



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ**

DEPARTMENT OF COMPUTER SYSTEMS

**VESTAVĚNÉ ZAŘÍZENÍ PRO POTŘEBY REGISTRACE**

EMBEDDED DEVICE FOR REGISTRATION PURPOSES

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**VOJTĚCH NOVOTNÝ**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Ing. ZDENĚK VAŠÍČEK, Ph.D.**

BRNO 2023

## Zadání bakalářské práce



147912

Ústav: Ústav počítačových systémů (UPSY)  
Student: **Novotný Vojtěch**  
Program: Informační technologie  
Specializace: Informační technologie  
Název: **Vestavěné zařízení pro potřeby registrace**  
Kategorie: Vestavěné systémy  
Akademický rok: 2022/23

### Zadání:

1. Seznamte se s vhodnou platformou pro realizaci vestavěných zařízení vybavených grafickým displejem umožňující interakci s uživatelem pomocí dotykové obrazovky.
2. Seznamte se s vhodnou grafickou knihovnou umožňující realizaci interaktivních uživatelských rozhraní jako je např. LVGL, TouchGFX apod.
3. Zpracujte studii pokrývající témata uvedená v bodu 1 a 2 zadání.
4. Navrhněte vestavěné zařízení umožňující registraci objektů zobrazených na displeji k určité entitě. Identifikace entity, pro kterou registrace proběhla bude je realizována pomocí RFID.
5. Navržené zařízení implementujte formou prototypu včetně podpory pro administraci, která bude umožňovat správu identifikátorů entit a získávání denních, týdenních a měsíčních reportů. Konfigurace a přístup k reportům bude realizována bezdrátově pomocí technologie Bluetooth nebo WiFi.
6. Ověřte funkčnost navrženého řešení. Vyhodnoťte jeho parametry a diskutujte možnosti dalšího rozšíření.

### Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 1 až 4 zadání.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Vašíček Zdeněk, doc. Ing., Ph.D.**  
Vedoucí ústavu: **Sekanina Lukáš, prof. Ing., Ph.D.**  
Datum zadání: 1.11.2022  
Termín pro odevzdání: 10.5.2023  
Datum schválení: 31.10.2022

## Abstrakt

Tato práce se zabývá návrhem a implementací vestavěného zařízení pro potřeby registrace. Navržený systém používá jako způsob identifikace entity RFID karty. Zařízení je realizováno pomocí mikrokontroleru ESP32. Konfigurace systému a generování reportů o provedených registracích je řešeno pomocí webového rozhraní, které je poskytováno pomocí webserveru běžícím na tomto zařízení.

## Abstract

This thesis deals with the design and implementation of an embedded device for registration purposes. The proposed system uses RFID cards as a method of entity identification. The device is implemented using ESP32 microcontroller. Configuration of the system and generation of reports on the registrations performed is handled through a web interface provided by a webserver running on the device.

## Klíčová slova

Vestavěné zařízení, registrační systém, ESP32, RFID, WiFi, LVGL

## Keywords

Embedded device, registration system, ESP32, RFID, WiFi, LVGL

## Citace

NOVOTNÝ, Vojtěch. *Vestavěné zařízení pro potřeby registrace*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. Ing. Zdeněk Vašíček, Ph.D.

# Vestavěné zařízení pro potřeby registrace

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana doc. Ing. Zdeňka Vašíčka, Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....  
Vojtěch Novotný  
9. května 2023

## Poděkování

Tímto bych chtěl poděkovat panu doc. Ing. Zdeňku Vašíčkovi, Ph.D. za odborné vedení, cenné připomínky a konzultace, které mi poskytl při vypracování této práce.



# Obsah

<b>1</b>	<b>Úvod</b>	<b>3</b>
<b>2</b>	<b>Registrační systémy</b>	<b>4</b>
2.1	Přístupové systémy . . . . .	5
2.2	Docházkové systémy . . . . .	6
2.3	Požizovací cena . . . . .	6
<b>3</b>	<b>Technologie RFID</b>	<b>8</b>
3.1	Princip komunikace . . . . .	8
3.2	Druhy tagů . . . . .	9
3.3	Frekvence a rozsahy . . . . .	10
<b>4</b>	<b>Platforma ESP32</b>	<b>12</b>
4.1	Architektura . . . . .	12
4.2	Dostupné moduly . . . . .	14
4.3	FreeRTOS . . . . .	14
4.4	Tvorba firmware . . . . .	15
<b>5</b>	<b>Zobrazovací zařízení pro vestavěné systémy</b>	<b>17</b>
5.1	Grafické knihovny pro vestavěné systémy . . . . .	18
<b>6</b>	<b>Návrh prototypu</b>	<b>20</b>
<b>7</b>	<b>Implementace</b>	<b>24</b>
7.1	Uživatelské rozhraní . . . . .	24
7.2	Ukládání dat . . . . .	26
7.3	Identifikace entity . . . . .	27
7.4	Administrační portál . . . . .	28
<b>8</b>	<b>Výsledky a diskuze</b>	<b>31</b>
8.1	Cenové zhodnocení prototypu . . . . .	31
8.2	Vliv použitého displeje na spotřebu energie . . . . .	32
8.3	Zhodnocení používání zařízení . . . . .	35
<b>9</b>	<b>Závěr</b>	<b>36</b>
	<b>Literatura</b>	<b>37</b>
<b>A</b>	<b>Srovnání modulů ESP32</b>	<b>39</b>

# Seznam obrázků

2.1	Zařízení na řízení přístupu od firmy EFG CZ, převzato z [4]	4
2.2	Docházkový systém od firmy EFG CZ, převzato z [3]	5
3.1	Diagram znázorňující komunikaci mezi prvky RFID systému, převzato z [10]	8
3.2	Ukázky pasivních (vlevo, převzato z [18]) a aktivních (vpravo, převzato z [10]) RFID tagů	9
3.3	Frekvenční rozsahy používané pro RFID, převzato z [10]	10
4.1	Funkční blokový diagram ESP32, převzato z [9]	14
5.1	Ukázka editoru SquareLine Studio, převzato z [7]	19
6.1	Návrh blokového diagramu vestavěného zařízení	20
6.2	Návrh uživatelského rozhraní vestavěného zařízení	21
6.3	Návrh uživatelského rozhraní konfigurace zařízení v administračním portálu	23
7.1	Ukázka uživatelského rozhraní na prototypu s e-paper displejem	25
7.2	Ukázka konfigurace v administračním portálu vestavěného zařízení	29
7.3	Ukázka vygenerovaného reportu v administračním portálu vestavěného zařízení	30
8.1	Ukázka naměřené spotřeby prototypu s e-paper displejem v režimu spánku	32
8.2	Ukázka naměřené spotřeby prototypu s e-paper displejem s aktivním administračním portálem	33

# Kapitola 1

## Úvod

V dnešním světě jsou registrační systémy nezbytnou součástí mnoha organizací, včetně škol, podniků a vládních institucí. Tradiční registrační systémy často spoléhají na ruční zadávání dat, což může být časově náročné a náchylné k chybám. K vyřešení těchto problémů se mnoho organizací obrací na registrační zařízení, která mohou automatizovat proces registrace a zlepšit přesnost zadaných dat.

Vestavěná zařízení jsou malé počítače s nízkou spotřebou, které jsou navrženy tak, aby vykonávaly konkrétní úkol, jako je monitorování teploty, nebo sběr dat. V posledních letech se vestavěná zařízení stávají stále oblíbenější kvůli jejich cenové dostupnosti, flexibilitě a snadnému použití.

Technologie RFID je široce používaná technologie pro autentizaci, která umožňuje bezdrátovou komunikaci mezi čtečkou a pasivním nebo aktivním štítkem. Zařízení na bázi ESP32 vyvinuté v této práci bude využívat technologii RFID k identifikaci a autentizaci entit a bude zaznamenávat data spojená s registracemi těchto entit. Zařízení dále umožní autorizovaným entitám zařízení konfigurovat a také vytvářet reporty o provedených registracích.

Tato práce je členěna následovně. V kapitole 2 jsou popsány obecné registrační systémy, jejich možnosti, fungování a pořizovací cena. V následující kapitole 3 je probrána technologie RFID, princip komunikace, druhy datových nosičů (tagů) a používané frekvence. Kapitola 4 je zaměřena na analýzu platformy ESP32. V rámci kapitoly jsou popsány parametry platformy ESP32, tvorba firmware pro platformu a operační systém FreeRTOS. V následující kapitole 5 jsou popsány některá zobrazovací zařízení dostupné pro vestavěné systémy, dále jsou v této kapitole popsány grafické knihovny, které mohou být použity v kombinaci s platformou ESP32. Návrh prototypu je popsán v kapitole 6, kde je návrh popsán po jednotlivých funkčních blocích. Kapitola 7 je věnována popisu implementačních detailů a zhodnocení kapacitního limitu pro záznamy na vestavěném zařízení. V poslední kapitole 8 jsou zhodnoceny vlastnosti a parametry obou variant prototypu.

## Kapitola 2

# Registrační systémy

Registrační systém se používá v aplikacích, kde je potřeba registrovat předem definované objekty k entitám. Získané informace o provedených registracích jsou shromažďovány a je možné s nimi dále pracovat. Systém umožňuje zobrazovat přehledy o tom, ke které entitě byl registrován vybraný objekt, vyhledávání v přehledech registrací a vytváření reportů (za časové období, vztažené k určité entitě, nebo objektu atd.).

Na základě získaných dat lze provádět předem definované akce, podle definovaných omezení. Například podle identifikátoru entity lze definovat množinu objektů, které k ní mohou být přiřazeny, nebo pokud má entitou vybraný objekt přiřazenou akci (otevření dveří, vydání položky ze skladu atd.) a entita má potřebné oprávnění pro požadovanou akci, tak akce bude provedena.

Jedním z možných rozdělení registračních systémů je na přístupové systémy a docházkové systémy.



Obrázek 2.1: Zařízení na řízení přístupu od firmy EFG CZ, převzato z [4]

## 2.1 Přístupové systémy

Přístupové systémy lze považovat za registrační systémy, kde je entitou uživatel s přístupovou kartou a objektem je označení budovy, nebo místnosti, kam chce uživatel vstoupit. Přehledy budou v tomto případě obsahovat záznamy o přístupech jednotlivých entit.

Tyto systémy mají podobu, jakou mají systémy řízení přístupu na obrázku 2.1. Podle složitosti lze zařízení přístupových systémů rozdělit do dvou kategorií, podle složitosti zařízení. Jednodušší zařízení se skládají pouze z mechanismu pro ověření identity uživatele (čtečka otisku prstů, RFID čtečka, ...). Složitější zařízení obsahují mimo mechanismu pro ověření identity uživatele také displej, což umožňuje zobrazit uživateli data, jako například aktuální čas, statistiky přístupů uživatele atd. Použití dotykového displeje v zařízení umožňuje pokročilejší interakci s uživatelem.

Správa přístupových systémů je většinou řešena centrálně, tedy po identifikaci uživatele na zařízení jsou informace odeslány pro vyhodnocení a zpět je poslána informace o povolení, nebo zamítnutí přístupu. Indikace výsledku vyhodnocení je na jednodušších zařízeních realizována pomocí LED diody. Pokud byl přístup povolen, led dioda se rozsvítí zeleně, pokud byl přístup zamítnut, led dioda se rozsvítí červeně. Na složitějších zařízeních je pro zpětnou vazbu pro uživatele použit displej. Následně, pokud byl uživateli přístup povolen, je aktivován elektromechanický zámek a uživatel je vpuštěn do prostoru.

Komunikace mezi přístupovým zařízením a řídicí jednotkou přístupového systému většinou probíhá po síti LAN, pomocí technologie WiFi, nebo pomocí technologie Ethernet. Použití technologie Ethernet umožňuje zařízení napájet pomocí *Power over Ethernet*, což minimalizuje počet kabelů, které vedou k zařízení. Touto možností jsou ale vybaveny většinou dražší zařízení, cenově dostupnější zařízení používají dodatečný zdroj napájení.



Obrázek 2.2: Docházkový systém od firmy EFG CZ, převzato z [3]

## 2.2 Docházkové systémy

Docházkové systémy slouží k elektronické evidenci pracovní doby. Tyto docházkové systémy mají firmám zjednodušit a zefektivnit zadávání a vyhodnocování dat. Moderní docházkové systémy umí například plánovat směny, evidovat práci na projektech atd. Zároveň tyto systémy umí hlídat splnění zákonných limitů, nebo zadaných požadavků.

Také docházkové systémy lze považovat za přístupové systémy, kde je entitou uživatel s identifikační kartou a objektem je akce spolu s časovým razítkem vzniku události. Přehledy budou v tomto případě obsahovat přehled jednotlivých událostí (například příchod, odchod, pauza, atd.).

Správa docházkových systémů je také většinou řešena centrálně, tedy po vybrání události a identifikaci uživatele, jsou tato data poslána a uložena do systému centrální správy dat. Tento přístup usnadňuje práci s daty (například generování reportů, analýza dat ...).

Moderní docházkové systémy komunikují pomocí technologie WiFi, nebo pomocí technologie Ethernet. Data z těchto systémů jsou ukládána do databáze, která může být umístěna na cloudu, nebo na serveru umístěném v podnikové síti.

Docházkové systémy jsou koncepčně velmi podobné, jako přístupové systémy zmíněné v kapitole 2.1. Často jsou tyto dvě funkcionality sloučeny do jednoho systému, který spravuje docházku i přístupy uživatelů. Příkladem systému, který kombinuje tyto funkcionality je systém *Aktion* od firmy EFG CZ, jak je uvedeno v [3, 4].

