových stránok. Pri tvorbe sú využité technológie HTML, CSS a JavaScriptu. Užívateľské rozhranie je tak dostupné v rámci lokálnej siete a tým nevyžaduje inštaláciu samostatnej aplikácie.

8.1 Implementácia softvéru pre vstavaný systém a modul ESP8266

Softvér pre čip ESP8266 je vyvíjaný v prostredí Arduino IDE doplnenom o knižnice poskytujúce základné funkcie spomenuté v kapitole 7. Všetky knižnice sú voľne dostupné na GitHubu¹. Vývojové prostredie poskytuje pri vývoji možnosť kompilácie zdrojových súborov a nahraťie novoimplementovaného firmvéru priamo do modulu ESP8266. Prostredie je určené primárne pre programovacie jazyky C a C++. Softvér pre modul je implementovaný v jazyku C++, vďaka čomu poskytuje veľkú kontrolu nad celým systémom. Ukladanie dát je implementované v jazyku C v samostatnom hlavičkovom súbore `file_operations.h`, ktorý umožňuje úplnú kontrolu nad spôsobom uloženia dát o evidencii spotreby vody.

¹Knižnice pre modul NodeMCU <https://github.com/esp8266/Arduino>

Softvér je rozdelený do dvoch častí. Prvá fáza je inicializačná a nachádza sa vo funkcii **Setup**. Funkcia **Setup** obsahuje časť programu, ktorá sa vykoná ihneď po spustení systému alebo po resete systému. Táto fáza sa vykoná len raz a slúži na inicializáciu celého systému, inicializáciu pripojených modulov, komunikačných rozhraní a inicializáciu programových premenných. Pri štarte systému sú najskôr inicializované komunikačné rozhrania UART, SPI, I2C a taktiež interný súborový systém a RTC modul. Následne sa čip pripojí k dostupnej Wi-Fi sieti, nastaví obsluhu pre všetky HTTP požiadavky a spustí webový server, čím sa ukončí inicializačná fáza viď obrázok č. 8.1. Po skončení inicializácie a ukončení funkcie **Setup** sa začne cyklicky vykonávať fáza behu softvéru.



Obr. 8.1: Znáznornenie logickej štruktúry implementovaného programu a jeho rozdelenia na inicializačnú fázu a fázu behu programu.

Funkcia **Loop** je vykonávaná periodicky v nekonečnej slučke počas celej doby činnosti systému. Pri vstavaných systémoch nie je periodické vykonávanie v nekonečnom cykle nežiadúce, nakoľko sa od systému nevyžaduje žiadna iná funkcionálna ako tá, na ktorú je systém vyvinutý a prispôbený. Vykonávanie funkcie **Loop** je skončené až resetom alebo

vypnutím celého systému. Vo vstavanom systéme sa vo funkcii Loop cyklicky vykonávajú a kontrolujú iba nevyhnutné veci zabezpečujúce správne fungovanie celého systému.

Vo fáze behu vo funkcii Loop softvér najskôr skontroluje, či je Wi-Fi pripojenie stále aktívne a nedošlo k jeho odpojeniu. V prípade, ak bolo pripojenie stratené, dôjde k opätovnému pokusu o pripojenie, čím sa zamedzí výpadku systému pri chvíľkovej nedostupnosti prístupového bodu napríklad z dôvodu straty napájania prístupového bodu. Ak je Wi-Fi pripojenie stále aktívne, skontrolujú a obslúžia sa prichádzajúce HTTP požiadavky. Obsluha je vykonávaná podprogramom, ktorý zachytí a spracuje prichádzajúcu požiadavku tak, že najskôr porovná či je schopný danú požiadavku obslúžiť. Následne otvorí daný súbor v lokálnom súborovom systéme a začne súbor grafického užívateľského rozhrania odosielať cez aktívne TCP spojenie. To je po prenesení zobrazené vo webovom prehliadači mobilného zariadenia. Rovnakým spôsobom sú prenášané aj súbory obsahujúce informáciu o spotrebe užívateľov.

V prípade, ak nedochádza k obsluhu žiadnej z predošlých udalostí, softvér si vyžiada od RTC modulu hodnotu aktuálneho dňa, pričom hodnota 0 zodpovedá dátumu 1. 1. 1970. Pri získavaní hodnoty dňa dôjde najskôr ku kontrole validity dátumu. Následne je z časovej známky získaný index aktuálneho dňa a ten je porovnaný s uloženou hodnotou dňa. V prípade, ak došlo k zmene dňa, je hodnota aktuálneho dňa aktualizovaná. Po aktualizácii časovej známky čip zahájí komunikáciu s vodomermom a zistí, či došlo k zmene počítadla na vodomeri. Komunikácia prebieha cez sériové rozhranie. V prípade zmeny stavu vodomera je uložená nová hodnota počítadla ako aktuálna. Vodomer odosiela údaje o aktuálnom stave počítadla v pravidelných intervaloch. Rozdiel medzi predošlou a súčasnou hodnotou sa pripočíta k aktuálne evidovanej karte na príslušný deň označený indexom aktuálneho dňa.

Na koniec cyklického vykonávania programu softvér overí, či nedošlo k priloženiu inej užívateľskej karty. Ak je priložená identifikačná karta, softvér prečíta 4 bajty reprezentujúce unikátne identifikačné číslo a porovná, či ide o inú kartu ako bola posledná priložená karta. Ak ide o inú kartu, softvér zapíše do Flash pamäte dáta v štruktúre `Record` o spotrebe vody načítané k predošlej karte. Vyhľadá novú kartu v uloženom binárnom súbore `archiv.bin`. V prípade evidencie záznamu s rovnakým identifikačným číslom, údaje o karte načíta do štruktúry `Record`. Tým, že softvér má načítanú vždy práve jednu aktívnu kartu, nedochádza k nadmernému zápisu pri pravidelnej aktualizácii spotreby vody do flash pamäte. A tým sa markantne zníži počet prepisov jednotlivých buniek, čo predĺži dobu uchovávaní dát v pamäti a životnosť celého vstavaného systému.

8.2 Implementácia dátových štruktúr slúžiacich na uchovanie evidenčných dát

Podprogramy pre uloženie údajov o spotrebe vody jednotlivých užívateľov sú implementované v hlavičkovom súbore `file_operations.h`. Súbor obsahuje funkcie potrebné pre prístup a zápis údajov do binárneho súboru uloženého v pamäti flash. Funkcie sú implementované v programovacom jazyku C. Knižnica umožňuje:

- vrátenie indexu zvolenej karty
- výpis všetkých uložených kariet
- výpis všetkých hodnôt zadanej karty do konzoly

- načítanie zadanej karty do štruktúry Record
- uloženie karty zo štruktúry Record do flash pamäte
- výpis všetkých hodnôt zadanej karty vo formáte JSON
- výpis spotreby všetkých kariet za zvolené obdobie vo formáte JSON

Funkcie operujú so zvyšnou dostupnou pamäťou v čipe ESP8266. Pri implementácii záznamov o užívateľoch bol počet stanovený na 100 sektorov v súbore. Ďalším zvoleným parametrom bol počet evidovaných dní jedného užívateľa. Hodnota bola stanovená na 1000 evidenčných blokov pre spotrebu vody a jeden evidenčný blok pre uloženie identifikátora bezkontaktnéj karty. Sektor s 1001 blokmi s veľkosťou 4 Bajty má veľkosť 4004 bajtov. Veľkosť celého binárneho súboru archivujúceho spotrebu je 400,4 kB, čo je približne 391 KiB.

