VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Přístupový systém založený na detekci otisku prstu
Access System Based on Fingerprint Detection

2015 **Bc. Filip Osadník**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Zadání diplomové práce

Student: **Bc. Filip Osadník**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2612T025 Informatika a výpočetní technika

Téma: Přístupový systém založený na detekci otisku prstu
Access System Based on Fingerprint Detection

Zásady pro vypracování:

Navrhněte a realizujte přístupový systém umožňující ovládání dveřního zámku. Sytém bude jako vstupní data využívat digitalizovaný otisk prstu. Sejmutý otisk bude analyzován nadřazeným systémem. Nadřazený systém vyhodnotí zdali je možné uvolnit dveřní zámek či nikoli.

1. Prostudujte základní teorii týkající se otisků lidských prstů, především se zaměřte na způsoby kvalifikace otisků a možnosti využití při autentizaci.
2. Prozkoumejte možnosti komerčně dostupných snímačů otisků prstů a navrhněte vhodný snímač pro další použití.
3. Navrhněte obvodové schéma které bude realizovat snímač otisků prstů a s nezbytnými obvody pro přenos do nadřazeného systému. Schéma realizujte na kontaktním poli.
4. Navrhněte a realizujte potřebný software pro vyhodnocení sejmutých otisků prstů. Zhodnoťte zdali nebude možné využít již existující knihovny.
5. Proveďte testování celého systému. Zhodnoťte jeho spolehlivost a bezpečnost.

Seznam doporučené odborné literatury:

[1] Davide Maltoni & Dario Maio & Anil K. Jain & Salil Prabhakar, "Handbook of Fingerprint Recognition", 2009, ISBN 978-1848822535
[2] Nalini Ratha & Ruud Bolle, "Automatic Fingerprint Recognition Systems", 2003, ISBN 978-0387955933
[3] Lucio Di Jasio, "Programming 16-Bit PIC Microcontrollers in C, Second Edition: Learning to Fly the PIC
24", 2011, ISBN 978-1856178709
[4] Fred Eady, "Networking and Internetworking with Microcontrollers", 2004, ISBN 978-0750676984

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. David Seidl, Ph.D.**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2015

doc. Dr. Ing. Eduard Sojka
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Dne: *1.5.2015*

.....

Podpis

Poděkování

Rád bych poděkoval Ing. Davidu Seidlovi, Ph.D. za odbornou pomoc a konzultaci při vytváření této diplomové práce.

Abstrakt

Diplomová práce se zabývá návrhem a realizací přístupového systému založeného na detekci otisku prstu.Výslednou implementací systému je realizace fyzického prototypu zařízení, jeho firmwaru a serverového softwaru pro identifikaci osob na základě rozpoznávání otisku prstu. Práce je rozdělena do 3 tematických kapitol, přičemž první kapitola se zabývá teoretickým nástinem funkce rozpoznávání otisků prstů a další dvě kapitoly se zaměřují na návrh a realizaci systému samotného.

Klíčová slova

Rozpoznávání otisku prstu, otisk prstu, přístupový systém, RFID, identifikace otisku, klientská jednotka

Abstract

This master's thesis is focused on designing and implementation of access system based on fingerprint detection. The outcome of the implementation is production of physical prototype of the device, it's firmware and server software for identification of person based on fingerprint recognition. The thesis is divided into 3 chapters, where the first one focuses on theoretical way of describing how fingerprint recognition works and the other two chapters deals with design and implementation of the system itself.

Key words

Fingerprint recognition, fingerprint, access system, RFID, fingerprint identification, client unit

Seznam použitých zkratek

Zkratka	Anglický význam	Český význam
PIN	Personal Identification Number	Osobní identifikační číslo
CCD	Charge-Coupled Device	Zařízení s vázanými náboji
CMOS	Complementary Metal Oxide Semiconductor	Komplementární kov-oxid-polovodič
LED	Light-Emitting Diode	Světlo-emitující dioda
SPI	Serial Peripheral Interface	Sériové periferní rozhranní
LCD	Liquid-Crystal Display	Displej z tekutých krystalů
I2C	Inter-Integrated Circuit	Meziobvodová sběrnice
ASCII	American Standard Code for Information Interchange	Americký standardní kód pro výměnu informací
IEEE	Institute of Electrical and Electronics Engineers	Institut pro elektrotechnické a elektronické inženýrství
MAC	Media Access Control	Řízení přístupu k médiu
TCP/IP	Transmission Control Protocol/Internet Protocol	Přenosový protokol síťové vrstvy
MIPS	Million Instruction Per Second	Milion instrukcí za sekundu
RFID	Radio Frequency Identification	Identifikace na rádiové frekvenci
ANSI	American National Standards Institute	Americký národní standardizační institut
DBMS	Database management system	Systém řízení báze dat

Obsah

1 Úvod

Omezení přístupu neautorizovaným osobám bývá se vzrůstajícími požadavky na bezpečnost čím dál důležitější disciplínou. Přístupové systémy se právě na tento úkol specializuji a umožňují omezit vstup pouze autorizovaným osobám. Ať už jde o omezení přístupu do nějaké místnosti nebo k nějakému prostředku, tak vždy je nutno identitu vstupující osoby nějakým definovaným způsobem ověřit. Konkrétní způsoby ověření pak lze v případě přístupových systémů rozdělit do následujících 3 skupin [1]:

- Znalost nějaké fráze (PIN kód, heslo)
- Držení specifické věci (přístupový token, identifikační karta)
- Biometrická identifikace

Všechny zmíněné způsoby ověření mají své klady a zápory týkající se bezpečnosti a pohodlnosti použití. První dva zmíněné způsoby jsou však potenciálně zneužitelné. Autentifikační heslo lze odchytit při zadávání do systému, či případně může pověřená osoba tuto frázi vyzradit dalším osobám, které tímto získávají přístup do daného systému také. Úzce s tímto souvisí také druhý způsob identifikace a to pomocí přístupového tokenu, který může být odcizen nebo zcela záměrně předán dalším osobám. I v tomto případě pak smysluplnost přístupového systému zaniká, protože přístup do systému získávají i neautorizované osoby.

Dalším aspektem při využívání přístupových systému je také pohodlnost obsluhy. Znalost nějaké fráze nebo hesla zde naráží na nedokonalost paměti lidského mozku a jeho zapomnětlivost. Poměrně často se tedy stává, že přístup do systému je zamítnut i autorizované osobě, protože tato osoba zadá heslo chybné nebo si na něj vůbec nevzpomene. Do podobného problému se také dostává způsob autentifikace pomocí držení specifické věci. V tomto případě může autorizovaná osoba zapomenout identifikační kartu vzít sebou a dostává se tak do naprosto stejného problému - přístup do systému jí je zamítnut.

Všechny tyto zmíněné neduhy řeší způsob ověření pomocí biometrických dat [2]. Biometrická data obsahují informace, které jsou pro daného člověka více, či méně jedinečné. Pověřená osoba nemůže předat přístup do systému jiné osobě, protože biometrická data jsou pro ní unikátní. Tímto je dosaženo vysoké bezpečnosti celého přístupového systému. Taktéž otázka pohodlnosti ovládání je tímto způsobem řešena, protože biometrická data jsou povětšinou součástí lidského těla a nevyžadují konstantní nošení specifického hardwaru nebo nutnost zapamatování si nějaké informace.

Tato diplomová práce se dále zabývá právě metodou ověření v přístupovém systému pomocí biometrických dat a návrhem fyzického prototypu zařízení, který bude schopen formu biometrické identifikace na bázi rozpoznávání otisků prstů využít.

2 Biometrická identifikace

Biometrická identifikace osoby je založena na principu sledování vlastností nebo identifikátorů, které jsou jedinečné pro každého člověka [1]. Identifikování osoby podle biometrického rozpoznávání lze rozdělit jako rozpoznávání anatomické (např. otisk prstu, snímání tváře, sítnice a DNA) a rozpoznávání chování (např. hlas nebo podpis). Tyto metody biometrického ověřování mají rozdílnou úroveň jedinečnosti. Kupříkladu podpis jedince dokáže profesionální padělatel napodobit tak dokonale, že vyhodnocovací software vyhodnotí podpis jako autentický. Proti tomu snímání sítnice se považuje za nejbezpečnější způsob ověření identity osoby, protože se předpokládá, že obraz struktury sítnice má každý člověk jedinečný a současně je velmi obtížné tento obraz potenciálním útočníkem replikovat [1]. My se však dále zaměříme výhradně na biometrickou identifikaci založenou na detekci otisku prstů.

2.1 Daktyloskopie

Daktyloskopie je nauka zkoumající obrazce a stopy papilárních linií na vnitřní straně prstů a na dlaních a chodidlech. Zkoumání těchto stop přivedlo odbornou veřejnost k vytvoření třech základních předpokladů, kterými se identifikace osob řídí [3]:

1. Neexistují dva jedinci, kteří by měli naprosto identické obrazce otisků prstů. Na základě statistické analýzy je prokázáno, že je velmi malá pravděpodobnost existence dvou osob, které by měly naprosto identické obrazce papilárních linií. Pokud vezmeme v úvahu, že na Zemi se nachází přibližně 6,5 miliardy obyvatel, tak prostým vynásobením 10 prstů dostaneme 65 miliard unikátních otisků prstů.
2. Papilární linie jsou obtížně odstranitelné. Odstraněním svrchní kůže prstu dojde pouze k dočasnému odstranění vzoru obrazu papilárních linií. Pokud by jedinec chtěl tuto vrstvu trvale odstranit, musel by odstranit zárodečnou vrstvu kůže. V opačném případě dojde po zahojeni poraněné kůže k vytvoření stejného obrazce.
3. Přes celý život člověka jsou otisky prstů neměnné. Obrazce papilárních linií se utvářejí již v 7. měsíci vývoje plodu a je dokázáno, že jsou po celý život člověka prakticky neměnné. Malé změny mohou být v průběhu života způsobeny poraněním, např. pořezáním nebo pohmožděním článku prstu.