## 2.3 Pořizovací cena

Pro zhodnocení pořizovací ceny vybraných registračních systémů byla použita hypotetická firma střední velikosti, která má 40 zaměstnanců rozdělených do 4 oddělení. Tato firma potřebuje docházkový systém s možností analyzovat data evidence pracovní doby. Na základě požadavků hypotetické firmy, jsem vybral několik docházkových systémů pro porovnání.

### Docházkový systém od společnosti vistech

Docházkový systém je koncipován jako síť terminálů centrálně spravovaných software *SW TimeBox*. Společnost vistech nabízí škálu terminálů, pro evidenci pracovní doby, tyto terminály se mezi sebou liší použitelnými způsoby ověření identity uživatele a podporou technologií Ethernet a WiFi. Terminály, které podporují Ethernet, mohou být napájeny pomocí Power over Ethernet, nebo pomocí dodatečného zdroje napájení. S každým zakoupeným terminálem, je zahrnutá licence software *SW TimeBox*, která podporuje až 100 pracovníků. Dále systém podporuje export pracovních výkazů ve formátu csv, nebo do excelové tabulky [22, 21].

Za předpokladu, že hypotetická firma nepotřebuje připojit terminály přes WiFi a autentizace uživatele přes RFID tag je dostačující, tak firmě stačí docházkový terminál *Time BOX X4*, který aktuálně na e-shopu firmy vistech stojí 13903 Kč včetně DPH. Dále je potřeba zakoupit RFID tagy, pro 40 RFID karet na e-shopu firmy vistech stojí 3392 Kč včetně DPH. Tedy při umístění jednoho terminálu do každého oddělení je výsledná cena pořízení systému od společnosti vistech 59004 Kč včetně DPH [23].

## Docházkový systém TimeMoto

Docházkový systém TimeMoto, je koncipován podobně, jako systém od společnosti vistech, tedy skládá se ze sítě terminálů, které se starají o evidenci pracovní doby. Data z terminálů se potom posílají po síti buď na počítač s nainstalovaným software *TimeMoto PC Software*, nebo na *TimeMoto Cloud*. Společnost Safescan nabízí několik druhů terminálů, které se mezi sebou liší použitelnými způsoby ověření identity uživatele (RFID tag, čtečka otisku prstu . . .). Dále, při použití *TimeMoto Cloud*, systém podporuje evidenci docházky z webového prohlížeče, nebo pomocí aplikace v mobilním telefonu. Tento systém také podporuje generování reportů, upravování reportů a také export do excelové tabulky [15].

Pořízení tohoto docházkového systému pro hypotetickou firmu lze provést dvěma způsoby. První varianty zahrnuje pořízení terminálu *TIMEMOTO TM-616*, který by měl pokrýt požadavky hypotetické firmy, pokud je autentizace uživatele pomocí RFID, nebo pomocí pinu dostačující. Tento terminál je dostupný na e-shopu safescan za 15304,08 Kč včetně DPH. Ke každému terminálu je přibaleno pět RFID karet. Tedy při umístění jednoho terminálu do každého oddělení, je potřeba ještě zakoupit další RFID karty, ty jsou nabízeny po sadách s 25 kartami za 1696,42 Kč včetně DPH. K tomuto vybavení je ještě potřeba pořídit software *TIMEMOTO PC PLUS*, který je potřeba pro shromažďování dat a vytváření reportů. Pořízení tohoto software stojí 7234,59 Kč včetně DPH. Tedy výsledná ceny první varianty je 70147,33 Kč včetně DPH [14].

Druhá varianta nezahrnuje pořízení terminálů, ale pořízení předplatného *TIMEMOTO CLOUD*, tedy uživatelé by používali k evidenci pracovní doby webový prohlížeč, nebo mobilní aplikaci. Tato varianta pro 40 zaměstnanců vyjde na 10360 Kč na rok včetně DPH [16].

## Docházkový systém sloneek

Docházkový systém saloneek je koncipován odlišně od systému od společnosti vistech. Tento systém je koncipován, jako webový portál, tedy nevyužívá klasických terminálů jako systém od společnosti vistech. U systému sloneek lze nakonfigurovat, jako docházkový terminál nakonfigurovat jakékoliv chytré zařízení s dotykovou obrazovkou. Tento systém také podporuje zaznamenávání příchodů a odchodů pomocí mobilní aplikace, nebo pomocí webového rozhraní. Systém sloneek dále podporuje tvorbu reportů, jejich přizpůsobení a také jejich export do excelové tabulky. Tento docházkový systém nemá pevnou pořizovací cenu, ale funguje na základě předplatného. Předplatné se dělí na několik kategorií, dělených podle schopností systému [19].

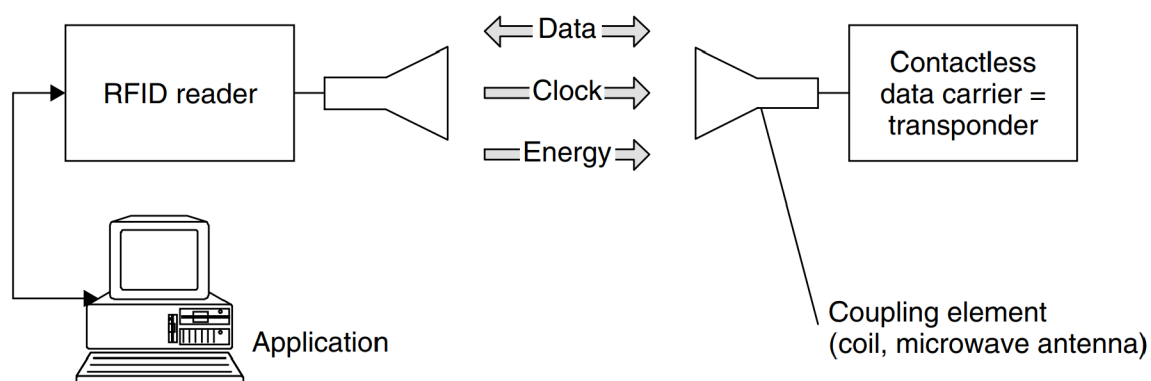
Z požadavků vyplývá, že by hypotetické firmě měla stačit základní verze předplatného *Essential*, která stojí 4 € měsíčně na uživatele. Podle aktuálního kurzu 4 € odpovídají 94,94 Kč. Tedy pro 40 uživatelů by to znamenalo částku 45571,2 Kč na rok [19].



## Kapitola 3

# Technologie RFID

Zkratka RFID (Radio-frequency identification) znamená identifikace pomocí rádiové frekvence, tj. informaci přenášenou rádiovými vlnami, tato informace *Data* je uložena na elektronickém nosiči dat – transpondéru (tagu). RFID systém se tedy skládá ze dvou komponent, transpondéru (tagu) a čtečky. Případná komunikace poté probíhá mezi těmito komponentami a získaná data jsou dále zpracována [10].



Obrázek 3.1: Diagram znázorňující komunikaci mezi prvky RFID systému, převzato z [10]

### 3.1 Princip komunikace

Výměna dat mezi nosičem dat a čtečkou probíhá bez použití galvanických kontaktů, místo toho se využívá magnetické nebo elektromagnetické pole. Když tag, který obvykle nemá vlastní zdroj napětí (baterii), není v interakční zóně čtečky (oblast působení elektromagnetického pole čtečky), je zcela pasivní. Tag je aktivován pouze tehdy, když se nachází v interakční zóně čtečky. Na diagramu 3.1 je tento přenos energie znázorněn šipkou s popisem *Energy*. Hodinový signál a data pro tag jsou dodávána prostřednictvím čtečky (bezkontaktně). Na diagramu 3.1, je přenos hodinového signálu znázorněn šipkou s popisem *Clock* a přenos dat je znázorněn obousměrnou šipkou s popisem *Data*. Základní technický postup vychází z oboru radiotechniky a radarové techniky [10].



## 3.2 Druhy tagů

RFID tagy se vyrábí v mnoha různých provedeních, podle požadavků konkrétních aplikací. Tag potřebuje energii pro komunikaci s čtečkou. Podle toho, jakým způsobem tag získává energii pro přenos dat můžeme rozdělit tagy na pasivní a aktivní.



Obrázek 3.2: Ukázky pasivních (vlevo, převzato z [18]) a aktivních (vpravo, převzato z [10]) RFID tagů

### Pasivní tagy

Pasivní RFID tag nemá vestavěný zdroj energie, a tedy získává energii z přijímaných rádiových vln, vysílaných čtečkou. Množství přijaté energie je dostačující pro napájení integrovaného obvodu v pasivním tagu, omezuje ale dodatečné funkce pasivního tagu, například aktivní vysílač. Když pasivní tag není v interakční zóně čtečky, nemá žádné napájení, a tedy není schopen předávat žádná data. Proto je zbytečné dávat na pasivní tagy další snímače (teploty, tlaku atd.) [12].

Často se pasivní tagy vkládají do samolepicích štítků, které se snadno a rychle připevňují, nebo někdy do samotných předmětů. Díky své nízké ceně se pasivní tagy hodí v aplikacích, pro které se štítky nedají opakovaně použít [12].

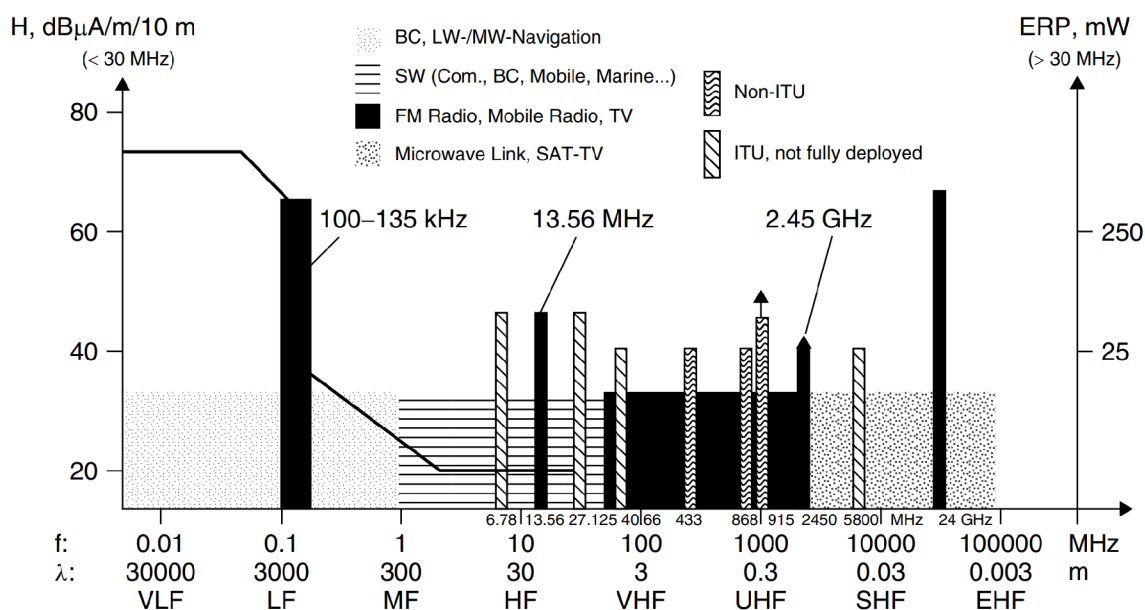
### Aktivní tagy

Aktivní RFID tag má vestavěný zdroj energie, obvykle baterii, a aktivní vysílač. Integrovaný obvod aktivního tagu může obsahovat větší výpočetní výkon pro implementaci dalších funkcí, jako je například manipulace s daty. Aktivní tag používá pro napájení integrovaného obvodu a vysílače vestavěný zdroj energie. K přenosu dat nepotřebuje rádiové vlny z čtečky, tudíž nepotřebuje ani čtečku. Aktivní tag může být nastaven tak, aby vysílal svá data v předem nastaveném čase, periodicky nebo při výskytu určité události. Jeho typický čtecí dosah je 90 až 220 metrů. Dosah čtení závisí na vestavěném zdroji energie a typu aktivního vysílače [12].

Aktivní tag může mít vestavěné snímače nebo k němu mohou být připojeny externí snímače. S větším výpočetním výkonem může aktivní tag shromažďovat data ze senzorů a před vysláním je lokálně zpracovávat. Aktivní tagy se často používají v RTLS systémech (lokalizační systémy v reálném čase) [12].

### 3.3 Frekvence a rozsahy

Vzhledem k tomu, že systémy RFID generují a vyzařují elektromagnetické vlny, jsou z právního hlediska klasifikovány jako rádiové systémy. Provoz systémů RFID nesmí za žádných okolností narušit nebo narušovat funkci jiných rádiových služeb, zejména komunikace složek integrovaného záchranného systému. Nutnost brát ohled na ostatní rádiové služby významně omezuje rozsah vhodných provozních frekvencí, které má systém RFID k dispozici viz. obrázek 3.3 [10].



Obrázek 3.3: Frekvenční rozsahy používané pro RFID, převzato z [10]

#### Srovnání frekvencí u technologie RFID

Technologie RFID funguje v různých frekvenčních rozsazích v závislosti na aplikaci a oblasti světa. Existují čtyři hlavní frekvenční rozsahy RFID: nízké frekvence (na obr. 3.3 LF), vysoké frekvence (na obr. 3.3 HF), ultra vysoké frekvence (na obr. 3.3 UHF) a mikrovlnné frekvence (na obr. 3.3 SHF).