Funkcie pracujú so súborom pomocou podprogramov `seek` a `readBytes`. Tie umožňujú posuv v súbore a čítanie súboru po bajtoch. Pri čítaní po bajtoch je potrebná konverzia na typ `unsigned int`.

Pri výpise spotreby všetkých kariet za určité obdobie je potrebné prejsť všetky sektory užívateľov v súbore a spočítať všetky bloky v zvolenom rozsahu. Rozsah je definovaný dvoma hodnotami. Hodnota `start` definuje poslednú položku v rozsahu, ktorá má najvyšší index. Hodnota `count` definuje veľkosť zvoleného rozsahu. Ak sa bloky nachádzajú uprostred sektora, funkcia vynechá všetky predchádzajúce bloky, sčíta zvolený rozsah a vynechaním zvyšných blokov sa presunie na začiatok ďalšieho sektora, kde sa postup opakuje. V prípade, ak sa rozsah nachádza zároveň na začiatku a na konci blokov, je potrebné vykonať súčet od nultého bloku po blok označený hodnotou `start`, preskočiť nepotrebné bloky a dokončiť súčet na posledných blokoch presahujúceho rozsahu. Výsledné hodnoty sú ukladané do súboru `items.json`, ktorý sa generuje zároveň pri vyhodnocovaní spotreby jednotlivých kariet.

Súbor ďalej obsahuje aj funkcie pre prácu so samotnou štruktúrou `Record` v operačnej pamäti. Funkcie umožňujú zapísať hodnotu na konkrétny index alebo pripočítať hodnotu spotreby vody k už uloženej hodnote na konkrétnom indexe. Snahou implementovaných funkcií je minimalizovať počet prepisov v pamäti. Dáta sú vždy aktualizované v operačnej pamäti. Zapisované do pamäte sú až keď je to nevyhnutné, čím sa zvyšuje jej životnosť.

8.3 Implementácia webového grafického užívateľského rozhrania

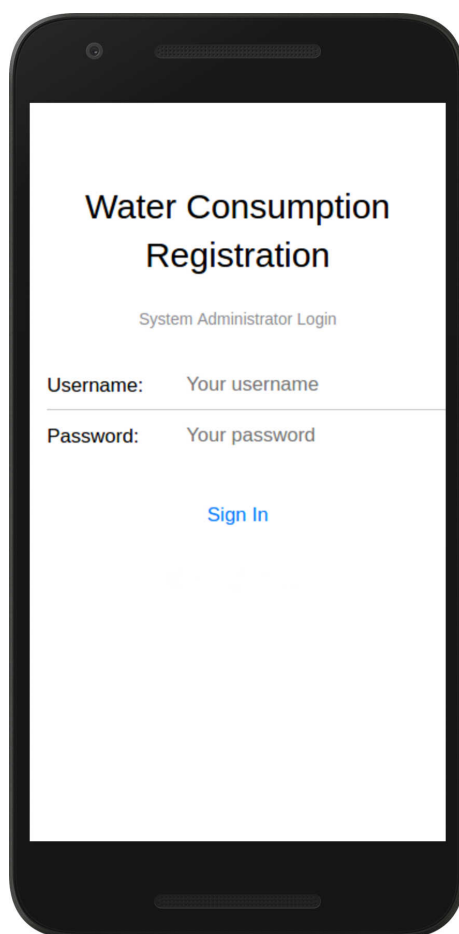
Grafické užívateľské rozhranie, ktoré správca využíva pri interakcii so vstavaným systémom, je implementované v podobe webovej stránky vytvorenej vo voľne dostupnom Frameworku7².

Framework7 sa špecializuje práve na mobilné zariadenia. Výhodou užívateľského rozhrania je, že je uložené priamo vo flash pamäti vstavaného systému. Nevyžaduje tak od užívateľa inštaláciu dodatočnej aplikácie, čím je nezávislá na platforme mobilného zariadenia. Použitie Frameworku7 vyžaduje prídanie súborov obsahujúce kaskádové štýly CSS a JavaScriptovú knižnicu. Súbory sú pomenované `fm7.css`, `fm7colors.css` a `fm7.js`. Pre zrýchlenie odosielania súborov po sieti boli zvolené minimalizované verzie s odstránenými

²Verzia Frameworku7 voľne dostupná na stránke <https://framework7.io/>

prebytočnými bielymi znakmi a komentármi. Grafické rozhranie je implementované v súboroch `index.html`, `item_info.html` a `my-app.js`. Súbor `index.html` obsahuje hlavnú obrazovku spolu s prihlasovacou obrazovkou. Súbor `item_info.html` obsahuje implementáciu zobrazenia detailného náhľadu jednej karty. V súbore `my-app.js` sa nachádzajú všetky podporné skripty zabezpečujúce interaktivitu celého užívateľského rozhrania.

Prihlasovacia obrazovka je pridaná do grafického rozhrania prostredníctvom vloženia "Login Screen Layoutu" do hlavného súboru `index.html`. Prihlasovacia obrazovka je zobrazená správcovi ihneď po načítaní webovej stránky. Pre vytvorenie prihlasovacej stránky bola použitá trieda "login-screen" vid' obrázok č. 8.2. Správca systému sa musí najskôr prihlásiť pomocou mena a hesla. Pre účely prototypu bolo zvolené užívateľské meno admin a heslo admin.