Právě tyto skutečnosti vytvořily podmínky pro využívání analýzy otisku prstů při usvědčování zločinů v kriminalistice, protože se považují za dostatečně jedinečný identifikátor konkrétní osoby.

2.1.1 Anatomie lidského otisku prstu

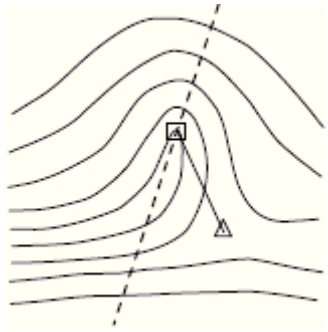
Otisk lidského prstu je složen z řady papilárních linií, které plynou souběžně vedle sebe a vytváří tak charakteristický vzor složený z hřebenů a brázd [3]. Někdy tyto souběžné linie vytváří místní jedinečnost v otisku jako např. určitý oblouk, smyčku apod.. Typický vzor otisku těchto linií nejlépe charakterizuje následující obrázek (Obr. 1):

Obr.1: Snímek otisku prstu [2]

Z obrázku je patrné, že hřebeny linií jsou tvořeny tmavými linkami a naopak brázdy tvoří světlé prostory mezi jednotlivými liniemi. Další zkoumání hřebenů papilárních linií se děli do tří skupin, dle úrovně zkoumaných detailů:

1. Zběžné zkoumání

   V první úrovni se zkoumá orientace hřebenu linie a četnost prokládání jednotlivých linií [1]. Hřebeny linií obvykle plynou vedle sebou, ale na několika ojedinělých místech obvykle vytváří náhle změny toku linií, které formují nějaký specifický tvar. Tyto tvary vytvářejí v otisku jedinečnosti, které se následně používají pro identifikaci osoby. Některé tyto formy tvarů jsou zachyceny na obrázcích (Obr. 2) a (Obr. 3):

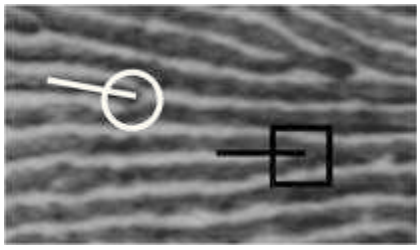


Obr.2: Tvar jedinečnosti v otisku typu "Levá smyčka" [2]

Obr. 3: Tvar jedinečnosti v otisku typu "Dvoudeltový ovál" [2]

2. Detailní zkoumání

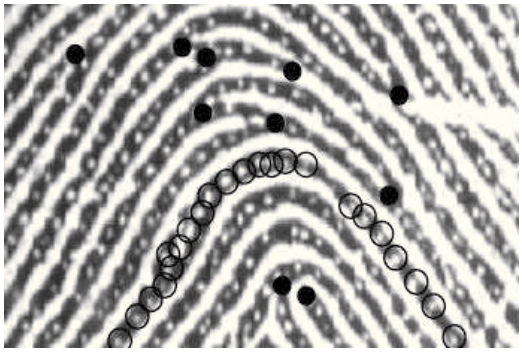
   V druhé úrovni se zkoumá kromě orientace a četnosti také náhlá zakončení, zdvojení, začátek nebo spojení hřebenů jednotlivých linií [1]. Tyto náhlé změny jsou opět individuální pro danou osobu a přispívají k její identifikaci. Ukázku zdvojení papilární linie a náhlého zakončení zachycuje (Obr. 4):



Obr. 4: Náhlé zakončení a zdvojení papilární linie [1]

3. Velmi detailní zkoumání

   V poslední úrovni se zkoumá, mimo informací zachycených v předchozích dvou úrovních, také rozměry jednotlivých hřbetů linií, tvary hřbetů linií a také póry, šrámy a záhyby hřbetů jednotlivých linií [2]. Takové informace dále zlepšují identifikaci konkrétní osoby a snižují riziko chybovosti. Póry a šrámy na papilárních liniích zachycuje následující obrázek (Obr. 5):



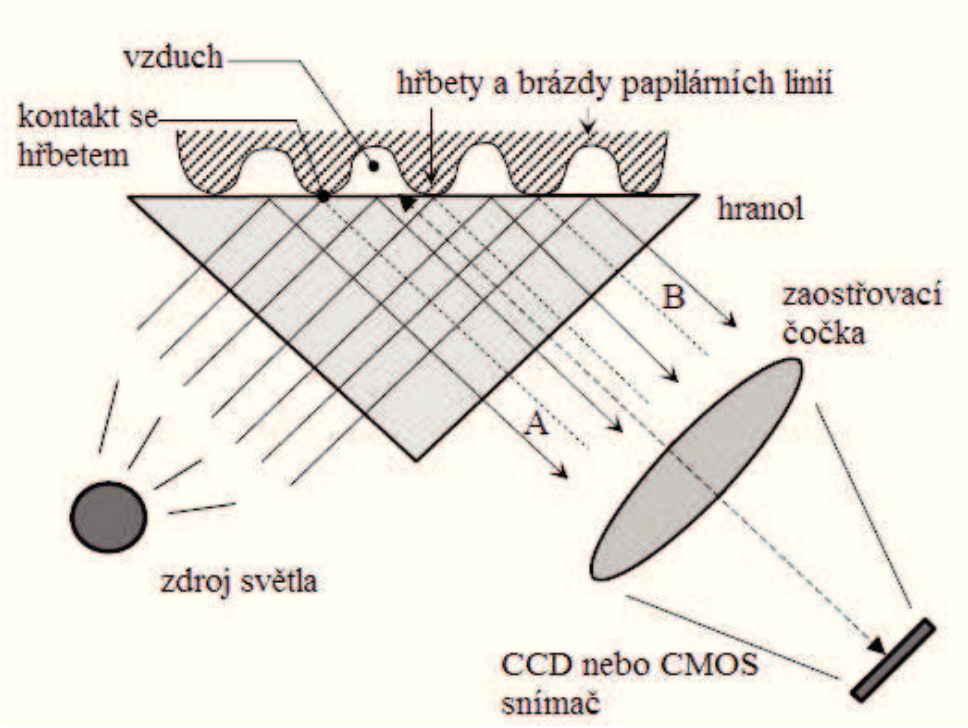
Obr. 5: Póry na hřbetech papilárních linií [2]

2.2 Elektronické snímání otisků

Elektronické snímání otisků je založeno na principu senzorického snímání plochy mezi podložkou a jednotlivými papilárními liniemi prstu. Toto snímání je vykonáváno za pomocí elektronického senzoru, který změří množství tloušťky kůže v daném bodě a tuto informaci dále převede do podoby dvoudimenzionálního obrazu. V dnešní době existuje celá řada způsobů snímání otisků založených na rozdílných fyzikálních principech. Tato kapitola popisuje některé nejpopulárnější způsoby snímání otisků.

2.2.1 Optický senzor

Optické snímání snímá odrazivost kůže prstu položeného na hranolu, který je z jedné strany pod určitým úhlem nasvícen, z opačné strany je umístěna optická čočka a za ní CCD nebo CMOS snímač pro snímání odraženého světla [2] (Obr. 6). Zdrojem světla může být laserová nebo LED dioda.



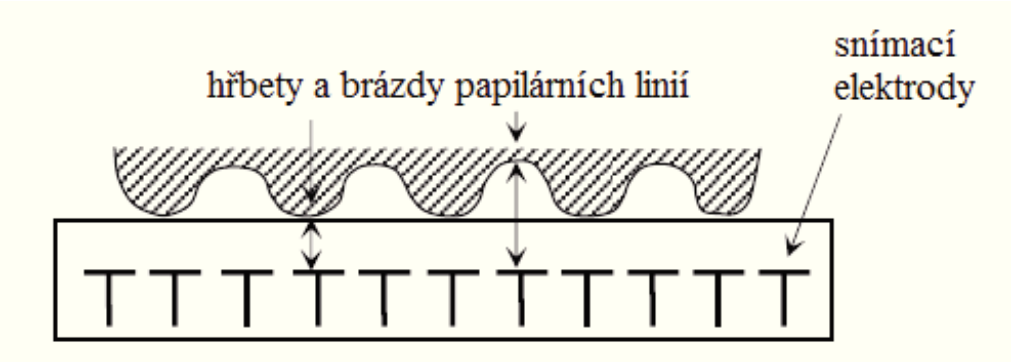
Obr. 6: Princip funkce optického snímače [2]

Když světlo dopadne na hřbet papilární linie dojde k jeho absorpci a snímač zaregistruje na výstupu tmavý bod. V případě, že světlo dopadne na brázdu mezi papilárními liniemi, tak se odrazí na snímač a na výstupu se projeví jako světlý bod. Výhoda takového řešení je poměrně nízká cena

senzoru a s ní související vysoká použitelnost pro nenáročné aplikace. Nevýhoda je potenciálně velká velikost senzoru jako celku, protože je nutno vzít v úvahu, že spojná optická čočka potřebuje nějakou minimální zaostřovací vzdálenost. Senzory, které tuto minimální zaostřovací vzdálenost nerespektují mohou do výsledného obrazu zanést výrazné zkreslení.