Tabulka 3.1: Srovnání frekvencí u RFID technologie, převzato z [13]

Frekvence	Objem dat	Čtecí rychlost	Dosah	Využití
125 – 134 kHz	malý	nízká	0,9 – 1,5 m	přístupové systémy, sledování sklad. zásob
13.56 MHz	malý – střední	střední	0,3 – 0,9 m	sledování položek, přístupové systémy
860 – 960 MHz	malý – střední	vysoká	0,3 – 9 m	sledování palet a beden (průmysl)
2.45 a 5.4 GHz	střední	vysoká	0,3 – 91 m	vybírání mýtného, sledování palet

Nízkofrekvenční RFID pracuje ve frekvenčním rozsahu 125 kHz až 134 kHz a používá se hlavně pro řízení přístupu, sledování zvířat a identifikační aplikace. Má kratší dosah čtení, obvykle do několika centimetrů, a je méně ovlivněn rušením z jiných zdrojů. Při použití nízkých frekvencí je také méně pravděpodobné, že bude přenos dat ovlivněn rušením, takže je vhodný pro použití v prostředích s horšími podmínkami [13].

Vysokofrekvenční RFID pracuje ve frekvenčním rozsahu 13,56 MHz a běžně se používá pro aplikace bezkontaktních čipových karet, řízení přístupu a sledování zásob. Má delší dosah čtení než nízkofrekvenční RFID, obvykle do 1 m, ale může být náchylnější k rušení. Tato náchylnost k rušení činí vysokofrekvenční RFID méně vhodným pro použití v prostředích s horšími podmínkami [13].

Ultra-vysokofrekvenční RFID pracuje ve frekvenčním rozsahu 860 MHz až 960 MHz a používá se pro řízení dodavatelského řetězce, sledování zásob, sledování palet a sledování majetku. Má delší čtecí dosah než nízkofrekvenční i vysokofrekvenční RFID, obvykle až několik metrů, a je méně náchylné na rušení [13].

Mikrovlnná frekvence RFID pracuje ve frekvenčním rozsahu 2,4 GHz a používá se pro vysokorychlostní aplikace RFID s dlouhým dosahem, jako je například výběr mýtného a sledování vozidel. Má delší čtecí dosah než ultra-vysokofrekvenční RFID, typicky až desítky metrů, a je méně náchylný na rušení z jiných zdrojů [13].

Volba frekvence RFID závisí na několika faktorech, včetně aplikace, vzdálenosti mezi čtečkou a štítkem a prostředím, ve kterém bude systém RFID fungovat. Tyto faktory je potřeba pečlivě zvážit při výběru frekvence RFID, pro specifickou aplikaci, aby byly zajištěny optimální podmínky pro technologii RFID.

## Kapitola 4

# Platforma ESP32

ESP32, vytvořený společností Espressif Systems, je systém na čipu (System on Chip – SoC) s vestavěnou podporou pro WiFi a Bluetooth komunikaci. ESP32 byl navržen pro mobilní zařízení, nositelnou elektroniku a IoT aplikace (aplikace internetu věcí). Platforma dosahuje velmi nízké spotřeby energie díky funkcím pro úsporu energie, jako je například: více režimů napájení a dynamické škálování spotřeby [9].

### 4.1 Architektura

ESP32 je založený na dvoujádrovém nebo jednojádrovém (podle varianty) mikroprocesoru *Tensilica Xtensa LX6* s taktem až 240 MHz a sekundárního koprocesoru s velmi nízkou spotřebou. ESP32 dále disponuje pamětí ROM o velikosti 448 KB a pamětí SRAM o velikosti 520 KB [9].

ESP32 disponuje i mnoha vestavěnými periferiemi, které jsou významným přínosem pro vývojáře IoT aplikací. Díky těmto periferiím mohou vývojáři snadno připojit zařízení, jako jsou senzory, přístroje pro měření, přenosné zařízení pro záznam dat a další. Mezi těmito periferiemi jsou například WiFi a Bluetooth moduly, které poskytují bezdrátové připojení k internetu a dalším zařízením [9]. V obrázku 4.1 jsou zobrazeny periferie ESP32, některé z nich budou v této kapitole podrobněji popsány.

Další důležitou periferií na ESP32 jsou analogově-digitální převodníky (ADC) a digitálně-analogové převodníky (DAC). Tyto převodníky umožňují měření analogových vstupů a výstupů, což je klíčové pro měření různých fyzikálních veličin, jako je například teplota, vlhkost, světelná intenzita a další. Díky těmto převodníkům je ESP32 ideální pro využití v aplikacích, které vyžadují měření analogových signálů [9].

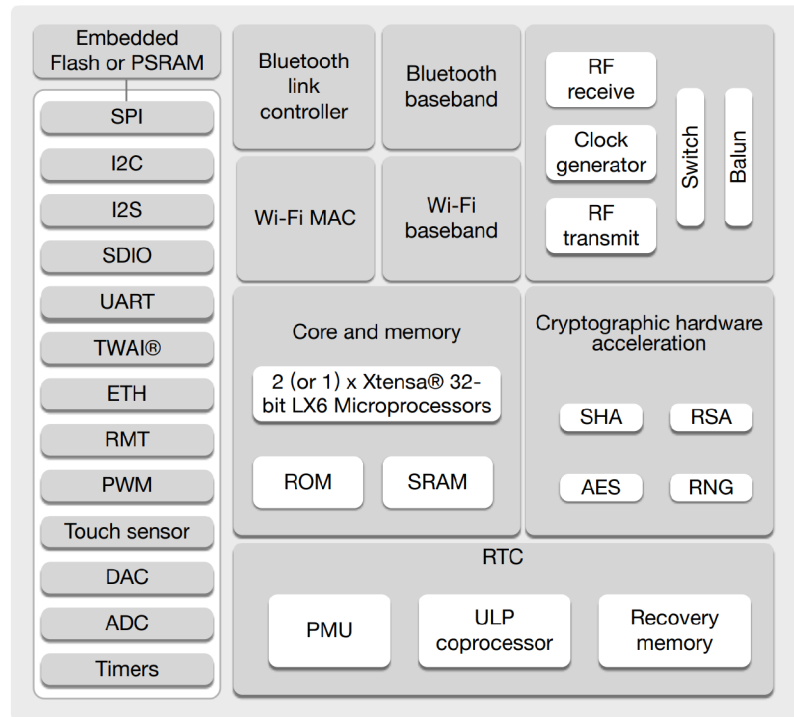
ESP32 také disponuje velkým počtem GPIO pinů, které jsou programovatelné a podporují různá rozhraní, jako jsou například SPI, I2C a UART. Tyto vstupy a výstupy umožňují vývojářům snadno připojit různá rozhraní a periferie [9].

Kromě toho, ESP32 obsahuje hardwarové akcelerátory pro kryptografii AES a SHA, což zajišťuje bezpečnost výměny dat. ESP32 také obsahuje bezpečné spouštění a bezpečné úložiště pro kryptografické klíče a další citlivá data, což umožňuje použití této platformy pro aplikace, které vyžadují bezpečnou komunikaci a ukládání dat. Díky těmto bezpečnostním funkcím a možnostem správy paměti je ESP32 ideální pro širokou škálu aplikací, včetně IoT zařízení, průmyslové automatizace a spotřební elektroniky [9].

Kromě toho ESP32 obsahuje také hardwarové akcelerátory pro kryptografii AES a SHA, bezpečné spouštění a bezpečné úložiště pro kryptografické klíče a další citlivá data. Díky

tomu je platforma ESP32 vhodná pro aplikace, které vyžadují bezpečnou komunikaci a ukládání dat [9].

Celkově je architektura ESP32 navržena tak, aby poskytovala vysoký výkon a efektivitu, s bohatou sadou vestavěných periférií a širokou škálou možností komunikace a připojení. Díky bezpečnostním funkcím a možnostem správy paměti je vhodný pro širokou škálu aplikací, včetně zařízení internetu věcí (IoT), průmyslové automatizace a spotřební elektroniky.



Obrázek 4.1: Funkční blokový diagram ESP32, převzato z [9]

## 4.2 Dostupné moduly

Modul je malá deska plošných spojů, která obsahuje SoC od společnosti Espressif Systems, flash paměť (do které se nahrává firmware), oscilátor pro generování RTC (hodiny reálného času) signálu a anténu (buď integrovanou v desce plošných spojů, nebo možnost připojit anténu formou IPEX konektoru), která slouží pro bezdrátovou komunikaci. Společnost Espressif Systems nabízí širokou škálu modulů navržené pro různé aplikace (podle aplikace se liší schopnosti modulu). V tabulce A.1 je možné vidět porovnání vybraných dostupných modulů.

## 4.3 FreeRTOS

FreeRTOS je open-source operační systém reálného času (RTOS), který je určený pro vestavěné systémy a mikrokontrolery. FreeRTOS je navržený pro použití s mikrokontrolery, u kterých je důležitá nízká spotřeba energie a minimální režie při zpracovávání úloh. Dále existuje sesterský projekt SafeRTOS, který zahrnuje certifikace pro lékařské, automobilové a průmyslové odvětví [17].

FreeRTOS poskytuje rozhraní pro vytváření nezávislých úloh (task), které se mohou provádět současně. Každá úloha může mít vlastní prioritu, přidělenou paměť i dobu provádění úlohy. Dále poskytuje rozhraní pro správu jednotlivých úloh, tedy mechanismy pro komunikaci mezi úlohami, jako jsou fronta (queue), semaforey a mutexy, které umožňují sdílení dat a synchronizaci [17].

Platforma ESP32 používá operační systém ESP-IDF FreeRTOS, což je implementace systému FreeRTOS založená na Vanilla FreeRTOS v10.4.3, která obsahuje významné úpravy



v aplikačním rozhraní a přidanou podporu SMP (Symmetric Multiprocessing) u dvoujádrových čipů a modulů ESP32 (ESP32 a ESP32-C3) [5].

## 4.4 Tvorba firmware

Cest, jak vytvořit a následně nahrát firmware na platformu ESP32 je více, pro tvorbu firmware nejjednodušší je použít jeden z dostupných aplikačních rámců (frameworků) v kombinaci s některým podporovaným vývojovým prostředím (IDE – Integrated Development Environment [5]).

Mezi programovací jazyky, ve kterých je možné firmware vytvořit se řadí hlavně jazyky *C* a *C++*. Další možností je tvorba firmware pomocí jazyka *MicroPython* (implementace jazyka, která je z velké části kompatibilní s jazykem Python 3.4 a optimalizovaná pro mikrokontrolery<sup>1</sup>), tato možnost sice přináší výhody vyššího programovacího jazyka, ale *MicroPython* je méně používaný, a tedy existuje pro něj méně dostupných materiálů.

### Framework Arduino-ESP32

Framework Arduino-ESP32 podporuje integraci do vývojových prostředí jako jsou Arduino IDE, nebo PlatformIO IDE. Výhodou frameworku Arduino-ESP32 je, že poskytuje kompatibilitu s knihovnými funkcemi, které jsou dostupné pro platformu Arduino (jednoděskový počítač založený na mikrokontrolerech ATmega od firmy Atmel), jako jsou například *Wire*, *SPI*, nebo *SD* [6].

### Framework ESP-IDF

ESP-IDF (Espressif IoT Development Framework) je vývojový rámec pro mikrokontrolery řady ESP32 a ESP32-S společnosti Espressif Systems. Mikrokontrolery vydané před rokem 2016 (ESP8266 a ESP8285) používají vývojový rámec RTOS SDK. Vývojový rámec ESP-IDF používá FreeRTOS jako základní operační systém reálného času (RTOS) pro správu plánování úloh a zdrojů v mikrokontroleru [5].

ESP-IDF poskytuje sadu knihoven a nástrojů pro vývoj aplikací pro mikrokontrolery. Také obsahuje sadu ovladačů pro běžné periferie, jako jsou WiFi, Bluetooth a senzory. Kromě toho poskytuje podporu pro oblíbená vývojová prostředí, jako jsou Eclipse a Visual Studio Code [5].

Jednou z klíčových výhod použití ESP-IDF se systémem FreeRTOS je, že poskytuje bohatou sadu aplikačních rozhraní pro správu a plánování úloh a také pro správu komunikace a synchronizace mezi úlohami. To usnadňuje vývoj aplikací v reálném čase, které vyžadují přesné načasování a koordinaci mezi úlohami [5].

### Ukládání dat

Pro ukládání dat do flash paměti, která je nevolatilní (paměť při ztrátě napájení uchovává informace), lze použít souborový systém SPIFFS (SPI Flash File System). Tento souborový systém je určený pro vestavěné zařízení s SPI NOR flash pamětí. Po inicializaci souborového systému SPIFFS lze pro práci se soubory použít standardní funkce jazyka C [1].

---

<sup>1</sup><https://github.com/micropython/micropython>

## **IDF komponenty**

IDF komponenty jsou modulární části samostatného kódu, které jsou zkompileovány do statických knihoven (soubory s příponou *.a*) a propojeny do aplikace. Některé z nich poskytují samotný framework ESP-IDF, jiné mohou pocházet z jiných zdrojů [5].



## Kapitola 5

# Zobrazovací zařízení pro vestavěné systémy

Zobrazovací zařízení jsou důležitou součástí vestavěných systémů a umožňují uživateli interakci s daným zařízením pomocí grafického rozhraní. Platforma ESP32 je jedním z populárních mikrokontrolerů využívaných ve vestavěných systémech a nabízí mnoho možností pro připojení zobrazovacích zařízení.