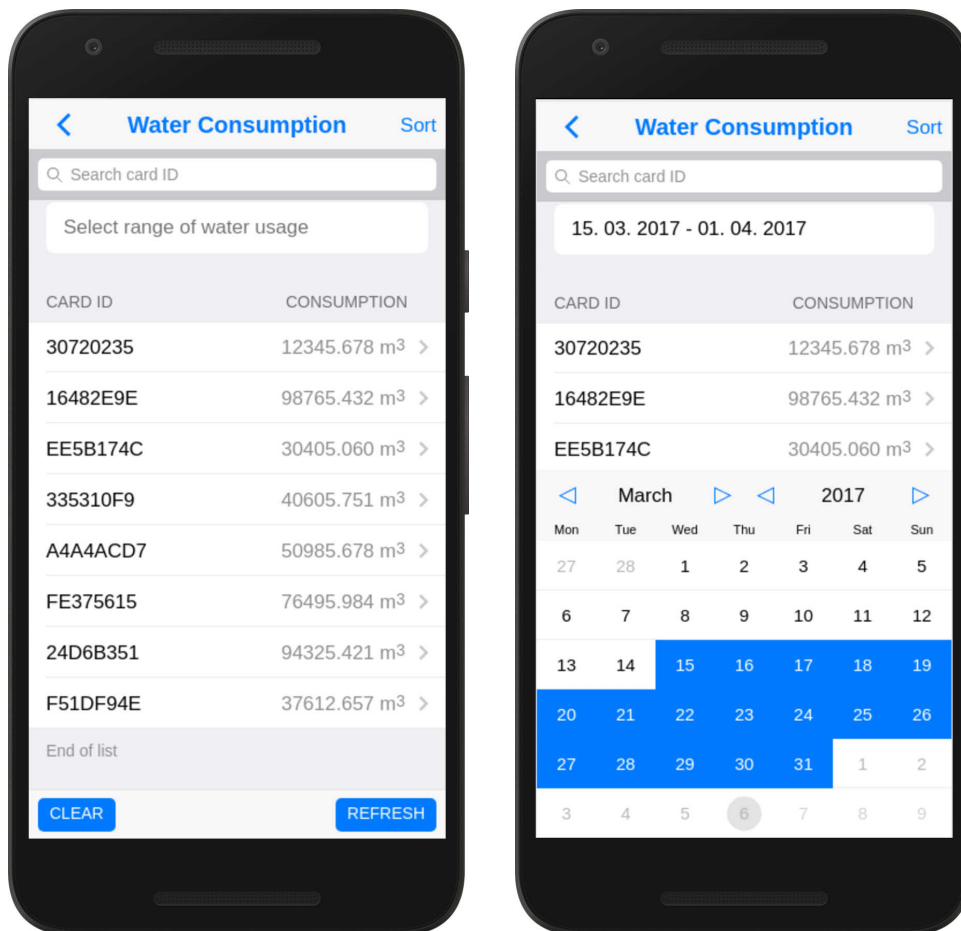
Obr. 8.2: Prihlasovacia obrazovka grafického užívateľského rozhrania.

Ústrednou časťou hlavnej obrazovky je zoznam všetkých kariet registrovaných v systéme. Pre implementáciu zoznamu bol zvolený "Swipeable List". Ten umožňuje prehľadné zobrazenie všetkých kariet spolu so spotrebou vid' obrázok č. 8.3. Potiahnutím je možné príslušnú kartu zmazať. Zoznam je generovaný pomocou JavaScriptu v závislosti na prijatom súbore `items.json`, ktorý obsahuje údaje o spotrebe všetkých kariet.

Súčasťou hlavnej obrazovky grafického rozhrania je tiež kalendár, v ktorom je možné zadať obdobie, za ktoré sa má spotreba prepočítať. Okrem samotného vytvorenia inštancie kalendára, je kalendár modifikovaný aj pre možnosť výberu rozsahu zvolených dní. Zároveň

je zablokovaná možnosť označiť dni v budúcnosti za aktuálnym dňom, ktoré ešte neobsahujú údaje o spotrebe vody.

Hlavná obrazovka ďalej obsahuje vyhľadávacie pole. Vyhľadávanie je implementované s pomocou "Search Bar Layoutu". Ten je umiestnený nad kalendárom a umožňuje vyhľadávanie naprieč celým zoznamom zobrazených kariet. O samotný algoritmus vyhľadávania sa stará Framework7. Vyhľadávaný je vždy akýkoľvek názov obsahujúci zadaný podreťazec znakov.

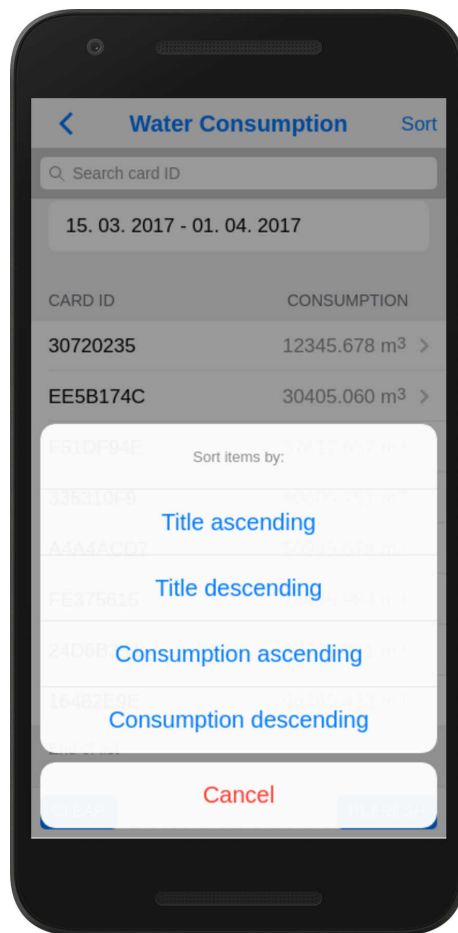


Obr. 8.3: Hlavná obrazovka grafického užívateľského rozhrania. Vľavo zoznam evidovaných kariet. Vpravo výber obdobia v kalendári.

Navigačná lišta, umiestnená na vrchu obrazovky, ponúka možnosť výberu zoradenia kariet. Karty je možné radiť podľa názvu alebo spotreby vzostupne a zostupne viď obrázok č. 8.4. Radenie je implementované v súbore `my-app.js`. Identifikačné čísla kariet sú z hexadecimálnej hodnoty prevedené na číslo a zoradené podľa veľkosti číselnej hodnoty nie abecedne. Po kliknutí v navigačnej lište na názov systému sa zobrazia základné informácie o systéme.

V dolnej nástrojovej lište hlavnej obrazovky sú umiestnené tlačidlá Refresh a Clear. Tlačidlá sú implementované pomocou triedy "Button". Uvedené tlačidlá slúžia na vynulovanie spotreby vody zobrazenej v zozname a na znovunačítanie dát o spotrebe.

Detailný náhľad je určený pre zobrazenie konkrétnej karty. Náhľad je implementovaný v súbore `item_info.html`. Hlavným prvkom detailného náhľadu je zobrazený graf. Graf



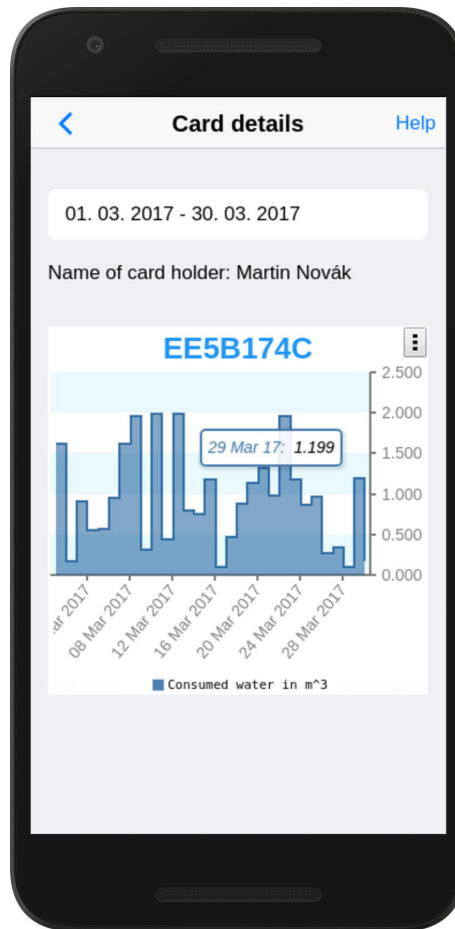
Obr. 8.4: Zobrazenie akčného zoznamu s nastavením usporiadania.

o spotrebe je implementovaný pomocou knižnice CanvasJS³. Knižnica CanvasJS sa špecializuje na tvorbu a vykresľovanie grafov.