2.2.2 Kapacitní senzor

Kapacitní senzor je tvořen polem elektrod, které jsou uzavřeny do pouzdra křemíkového čipu [1]. Na dotykovou plochu takového senzoru se položí prst, který se chová jako druhá elektroda a vytváří tak miniaturní kondenzátor. Výsledná kapacita jednotlivých snímacích bodů se pak liší v závislosti na tom, zda je na konkrétním snímacím bodu hřbet nebo brázda papilární linie.



Obr. 7: Princip funkce kapacitního snímače [2]

Kapacitní senzor je výhodný v poměrně malém výsledném pouzdru produktu. Nevýhoda použití popsaného senzoru je však v riziku poškození snímací plochy vlivem vysokého statického napětí při pokládání prstu na snímací plochu. Pro bezchybnou funkci musí mít senzor správně vyřešenou ochranu před elektrostatickým napětím.

2.2.3 Ultrazvukový senzor

Ultrazvukový snímač funguje na principu snímání odraženého zvuku vygenerovaného ultrazvukovým vysílačem směrem k prstu uživatele [2]. Typický senzor se tak skládá z ultrazvukového vysílacího a přijímacího modulu. Vysílač vysílá směrem k dotykové ploše ultrazvukový signál, který se v závislosti na vzdálenosti od hřebene nebo brázdy papilární linie vrací s odlišnou amplitudou. Tohoto efektu je využito pro vytvoření výsledného snímku. Ultrazvukové senzory poskytují kvalitní obrazy otisku, protože ultrazvukové vlny mají schopnost procházet přes

drobné nečistoty na dotykové ploše. Nevýhoda je však v délce trvání snímání jednoho prstu a prozatím poměrně vysoké ceně senzoru.

2.2.4 Piezoelektrický senzor

Snímání na principu piezoelektrického jevu je další metodou získávání snímku otisku prstu. Snímač pracuje na principu vyhodnocování drobných změn v generovaném proudu piezoelektricky aktivního materiálu [2]. Každý snímaný bod na dotykové ploše senzoru představuje vlastní obvod oddělený od ostatních. Tímto způsobem lze detekovat, zda se na daném senzorickém bodu nachází hřbet nebo brázda papilární linie otisku prstu, protože hřbet vytváří větší tlak na piezoelement, který následně generuje větší množství proudu obvodem. Nevýhoda použití tohoto způsobu však souvisí s nedostatečnou rozlišovací schopností jednotlivých úrovní papilární linie a proto metoda piezoelektrického snímání generuje pouze binární obraz otisku prstu.

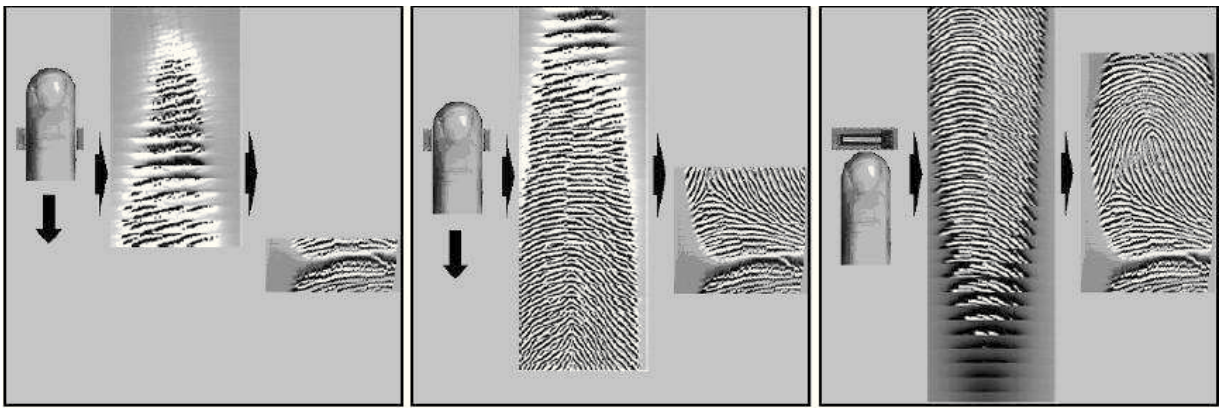
2.2.5 Způsoby snímání otisku prstu

Snímače otisků prstů se dělí na dvě skupiny v závislosti na způsobu, jakým dochází ke snímání otisku prstu. Tyto dvě skupiny jsou:

- Snímače pracující na bázi snímání posuvem prstu přes aktivní plochu snímače
- Snímače pracující na bázi snímání dotykem prstu k aktivní ploše snímače

První metoda se stala velmi populární v přenosných zařízeních, protože poměrně výrazně šetří prostor potřebný pro zabudování snímače do zařízení. Snímač tak nemusí mít aktivní plochu přes celou délku bříška prstu a aktivní plocha může být redukována a to až teoreticky na velikost pouhé jedné řady snímacích bodů [2]. Snímání je vykonáno posuvem prstu po aktivní snímací ploše senzoru.

Snímač v periodických intervalech snímá a zasílá na výstup snímek představující jednu část celkového obrazu otisku prstu. Tyto části jsou softwarovou cestou spojeny a vzniká celkový snímek otisku prstu (Obr. 8).



Obr. 8: Snímač pracující na bázi snímání posuvem prstu přes aktivní plochu snímače [2]

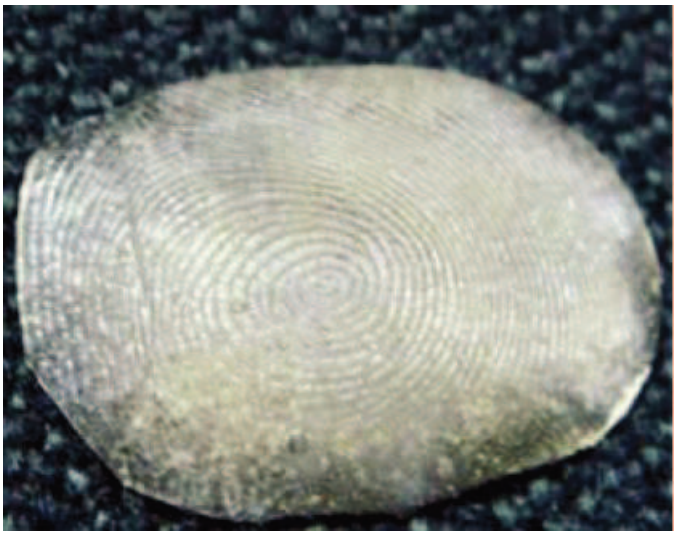
Podmínkou tohoto řešení je však pokud možno konstantní rychlost posuvu prstu přes snímač, což je však málokdy zcela dosažitelné. Vzhledem k této podmínce nemůže být pro zpětnou korekci při změnách rychlosti pohybu prstu redukována velikost snímací plochy na pouhou jednu řadu snímacích bodů, ale je nutné přidat několik řad snímacích bodů navíc právě pro možnost zpětné korekce při náhlých změnách rychlosti. Také při tomto způsobu řešení přichází v úvahu zkreslení vznikající vlivem rozpínající a stahující-se kůže na špičkách prstů během posuvu po aktivní ploše snímače [2]. Tyto dvě záporné vlastnosti bohužel představují pro začínající uživatele snímače pracujícího na bázi snímaní posuvem prstu poměrně velké problémy a dochází často k chybným vyhodnocením.

Druhá metoda využívá plnou plochu snímače. Uživatel zde není nucen prst posouvat a nemůže docházet k chybám vlivem náhlých změn rychlosti posuvu [2]. Tímto je dosaženo maximální pohodlnosti ovládání i pro začínající uživatele. Nevýhodou takového řešení však je větší cena dotykového snímače, protože aktivní plocha musí být mnohem větší a pojmout celé bříško prstu. Nutno také vzít v úvahu potenciální nevýhodu proti snímači pracujícího na bázi snímaní posuvem, kdy dochází k samovolnému čištění aktivní plochy snímače během každého sejmutí otisku. Na dotykových snímačích mohou v průběhu používání ulpívat nečistoty, které se poté samozřejmě projeví i ve výstupním obrazu otisku prstu.

2.2.6 **Detekce falešného otisku**

Potenciální hrozbou v bezpečnosti využívání přístupových systémů na principu snímání otisku prstu je možnost vytvoření falzifikátu původního otisku, který bude předložen snímacímu zařízení [1]. Takový falzifikát, bude-li patřičně vyhovující kvality, může umožnit útočníkovi přístup do systému. Falešný otisk prstu je relativně obtížné detekovat a představuje reálnou hrozbu pro přístupové systémy

založené na snímání otisků prstů. Útočníkovi postačí získat otisk pověřené osoby (teoreticky i např. latentní otisk prstu po předchozím použití dotykového snímače) a takto získaný otisk využije k vytvoření falzifikátu z vhodného materiálu. Mezi tyto materiály patří jakékoliv, které svým fyzikálními vlastnostmi napodobují vlastnosti lidské kůže na prstech. Jde např. o silikon, latex, želatinu nebo vosk [5]. Takto vytvořený falzifikát otisku (Obr. 9) může být dostatečně podobný s originálním otiskem prstu uživatele a software jej vyhodnotit jako autentický.