Platforma ESP32 podporuje různé typy zobrazovacích zařízení, jako jsou OLED displeje, TFT displeje, LCD displeje a e-paper displeje. ESP32 má dále vestavěnou podporu pro širokou škálu rozhraní pomocí kterých lze připojit displej, mezi tato rozhraní patří například I2C, SPI, nebo UART. Každé zobrazovací zařízení může mít jiné specifikace a způsob ovládání, toto je potřeba zohlednit při výběru grafické knihovny a při návrhu grafického rozhraní, a tedy i při výběru zobrazovacího zařízení pro danou aplikaci.

Komunikace mezi mikrokontrolerem a řadičem displeje může být realizována různými způsoby a závisí na použitém typu displeje a jeho řadiči. K tomuto účelu jsou dostupné knihovny, které usnadňují ovládání displeje. Tyto knihovny jsou většinou tvořeny pro použití s konkrétním řadičem.

V následující sekci budou představeny některé typy zobrazovacích zařízení, které jsou kompatibilní s platformou ESP32.

### LCD displej

LCD (Liquid Crystal Display) je typ displeje, který k vytvoření obrazu používá tekuté krystaly. Tyto krystaly jsou vloženy mezi dvě vrstvy polarizačního materiálu a po přivedení elektrického proudu se natočí tak, aby světlo propouštěly nebo blokovaly. Přivedením elektrického proudu na jednotlivé pixely displeje lze vytvářet pokročilejší obrazy.

Mezi výhody LCD displejů patří dostatečně malá spotřeba pro použití v přenosných zařízeních. Dále je tento druh displejů je dostupný ve velké škále možných velikostí, tvarů a rozlišení, což je činí vhodnou volbou pro celou řadu aplikací. Dále jsou LCD displeje schopny zobrazit plné barvy, což je činí ideálními pro aplikace, kde jsou barvy důležité. Tyto displeje jsou využívány například v televizích, počítačových monitorech a digitálních hodinkách.

Mezi nevýhody tohoto typu displejů patří omezené pozorovací úhly, což znamená, že při pohledu na displej z určitých úhlů se může obraz jevit, jako ztmavený, nebo zkreslený. Dále tento typ displejů trpí takzvaným *motion blur*, tedy k rozmazanému obrazu, nebo nežádoucím efektům na obrazu, při rychle pohybujících se objektech v obraze.

## E-paper displej

E-paper displej je typ displeje, který byl navržen tak, aby napodoboval vzhled inkoustu na papíře. Displej používá k vytvoření obrazu mikro kapsle naplněné nabitými částicemi a kapalinou. Nabité částice se po vystavení elektromagnetickému poli přeuspořádají a způsobí, že se mikro kapsle jeví, jako černá, nebo bílá. Tento jev na displeji vytváří požadovaný obraz.

Jednou z výhod e-paper displejů je jejich nízká spotřeba energie, které dosahují díky tomu, že k udržení obrazu nevyžadují konstantní přísun energie, což je ideální pro zařízení napájené baterií, které jsou určeny k dlouhodobému používání na jedno nabití (elektronické čtečky knih atd.). Dále je tento typ displejů typicky používán v aplikacích, kde je důležitá dobrá čitelnost.

Mezi nevýhody tohoto typu displejů patří relativně nízká obnovovací frekvence, kterou má e-paper displej oproti ostatním druhům displejů. Toto dělá e-paper displej nevhodným řešením pro zobrazování animací, nebo videa.

## 5.1 Grafické knihovny pro vestavěné systémy

Grafické knihovny pro vestavěné systémy jsou navrženy s ohledem na omezení (nižší výpočetní výkon a paměťové omezení) vestavěných zařízení. Tyto knihovny poskytují nástroje, které usnadňují tvorbu grafického rozhraní pro vestavěné zařízení. Některé knihovny umožňují využít WYSIWYG (What You See Is What You Get) editoru, který oproti psaní kódu s uživatelským rozhraním umožňuje hned vidět výsledek, což značně usnadní práci (například při drobných úpravách umístění objektů).

V následující sekci budou představeny grafické knihovny, které mohou být použity v kombinaci s platformou ESP32.

### LVGL

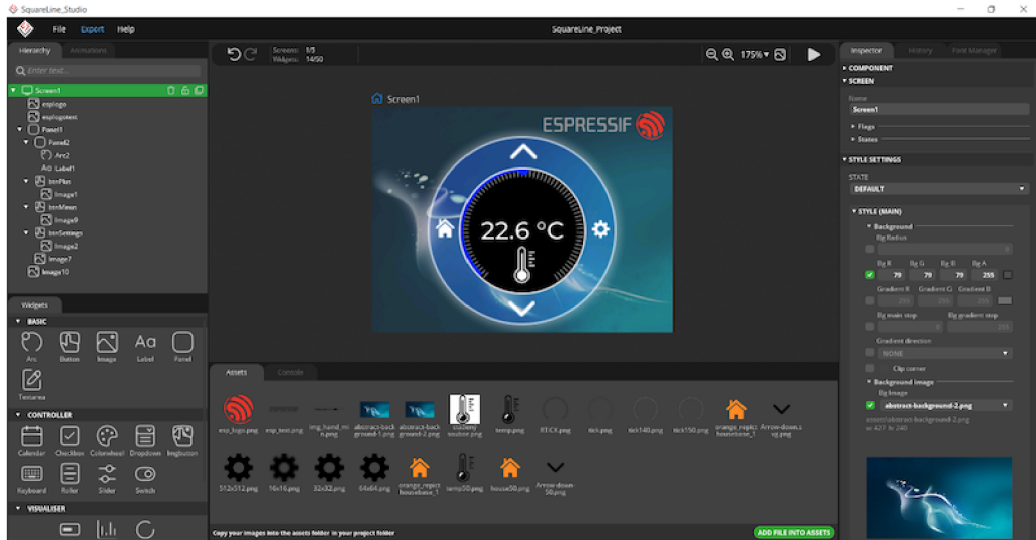
LVGL (Light and Versatile Graphics Library) je open source (software s otevřeným zdrojovým kódem) grafická knihovna, s rozsáhlou dokumentací, která umožňuje vytvářet uživatelská rozhraní pro vestavěné systémy, pomocí jazyka *C*, *C++*, nebo *Micropython*, bez vnějších závislostí, která je zdarma i pro komerční účely. LVGL je podporována předními výrobci, jako jsou například Arm, STM32, NXP, Espressif, Arduino, Adafruit a mnoho dalších. Knihovna LVGL podporuje monochromatické, e-paper, OLED, i LCD displeje, které mohou mít i dotykovou vrstvu. Z hlediska paměťové náročnosti potřebuje knihovna LVGL 32 kB RAM a 128 kB Flash paměti [11].

LVGL poskytuje širokou škálu grafických prvků, jako jsou tlačítka, textová pole, grafy, obrázky a mnoho dalšího, které mohou být použity pro vytvoření moderního uživatelského rozhraní. Knihovna LVGL také nabízí pokročilé funkce, jako jsou animace, podpora dotykových obrazovek a možnost přizpůsobení vzhledu GUI [11].

Knihovna LVGL byla navržena tak, aby byla snadno použitelná a přizpůsobitelná různým platformám a architekturám. Podporuje různé grafické adaptéry, jako jsou framebuffer, SDL, OpenGL a mnoho dalších. LVGL také obsahuje mnoho ukázkových projektů, které mohou být použity jako výchozí bod pro vývoj GUI ve vestavěných systémech [11].

Grafické rozhraní lze tvořit buď popsáním grafických objektů v kódu manuálně, nebo pomocí WYSIWYG editoru uživatelského rozhraní *SquareLine Studio*, za pomoci kterého lze kód vygenerovat. Popis grafických objektů lze tvořit za pomoci jazyků *C*, *C++*, nebo *Micropython* [11].

Díky své flexibilitě, široké funkčnosti a podpoře různých platform je knihovna LVGL stále populárnější volbou pro vývoj moderních uživatelských rozhraní ve vestavěných systémech a mikrokontrolerech. Další výhodou LVGL je jeho aktivní komunita a dokumentace. LVGL má rozsáhlou dokumentaci a online fóra, kde mohou vývojáři najít rady a odpovědi na své otázky. Komunita LVGL také pravidelně aktualizuje knihovnu a přidává nové funkce a vylepšení.



Obrázek 5.1: Ukázka editoru SquareLine Studio, převzato z [7]

## uGFX

uGFX je také open source grafická knihovna, implementovaná v jazyce C, ale může být použita i v projektu v C++. Dokumentace je Knihovna je zdarma pro nekomerční účely. Pro komerční účely je potřeba jednorázově zakoupit licenci. uGFX podporuje monochromatické, e-paper, OLED, nebo LCD displeje, které mohou mít i dotykovou vrstvu [20].

Knihovna uGFX také podporuje tvorbu grafického rozhraní buď popsáním grafických objektů v kódu, nebo pomocí WYSIWYG editoru *μGFX-Studio*, který je aktuálně dostupný pouze v beta verzi.

## FabGL

FabGL je také open source grafická knihovna, která je vytvořená primárně pro platformu ESP32, implementovaná v jazyce C++. Knihovna podporuje LCD a OLED displeje připojené pomocí I2C, SPI. Navíc ještě knihovna podporuje displeje připojené pomocí VGA. Oproti výše zmíněným knihovnám, FabGL nepodporuje e-paper displeje. FabGL nepodporuje displeje s dotykovou vrstvou, ale podporuje vstupy z PS/2 klávesnice a myši. Grafické rozhraní pro knihovnu FabGL lze tvořit pouze popsáním grafických objektů v kódu [24].

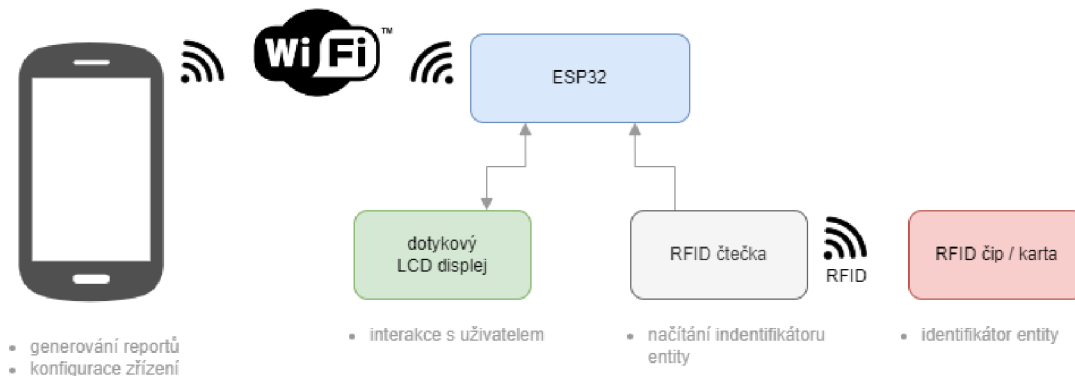
Knihovna FabGL byla navržena s hlavním záměrem emulace starších počítačových platforem, jako je například IBM PC. Pro tyto účely poskytuje knihovna třídy pro emulaci zařízení, například *Intel 8080 CPU*, *MOS 6502 CPU*, nebo *VIA 6522* [24].

## Kapitola 6

# Návrh prototypu

Cílem zařízení je umožnit registraci objektů zobrazených na displeji k určité entitě. Identifikace entity bude realizována pomocí technologie RFID. Zařízení musí dále obsahovat podporu pro administraci, za použití bezdrátové technologie (Bluetooth nebo WiFi). Na základě těchto požadavků jsem navrhl koncept zařízení znázorněný v blokovém diagramu na obrázku 6.1.

Blokový diagram znázorňuje propojení jednotlivých částí vestavěného zařízení pro potřeby registrace. Zařízení se skládá ze 3 hlavních částí. Dotykový displej je částí zařízení, která má na starosti zobrazení uživatelského rozhraní a interakci s uživatelem. Další částí zařízení je RFID čtečka, která má na starosti čtení identifikátoru entity. Další částí zařízení je mikrokontroler ESP32, který má na starosti komunikaci s displejem a RFID čtečkou, ukládání dat registrací a správu komunikace webové administrace.



Obrázek 6.1: Návrh blokového diagramu vestavěného zařízení

## Uživatelské rozhraní

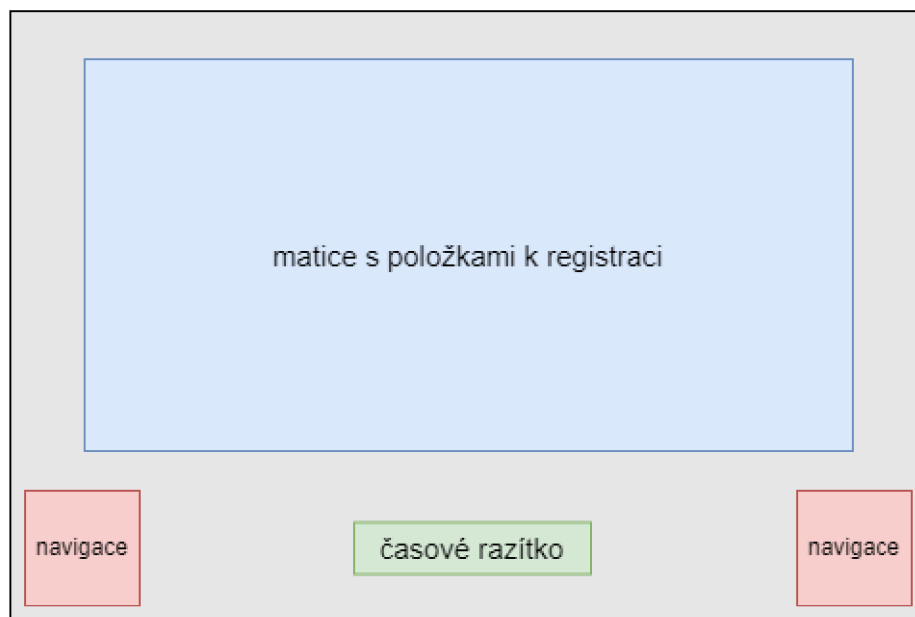
Z grafických knihoven zmíněných v kapitole 5 byla v rámci kapitoly identifikována jako nejvýhodnější varianta knihovna LVGL, a proto bude využita pro implementaci grafického rozhraní. Jednou z největších výhod je snadná integrace v prostředí ESP32 a rozsáhlá dokumentace.