Použitie knižnice CanvasJS si vyžaduje prídanie súboru `canvas.js`, ktorý zabezpečuje podporu pre prácu s grafmi a ich vykresľovanie. Graf je nakonfigurovaný pre automatický výber zobrazovanej mierky. Grafická knižnica podporuje tlač alebo uloženie aktuálneho grafu. Zobrazenie grafu je nakonfigurované v súbore `my-app.js`. Konfigurácia obsahuje nastavenia pre popis osí, nastavenia veľkosti intervalov a formátovanie dátumu. Pre túto aplikáciu bol zvolený typ grafu "StepArea". Hodnoty vykresľované v grafe sú uložené v súbore `values.json`. Súbor obsahuje záznam všetkých hodnôt spotreby vody pre zvolenú kartu. Implicitne je v grafe zobrazených posledných 10 dní evidovaného obdobia.

Detailný náhľad spotreby vody taktiež obsahuje inštanciu kalendára. Zvolením časového intervalu sú v súbore `values.json` vybrané správne hodnoty a tie sú následne vykreslené v grafe vid' obrázok č. 8.5. V detailnom náhlade je tiež možnosť zobrazit meno držiteľa karty. Mená užívateľov sú uložené v súbore s príponou `.info`, ktorý je pomenovaný názvom identifikačnej karty. Náhľad obsahuje v navigačnej lište inštrukcie pre prácu so systémom a možnosť návratu na hlavnú obrazovku.

³Knižnica CanvasJS voľne dostupná na stránke <http://canvasjs.com/>



Obr. 8.5: Detailný náhľad spotreby jedného užívateľa.

8.4 Princíp komunikácie webového rozhrania a navrhnutého prototypu

Komunikácia s mobilným zariadením prebieha pomocou HTTP protokolu. Čip ESP8266 na vyžiadanie začne odosielať webovú stránku vytvorenú vo voľne dostupnom Frameworku7. Všetky súbory sú uložené v pamäti flash. Po zadaní správnej adresy webového serveru v lokálnej LAN sieti, čip ESP8266 začne odosielať súbor `index.html`, v ktorom sú definované zvyšné súbory obsahujúce skripty a kaskádové štýly. To má za následok odoslanie súborov `fm7.css`, `fm7colors.css`, `fm7.js`, `canvas.min.js` a `my-app.js` na stranu klienta. Súbory sú potrebné pre celkové vykreslenie a interaktivitu užívateľského rozhrania. Výsledné vykreslenie rozhrania prebieha až na strane klienta.

Rozhranie poskytuje správcovi systému výber obdobia v kalendári. Po zadaní obdobia v kalendári si stránka vyžiada údaje o spotrebe všetkých evidovaných kariet v systéme. Klient odošle HTTP GET žiadosť na webový server. Formát odosielanej žiadosti je `items.json?day=x&amount=y`. Súbor `items.json` špecifikuje serveru akú činnosť má s dátami o spotrebe vykonať. Parameter `day` udáva posledný deň v zvolenom rozsahu kalendára a parameter `amount` udáva počet dní v zvolenom rozsahu.

Server po prijatí požiadavky najskôr spracuje všetky karty a spočíta spotrebu za zvolené obdobie. Zo spracovaných dát je dynamicky vygenerovaný súbor vo formáte JSON a ten

je odoslaný späť do webového prehliadača. Súbor `items.json` pozostáva z kolekcie štruktúr, pričom každá štruktúra obsahuje meno a údaj o spotrebe karty. Parameter "id" obsahuje mená kariet v hexadecimálnom tvare. Parameter "value" obsahuje spotrebu vody. Súbor je interpretovaný pomocou javascriptu na strane správcu a ponúka mu tak prehľad spotreby jednotlivých kariet za zvolené obdobie.

Pri zobrazení detailného náhľadu karty je odoslaná požiadavka na súbor `item_info.html`, ktorý obsahuje elementy vykresľujúce detailné zobrazenie. Klient si taktiež vyžiada súbor `values.json` a súbor `name.json`. Požiadavka má formát `values.json?id=x`, kde parameter "id" obsahuje názov identifikačnej karty v hexadecimálnom tvare. Hodnoty sú dynamicky spracované na strane serveru v čípe ESP8266. Ten otvorí binárny súbor `archiv.bin`, vyhledá záznam pre zvolenú kartu a odošle súbor `values.json` obsahujúci číselnú kolekciu hodnôt spotreby. Z kolekcie sa vyberú správne hodnoty zvolené v kalendári a tie sú zobrazené v grafe.

Požiadavka na súbor `name.json` má formát `name.json?id=x`, parameter `x` obsahuje názov karty. Server pri spracovaní porovná názov karty s uloženými súbormi vo flash pamäti. V prípade, ak nájde súbor s príponou `.info` s rovnakým názvom, obsah súboru s uloženými údajmi o majiteľovi karty odošle na stranu klienta.

Bezpečnosť komunikácie pokrýva predovšetkým zabezpečené Wi-Fi pripojenie pomocou štandardu IEEE 802.11i a používa zabezpečenie WPA/WPA2 PSK. Zabezpečenie sprístupnenia dát zahŕňa potreba autentifikácie správcu pomocou loginu a hesla.

Kapitola 9

Parametre a optimalizácia navrhnutého prototypu

Navrhnutý prototyp je po hardvérovej stránke schopný spĺňať všetky požadované funkcie. Vstupné napájanie sa pohybuje medzi 2.5 V – 5.0 V. Rozšíriť rozsah podporovaných napätí je možné pridaním miniatúrneho stabilizátora na báze spínaného zdroja, ktorých je na trhu dostatok. Spotreba výsledného prototypu sa pohybuje okolo hodnoty 145 mA. Pripojenie k prístupovému bodu je stabilné. Nedostatky spojené so stratou napájania boli odstránené pridaním záložnej batérie.