Obr. 9: Falešný otisk z želatiny[4]

Zjištění, zda je použit otisk originální nebo jeho falzifikát lze teoreticky docílit detekováním pulzu srdce během snímání otisku prstu nebo detekováním potu na pórech hřebenu papilárních linií, či elektrické vodivosti plochy prstu [1]. Mezi nejjednodušší způsoby detekce živého prstu patří snímání teploty prstu na aktivní ploše senzoru. Falzifikát bude mít teplotu pokojovou, tedy v normálním prostředí nižší, než živý lidský prst. Alternativní přístup k detekci falzifikátu je hledání přítomnosti zápachu chemikálií použitých pro vytvoření falešného otisku pomocí elektronických čidel pachu. Zcela jiný přístup je pak samozřejmě kombinace vstupních údajů pro vyhodnocení identity osoby do přístupového systému s nějakou jinou formou biometrické identifikace (např. snímání a rozpoznávání obličeje).

2.3 Způsoby vyhodnocování otisků

Vyhodnocovací algoritmus je klíčovou komponentou celého přístupového systému. Zajišťuje porovnávání otisku prstu uživatele systému s kolekcí vzorů otisků uložených v databázi. V případě, že algoritmus najde dostatečnou podobnost nebo shodu vstupního otisku s otiskem vzorovým, pak může

uživatele vpustit do systému. Většina algoritmů vyhodnocování otisků nepracuje s původními otisky, ale verifikuje vstupní otisk převedením do formy vhodné pro porovnávání jednotlivých jedinečností [2]. Tento proces se pro daný konkrétní algoritmus označuje jako získání jedinečností z otisku. Taktéž existující databáze vzorů nebývá tvořena původními černobílými otisky, ale jejich upravenými podobami, které obsahují již z původního otisku získané jedinečnosti. Takovým způsobem lze významně zrychlit průběh verifikace vstupního otisku s kolekcí otisků vzorových.

Algoritmus vyhodnocování musí počítat s určitou chybovostí při porovnávání, protože získané otisky jsou zatíženy problémem fyzické součinnosti uživatele přístupového systému, který musí poskytnout snímací komponentě pokud možno stále stejně dislokovaný prst na aktivní ploše snímače. V praxi je toto velmi obtížně dosažitelné a vznikají v různé míře pokaždé jinak zkreslené snímky otisku stejného prstu. Mezi některá nejznámější zkreslení, s jakými je nutno počítat, patří:

- Rotace prstu

  Uživatel velmi často pokládá prst na snímací plochu senzoru jinak natočen. V praxi lze dosáhnout různého natočení prstu a to až v úhlu 20° proti zamýšlenému středu senzorické plochy, se kterou výrobce senzoru počítal [2]. Toto natočení prstu samozřejmě ovlivňuje i celkový výsledný snímek otisku prstu, který senzor vyprodukuje. Algoritmus musí s tímto typem zkreslení vždy počítat a případně proces získání jedinečností a verifikace otisku opakovat pod různě natočeným úhlem původního snímku.
- Rozložení prstu na snímací ploše

  Uživatel nepřesně položí prst na snímací plochu senzoru. Rozpoložení prstu přitom hraje klíčovou roli na získání přesného snímku otisku prstu. I poměrně malá změna v rozložení prstu na aktivní ploše senzoru zde vytvoří velkou změnu ve výsledném snímku, protože aktivní senzorická plocha je poměrně malá a je vytvořena specificky proto, aby pojala pouze bříško prstu, kde se identifikační papilární linie nacházejí [2]. I velmi malá změna v rozpoložení prstu na aktivní ploše proto může způsobit, že se část papilárních linií prstu dostane mimo hranici této aktivní plochy a senzor nebude schopen jí zachytit. Takový problém samozřejmě nelze řešit softwarovou cestou, ale je nutné zajistit, aby uživatel pokládal prst pokud možno na střed aktivní plochy snímače.Toho lze dosáhnout například vytvořením drážkové plochy se speciálním vybráním přímo pro prst. Tato plocha je potom zakončena mírně za aktivní plochou snímače a zajišťuje tak přesné položení bříška prstu na střed aktivní plochy snímače s minimální možností úhybu prstu do stran.

- Nečistoty na aktivní ploše snímače
  Používáním dotykových snímačů dojde po nějaké době k ulpění nečistot na aktivní ploše snímače. Tyto nečistoty mohou mít formu latentních otisků, potu, nečistoty prostředí apod. Všechny nečistoty na aktivní ploše vytvářejí zkreslení a ruchy ve vytvořeném otisku prstu [2]. Algoritmus může z části některá zkreslení filtrovat, avšak pokud je aktivní plocha senzoru příliš znečištěná, může to vést až k chybně vyhodnoceným otiskům a nemožnosti se do systému autentifikovat. Náprava je zde jednoduchá a je založena na pravidelném čištění aktivní plochy snímače.
- Zkreslení vlivem akce uživatele
  Uživatel např. neúmyslně aplikuje tlak na prst položený na aktivní ploše snímače. Tlakem prstu dojde k mírné deformaci kůže na prstech a s ní souvisejících změnách na papilárních liniích. Takové změny se přenesou i na výstupní obraz otisku prstu a vznikne tímto zkreslení. Algoritmus může do jisté míry tento typ zkreslení eliminovat, avšak nejlepší prevencí proti problémům s tímto zkreslením je důkladná a detailní instruktáž uživatele o správném použití senzoru přístupového systému.

Vzhledem k těmto zkreslením při porovnávání bylo nutné zavést do biometrických systémů ukazatele, které porovnávají poměry četnosti chybných detekcí algoritmů [1]. Jde především o:

1. FAR (False Acceptance Rate) - poměr vyhodnocení otisků, které byly nesprávně vyhodnocené jako autentické vzhledem k celkovému počtu vyhodnocení.
2. FRR (False Rejection Rate) - poměr vyhodnocení otisků, které byly nesprávně vyhodnocené jako neautentické vzhledem k celkovému počtu vyhodnocení.
3. ERR (Equal Error Rate) - poměr shody mezi FAR a FRR

Na základě těchto ukazatelů mohly být vyhodnocovací algoritmy testovány a následně nesprávně vyhodnocující nebo jinak nedokonalé postupy s vysokými hodnotami FAR nebo FRR vynechány, či upraveny do funkční podoby.

2.3.1 Algoritmus pracující na základě extrakce jedinečností

Nejvíce rozšířeným algoritmem pro porovnávání vstupních otisků proti jejím vzorům je algoritmus porovnávající extrahované jedinečnosti z otisku. Tento algoritmus pracuje na principu porovnávání podobností jedinečností. Činnost algoritmu je rozdělena na tři oddělené operace:

1. Zarovnání otisku

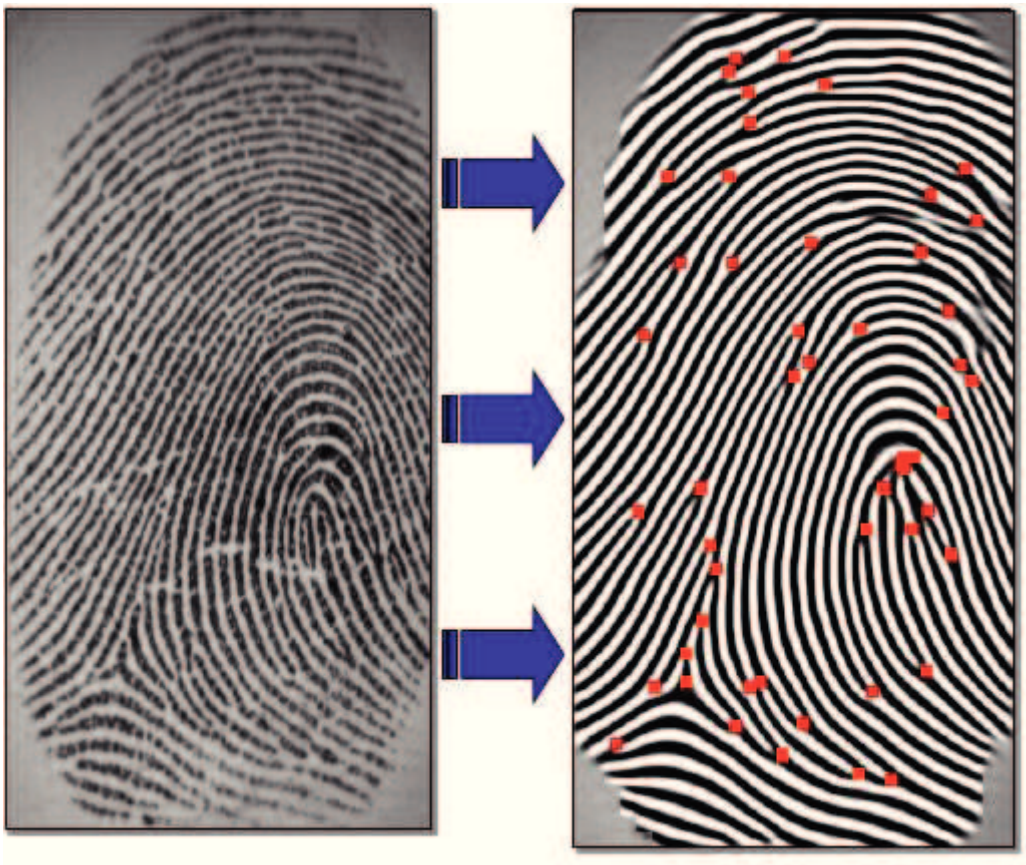
   V této fázi dochází k zarovnání otisku proti jeho vzoru. Jde o úkon potřebný, protože jak už bylo výše uvedeno, tak uživatel systému pokládá prst na aktivní plochu pokaždé jinak natočený a pokaždé mírně posunutý. Tento krok tedy slouží k eliminaci tohoto posuvu a natočení. Pro tyto účely se nejčastěji používá algoritmus zobecněné Houghovy transformace[2].