Interakce se zařízením bude realizována, pomocí dotykového displeje. Uživatel bude moci přepínat mezi obrazovkami s předem definovanými objekty k registraci. Speciálním přípa-

dem je obrazovka s nastavením, kde bude zobrazen přepínač pro povolení portálu s webovou administrací. Po vybrání objektu se zařízení přepne na obrazovku pro identifikaci entity a po úspěšné identifikaci entity se zařízení přepne na obrazovku s informací o potvrzení provedené registrace. Uživatelské rozhraní bylo navrženo univerzálně tak, aby s ohledem na spotřebu bylo možné zaměnit zobrazovací prvek. Přibližné rozložení uživatelského rozhraní lze vidět na obrázku 6.2, který obsahuje wireframe uživatelského rozhraní.

Tlačítka s položkami k registraci budou uspořádány do matice o rozměrech, které si uživatel nadefinuje v administračním portálu. Dále bude uživatelské rozhraní obsahovat prvek, který bude zobrazovat aktuální čas. Dalšími prvky jsou navigační tlačítka (na obrázku 6.2 objekty navigace), která budou sloužit k přepínání mezi jednotlivými obrazovkami. Pomocí těchto navigačních tlačítek, bude uživatel moci přepínat mezi obrazovkami s položkami k registraci a také se bude moci dostat na obrazovku **nastavení**. Obrazovka nastavení bude obsahovat přepínač pro zapnutí administračního portálu.

Tedy celý postup registrace z hlediska uživatelského rozhraní je následující. Uživatel si na dotykovém displeji vybere položku, kterou chce registrovat. Poté se na displeji vykreslí obrazovka znázorňující časový interval pro přiložení RFID tagu a nakonec, pokud byla registrace úspěšná, se zobrazí obrazovka se zprávou indikující úspěšné provedení registrace. Obrazovka s indikací úspěšné registrace bude zároveň obsahovat informaci o provedených registracích uživatele za určitý časový interval.



Obrázek 6.2: Návrh uživatelského rozhraní vestavěného zařízení

Vzhledem k tomu, že objekty k registraci budou definovány v administrátorském portálu, tak je možné všechny obrazovky s objekty generovat. Uživatel si ve webové administraci určí všechny objekty, kolik jich bude na jedné obrazovce ve sloupci a kolik v řádku. S těmito informacemi lze podle vzorce 6.1 spočítat výsledný počet obrazovek, které je potřeba vygenerovat. Velikost všech tlačítek objektů bude stejná a řídí se hodnotou zadanou uživatelem v administračním portálu.

$$\text{počet obrazovek} = \frac{\text{celk. počet tlačítek}}{\text{počet tl. v řadě} * \text{počet tl. ve sloupci}} \quad (6.1)$$

## Identifikace entity

Pro identifikaci entity bude využita technologie RFID, konkrétně pro tuto aplikaci byl zvolen pasivní tag používající ke komunikaci frekvenci 13,56MHz. Pasivní tag byl zvolen, protože v této aplikaci nejsou zapotřebí senzory umístěné na tagu, ani vylepšená čtecí vzdálenost potřeba. Komunikační frekvence 13.56MHz byla zvolena kvůli dostačujícím parametrům vzdálenosti čtení a nízké pořizovací ceně (tagů i čteček).

Entita se po vybrání objektu na displeji identifikuje pomocí RFID tagu, následně bude v zařízení vytvořen a uložen záznam o registraci.

## Hodiny reálného času

Vzhledem k tomu, že systém bude ukládat informace o registraci s časovým razítkem, je důležité získat a udržet v zařízení aktuální čas. Pro získání aktuálního času bude použit protokol *SNTP*. Modul ESP32 obsahuje několik oscilátorů, které lze použít jako zdroj RTC signálu, liší se hlavně přesností a spotřebou. Kvůli tomu, že každý oscilátor má určitou chybovost, proto bude potřeba periodicky synchronizovat aktuální čas pomocí protokolu *SNTP*.

## Ukládání dat

V systému bude potřeba ukládat dva druhy dat, data o provedených registracích a aplikační data. Aplikační data bude potřeba uchovávat i po vypnutí zařízení, proto musí být uložena v nevolatilní flash paměti.

## Databáze

Pro ukládání informací o registracích bude použita databáze FlashDB, která je navržena pro použití ve vestavěných zařízeních. FlashDB poskytuje dva režimy ukládání dat režim key-value (klíč-hodnota – primárně pro ukládání parametrů, nebo konfigurací) a režim time-series (časová série – primárně pro ukládání strukturovaných dat, například data ze senzorů teploty, vlhkosti vzduchu atd.) [2].

Ukládání obrázků položek k registraci pro grafický režim vestavěného zařízení pro potřeby registrace byl zvolen souborový systém SPIFFS, který umožňuje ukládat soubory na flash paměť a potom je možné provádět operace se soubory uloženými v tomto souborovém systému pomocí standardních funkcí pro operace se soubory jazyka C.

Pro tuto aplikaci lze využít obou režimů ukládání dat, režim časové série je vhodné použít pro ukládání jednotlivých záznamů o registracích, kde je potřeba ukládat časové razítko, název objektu a identifikátor entity. Režim key-value je vhodný pro ukládání dat z administračního portálu (například: počet tlačítek v řadě, počet tlačítek ve sloupci atd.), tady je potřeba ukládat pouze klíč a jemu přiřazenou hodnotu.

## Webová administrace

Webová administrace bude realizovaná pomocí komponenty webového serveru dostupné na platformě ESP32. Uživatel bude moci v tomto administračním portálu definovat objekty k registraci, konfigurovat parametry zařízení a generovat reporty za specifikované časové období.



Na obrázku 6.3 lze vidět, že administrační portál se bude dělit na 3 části (karty): domovskou stránku, konfigurace zařízení a generování reportů. Po připojení zařízení do administračního portálu se uživateli zobrazí domovská stránka, která obsahuje krátký popis vysvětlující funkčnost zařízení. Z domovské stránky se bude moct uživatel dostat do všech ostatních částí administračního portálu. Na kartě s konfigurací zařízení bude jednoduchý formulář, který bude obsahovat název parametru spolu s textovým polem sloužícím pro zobrazení aktuální hodnoty, nebo změnu aktuální hodnoty.

Pokud uživatel bude chtít provádět některé z akcí, které je možné provést ve webové administraci, tak v nastavení na vestavěném zařízení povolí přístupový bod (přístupový bod není povolen pořad, kvůli úspoře energie). K tomuto přístupovému bodu připojí uživatel nějaké zařízení (notebook, tablet, nebo mobilní telefon) a poté zadá ve webovém prohlížeči IP adresu vestavěného zařízení.

Home Items Data

**Screen configuration:**

Number of rows per screen:

Number of columns per screen:

Button width (px):

Button height (px):

Items:

**Device configuration:**

Sleep interval between time synchronizations (ms):

SSID of network used for time synchronization:

Password of network used for time synchronization:

Administration portal timeout (ms):

SSID of administration portal:

Password of administration portal:

Graphical mode enable:

Sleep mode enable:

Obrázek 6.3: Návrh uživatelského rozhraní konfigurace zařízení v administračním portálu

# Kapitola 7

## Implementace

Z vývojových frameworků zmíněných v kapitole 4.4 byl v rámci kapitoly identifikována jakko nejvýhodnější varianta framework ESP-IDF.

### Struktura programu

U rozvržení jednotlivých souborů a složek projektu jsem se rozhodl použít výchozí rozvržení projektu použité frameworkem ESP-IDF. Projekt má tedy následující rozložení:

- `build` – obsahuje pomocné soubory vygenerované během sestavování a výsledný binární soubor
- `components` – obsahuje použité IDF komponenty
- `main` – obsahuje soubor `main.c` (vstupní bod projektu) a další soubory se zdrojovým kódem programu

Zdrojový kód programu je rozdělen do souborů podle jednotlivých funkčních celků, které jsou popsány v následujících částech této kapitoly.

### 7.1 Uživatelské rozhraní

Inicializace knihovny LVGL je realizována pomocí tasku, který také obstarává inicializaci ovladačů displeje a zpracování událostí (překreslení obrazovky atd.). Tento task byl převzat ze vzorového projektu na použití knihovny LVGL a následně upraven pro použití. Jelikož má každý prototyp různý displej, je potřeba použít pro každý displej jiné ovladače. Pro e-paper displej byl zvolen ovladače displeje ze sady `lvgl_epaper_drivers`<sup>1</sup> a pro LCD prototyp byl použit ovladač displeje ze sady `lvgl_esp32_drivers`<sup>2</sup>

Jelikož knihovna LVGL nepodporuje ukládání dodatečných dat do struktur objektů uživatelského rozhraní, bylo potřeba implementovat vlastní datové struktury pro tyto účely. Datové struktury `screen_t` a `button_t` jsou jednosměrné lineární seznamy, které jsou implementovány v souboru `uiData.c`, včetně pomocných funkcí pro práci s nimi. Tyto struktury slouží pro ukládání dat potřebných pro navigaci uživatelským rozhraním zařízení a registraci objektu k entitě.

---

<sup>1</sup>[https://github.com/martinberlin/lvgl\\_epaper\\_drivers](https://github.com/martinberlin/lvgl_epaper_drivers)

<sup>2</sup>[https://github.com/lvgl/lvgl\\_esp32\\_drivers](https://github.com/lvgl/lvgl_esp32_drivers)



Zařízení zobrazuje položky k registraci formou tlačítka s textem (názvem položky), který je uchovávan v databázi. Zařízení také podporuje zobrazení položek k registraci ve formátu tlačítka s obrázkem, kde si uživatel nadefinuje obrázky, které budou zobrazeny jako položky. Tyto obrázky uživatel umístí do složky `spiffs` a s nahráním firmware jsou umístěny do SPIFFS oddílu na mikrokontroleru ESP32. V tomto režimu se do záznamu o registraci použije název souboru obrázku.

Obrázky pro navigační tlačítka a pro testování grafického režimu vestavěného zařízení pro potřeby registrace byly použity *Material Icons*<sup>3</sup>, které jsou dostupné pod licencí Apache 2.0

Jelikož knihovna LVGL ve verzi, která byla portovaná pro ESP32 (verze 7.11) nepodporuje obrázky ve formátu PNG, bylo zapotřebí při implementaci použít knihovnu `lv_lib_png`<sup>4</sup> (LVGL implementace knihovny `lodepng`), která umožňuje použití obrázků ve formátu PNG s knihovnou LVGL.

Samotné uživatelské rozhraní je implementované v souboru `ui.c`. Uživatelské rozhraní je generováno nově podle dat vyčtených z databáze po každém zapnutí zařízení, toto probíhá zavoláním inicializační funkce `initViews`, která inicializuje všechny potřebné styly grafických prvků a následně zavolá funkci `initUI`, která podle dat z databáze vygeneruje potřebný počet obrazovek (podle vzorce 6.1), které podle předvolby naplní buď objekty vygenerovanými podle obrázků uložených v systému SPIFFS, nebo objekty vygenerovanými podle řetězce uloženého v databázi.

Při generování uživatelského rozhraní jsou objekty s položkami vkládány na jednotlivé obrazovky dokud nedojde k zaplnění obrazovky, podle uživatelem definovaných počtů řad a sloupců, poté se začne plnit další obrazovka, dokud nedojde k vygenerování objektů pro všechny požadované položky.

Výsledné uživatelské rozhraní lze vidět na obrázku 7.1.