Pre skrátenie intervalu odosielania webového rozhrania zo strany serveru ku klientovi bola zvolená optimalizácia formou kompresie obsahu. Súbor sú komprimované do formátu gzip. Webový prehliadač umožňuje prijímanie takto komprimovaného obsahu bez vplyvu na výsledné zobrazenie. Zmenšením veľkosti odosielaných dát bol čas prenosu skrátený na štvrtinu. Čas odoslania stránky s užívateľským rozhraním na stranu klienta sa pohybuje medzi 3 s - 3.5 s. Veľkosť prenášaných dát bola zredukovaná z 856.7 kB na 178.4 kB.

Súbor, ktorý prenáša spotrebu na všetkých kartách, má 4.7 kB a jeho vytvorenie a spracovanie všetkých hodnôt za zvolené obdobie v ESP8266 trvá približne 550 ms. Súbor, ktorý prenáša detailný náhľad jednej karty, má veľkosť 7 kB. Jeho prijatie od odoslania požiadavky trvá približne 350 ms. Napriek tomu, že súbor detailnej spotreby jednej karty má väčšiu veľkosť ako súbor obsahujúci spotrebu všetkých kariet, jeho odoslanie trvá kratšie ako v prvom prípade. Pri prenose súboru items.json je potrebné prejsť vybrané hodnoty a vykonať väčší počet operácií pre získanie spotreby pre každú kartu.

Kapitola 10

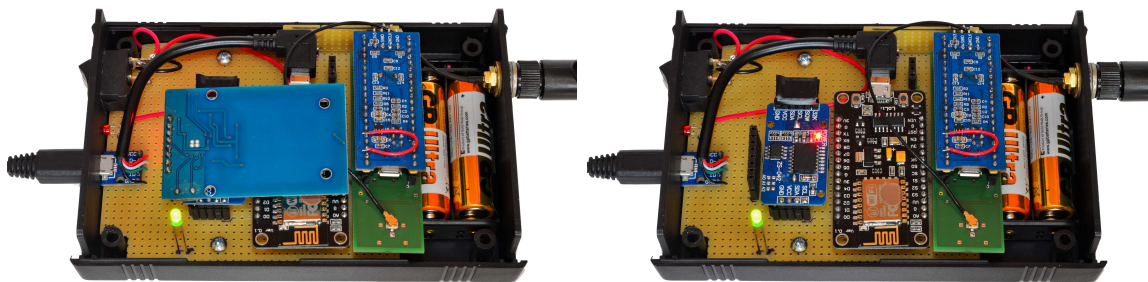
Finálna podoba prototypu a jeho využitie

Práca zahŕňa aj vytvorenie konštrukčného riešenia navrhnutého a implementovaného prototypu. Prototyp je umiestnený v uzatváratelnom plastovom obale. Komponenty sú vodivo prepojené pomocou PCB dosky plošných spojov. Prototyp je vybavený záložným napájaním. Záložné napájanie je ovládané pomocou spínača umiestneného na obale prototypu a aktivita záložného napájania je signalizovaná pomocou červenej LED diódy umiestnenej vedľa spínača viď obrázok č. 10.1. Záložné napájanie pozostáva z boxu na 3 alkalické batérie typu AA s napätím 4,5 V, ktoré je možné užívateľsky vymeniť viď obrázok č. 10.2.



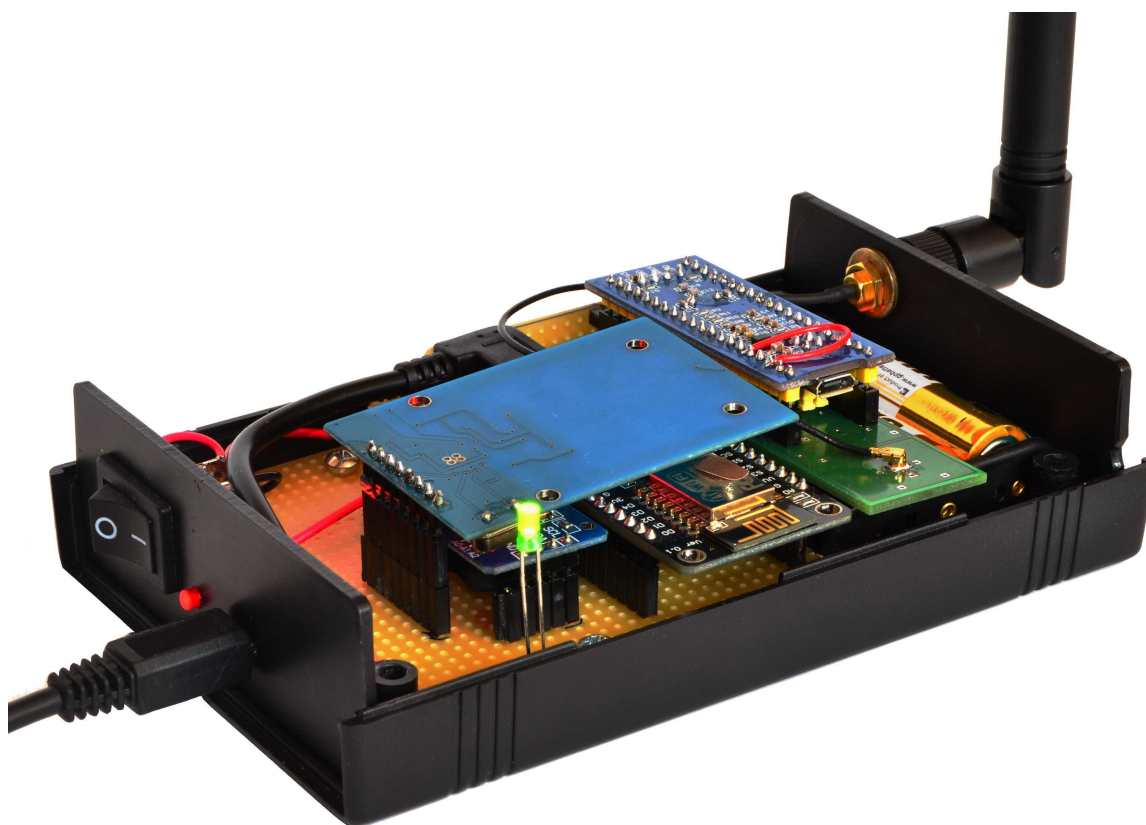
Obr. 10.1: Konštrukcia výsledného prototypu so spusteným záložným napájaním.

Konštrukcia prototypu je vybavená vstupom pre MikroUSB konektor. Konektor umožňuje programovanie a samotné napájanie všetkých súčastí. Konektor je vo vnútri pripojený k modulu s čipom ESP8266 a ten napája ďalšie komponenty. Vnútorne rozmiestnenie komponentov sa snaží brať ohľad na jednoduchosť vzájomného prepojenia. RFID modul pre bezkontaktné karty bol umiestnený tesne pod povrchom plastového krytu. To umožnilo aktívny dosah pri komunikácii s bezkontaktnou kartou na vzdialenosť približne 1,5 cm viď obrázok 10.4.