2. Párování jedinečností otisku

   Jedinečnosti jsou extrahovány z drobných detailů (Obr. 10) na hřebenech papilárních linií. Zejména jde o náhlá zakončení, začátky nebo bifurkace hřbetů papilárních linií [5]. Z takových drobných detailů je vytvořena mapa jedinečností, která definuje relativní poměry vzdáleností a orientací mezi jednotlivými jedinečnostmi. Algoritmus poté pouze porovnává, jaký existuje největší možný počet shod mezi získanou extrahovanou mapou jedinečností a vzorovou mapou jedinečností uloženou v databázi. Jelikož ale činnost zarovnávacích algoritmů není dokonalá, tak se velmi často stává, že se otisk nezarovná naprosto stejně. Taktéž snímací schopnosti senzoru přístupového systému nejsou schopny vždy vyprodukovat stejně detailní a korektní snímek. Chybovost nedokonalým zarovnáním vstupního snímku je nutno v porovnávacím algoritmu zohlednit a zanést do výpočtu určitou toleranci v úhlu natočení a vzdálenosti mezi drobnými detaily v získané a vzorové mapě jedinečností. Vhodně nastavená hranice při vyhodnocování je např. limit natočení +- 20° a limit vzdálenosti +-15 pixelů od referenčního bodu, kde se jedinečnost v mapě nachází.

3. Vyhodnocení autentičnosti otisku

   Z předchozího kroku je získán počet spárovaných jedinečností, z kterého je následně vytvořen procentuální poměr správně spárovaných detailů. Tento poměr je potřeba porovnat s nastaveným limitem, protože tento, ale i celá řada dalších porovnávacích algoritmu, nevracejí uživateli procentuální shodu, ale binární hodnotu indikující, zda je otisk autentický se vzorem nebo není. Limit shody musí být pečlivě nastaven, protože příliš nízká hodnota vytváří v systému velké množství chybně vyhodnocených otisků jako správných (vysoký FAR), ale příliš vysoká hodnota naopak zanáší do systému riziko špatně vyhodnocených autentických otisků jako nesprávných (vysoký FRR) [2]. Z toho také plyne, že příliš nízká hodnota umožňuje potenciálnímu útočníkovi snadněji prolomit přístupový systém pomocí falešného otisku, naopak vyšší hodnota limitu poskytuje systému potenciálně vyšší míru zabezpečení.



Obr. 10: Extrakce jedinečností z otisku prstu [5]

2.4 Způsoby uchovávání vzorových otisků

V každém přístupovém systému musí být uloženy vzorové otisky, na základě kterých je možné pověřené osoby identifikovat. Vzory otisků mohou být, dle povahy systému, uloženy buďto přímo v přístupovém systému sestávajícím z jediného modulu, anebo je lze uchovat v nadřazeném zařízení (serveru), který vzorové otisky nejčastěji uchovává ve formě relační databáze. Vzhledem k tomu, že v České republice se otisk prstu, jakožto biometrická informace, považuje dle zákona č. 101/2000 Sb.(Zákon o ochranně osobních údajů) za citlivý osobní údaj, který umožňuje přímou identifikaci subjektu, tak je také nutné s tímto otiskem prstu jako s citlivým osobním údajem zacházet. Stanovisko Úřadu pro ochranu osobních údajů č. 3/2009 zde navíc stanoví, že:

> *Je žádoucí, aby šablony byly před uložením v systému zpracovávány matematickými operacemi (kódování, algoritmy nebo hash funkce) tak, aby nebyly volně čitelné nebo zpětně rekonstruovatelné.*

Z takto vzneseného požadavku je nadmíru jasné, že není možné v systému uchovávat biometrické údaje (tedy i otisky prstů) v nezměněné podobě, z jaké je systém získá přímo ze snímače

otisku prstů. Požadavek však lze považovat za diskutabilní u přístupových systému sestávajících z jediného modulu, které jsou často postaveny pouze z jednoho monolitického počítače, jenž lze dostatečně zabezpečit na hardwarové úrovní proti možnému čtení obsahu pracovní paměti. Nicméně jde o požadavek zadaný správním úřadem a je tak nutné jej respektovat. Ochrana existujících vzorových otisku prstů proti možnému čtení neautorizovanou osobou je obvykle realizována dvěma způsoby. Jedná se o reverzibilní algoritmy pro možnou rekonstrukci původního snímku získaného snímačem otisků a nereverzibilní algoritmy, které vstupní data převedou do podoby, ze které již není možné původní snímek rekonstruovat.

2.4.1 **Reverzibilní uchovávání otisků**

Metoda reverzibilního uchovávání vzorových otisků umožňuje převést otisky do šifrované podoby, ze které není možné zpětně extrahovat původní snímek bez znalosti použitého kryptografického algoritmu a jeho klíče. Pro šifrování se používají standardní symetrické šifrovací algoritmy (např. AES nebo 3DES). Ochranu založenou na reverzibilních šifrovacích algoritmech lze obecně rozdělit na dva způsoby použití kryptografického klíče:

a) Klíčem je běžné heslo

   Klasická metoda šifrování za pomocí tajné fráze (hesla) je nejjednodušší formou ochrany otisků před zneužitím. Vstupní otisk je za pomocí kryptografického algoritmu šifrován klíčem, jež může být poskytnut i v textové formě. Výstupní šifrovaná podoba otisku je poté uložena v databázi. Pokud potenciální útočník získá přístup k databázi, tak za předpokladu použití dostatečně silného hesla, které bude odolné vůči slovníkovým útokům, a předpokladu využití dostatečně robustního šifrovacího algoritmu, je pro útočníka prakticky nemožné rekonstruovat původní snímek otisku bez znalosti šifrovacího klíče [2]. Taková metoda je proto velmi jednoduchým a vhodným základním prvkem ochrany vzorových otisků v přístupovém systému. Nevýhodou šifrování za pomocí hesla však je fakt, že v případě přístupového systému, u kterého se předpokládá využití více osobami, lze jen velmi obtížně využít pro každý uložený otisk unikátní heslo. Proto se pro šifrování všech otisků v databázi využívá většinou jediné hlavní heslo. V případě, že se útočníkovi podaří takové heslo získat, pak ihned získává přístup ke všem otiskům v celé databázi přístupového systému. Nabízí se zde uchování šifrované podoby jedinečného hesla v databázi společně s šifrovaným otiskem, ale toto heslo by nakonec stejně muselo být šifrováno jediným hlavním heslem, se kterým aplikace může pracovat.

b) Klíčem je hash samotného otisku

Reverzibilní metoda založená na šifrování výstupem hashovací funkce významně vylepšuje metodu založenou na šifrování otisku heslem o unikátnost kryptografického klíče. Vstupní otisk je nejprve podroben zpracování hashovací funkcí, která vygeneruje klíč o fixní velikosti, který koresponduje právě jednomu otisku [2]. Získaný klíč se podobně jako v předchozím případě použije k zašifrování snímku otisku prstu a výstupní šifrovaná podoba se uloží do databáze. V případě dešifrování se opět nejdříve snímek otisku využije k vygenerování kryptografického klíče a až poté dojde k samotnému dešifrování. Použitá hashovací funkce musí být dostatečně odolná vůči drobným změnám na papilárních liniích. V opačném případě by mohlo dojít k nemožnosti dešifrovat původní obsah, protože hashovací funkce by generovala rozdílný kryptografický klíč. Výhoda této metody je právě v jedinečnosti klíče, který je pro šifrování použit. Pokud se útočníkovi povede k nějakému jednomu konkrétnímu otisku získat použitý šifrovací klíč, pak nedojde k ohrožení ostatních šifrovaných otisků, protože, jak již bylo řečeno, kryptografický klíč je pro každý otisk jedinečný. Nevýhoda spočívá v obtížné implementaci hashovací funkce, která musí splňovat současně dvě kritéria. Hashovací funkce musí být současně dostatečně unikátní (každý otisk musí produkovat rozdílný výstupní hash) a také dostatečně odolná vůči drobným změnám na papilárních liniích vlivem např. úrazu.