Obrázek 7.1: Ukázka uživatelského rozhraní na prototypu s e-paper displejem

<sup>3</sup><https://fonts.google.com/icons?icon.set=Material+Icons>

<sup>4</sup>[https://github.com/lvgl/lv\\_lib\\_png](https://github.com/lvgl/lv_lib_png)

## 7.2 Ukládání dat

Podle druhu ukládaných dat lze rozdělit kapitolu na několik částí:

### Konfigurační data

Pro ukládání konfiguračních dat byla použita databáze FlashDB, konkrétně režim KVDB. Knihovna pro interakci s KVDB byla implementována v souboru `db.c`. Databáze je inicializovaná vždy po zapnutí zařízení funkcí `db_init`, jejíž volání se nachází ve funkci `app_main` v souboru `main.c`

Hodnoty konfiguračních dat lze měnit v administračním portálu. V případě poškození integrity dat, lze obnovit výchozí hodnoty zavoláním funkce `db_firstInit`. Mezi ukládaná konfigurační data patří:

- `buttons` – řetězec s položkami k registraci
- `btn_rows` – počet řad tlačítek na 1 obrazovce
- `btn_cols` – počet sloupců tlačítek na 1 obrazovce
- `btn_width` – šířka tlačítka
- `btn_height` – výška tlačítka
- `sntp_timer` – časový interval, ve kterém zařízení periodicky získává aktuální čas
- `ap_timeout` – časový interval, po kterém se administrační portál automaticky vypne
- `ap_ssid` – SSID pro přístupový bod administračního portálu
- `ap_pwd` – heslo k přístupovému bodu administračního portálu
- `sta_ssid` – SSID přístupového bodu pro získávání aktuálního času
- `sta_pwd` – heslo k přístupovému bodu pro získávání aktuálního času

### Záznamy o provedených registracích

Pro ukládání záznamů o provedených registracích byla použita databáze FlashDB, konkrétně režim TSDB. Knihovna pro interakce s TSDB byla implementována v souboru `db.c`. Jak bylo zmíněno v kapitole 6, o provedených registracích jsou ukládány tyto data:

- *časové razítko*
- *identifikátor entity*
- *identifikátor objektu*

Velikost časového razítka je daná velikostí datového typu `fdb_time_t`, který je definovaný jako `int32_t`, tedy na platformě ESP32 má velikost 4 B. Velikost identifikátoru entity byla zvolena na 7 B, protože RFID tagy použitého typu (ISO14443A) mají identifikátor dlouhý 4 B, nebo 7 B (podle typu karty). Identifikátor objektu je do databáze vkládán jako řetězec, jelikož na platformě ESP32 má datový typ `char` velikost 1 B lze do identifikátoru objektu o velikosti 20 B zahrát řetězec o délce až 20 znaků, což by měla být dostačující

délka pro identifikátor objektu.

Pro FlashDB je vyhrazen oddíl, který má, ve výchozím stavu, velikost 12582912 B (12 MiB). Podle výpočtu 7.1 získáme celkový počet záznamů, které může vestavěné zařízení pro potřeby registrace uchovávat. Pokud odečteme od výchozí velikosti oddílu vyhrazeného pro databázi, velikost části oddílu, kterou zabírá KVDB, což je 16384 B, získáme použitelnou paměť pro TSDB o velikosti 12566528 B. Velikost jednoho záznamu v TSDB má velikost 31 B. Data FlashDB potřebná pro režii jsou zhruba 6 kiB na 30000 záznamů. Tedy teoreticky bez režie může zařízení uchovat až 405371 záznamů. Z toho vyplývá, že pro režii bude potřeba zhruba 83020 B paměti. Tedy výsledná velikost paměti pro ukládání záznamů o registraci je 12483507 B, což odpovídá zhruba **402693** záznamům.

$$\text{počet záznamů} = \frac{\text{velikost oddílu} - \text{data KVDB} - \text{režie TSDB}}{\text{velikost záznamu v TSDB}} \quad (7.1)$$

Pokud budeme předpokládat použití zařízení pro potřeby registrace v prostředí středního podniku, který potřebuje registrovat zhruba 100 položek denně. Tak podle výpočtu 7.2 by zařízení mělo mít dostatečnou paměť na necelých 16 let provozu.

$$\frac{402693 \text{ záznamů} / 100 \text{ záznamů na den}}{252 \text{ prac. dní v roce}} = 15,980 \text{ let} \quad (7.2)$$

### Obrázky položek k registraci

K ukládání obrázků položek k registraci byl použit souborový systém SPIFFS. Pro operace s těmito daty nebyla potřeba vytvořit žádná knihovna, protože ESP-IDF FreeRTOS poskytuje aplikační rozhraní pro inicializaci souborového systému SPIFFS. Inicializace souborového systému SPIFFS probíhá vždy po zapnutí zařízení funkcí `fs_init`, jejíž volání se nachází ve funkci `app_main` v souboru `main.c` Po inicializaci se pro práci se soubory používají standardní funkce jazyka C. Obrázky jsou ukládány do systému SPIFFS při nahrávání firmware do zařízení.

Pro systém SPIFFS je vyhrazen oddíl o velikosti 1048576 B (1 Mib), ale po inicializaci souborového systému SPIFFS zbývá volné místo pro soubory o velikosti 956561 B. Testovací obrázek o velikosti 96 x 96 pixelů zabírá paměť o velikosti 277 B. Pokud použijeme tuto hodnotu pro výpočet, dostaneme počet 3453 položek, které je teoreticky možné do oddílu uložit.

## 7.3 Identifikace entity

K identifikaci entity byla použita technologie RFID, konkrétně vývojová deska s vestavěnou anténou a s čipem PN532 od společnosti NXP. Tato deska umožňuje čtení a zápis dat z RFID zařízení na frekvenci 13,56 MHz. Komunikace mezi deskou a platformou ESP32 probíhá po sběrnici I2C. Jelikož je čip PN532 poměrně hojně používaný, bylo možné pro komunikaci s ním použít open source knihovnu `esp-pn532`<sup>5</sup>, což je port knihovny *Adafruit PN532* (která byla navržena pro platformu Arduino) na platformu ESP32. Inicializace knihovny a

<sup>5</sup><https://github.com/lucafacchin/esp-pn532>

komunikace mezi platformou ESP32 a RFID čtečkou je implementována v tasku `rfid_task`, který se nachází v souboru `rfidReader.c`.

Komunikace mezi taskem zajišťujícím vyčítání identifikátoru entity (`rfid_task`) a taskem zajišťujícím obsluhu GUI (`guiTask`) je realizována pomocí aplikačního rozhraní *FreeRTOS front*. Pro tyto účely byla vytvořena struktura `message_t`, která obsahuje vyčtený identifikátor entity a délku identifikátoru. Po vyčtení identifikátoru entity je naplněna struktura `message_t` a přidána do fronty pomocí makra `xQueueSendToBack`. V tasku `guiTask` se periodicky kontroluje, zda ve frontě přibyla nějaká zpráva, pomocí makra `xQueueReceive`. Pokud ve frontě zpráva přibyla, tak se zpráva zpracuje, tedy vyčtou se z ní data a uloží se do databáze záznam o registraci. Nakonec se uživateli zobrazí potvrzení o jeho provedené registraci. V rámci této obrazovky s potvrzením provedení registrace je uživateli zobrazena informace o provedených registracích uživatele v rámci týdne. Tedy z databáze se vyfiltrují registrace spojené s uživatelem od začátku týdne, až do aktuální registrace (včetně), jsou spočítány a následně prezentovány po úspěšně provedené registraci.

## 7.4 Administrační portál

Administrační portál je realizován pomocí technologie WiFi, tedy pomocí webserveru hostovaného na platformě ESP32. Po zapnutí administračního portálu v nastavení na vestavěném zařízení, je zapnuta WiFi a následně zapnut mód přístupového bodu. Uživatel se poté může na přístupový bod připojit pomocí libovolného zařízení, které má schopnost se připojit k přístupovému bodu a zároveň je schopné zobrazit webové stránky, například mobilní telefon, nebo notebook. Uživatel potom zadá IP adresu vestavěného zařízení, tedy: `192.168.4.1` a načtou se mu webové stránky hostované na zařízení, kde může konfigurovat vestavěné zařízení a generovat reporty o provedených registracích. Po zapnutí administračního portálu se spustí časovač, jehož hodnota je nastavitelná v administračním portálu pod hodnotou `ap_timeout`. Po doběhnutí časovače se administrační portál automaticky vypne, kvůli úspoře energie.

Implementace administračního portálu se nachází v souboru `webserver.c`. Platforma ESP32 nabízí řadu knihovných funkcí pro implementaci webserveru, což značně zjednodušuje integraci webserveru do aplikace. Pro přenos dat mezi vestavěným zařízením a zařízením, ze kterého je prováděna administrace, bylo zvoleno aplikační rozhraní REST. Na vestavěném zařízení jsou registrované URI handlers pro jednotlivé požadavky.

### Konfigurace zařízení

Uživatel může v administračním portálu konfigurovat jednotlivé aspekty fungování zařízení. Jak je vidět na obrázku 7.2, nastavení se skládá z popisku parametru a textového pole, kde má uživatel možnost změnit hodnotu parametru. Mezi možnosti konfigurace obrazovky patří volba, jak bude vypadat vygenerované uživatelské rozhraní, pomocí nastavení počtu řádků a sloupců tlačítek objektů na jedné obrazovce, nebo rozměry tlačítek objektů na vestavěném zařízení. Dále si uživatel může nastavit k jakému přístupovému bodu se bude vestavěné zařízení připojovat pro získání aktuálního času, pomocí protokolu SNTP. Toto nastavení je v administračním portálu pod položkou `SSID of network used for time synchronization`, kde uživatel zadá SSID přístupového bodu a pod položkou `Password of network used for time synchronization`, kde uživatel zadá heslo k přístupovému bodu.



The screenshot shows a web-based configuration interface for a device. At the top, there are navigation tabs: 'Home', 'Items', and 'Data'. The main content is divided into two sections:

- Screen configuration:**
  - Number of rows per screen:
  - Number of columns per screen:
  - Button width (px):
  - Button height (px):
  - Items:
- Device configuration:**
  - Sleep interval between time synchronizations (ms):
  - SSID of network used for time synchronization:
  - Password of network used for time synchronization:
  - Administration portal timeout (ms):
  - SSID of administration portal:
  - Password of administration portal:
  - Graphical mode enable:
  - Sleep mode enable:

At the bottom of the configuration area, there is a 'Save' button.

Obrázek 7.2: Ukázka konfigurace v administračním portálu vestavěného zařízení

Tyto změny konfigurace vychází z ukládaných dat v KVDB zmíněných výše. Po stisknutí tlačítka *save* se vytvoří POST požadavek, který je následně na vestavěném zařízení zpracován. Po přijetí POST požadavku ze stránky s konfigurací zařízení, jsou jednotlivé hodnoty zadané uživatelem porovnány s hodnotami dat v KVDB databázi a pokud byly hodnoty změněny jsou staré hodnoty v KVDB databázi nahrazeny novými hodnotami a následně reflektovány v konfiguraci zařízení (překreslení uživatelského rozhraní atd.).

## Generování reportů

Uživatel může také v administračním portálu generovat reporty z provedených registrací. Pro vygenerování reportů uživatel spustí administrační portál, jak bylo zmíněno výše. V záložce *data* poté uživatel najde formulář, kde vyplní, od jakého data až po jaké datum chce vygenerovat report. Hodnoty *od* a *do* tvoří otevřený interval ze kterého se vezmou záznamy o registracích a vytvoří se z nich tabulka. Tabulka s reportem obsahuje časové razítko, identifikátor entity a identifikátor objektu. Pokud ve zvoleném časovém intervalu nebyly provedeny žádné registrace, bude vygenerovaná tabulka obsahovat pouze záhlaví.

Po stisknutí tlačítka pro vygenerování reportu je odeslán do vestavěného zařízení POST požadavek, který obsahuje informace zadané uživatelem. Po přijetí POST požadavku vestavěným zařízením, je požadavek zpracován a následně poslán požadavek do databáze pomocí funkce `db_filterTsdBValues`, tato funkce vybere z databáze položky, jejichž časové razítko se nachází ve zvoleném intervalu. Poté se pomocí callback funkce tato data zpracují a vytvoří se z nich report, který se následně pošle uživateli. Tento report se uživateli zobrazí v administračním portálu.

## Generated report (01.03.2023 - 26.04.2023):

Timestamp	UID	Item
21:54 25.04.2023	0x3ffd	btn3
21:54 25.04.2023	0x3ffd	btn1
21:54 25.04.2023	0x3ffd	btn3
21:59 25.04.2023	0x3ffd	btn2
22:12 25.04.2023	0x3ffd	btn2

Obrázek 7.3: Ukázka vygenerovaného reportu v administračním portálu vestavěného zařízení

## Synchronizace času

Synchronizace času je důležitá pro správnou hodnotu časového razítka při registraci objektu. Po každém zapnutí zařízení se načte aktuální čas, pomocí síťového protokolu SNTP. Volání funkcí spojených se synchronizací času je implementováno do funkce `sntp_start` v souboru `sntp.c`. V případě nepovedené synchronizace, způsobené přítomností zařízení mimo dosah přednastaveného přípojného bodu pro získávání času, bude časové razítko zobrazené na obrazovce mít výchozí hodnotu unixového času, tedy *1. ledna 1970 00:00:00*.

Synchronizace času v 95 % případů proběhne ještě před inicializací grafického rozhraní, ve zbývajících 5 % proběhne zhruba do 2 sekund, po inicializaci grafického rozhraní. Úspěšnost synchronizace času je na první pohled patrná, z hodnoty časového razítka zobrazeného na obrazovkách s položkami k registraci.

## Režim spánku

Kvůli minimalizaci spotřeby vestavěného zařízení byla použita funkce režimu spánku, která je dostupná na platformě ESP32. Tedy když zařízení není po nějakou dobu využíváno, tak se samo uvede do režimu spánku. Platforma ESP32 podporuje několik druhů uvedení zařízení do režimu spánku a jeho následné probuzení. Pro implementaci vestavěného zařízení pro potřeby registrace bylo zvoleno probuzení pomocí změny úrovně na GPIO pinu, tedy na pin, který generuje signál přerušení od čtečky RFID tagů. To znamená, že pokud je zařízení v režimu spánku, a uživatel si přeje provést registraci, tak po přiložení RFID tagu k čtečce se zařízení probudí z režimu spánku. V rámci další úspory energie je u prototypu s LCD displejem v režimu spánku vypínáno podsvícení displeje. Po probuzení z režimu spánku je napájení podsvícení displeje obnoveno. U prototypu s e-paper displejem není využíváno, neboť displej nedisponuje podsvícením.