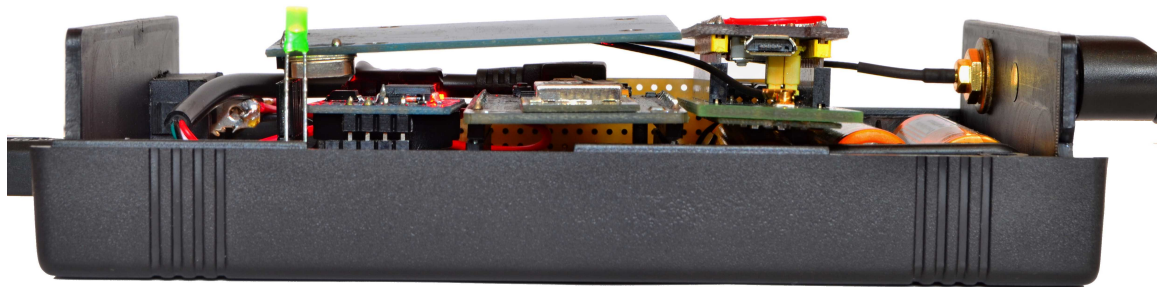
Obr. 10.2: Vnútro prototypu zhora. Vľavo prototyp obsahujúci RFID modul. Vpravo zobrazenie modulov NodeMCU, RTC a prijímača CC1101.

Ďalej je prototyp vybavený signalizačnou zelenou LED diódou. Dióda je pripojená k pinu číslo D3 modulu NodeMCU označeného ako GPIO0. Dióda indikuje pripojenie modulu k sieti Wi-Fi. Ak je sieť dostupná, dióda svieti. V prípade, ak prebieha pripájanie alebo je modul odpojený, dióda bliká. Správca systému tak má možnosť skontrolovať stav pripojenia prototypu k Wi-Fi sieti. Zoznam všetkých použitých komponentov je k dispozícii v prílohách č. C.1.



Obr. 10.3: Prototyp s otvoreným krytom.

Implementovaný prototyp je určený najmä pre správcov v ubytovacích zariadeniach, menších firemných priestoroch alebo pri správe bytových jednotiek, kde vzniká potreba evidencie spotreby vody viacerých užívateľov samostatne. Správca tým získa možnosť jednoduchšej evidencie všetkých užívateľov v systéme. Mierna modifikácia systému umožňuje



Obr. 10.4: Bočný pohľad na prototyp s umiestneným RFID modulom.

jeho prevádzku aj v súkromných domácnostiach. Systém okrem samotnej evidencie poskytuje aj určitú možnosť kontroly nad neželanými únikmi vody v prípade prasknutých potrubí alebo pretekajúcich vodovodných uzáverov.

Základnou požiadavkou na infraštruktúru pri nasadení prototypu v reálnej prevádzke je inštalovaný vodomer podporujúci bezdrôtovú komunikáciu prostredníctvom protokolu WM-Bus. Súčasne potrebuje dostupný Wi-Fi prístupový bod, prostredníctvom ktorého je možné zobrazovať spotrebu na mobilnom zariadení. Pripojenie k Wi-Fi vyžaduje konfiguráciu prístupových údajov pri nahrávaní firmvéru samotného prototypu. Prototyp vyžaduje umiestnenie v dosahu oboch sietí a pripojenie napájania k elektrickej sieti. Taktiež je vhodné prototyp dodatočne vybaviť batériami, ktoré v prípade výpadku napájania zabezpečia nepretržitú činnosť celého systému. Doplňujúce obrázky je možné nájsť v prílohe č. [B.1](#) a [B.2](#).

Kapitola 11

Záver

Cieľom tejto práce bolo predovšetkým navrhnuť a implementovať funkčný prototyp vstavaného systému určeného pre evidenciu spotreby vody. Systém umožňuje jeho správcovi prehľadné zobrazenie a interakciu v jeho mobilnom telefóne. Návrh pozostáva ako z hardvérovej tak aj softvétovej časti.

Systém je založený na platforme ESP8266, ktoré je vysoko integrované Wi-Fi SoC riešenie. ESP8266 pracuje ako samostatná aplikácia, čo umožňuje rýchly štart celého modulu. ESP8266 pri svojej činnosti využíva modul reálneho času a čítačku bezkontaktných kariet RFID.

Výsledný systém je navrhnutý s ohľadom na dlhodobú prevádzku, minimalizáciu počiatočných nákladov a jednoduchú réžiu. Implementované riešenie umožňuje ukladať informácie o sto užívateľoch. Navrhnutý systém disponuje informáciami o spotrebe, ktoré pokrývajú obdobie 33 mesiacov, čo je obdobie dlhé viac ako dva a pol roka s hodnotou spotreby uloženou pre každý deň.

Implementovaný prototyp prináša nové, originálne a praktické riešenie pre evidenciu spotreby vody. Zariadenie tohto zamerania sa snaží priniesť inováciu do oblastí, kde je potrebné riešiť evidenciu spotreby vody jednotlivých užívateľov vo vlastnej réžii. Prototyp je zatiaľ navrhnutý ako zobrazovacie zariadenie, pričom prijíma a ukladá namerané hodnoty z vodomeru. Presnosť zaručuje samotný vodoměr, ktorý podlieha kalibračnému testovaniu.

Vytvorený prototyp by mohol byť v budúcnosti rozšírený aj o prístup jednotlivých užívateľov. Tí by tak mali možnosť sami v reálnom čase sledovať vlastnú spotrebu vody s využitím napríklad možnosti notifikácie alebo nastavenie limitov spotreby. To by mohlo pomôcť obmedziť prípadné úniky, čo by malo zároveň pozitívny dopad na životné prostredie.

Ďalším smerom je možné rozšíriť modul o výkonnejšie MCU a sprístupniť tak možnosť interakcie s evidenčným systémom prostredníctvom internetu, ktorý by umožňoval pokročilejšiu filtráciu a detailnejšie zobrazenie jednotlivých období. Tým by sa toto zariadenie výraznejšie začlenilo do rýchlo sa rozrastajúcej rodiny IoT zariadení.