2.4.2 Nereverzibilní uchovávání otisků

Nereverzibilní transformace otisků využívá přímo hashovací funkci generující výstupní hash, který je napříč celou databází otisků unikátní [2]. Typický průběh ukládání je uskutečněn tak, že výstup ze snímače otisků je zpracován hashovací funkcí a výstupní hash je uložen do databáze ostatních hashů. Takový výstupní hash je sestaven z jedinečností ve vstupním snímku otisku a jde o transformaci informačně ztrátovou. Potenciální útočník sice může získat obsah databáze, avšak původní snímek nikdy zpětně nebude schopen zrekonstruovat. Tento způsob transformace však naráží na stejné problémy jako v případě využití hashovací funkce při získávání unikátního biometrického klíče pro reverzibilní metodu uchovávání otisků. Vzhledem k tomu, že i dva otisky, které jsou ze snímače získány ihned po sobě nejsou vzhledem k odlišnému rozpoložení bříška prstu na aktivní ploše snímače zcela totožné, tak je poměrně obtížné vytvořit hashovací funkci, která bude schopna produkovat vždy stále stejný výstupní hash pro stejný snímaný prst. Jedním z takových hashovacích algoritmů je např. algoritmus hledající nejkratší cestu grafem (Obr. 11). Graf je tvořen body označující místa jedinečností papilárních linií. Cílem algoritmu je najít nejkratší cestu ze středu otisku prstu

(jádro) přes všechny detekované jedinečnosti papilárních linií a z této získané cesty je následně produkován výstupní hash [6]. Podstatnou výhodou této hashovací funkce je určitá odolnost vůči drobným změnám na papilárních liniích a velká odolnost vůči natočení (rotaci) snímku, což se projevuje v poměrně stabilních hodnotách výstupního hashe pokud je vstupní snímek zatížen nějakou formou zkreslení.

Obr. 11: Hashovací funkce hledající nejkratší cestu grafem [6]

3 Návrh přístupového systému

Cílem této diplomové práce je navrhnout a realizovat zařízení, které bude schopné vykonávat funkci přístupového systému. Systém bude autorizovat uživatele na základě snímání otisku prstů, které budou vyhodnocovány v nadřazeném systému. Při srovnání s komerčně dostupnými přístupovými systémy byly zavedeny následující minimální požadavky, které systém bude muset splňovat:

- Systém bude sestaven ze dvou hlavních komponent - nadřazeného systému a vstupně výstupní jednotky
- Otisky prstů budou vyhodnocovány v nadřazeném systému
- Uživateli systému budou zobrazeny informativní hlášení o stavu autorizace
- Systém bude schopen ovládat externí zátěž (např. elektrický zámek)
- Nadřazený systém bude schopen přijímat a vyhodnocovat data z více jednotek

Je zřejmé, že vhodným nadřazeným systémem (s dostatečnou výpočetní kapacitou) je minimálně osobní počítač, ideálně pak server dedikovaný přímo pro autoritativní funkci nadřazeného systému. Činnost nadřazeného systému proto bude realizována jako standardní aplikace pro platformu x86 osobního počítače. Z hardwarového úhlu pohledu tedy zbývá navrhnout vhodnou vstupně - výstupní jednotku primárně pro snímání otisků a ovládání externí zátěže.

V této sekci se budeme dále zabývat právě teoretickým návrhem popsaného přístupového systému.

3.1 Komponenty vstupně - výstupní jednotky

Pro zajištění nastavených minimálních požadavků pro přístupový systém je nutné, aby vstupně výstupní jednotka obsahovala jisté komponenty, které splnění těchto požadavků naplní. K zajištění schopnosti přijímat otisky prstů bude nutné zvolit vhodný vstupní snímací prvek. K možnosti zobrazení stavových informací pro uživatele systému bude nutné, ať jednotka obsahuje nějakou formu zobrazovače. Dále bude nezbytné řešit vhodnou formu komunikace s nadřazenou jednotkou. Vzhledem k požadavku schopnosti vyhodnocování a ovládání více vstupně výstupních jednotek se jako vhodná forma fyzického komunikačního rozhranní jeví počítačová síť (Ethernet), a proto bude nezbytné zvolit vhodný ethernetový řadič, který bude schopen signály fyzické vrstvy ethernetu transformovat do vhodné podoby pro další zpracování. Poslední hardwarovou komponentou vstupně - výstupní jednotky je mikroprocesor, který bude schopen celkovou činnost jednotky zajišťovat a

ovládat.V této stati se zaměříme na výběr vhodných komponent pro zajištění minimální požadované funkčnosti vstupně - výstupní jednotky.

3.1.1 Výběr snímače otisku prstu

Snímač otisku prstu představuje v námi zamýšleném přístupovém systému klíčovou komponentu, protože generuje vstupní biometrická identifikační data, které nadřazený systém musí zpracovat a vyhodnotit. Ideálním snímačem pro náš přístupový systém bude snímač kapacitní pracující na bázi snímání dotykem prstu k aktivní ploše snímače. Takový snímač má relativně přijatelné nároky na prostorové dispozice a taktéž eliminuje artefakty při snímání vznikající u snímače pracujícího na bázi snímání posuvem prstu přes aktivní plochu snímače. V neposlední řadě má taktéž použití kapacitního snímače výborný poměr výkon/cena ve srovnání s ostatními způsoby snímání otisků prstů.

Vhodným komerčním produktem byl proto zvolen senzor FPC1011F3 vyráběný švédskou společností Fingerprint Cards AB. Jedná se o kapacitní senzor s rozlišením výstupního snímku 152 x 200 obrazových bodů [7]. Senzor komunikuje přes rozhranní SPI a pro daný účel má prakticky zanedbatelnou vlastní spotřebu, jejíž hodnota činí přibližně 23mW.

3.1.1.1 Popis komunikace se snímačem otisku prstu

Snímač otisků používá pro komunikaci výhradně sériové SPI rozhranní, z kterého získává také hodinový signál pro interní účely. Senzor používá vlastní instrukční sadu pro ovládání a výčet dat [7]. Tuto instrukční sadu blíže charakterizuje následující tabulka:

Tabulka 1: Instrukční sada snímače otisků

Instrukce	*Kód instrukce*	*Popis*
rd_sensor	*0x11*	*Sejme otisk a data uloží do fronty SPI rozhranní*
rd_spidata	*0x20*	*Čtení dat z fronty SPI rozhranní*
rd_spistat	*0x21*	*Čtení interního STATUS registru*
rd_regs	*0x50*	*Čtení všech registru, data jsou uložena do fronty SPI rozhranní*
wr_drivc	*0x75*	*Zápis do DRIVC registru*
wr_adcref	*0x76*	*Zápis do ADCREF registru*
wr_sensem	*0x77*	*Zápis do SENSEMODE registru*
wr_fifo_th	*0x7C*	*Zápis do FIFO_TH registru*
wr_xsense	*0x7F*	*Zápis do XSENSE registru*
wr_ysense	*0x81*	*Zápis do YSENSE registru*
wr_xshift	*0x82*	*Zápis do XSHIFT registru*
wr_yshift	*0x83*	*Zápis do YSHIFT registru*
wr_xreads	*0x84*	*Zápis do XREADS registru*

Snímač umožňuje celou řadu konfigurací včetně např. pouze selektivního snímání určité části aktivní plochy snímače. Bližší specifikace jednotlivých registrů snímače včetně veškerých funkcí jsou k dispozici v datovém listu výrobku.

Samotné čtení snímku otisku prstu pak vypadá následovně:

1. Nejprve je nutné aktivovat snímač signálem CS (Chip Select).
2. Na začátku každého výčtu je nutné nastavit 3 registry na jejich základní hodnoty, konkrétně jde o:
    - registr DRIVC na hodnotu 255
    - registr ADCREF na hodnotu 3
    - registr SENSEMODE na hodnotu 0
3. Poté je možné sejmout otisk pomocí instrukce rd_sensor. Ve výchozím nastavení konfiguračních registrů je sejmuta celková plocha snímače.
4. Nyní je nutné vyčkat definovaný počet cyklu předtím, než se ve frontě SPI rozhranní objeví data snímku. Vzhledem k tomu, že celý senzor je taktován hodinovým signálem SPI, lze data číst nejdříve po 365 hodinových taktech SPI sběrnice pomocí instrukce rd_spidata.
5. Po kompletním vyčtení celého snímku (152 * 200 bajtů) je možné snímač deaktivovat signálem CS (Chip Select).

3.1.2 Výběr zobrazovací jednotky

Zobrazovací jednotka tvoří základní komunikační kanál s uživatelem systému. Umožňuje zobrazovat informativní zprávy, případně stavové informace o úspěchu nebo neúspěchu autorizace do přístupového systému. Pro daný systém musí být zobrazení dostatečně přehledné a čitelné i za extremních světelných podmínek. Toto bezezbytku splní podsvícený displej z tekutých krystalů (LCD).

K tomuto účelu byl vybrán LCD displej značky MIDAS řady MCCOG22005A6W, což je dvouřádkový textový displej s LED podsvícením, který disponuje na každém řádku 20 znaky [8]. Displej komunikuje přes I2C rozhranní a je relativně dobře čitelný jak na přímém slunečním světle, tak v úplné tmě, kdy zdrojem světla zajišťující aspoň minimální čitelnost je LED podsvícení.

3.1.2.1 ***Popis komunikace se zobrazovačem***

LCD zobrazovač využívá interní řadič LCD displeje ST7036i a komunikuje pouze přes sběrnici I2C [8]. Každé zařízení na této sběrnici má přidělenou I2C adresu. Pro zvolený zobrazovač byl výrobcem přidělen rozsah I2C adres 0x78 - 0x7E. Zobrazovač využívá pro komunikaci vlastní instrukční sadu, která je důkladně popsána v datovém listu k zobrazovači a nevyžaduje tak další komentář. Komunikace se zobrazovačem sestává z jednotlivých příkazů. Každý příkaz vyžaduje řídící a datový bajt, kde řídící bajt udává, zda se bude jednat o vnitřní konfigurační instrukci nebo zápis do znakové paměti zobrazovače a taktéž, zda po dokončení právě probíhajícího příkazu bude zápis dat pokračovat dalším příkazem, či bude další zapisování dat ukončeno. Datový bajt obsahuje přímo zapisovaná data a v případě zápisu vnitřní konfigurační instrukce její signaturu nebo eventuálně v případě přímého zápisu do textové paměti zobrazovače samostatné ASCII znaky.