## Kapitola 8

# Výsledky a diskuze

V rámci práce byly zhotoveny dvě verze prototypu. Funkcionalita obou verzí prototypu je totožná, oba prototypy se liší pouze použitým displejem. Pro prototyp byly vybrány displeje typu LCD a e-paper. Nejprve byl zhotoven prototyp s LCD displejem, na tomto prototypu byl odladěn firmware vestavěného zařízení. Následně byl tento firmware uzpůsoben pro použití s e-paper displejem a přenesen na druhý prototyp. Oba prototypy jsou funkční a poskytují stejné funkcionality.

V následující části budou představeny a porovnány obě verze prototypu na základě jejich příslušných vlastností, dále budou diskutovány možnosti dalšího rozšíření, včetně vylepšení a nových funkcí.

Tabulka 8.1: Tabulka s odhadem pořizovací ceny prototypů

<i>var. prototypu</i>	<i>položka</i>	<i>cena prototypu</i>	<i>sériová výroba</i>
LCD	vývojová deska ESP-WROOM-32	208 Kč	54,11 Kč
	LCD displej (320x480)	358 Kč	144,80 Kč
	PN532 RFID čtečka	218 Kč	53,24 Kč
	celková cena	<b>784 Kč</b>	<b>252,15 Kč</b>
e-paper	vývojová deska ESP-WROOM-32	208 Kč	54,11 Kč
	e-paper displej (400x300)	1018 Kč	217,53 Kč
	PN532 RFID čtečka	218 Kč	53,24 Kč
	celková cena	<b>1444 Kč</b>	<b>324,88 Kč</b>

### 8.1 Cenové zhodnocení prototypu

Stanovení ceny prototypů registračního zařízení je poněkud obtížné, z důvodu možnosti použití novějších komponentů a dostupnost použitelných komponentů. Pro zjednodušení bude předpokladem použití co nejpodobnějších komponent, které byly použity pro tvorbu prototypů. Dále bude pro zjednodušení vyhodnocení ceny použit jeden internetový obchod, jako zdroj komponent. Stanovení přibližné ceny obou prototypů je znázorněno v tabulce 8.1, ceny v tabulce jsou uvedeny s DPH. Pro odhad ceny prototypů byly použity aktuální ceny komponent z internetového obchodu laskakit<sup>1</sup>. Tento odhad ceny prototypu je v tabulce ve sloupci *cena prototypu*. Pro odhad výrobní ceny v sériové výrobě byl použit internetový

<sup>1</sup><https://www.laskakit.cz/>

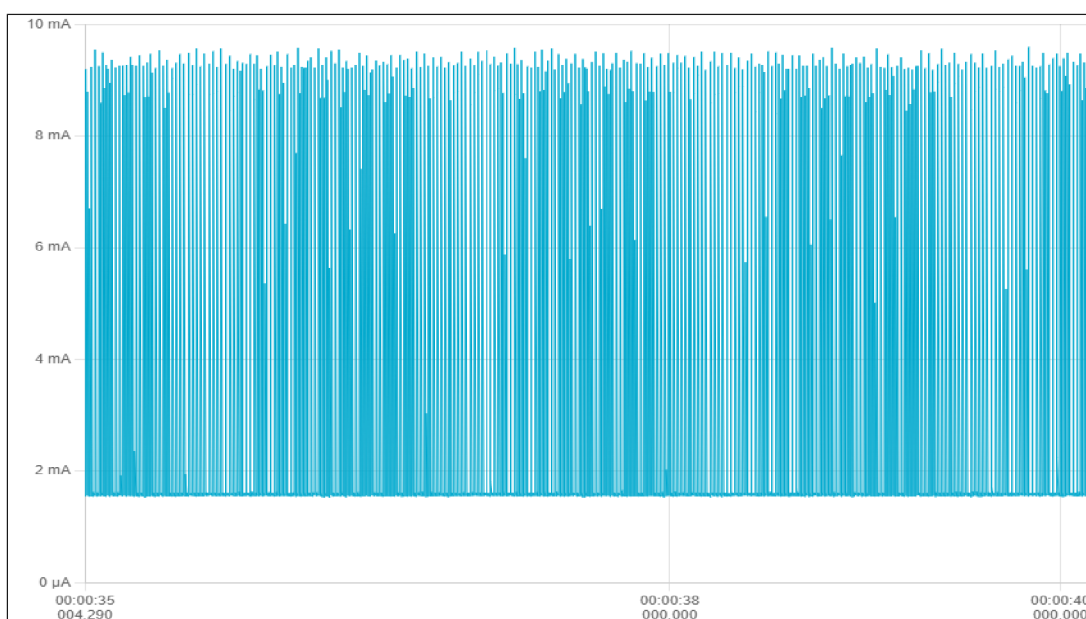
obchod alibaba<sup>2</sup>, tento odhad je umístěn v tabulce 8.1 ve sloupci *sériová výroba*. Pro získání ceny jednoho kusu byla vzata cena pro 100 kusů (zohlednění nižší marže) a tato cena byla vydělena počtem kusů, tedy 100.

Z hodnot v tabulce 8.1 vyplývá, že prototyp, který pro zobrazování používá e-paper displej je výrazně dražší, toto je dáno skutečností, že e-paper displeje jsou obecně dražší, než LCD displeje. Avšak u odhadované ceny sériové výroby tento rozdíl není tak velký.

Vestavěné zařízení pro potřeby registrace by bylo možné použít pro splnění požadavků hypotetické firmy zmíněné v sekci *pořizovací cena* kapitoly 2. Zařízení by mělo teoreticky zvládat registrace 40 uživatelů bez problémů. Tedy pro použití jednoho terminálu na každé oddělení a při použití prototypu s LCD displejem (stejný typ displeje, jako mají terminály zmíněné v kapitole 2) je výsledná cena tohoto řešení 1008,6 Kč včetně DPH. Problém tohoto řešení je, že je braná v potaz cena prototypu. Za předpokladu, že po transformaci prototypu do více komerčně akceptovatelného řešení bude jedno vestavěné zařízení pro potřeby registrace stát zhruba 2000 Kč, tak poté by byla výsledná cena tohoto řešení 8000 Kč včetně DPH. Což je ve srovnání s ostatními variantami zmíněnými v kapitole 2 výrazně nižší cena.

## 8.2 Vliv použitého displeje na spotřebu energie

Jelikož se prototypy liší použitým displejem je možné změřit vliv použitého displeje na spotřebu energie prototypu. Pro měření spotřeby byl zvolen měřicí přípravek *Power Profiler II* od společnosti Nordic Semiconductor. Zařízení umí měřit proudy od 200 nA až do 1 A, tedy by mělo mít dostačující parametry pro změření spotřeby obou prototypů.



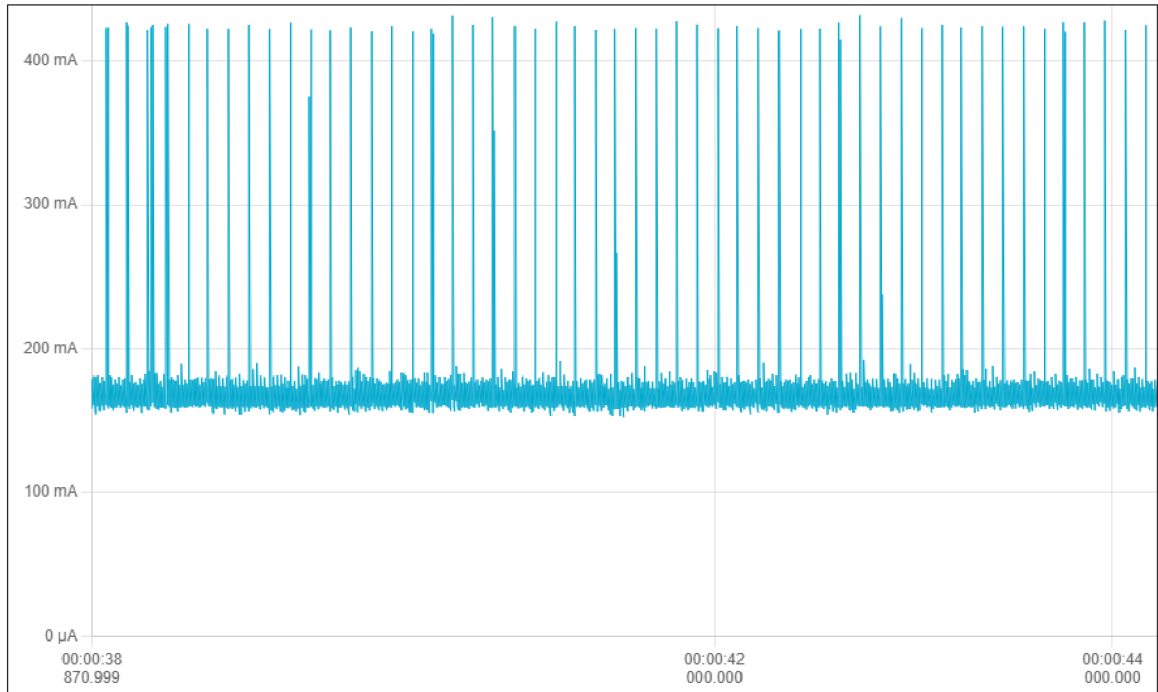
Obrázek 8.1: Ukázka naměřené spotřeby prototypu s e-paper displejem v režimu spánku

Oba prototypy byly modifikovány (včetně firmware), pro určení vlivu jednotlivých komponent na spotřebu zařízení. U obou prototypů je na desce plošných spojů přítomný konektor pro napájení desky z baterie, který byl použit pro napájení prototypů a měření spotřeby

<sup>2</sup><https://www.alibaba.com/>



energie, kvůli možnému ovlivnění výsledků USB převodníkem. V rámci měření bylo nejprve spuštěno zaznamenávání dat o spotřebě, poté bylo zapnuto napájení prototypu. Aby se zajistilo ustálení hodnot, byla délka všech měření nastavena na jednu minutu. Jelikož naměřené hodnoty výrazně měnily (jak lze vidět na obrázku 8.1), byly naměřené hodnoty průměrovány. Výsledky měření jsou zaznamenány v tabulce 8.2.



Obrázek 8.2: Ukázka naměřené spotřeby prototypu s e-paper displejem s aktivním administračním portálem

Z výsledků v tabulce 8.2, lze usoudit, že největší vliv na spotřebu má knihovna LVGL, použití režimu spánku, používání administračního portálu a použitý typ displeje. Z naměřených hodnot vyplývá, že obě desky mají v základu podobnou spotřebu energie a neinicializovaný LCD displej bez podsvícení má vyšší spotřebu, než neinicializovaný e-paper displej.

Dále je z výsledků vidět, že z obou plně inicializovaných prototypů má nižší spotřebu prototyp s e-paper displejem, zhruba o 28,75 mA. Zdaleka největší vliv na spotřebu zařízení má povolení administračního portálu. U prototypu s LCD displejem rozdíl oproti spotřebě v normálním režimu činí 87,31 mA a u prototypu s e-paper displejem tento rozdíl činí 91,12 mA. Tyto vysoké rozdíly jsou dané spotřebou platformy ESP32 při používání technologie WiFi (hlavně odběrové špičky viditelné na obrázku 8.2), která je potřeba pro implementaci administračního portálu pomocí webserveru hostovaného na zařízení. Jelikož je administrační portál používán pouze ke konfiguraci zařízení a ke generování reportů, tedy nebude tato možnost povolena často.

Z naměřených hodnot vyplývá, že prototyp s e-paper displejem má v režimu spánku nižší spotřebu, než prototyp s LCD displejem a rozdíl činí 13,8 mA. V normálním režimu má prototyp s e-paper displejem také nižší spotřebu oproti prototypu s LCD displejem a rozdíl činí 28,75 mA. Nižší spotřeba prototypu s e-paper displejem je dána obecně nižší spotřebou e-paper displejů než LCD displejů.

Tabulka 8.2: Naměřené hodnoty spotřeby energie prototypů

<i>zařízení</i>	<i>položka</i>	<i>průměrný odběr (mA)</i>
RFID čtečka	RFID čtečka	2,14
prototyp s LCD displejem	pouze deska s ESP32	13,65
	deska s ESP32, displej	22,75
	deska s ESP32, displej, LVGL	109,52
	zařízení v režimu spánku	16,65
	zařízení s admin. portálem	196,83
prototyp s e-paper displejem	pouze deska s ESP32	13,44
	deska s ESP32, displej	16,22
	deska s ESP, displej, LVGL	80,77
	zařízení v režimu spánku	2,85
	zařízení s admin. portálem	171,89

Naměřené hodnoty v normálním režimu používání jsou oproti očekávání nepřiměřeně vysoké, obzvlášť u prototypu s e-paper displejem, toto, jak je vidět z naměřených hodnot v tabulce 8.2, je dané režii knihovny LVGL. Jelikož zařízení při běžném používání stráví většinu času v režimu spánku, tak je tato vyšší spotřeba kompenzována nízkou spotřebou v režimu spánku. Toto má samozřejmě vliv na možnou dobu používání zařízení při použití napájení z baterie.