Literatúra

- [1] RFID. [Online], [cit. 2017-03-17].
URL <http://endtimetruth.com/mark-of-the-beast/rfid/>
- [2] Devkit10. 2015, [Online], [cit. 2017-03-18].
URL http://www.esp8266.com/wiki/lib/exe/detail.php?id=nodemcu&media=devkit_1.0.jpg
- [3] NodeMCU. 2016, [Online], [cit. 2017-03-18].
URL <http://www.esp8266.com/wiki/doku.php?id=nodemcu>
- [4] Mifare 13.56Mhz RC522 RFID Card Reader Module. 2017, [Online], [cit. 2017-03-17].
URL <http://www.hotmcu.com/mifare-1356mhz-rc522-rfid-card-reader-module-p-84.html>
- [5] NodeMcu. c2014, [Online], [cit. 2017-03-18].
URL <http://www.nodemcu.com>
- [6] Framework7. c2016, [Online], [cit. 2017-03-18].
URL <https://framework7.io/>
- [7] Wireless M-Bus protocol software. c2016, [Online], [cit. 2017-03-18].
URL <http://www.ti.com/tool/WMBUS#TechnicalDocuments>
- [8] DS3231 AT24C32 IIC module. [cca 2016], [Online], [cit. 2017-03-18].
URL <http://img.banggood.com/images/upload/2012/jiangjunchao/SKU082052j.JPG>
- [9] Barr, M.: *Programming embedded systems in C and C++*. Sebastopol: O'Reilly, první vydání, 1999, ISBN 1565923545.
- [10] Catsoulis, J.: *Designing embedded hardware*. Sebastopol: O'Reilly, druhé vydání, 2005, ISBN 0596007558.
- [11] Espressif: *ESP8266 Technical Reference*. [Online], Verze 1.2 (2016), [rev. 2016], [cit. 2017-03-09].
URL http://espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf
- [12] Espressif: *ESP8266EX Datasheet*. [Online], Verze 5.3 (2016), [rev. 2016], [cit. 2017-03-09].
URL http://espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf

- [13] Filippov, M.: Memory Map. 2015-06-15, [Online], [cit. 2017-04-07].
URL <https://github.com/esp8266/esp8266-wiki/wiki/Memory-Map>
- [14] Finkenzeller, K.: *RFID handbook*. Chichester: Wiley, třetí vydání, 2010, ISBN 9780470695067.
- [15] Maxim Integrated Products, San Jose: *DS3231*. [Online], Verze 10 (2016), [rev. 2016], [cit. 2017-03-17].
URL <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>
- [16] McRoberts, M.: *Beginning Arduino*. New York: Apress, první vydání, 2010, ISBN 9781430232407.
- [17] Mohan, V.: An Introduction to Wireless M-Bus. [Online], [cit. 2017-04-20].
URL <http://pages.silabs.com/rs/634-SLU-379/images/introduction-to-wireless-mbus.pdf>
- [18] NXP SEMICONDUCTORS, Eindhoven: *MFRC522*. [Online], Verze 3.9 (2016), [rev. 2016-04-27], [cit. 2017-03-17].
URL https://www.nxp.com/documents/data_sheet/MFRC522.pdf
- [19] Seem, P.: *Wireless MBUS Implementation with CC1101 and MSP430*. Texas Instruments, Dallas, 10 2008, [Online], [cit. 2017-03-09].
URL <http://www.ti.com/lit/an/swra234a/swra234a.pdf>
- [20] White, E.: *Making embedded systems*. Sebastopol: O'Reilly, první vydání, 2012, ISBN 9781449302146.
- [21] Zeman, K.: Implementace komunikačního protokolu Wireless M-BUS v simulačním prostředí NS-3. 2015.
- [22] Zheng, F.; Kaiser, T.: *Digital signal processing for RFID*. Chichester: Wiley, první vydání, 2016, ISBN 9781118824313.

Zoznam použitých skratiek

| | |
|-------|--|
| Wi-Fi | IEEE 802.11 štandard pre pripojenie k bezdrôtovej lokálnej sieti |
| SoC | System on a Chip (systém na jednom čipe) |
| PCB | Printed Circuit Board (doska plošných spojov) |
| MCU | Microcontroller Unit |
| ADC | Analog to Digital Converter (analogovo digitálny prevodník) |
| WEP | Wired Equivalent Privacy |
| WPA | Wi-Fi Protected Access |
| TCP | Transmission Control Protocol |
| UDP | User Datagram Protocol |
| HTTP | Hypertext Transfer Protocol |
| FTP | File Transfer Protocol |
| SS | Slave Select |
| MOSI | Master Output Slave Input |
| MISO | Master Input Slave Output |
| SCLK | Serial Clock |
| CS | Chip Select |
| MSB | Most Significant Bit (najviac významný bit) |
| LSB | Least Significant Bit (najmenej významný bit) |
| TWI | Two Wire Interface |
| SDA | Serial Data Line |
| SCL | Serial Clock Line |
| ACIAs | Asynchronous Communication Interface Adapters |
| IoT | Internet of Things (internet vecí) |
| SRD | Short Range Devices (zariadenia s krátkym dosahom) |
| RGB | Red Green Blue (aditívny farebný model) |

Prílohy

Príloha A

Obsah priloženého pamäťového média

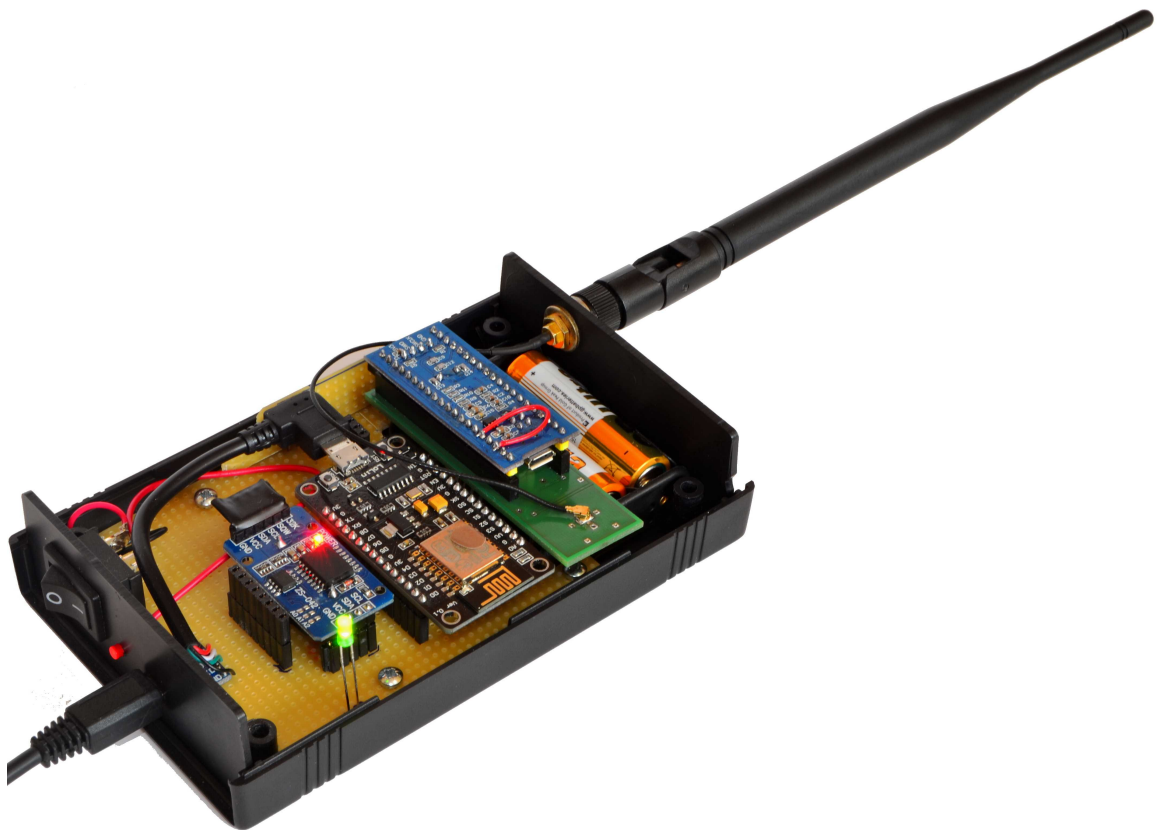
```
root
├── embedded_system
│   ├── data
│   │   ├── css
│   │   │   ├── fm7.css.gz
│   │   │   └── fm7colors.css.gz
│   │   ├── js
│   │   │   ├── canvas.js.gz
│   │   │   ├── fm7.js.gz
│   │   │   └── my-app.js.gz
│   │   ├── 373829278.info
│   │   ├── 812778037.info
│   │   ├── 3998947148.info
│   │   ├── archiv.bin
│   │   ├── index.html.gz
│   │   └── item_info.html.gz
│   ├── file_operations.h
│   └── embedded_system.ino
├── Poster.pdf
└── Bakalárska_práca.pdf
```