Podsvícení displeje není softwarově konfigurovatelné a vyžaduje pro řízení a provoz externí komponenty.

3.1.3 Výběr ethernetového řadiče

K zajištění funkce komunikace klientské vstupně výstupní jednotky s nadřazeným systémem je nutné zvolit vhodný řadič, který umožní komunikaci po počítačové sítí (Ethernetu). Ethernetový řadič je zařízení převádějící modulované signály fyzické vrstvy počítačové sítě do podoby vhodné pro zpracování mikrokontrolérem.

Pro tento účel byl vybrán ethernetový řadič ENC424J600 od firmy Microchip Technology Inc. Jde o SPI řadič kompatibilní se specifikací IEEE 802.3 s předprogramovanou unikátní MAC adresou. Řadič podporuje standard Fast Ethernet 100BASE-T pro maximální komunikační rychlost až 100 Mbit/s [9]. Výrobce poskytuje pro tento řadič softwarovou podporu v podobě již hotového zásobníku protokolů TCP/IP včetně vlastního ovladače pro samotný ethernetový řadič [10].

3.1.4 Výběr mikrokontroléru

Pro zajištění funkčnosti systému jako celku je nezbytné vybrat vhodný monolitický počítač - mikrokontrolér. Ten musí být schopen komunikovat se všemi zvolenými periferiemi a zajistit stabilní a reliabilní funkci klientské jednotky přístupového systému. Činnost mikrokontroléru tedy bude vázána na přečtení snímku otisku uživatele, jeho zaslání v šifrované podobě pomocí ethernetového řadiče do nadřazeného systému a následně příjmu odpovědi nadřazeného systému na umožnění,

případně zamezení přístupu k danému prostředku. Zvolený mikrokontrolér tedy bude muset splňovat několik kritérií:

- dostatečný počet vstupně/výstupních bran
- schopnost komunikace po sběrnici SPI a I2C
- dostatečný výpočetní výkon pro šifrovaný přenos dat

Jako vhodný mikrokontrolér splňující zadané podmínky byl vybrán výrobek společnosti Microchip Technology Inc. typ dsPIC33EP512GM304. Jedná se o 16 bitový mikrokontrolér s rychlostí zpracování instrukcí až 70 MIPS [11]. Mikrokontrolér obsahuje řadu dedikovaných modulů, mezi kterými jsou i moduly pro komunikaci se sběrnicemi SPI a I2C. Zvolený mikropočítač je také kompatibilní s TCP/IP zásobníkem stejného výrobce.

3.2 Funkce aplikace nadřazeného systému

Primární činností aplikace nadřazeného systému bude vyhodnocování otisků z jednotlivých klientských jednotek. K naplnění této požadované funkce bude nezbytné využít specializovaného softwaru určeného k extrakci jedinečností a identifikaci otisku. Předpokladem bude pouze uchovávání extrahovaných dat z otisku bez možnosti reverzibilního sestavení původního otisku. Splněním této podmínky dojde k výraznému zvýšení bezpečnosti celého systému a prakticky k vyloučení možnosti zneužití již uchovaného otisku prstu. K hlavním požadavkům aplikace nadřazeného systému tedy bude patřit:

- Příjem otisku, jeho vyhodnocení a následná interakce s klientskou jednotkou
- Schopnost obsluhy více klientských jednotek
- Přidávání, odebírání a editace uživatelů přístupového systému
- Možnost selektivního povolení přístupu osoby k dané místnosti, či prostředku
- Uchování informací o přístupech jednotlivých osob k prostředkům včetně času události
- Zajištění šifrované komunikace s klientskou jednotkou pro zamezení odposlechu citlivých dat
- Předpokladem je uložení informací o osobách a extrahovaných otiscích v databázovém systému

Vzhledem ke komplexnosti softwaru pro vyhodnocování otisků prstů bude pravděpodobně nutné využít již existujících knihoven či aplikačních rámců. Konkrétní programovací jazyk, který bude použit pro implementaci aplikace nadřazeného systému tedy bude značně vázán na použitý jazyk, ve kterém zvolená knihovna pro vyhodnocování otisků prstů bude naimplementována. Serverová aplikace pak bude muset být s touto implementací použité knihovny plně kompatibilní a schopna spolehlivé a bezpečné funkce s klientskými jednotkami.

4 Realizace přístupového systému

Ke splnění zadání této diplomové práce bude nezbytné navržený přístupový systém realizovat. Konkrétně půjde o navržení obvodového zapojení a sestavení klientské jednotky pro interakci s uživatelem systému, implementaci firmwaru pro tuto klientskou jednotku a také i implementaci softwaru pro nadřazený systém. V následující sekci se proto zaměříme na implementaci navrženého přístupového systému včetně otestování jeho spolehlivosti a celkové funkčnosti.

4.1 Sestavení klientské jednotky

Z individuálně zvolených klíčových komponent klientské jednotky bylo navrženo a sestaveno obvodové schéma zapojení. Zapojení vychází z typických nebo doporučených zapojení dílčích komponent, která jsou publikována v jednotlivých datových listech k periferiím a nepředpokládá se, že vyžadují dalšího komentáře. Celkové schematické znázornění zapojení klientské jednotky je k dispozici v elektronické příloze této diplomové práce.

Ke stávajícímu způsobu autorizace k přístupovému systému, tedy autentizace na základě rozpoznávání otisku prstu, byla z demonstračních důvodu možné existence zpětné kompatibility s již existujícími systémy přidána podpora autentizace osoby na základě držení specifické věci. Konkrétně jde o identifikační RFID čipovou kartu pracující na principu bezdrátového přenášení sekvence bajtů, které ji jednoznačně identifikují napříč spektrem všech vyrobených karet jednoho výrobce. Zvolená RFID čipová karta pracuje na frekvenci nosné vlny 125 kHz a je relativně bezpečná proti zneužití, přičemž vzdálenost, při které je možné uskutečnit výčet unikátních identifikačních dat nepřesahuje, za normálních okolnosti, přibližně 10 cm. Na základě tohoto požadavku bylo nutné přidat do návrhu klientské jednotky periferii, která bude schopna s RFID identifikační čipovou kartou komunikovat. Tímto zařízením je RFID modul pro kmitočet nosné vlny 125 kHz. Do schématu zapojení proto bylo dodatečně přidáno zařízení EM-18 od výrobce Shandong Mingwah Aohan Smart Tech Co., Ltd., které požadavek na komunikaci se zvolenou RFID čipovou kartou bezezbytku splní [12].

4.2 Firmware pro klientskou jednotku

Z požadavků vzniklých v návrhu klientské jednotky byl vytvořen firmware pro mikrokontrolér, který je obsažen v klientské jednotce a provádí veškeré operace spojené s její funkcí. V této části se zaměříme na firmware pro mikrokontrolér klientské jednotky. Zejména se blíže

seznámíme s některými částmi firmwaru spojenými s komunikací s externími komponentami klientské jednotky přístupového systému.

Celý firmware je implementován v jazyce ANSI C a je kompatibilní s překládačem Microchip MPLAB XC16, pro 16 bitové mikrokontroléry téhož výrobce [13]. Firmware využívá již existující implementace zásobníku protokolů TCP/IP od výrobce Microchip Technology Inc. pro komunikaci s externím ethernetovým řadičem pod názvem "Microchip TCP/IP Stack" [10]. Použitá verze tohoto zásobníku protokolů TCP/IP je verze 5.42.08, vydána dne 15. června 2013.

Kompletní sestavení firmwaru včetně zdrojových kódů je k dispozici v elektronické příloze této diplomové práce. Zdrojové kódy jsou přiloženy ve formě projektu pro vývojové prostředí MPLAB X verze 2.35 a vyšší.

4.2.1 Řízení displeje pomocí mikrokontroléru

Displej klientské jednotky představuje v přístupovém systému jedinou zpětnou vazbu, kterou jeho uživatel získá. Je proto velmi důležité, aby komunikace s uživatelem formou zobrazení informačních a varovných zpráv na displeji byla dostatečně přehledná a intuitivní. K dosažení tohoto cíle bude zapotřebí dostatečně robustní implementace části firmwaru zabývajícího se právě výpisem zpráv na displej.

Použitý displej komunikuje s mikrokontrolérem pomocí sběrnice I2C, která byla vyvinuta v roce 1982 společností Philips Semiconductor. Sběrnice I2C je dvouvodičová, poloduplexní sběrnice určená pro přenos dat zejména mezi komponentami vestavěných systému, a to na relativně krátké vzdálenosti [14]. První ze dvou vodičů sběrnice bývá označen SCL (Serial CLock) a generuje hodinový signál potřebný pro činnost sběrnice, druhý vodič, který se obvykle označuje jako SDA (Serial DAta), přenáší samotná data. Zařízení se na sběrnici dělí na zařízení typu "Master" a "Slave". Periférie typu "Master" jako jediná generuje hodinový signál pro celou sběrnici.