Jelikož *Power Profiler II* zaznamenává i spotřebovaný náboj, lze z těchto hodnot snadno spočítat výdrž zařízení, pokud by bylo napájeno z baterie. Pro toto použití je vhodnější prototyp, který používá e-paper displej, proto budou naměřené hodnoty z tohoto prototypu použity pro další výpočty. Prototyp s e-paper displejem spotřebuje na probuzení z režimu spánku, následný provoz po dobu 15 sekund a následné uvedení zpět do režimu spánku náboj o hodnotě 1,26 C. Oproti tomu v režimu spánku potřebuje prototyp s e-paper displejem pro stejnou dobu provozu náboj o hodnotě 43,19 mC. Pokud tyto hodnoty vydělíme dobou provozu, tedy hodnotou 15 sekund, dostaneme průměrný náboj potřebný na 1 sekundu provozu zařízení, tedy pro standardní režim tato hodnota vychází 84 mC a pro režim spánku tato hodnota vychází zhruba 2,879 mC. Za předpokladu provedení 30 registrací za den a při trvání jedné registrace zhruba 15 sekund, bude celková část dne zařízení strávená v normálním režimu rovna 450 sekundám, v rovnici 8.1 je tato hodnota označena jako  $t_{norm}$ , zbývajících 85 950 sekund dne, v rovnici 8.1 je tato hodnota označena, jako  $t_{sleep}$ , bude zařízení strávit v režimu spánku, který spotřebovává méně energie.

Tedy za předpokladu režimu používání zmíněného výše a použití baterie typu 18650 s kapacitou 6800 mAh, prototyp s e-paper displejem vydrží být napájen z baterie zhruba 85 dnů, jak vyplývá z výpočtu 8.1. Výsledná teoretická hodnota vycházející z výpočtu 8.1 je oproti očekávání vycházejících z naměřených hodnot v tabulce 8.2 poměrně překvapivá, avšak pokud by se ještě podařilo snížit spotřebu vestavěného zařízení (hlavně složku danou režii knihovny LVGL), mohla by být možná doba napájení zařízení z baterie ještě výrazně vyšší.

$$\begin{aligned}
 \frac{\text{kapacita baterie}}{\text{odběr v norm. rež.} * t_{norm} + \text{odběr v rež. spánku} * t_{sleep}} &= \text{výdrž zařízení} \\
 \frac{6,8Ah * 3600s}{0,084As * 450s + 0,002879 * 85950s} &= 85,819 \text{ dne}
 \end{aligned} \tag{8.1}$$

V pokračování této práce by bylo vhodné předělat inicializaci knihovny LVGL spolu s integrací ovladače displeje, pro docílení nižší režie spojené s displejem. Toto by bylo vhodné obzvláště u prototypu s e-paper displejem, kde nízká spotřeba, a z toho vyplývající možnost použití baterie pro napájení zařízení, byla hlavním důvodem použití e-paper displeje v jednom z prototypů.

### 8.3 Zhodnocení používání zařízení

Vliv na používání zařízení může mít několik různých komponent. V následující části bude popsán vliv těchto komponent a jak ovlivňují používání zařízení.

Displej je jedním z klíčových prvků při používání jakéhokoli elektronického zařízení a může mít zásadní vliv na jeho celkovou funkčnost a pocit uživatelů z používání tohoto zařízení. V případě prototypu s displejem typu e-paper je však nutné počítat s určitými omezeními. Prototyp s displejem typu e-paper se jeví pomalejší než druhý prototyp s LCD displejem. To je dáno především vlastnostmi displejů typu e-paper, které mají nízkou obnovovací frekvenci. Tento fakt musel být reflektován v návrhu uživatelského rozhraní, což vedlo k některým omezením a omezení v možnostech uživatelské interakce s displejem. To znamená, že nebylo možné vytvořit uživatelské rozhraní s vysokou odezvou nebo s animacemi, které jsou běžné u zařízení s displeji s vysokou obnovovací frekvencí.

Na druhou stranu, díky své nižší spotřebě energie, oproti prototypu s LCD displejem je tento prototyp ideální pro zařízení, které by mělo být používáno po delší dobu bez nutnosti nabíjení.

Kromě použitého typu displeje má také vliv na používání zařízení režim, v jakém se zařízení provozuje. Například v grafickém režimu má zařízení při zapnutí pomalejší inicializaci. Zpoždění není příliš výrazné (přibližně 1 sekunda navíc), ale tato pomalejší inicializace je způsobena načítáním obrázků položek k registraci do operační paměti zařízení, což je náročnější operace oproti načítání názvů položek k registraci z řetězce uloženého v databázi do operační paměti. Proto mohou uživatelé zaznamenat zpoždění při přepnutí do grafického režimu a při zapnutí zařízení. Avšak, jakmile je inicializace dokončena, používání zařízení by mělo být relativně plynulé (pořád platí omezení e-paper displejů) a bez výrazných zpoždění.

## Kapitola 9

# Závěr

Cílem této práce bylo navrhnout a ve formě prototypu realizovat vestavěné zařízení pro potřeby registrace s možností správy zařízení a generování reportů za určitý časový interval, pomocí technologie Bluetooth nebo WiFi.

Nejprve jsem se zaměřil na problematiku registračních systémů a jejich vlastností, technologii RFID, platformu ESP32 a problematiku zobrazovacích zařízení pro vestavěné systémy, včetně grafických knihoven pro tato zobrazovací zařízení. Tyto znalosti byly následně použity pro tvorbu návrhu zařízení, pomocí kterého byly následně zkonstruovány dvě verze vestavěného zařízení pro potřeby registrace, ve formě prototypu. Jeden z těchto dvou prototypů disponuje LCD displejem a druhý e-paper displejem.

Výsledné zařízení, které bylo vytvořeno, nabízí možnost registrace objektů k určité entitě a také podporuje identifikaci entity pomocí RFID karty. Díky tomu umožňuje ukládání dat o provedených registracích a následné generování reportů z těchto dat. Pro správu zařízení a generování reportů byl vytvořen administrativní portál, který je k dispozici prostřednictvím webového rozhraní.

Tyto výsledné prototypy splnil požadavky stanovené v kapitole věnované návrhu prototypu. Parametry obou prototypů byly vyhodnoceny a popsány v poslední kapitole práce. Bylo dosaženo cílů této práce, avšak pro komerční využití by bylo vhodné přizpůsobit prototypy pro nižší režii spojenou s displejem, a tedy i nižší potřebou energie zařízení. To by vyžadovalo úpravu inicializace knihovny LVGL spolu s integrací ovladače displeje. Taková úprava by měla výrazně snížit spotřebu energie vestavěného zařízení, a tedy i prodloužit dobu používání zařízení při napájení z baterie.

Jako rozšíření funkcionality tohoto zařízení by bylo vhodné implementovat nahrávání obrázků do zařízení pomocí administrativního portálu, což by přineslo významné zjednodušení procesu pro uživatele, kteří by mohli snáze nahradit stávající obrázky položek k registraci novými a přizpůsobit tak vzhled zařízení svým potřebám. Dalším užitečným rozšířením by byla možnost odesílat záznamy o registraci po síti na centrální server, což by umožnilo centralizovaný sběr dat z více registračních zařízení najednou. To by umožnilo efektivnější vyhodnocování údajů při použití vícero registračních zařízení. Tyto funkce by byly významným přínosem pro uživatele a přidaly by nové možnosti pro využití zařízení.

# Literatura

- [1] ANDERSSON, P. *SPIFFS* [online]. Peter Andersson, 2022 [cit. 2023-24-01]. Dostupné z: <https://github.com/pellepl/spiffs>.
- [2] ARMINK. *FlashDB* [online]. 2017 [cit. 2023-24-01]. Dostupné z: <https://github.com/armink/FlashDB>.
- [3] EFG CZ SPOL. S R.O.. *Chytrý docházkový systém* [online]. 2023 [cit. 2023-19-01]. Dostupné z: <https://www.aktion.cz/produkty/dochazkovy-system.html>.
- [4] EFG CZ SPOL. S R.O.. *Moderní kontrola přístupů* [online]. 2023 [cit. 2023-19-01]. Dostupné z: <https://www.aktion.cz/produkty/pristupovy-system.html>.
- [5] ESPRESSIF SYSTEMS. *ESP-IDF Programming Guide* [online]. 2020 [cit. 2023-17-01]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/release-v4.1/>.
- [6] ESPRESSIF SYSTEMS. *ESP32 Arduino Core's documentation* [online]. 2022 [cit. 2023-17-01]. Dostupné z: <https://espressif-docs.readthedocs-hosted.com/projects/arduino-esp32/en/latest/index.html>.
- [7] ESPRESSIF SYSTEMS. *Impressive UIs for ESP Projects with SquareLine Studio* [online]. 2022 [cit. 2023-20-01]. Dostupné z: [https://www.espressif.com/en/news/ESP\\_UIs\\_SquareLine\\_Studio](https://www.espressif.com/en/news/ESP_UIs_SquareLine_Studio).
- [8] ESPRESSIF SYSTEMS. *ESP Product Selector* [online]. 2023 [cit. 2023-18-01]. Dostupné z: <https://products.espressif.com/#/product-comparison>.
- [9] ESPRESSIF SYSTEMS. *ESP32 Series Datasheet* [online]. 4.1. 2023 [cit. 2023-16-01]. Dostupné z: [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf).
- [10] FINKENZELLER, K. *RFID handbook: fundamentals and application in contactless smart cards and identification*. 2. vyd. 2003. ISBN 0-470-84402-7.
- [11] LVGL LLC. *Light and Versatile Graphics Library* [online]. 2021 [cit. 2023-20-01]. Dostupné z: <https://github.com/lvgl/lvgl>.
- [12] RFID4U. *How to Select a Correct RFID Tag – Passive vs. Active* [online]. RFID4U, 2023 [cit. 2023-16-01]. Dostupné z: <https://rfid4u.com/rfid-passive-active/>.
- [13] RFID4U. *How to Select a Correct Tag – Frequency* [online]. RFID4U, 2023 [cit. 2023-16-01]. Dostupné z: <https://rfid4u.com/rfid-frequency/>.

- [14] SAFESCAN. *Docházkové přístroje s RFID nebo otiskem prstu* [online]. 2023 [cit. 2023-23-04]. Dostupné z: <https://www.safescan.com/cs-cz/store/dochazkove-pristroje>.
- [15] SAFESCAN. *Docházkový systém - TimeMoto TM616* [online]. 2023 [cit. 2023-23-04]. Dostupné z: <https://www.safescan.com/cs-cz/store/dochazkove-pristroje/timemoto-tm-616-dochazkovy-system>.
- [16] SAFESCAN. *TimeMoto registrační software v cloudu* [online]. 2023 [cit. 2023-23-04]. Dostupné z: <https://www.safescan.com/cs-cz/store/dochazkove-pristroje/timemoto-cloud>.
- [17] SERVICES, A. W. *About FreeRTOS Kernel* [online]. 2023 [cit. 2023-16-03]. Dostupné z: <https://www.freertos.org/RTOS.html>.
- [18] SIEMENS A.S.. *Transponders (ISO mode)* [online]. Siemens, 2022 [cit. 2023-19-01]. Dostupné z: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/10047940?tree=CatalogTree>.
- [19] SLONEEK EUROPE S.R.O.. *Docházkový systém - efektivní evidence pracovního času* [online]. 2023 [cit. 2023-23-04]. Dostupné z: <https://www.sloneek.cz/dochazkovy-system/>.
- [20] UGFX GMBH. *UGFX* [online]. 2021 [cit. 2023-20-01]. Dostupné z: <https://ugfx.io/index>.
- [21] VISTECH S.R.O.. *Docházkový systém Time BOX X4* [online]. 2023 [cit. 2023-23-04]. Dostupné z: [dochazkove-hodiny.cz/produkty/dochazkove-systemy/dochazkovy-system-timebox-x4.htm](https://dochazkove-hodiny.cz/produkty/dochazkove-systemy/dochazkovy-system-timebox-x4.htm).
- [22] VISTECH S.R.O.. *SW TimeBox* [online]. 2023 [cit. 2023-23-04]. Dostupné z: <https://www.dochazkove-hodiny.cz/sw-timebox.htm>.
- [23] VISTECH S.R.O.. *Vistech e-shop* [online]. 2023 [cit. 2023-23-04]. Dostupné z: <https://www.timebox.cz/e-shop.htm>.
- [24] VITTORIO, F. D. *FabGL* [online]. Fabrizio Di Vittorio, 2022 [cit. 2023-20-01]. Dostupné z: <https://github.com/fdivitto/fabgl>.

## Příloha A

# Srovnání modulů ESP32

Tabulka A.1: Srovnání vybraných dostupných modulů ESP32 [8]

<i>Modul</i>	<i>ESP32-WROOM-32E</i>	<i>ESP32-S2-WROOM-I</i>	<i>ESP32-S3-WROOM-1</i>	<i>ESP32-C3-WROOM-02</i>	<i>ESP32-C6-WROOM-1</i>
<i>Řada</i>	ESP32	ESP32-S2	ESP32-S3	ESP32-C3	ESP32-C6
<i>CPU</i>	Xtensa LX6	Xtensa LX7	Xtensa LX7	RISC-V	RISC-V
<i>Počet jader</i>	2	1	2	1	1
<i>Frekvence</i>	240 MHz	240 MHz	240 MHz	160 MHz	160 MHz
<i>SRAM</i>	520 KB	320 KB	512 KB	400 KB	512 KB
<i>ROM</i>	448 KB	128 KB	384 KB	384 KB	320 KB
<i>Flash</i>	4 MB	4 MB	4 MB	4 MB	4 MB
<i>GPIO</i>	26	37	36	15	23
<i>UART</i>	3	2	3	2	3
<i>SPI</i>	4	4	4	3	1
<i>I2C</i>	2	2	2	1	2
<i>I2S</i>	2	1	2	1	1
<i>MCPW</i>	2	0	2	0	0
<i>USB OTG</i>	0	1	1	0	0