Príloha B

Výsledný prototyp



Obr. B.1: Fotografia prototypu v puzdre s pripojenou anténou.



Obr. B.2: Fotografia prototypu zobrazujúca uloženie komponentov bez RFID modulu pre lepšiu viditeľnosť

Príloha C

Zoznam použitých komponentov

| Popis | Počet kusov |
|---|-------------|
| NodeMcu Lua development board based on ESP8266 CP2102 | 1 ks |
| MFRC-522 RC522 RFID RF card sensor module | 1 ks |
| DS3231 AT24C32 IIC Real time clock module | 1 ks |
| CC1101 wireless module 868 MHz | 1 ks |
| STM32F103C8 | 1 ks |
| USB CJMCU-micro interface board 5 V | 1 ks |
| Red LED diode 5 mm 2,1 V 20 mA | 1 ks |
| Green LED diode 5 mm 2,1 V 20 mA | 1 ks |
| Batrery box 3 x AA 4,5 V | 1 ks |
| Switch 12 V 16 A | 1 ks |

Tabuľka C.1: Tabuľka zobrazujúca použité komponenty.

Príloha D

Plagát



Vstavaný systém pre evidenciu spotreby vody

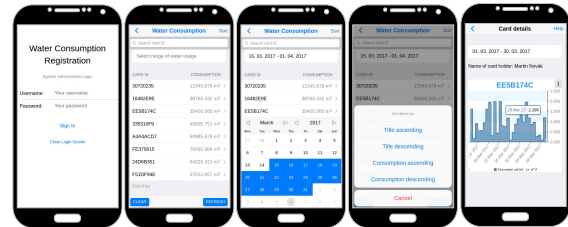
Vedúci: Ing. Zdeněk Vašíček, Ph.D.

Tomáš Hajdík
xhajdi01@stud.fit.vutbr.cz

Riešený problém

Cielom je návrh a implementácia funkčného prototypu vstavaného systému, ktorý je schopný evidovať množstvo spotrebovanej vody u jednotlivých užívateľov prostredníctvom bezdrôtového odpočtu stavu vodomera.

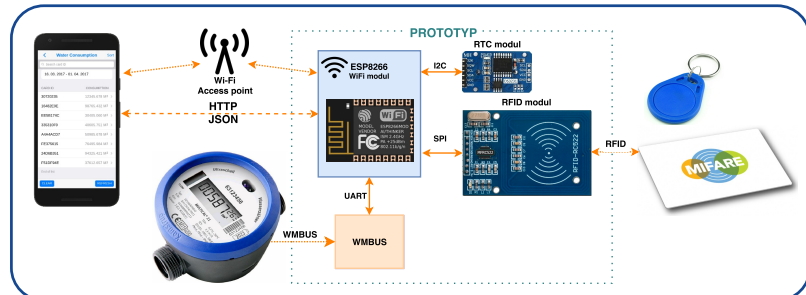
Užívateľia majú možnosť identifikácie bezkontaktnou RFID kartou. Vstavaný systém sprístupňuje dáta administrátorovi pomocou siete Wi-Fi. Správca systému interaguje so systémom pomocou mobilného telefónu a aplikácie bežiacej na module ESP8266.



Implementácia webového rozhrania zobrazujúceho identifikačné karty a spotrebu vody za zvolené obdobie.

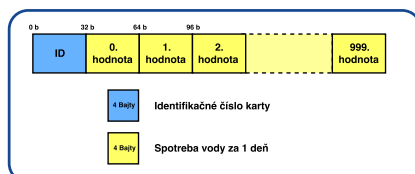
Prototyp a súčasti infraštruktúry

- ESP8266-12E modul
- MFRC522 RFID čítačka
- DS3231 RTC modul
- CC1101 WM-Bus modul
- Vodomer vybavený WM-Bus
- Wi-Fi prístupový bod
- Mobilný telefón
- čípkové identifikačné karty



Dátové štruktúry

Dáta o spotrebe jednotlivých užívateľov sú uložené v binárnom súbore. Pre dosiahnutie čo najväčšieho množstva evidovaných kariet s čo najväčšou dĺžkou evidovaného obdobia je súbor rozdelený na 100 sektorov usporiadaných lineárne za sebou s veľkosťou 4004 B. Prvý blok je vyčlenený pre uloženie identifikačného čísla karty. Zvyšné bloky obsahujú údaj spotreby vody za jeden deň. Hodnota aktuálneho dňa počítaná od 1. 1. 1970 modulu 1000 je index konkrétneho bloku.



Parametre navrhnutého riešenia

- 100 užívateľov
- 33 mesiacov dát o spotrebe každého užívateľa
- hodnota spotreby uložená pre každý deň
- 400 kB evidovaných dát
- grafické zobrazenie hodnôt spotreby
- krátka prístupová doba
- možnosť dlhodobej prevádzky
- výber zobrazovaného obdobia správcom systému

Návrh vstavaného systému

- **Hardvérová časť** sa venuje výberu vhodných komponentov, návrhu vzájomného zapojenia, spôsobu komunikácie, ale taktiež aj spôsobu uloženia dátových štruktúr obsahujúcich informácie o evidencii a spotrebe vody.
- **Softvérová časť** sa zaoberá najmä návrhom grafického užívateľského rozhrania a jeho distribúcie prostredníctvom webového serveru a siete Wi-Fi správcovi systému priamo do jeho mobilného telefónu.

Prínos implementovaného prototypu

- Nové originálne a praktické riešenie evidencie spotreby vody
- Prehľadné zobrazenie spotreby
- Interakcia pomocou mobilného telefónu
- Nízke investičné a prevádzkové náklady
- Evidencia spotreby vo vlastnej réžii
- Minimálne nároky na podpornú infraštruktúru
- Jednoduchá dostupnosť