V našem konkrétním případě je zařízením typu "Master" přímo mikrokontrolér a zařízení "Slave" je připojená zobrazovací jednotka. Obsluhu sběrnice řeší část firmwaru, nikoliv přímo dedikovaný modul mikrokontroléru. Kód řešící obsluhu I2C sběrnice se nachází jako separovaný modul jazyka C v souboru zdrojového kódu "I2C.c" a hlavičkovém souboru definic "I2C.h". Pro správnou funkci modulu je zapotřebí definovat tyto povinné konstanty:

1. konstanty `I2C_SCL_TRIS` a `I2C_SDA_TRIS` definující TRIS registry vývodů mikrokontroléru, které jsou použitý pro vodiče SDA a SCL

2. konstanty `I2C_SCL_PORT` a `I2C_SDA_PORT` definující PORT registry vývodu mikrokontroléru, které jsou použity pro vodiče SDA a SCL
3. konstanty `I2C_SCL_LAT` a `I2C_SDA_LAT` definující LAT registry vývodu mikrokontroléru, které jsou použity pro vodiče SDA a SCL

Ukázková implementace těchto konstant může vypadat např. následovně:

```
//konstanty definující TRIS registry
#define I2C_SCL_TRIS       TRISBbits.TRISB6
#define I2C_SDA_TRIS       TRISBbits.TRISB5

//konstanty definující PORT registry
#define I2C_SCL_PORT       PORTBbits.PORTB6
#define I2C_SDA_PORT       PORTBbits.PORTB5

//konstanty definující LAT registry
#define I2C_SCL_LAT        LATBbits.LATB6
#define I2C_SDA_LAT        LATBbits.LATB5
```

Po nadefinování potřebných konstant pro provoz modulu je možné přistoupit k jeho využití v kódu. K tomuto účelu slouží následující funkce I2C modulu:

```
void I2CInitialize();
void I2CStart();
void I2CStop();
unsigned char I2CSend(unsigned char byte);
unsigned char I2CRecv(unsigned char ack);
```

Jejich funkce jsou pravděpodobně již z názvu patrné, nicméně pojďme si nyní pro přesnost vysvětlit jejich charakteristické využití v kódu. Funkce `I2CInitialize` provede inicializaci a nastavení základních logických stavů na sběrnici před jejím prvním využitím. Tato funkce musí být v kódu zavolána jako první, a to před jakýmkoliv využitím I2C modulu. Další dvě funkce v modulu jsou `I2CStart` a `I2CStop`, které slouží pro zaslání start a stop podmínek pro zahájení nebo ukončení komunikace po sběrnici. Poslední dvě funkce I2C modulu jsou `I2CSend` a `I2CRecv`. Funkce `I2CSend` slouží k zaslání jednoho bajtu po I2C sběrnici, který je specifikován jako parametr funkce,

přičemž návratovou hodnotou této funkce je získaný potvrzovací bit. Funkce `I2CRecv` slouží pro získání jednoho bajtu z I2C sběrnice, zatímco jejím parametrem je potvrzovací bit, který má být vyslán na konci přenosu. Ten může mít, stejně jako v případě funkce `I2CSend`, jednu ze dvou hodnot definovaných v hlavičkovém souboru "I2C.h", a to `ACK` nebo `NACK`.

Pro výpis na displej byl vytvořen samostatný modul ovladače použitého LCD displeje. Zdrojový kód ovladače displeje se nachází v souboru "LCD_driver.c" a hlavičkovém souboru definic "LCD_driver.h". Ovladač využívá zmíněného I2C modulu pro komunikaci s jednotkou displeje a definuje celou řadu funkcí pro výpis dat na displej, jeho konfiguraci a správu, které vycházejí ze schopnosti řadiče displeje specifikovaných v datovém listu k výrobku.

Pojďme si nyní přiblížit ty nejdůležitější funkce, které klientská jednotka využívá pro výpis informativních zpráv na displeji:

- Funkce `displayReset` a `displayInit` slouží k uvedení displeje do základního nastavení a provedení jeho inicializace. Jejich detailní popis je patrný přímo z komentářů obsažených ve zdrojovém kódu modulu ovladače displeje.
- Funkce `displayClear` provede smazání všech znaků právě zobrazovaných na displeji. Interně tato funkce provádí smazání obsahu znakové operační paměti řadiče displeje, čímž logicky dochází také ke smazání zobrazovaných znaků na displeji. Řadič displeje disponuje pro tyto účely přímo specifickou instrukcí pro smazání této paměti.
- Funkce `displayWrite` provede zobrazení jednoho znaku definovaného parametrem "data" na displeji. Tento znak bude zapsán na právě používané místo operační znakové paměti displeje. Zobrazení na displeji tedy nemusí být kontinuální, ale znak může být zobrazen na jiném místě, než znak předchozí, dojde - li ke změně aktivní adresy operační znakové paměti displeje.
- Funkce `displayWriteString` je hlavní funkcí, kterou firmware klientské jednotky používá pro výpis informací na displeji. Účelem této funkce je výpis celého řetězce znaků na displeji. Řetězec je předán funkci jako parametr, přičemž se předpokládá, že jeho délka bude zjistitelná speciálním znakem na jeho konci (tzv. nulový znak). Tento fakt eliminuje potřebu předávání parametru délky řetězce.

Zaměřme se nyní na poslední zmíněnou funkci, která představuje poměrně jednoduchý způsob výpisu jednotlivých řetězců na displeji. Celá signatura funkce vypadá následovně:

```
int displayWriteString(char *str1, char *str2);
```

Parametr str1 definuje řetězec, který se má zobrazit na prvním řádku displeje, naopak parametr str2 definuje řetězec, jehož zobrazení je požadováno na druhém řádku displeje. Návratovou hodnotou funkce je v případě chyby nenulová hodnota. Pojďme se nyní podívat na detailní činnost zjednodušené verze této funkce:

```
int displayWriteString(char *str1, char *str2)
{
    int i, ack, len;

     //je - li první řetězec funkci předán, tak dojde k přepsání
     //aktuální hodnoty zobrazené na displeji
    if(str1 != NULL)
    {
        //nejprve je nutné nastavit aktivní adresu operační znakové
        //paměti displeje na začátek prvního řádku displeje
        displaySetAddressDDRAM(LCDFIRSTLINESTART);

       //zjistíme délku řetězce a upravíme počet cyklů výpisu znaků
        len = strlen(str1);

        //provede výpis tolika znaků, kolik se nachází v řetězci
        for(i=0; i<len; i++)//
        {
            ack = displayWrite((unsigned char)str1[i]);//výpis znaku

            //pokud došlo k chybě, přerušíme činnost a vrátíme chybu
            if(ack != LCDI2CERRORSUCCESS)
                return ack;
        }

        //operační znaková paměť displeje obsahuje více paměťových
        //buněk než je znaků na displeji. Ty slouží pro operace
        //posuvu displeje. Je nutné po předchozím řetězci, který
        //mohl zasahovat až do této oblasti paměti znaky přepsat
        for(i=len; i<LCDFIRSTLINEEND; i++)
            displayWrite((unsigned char)' ');
    }

    //vracíme po úspěšném výpisu nulovou hodnotu
    return LCDI2CERRORSUCCESS;
}
```

Ovladač displeje disponuje dalšími funkcemi, které ale nejsou pro výpis informačních zpráv na displeji relevantní. Tyto funkce jsou pro případné zájemce poměrně dobře okomentovány ve zdrojovém kódu ovladače a vycházejí přímo z datového listu k displeji [8].

4.2.2 Snímání otisků ze senzoru

Primárním cílem této diplomové práce je realizace přístupového systému pracujícího na principu identifikace osob na základě snímání otisku prstu. Je patrné, že ke splnění tohoto cíle bude nezbytné zajistit ve firmwaru výčet dat ze zvoleného senzoru otisků. Tato data reprezentují snímky otisků ve stupních šedi.

Vybraný senzor otisků komunikuje s mikrokontrolérem po sběrnici SPI [7]. Tato sběrnice (na rozdíl od sběrnice I2C) je plně duplexní a obecně může dosahovat vyšších rychlostí. Podobně jako u I2C sběrnice i zde jsou zařízení rozdělena na "Master" a "Slave". Zařízení typu "Master" generuje pro celou sběrnici hodinový signál a ovládá (aktivuje nebo deaktivuje) jednotlivá "Slave" zařízení na sběrnici. SPI je z pravidla čtyřvodičová sběrnice, kde jednotlivé vodiče jsou:

1. MOSI (Master Out Slave In) - Master výstup, Slave vstup
2. SCK (Serial Clock) - hodinový signál sběrnice
3. MISO (Master In Slave Out) - Master vstup, Slave výstup
4. CS (Chip Select) - aktivace/deaktivace připojeného zařízení

Obsluhu této sběrnice ve firmwaru klientské jednotky řeší přímo dedikovaný SPI modul mikrokontroléru. Toto řešení je v daném případě vhodnější než ovládání sběrnice přímo firmwarem mikrokontroléru, protože dosahuje podstatně vyšších rychlostí a neomezuje tolik mikrokontrolér v činnosti, při nutné obsluze sběrnice.

V kooperaci s SPI modulem mikrokontroléru byl vytvořen ovladač pro snímač otisků. Tento ovladač umožňuje výčet jednotlivých snímků ze senzoru. Ovladač snímače je k nalezení ve zdrojovém kódu pod souborem "FPC1011F3.c" a v hlavičkovém souboru definic "FPC1011F3.h". Ve zdrojovém kódu ovladače je definováno několik funkcí pro jeho provoz. Mezi ně patří:

```
void FPCReset();
void FPCInitialize();
void FPCReadImage();
int FPCCompare();
```

Funkce `FPCReset` slouží k vyvolání funkce RESET v připojeném senzoru otisků. Jde tedy o metodu zajišťující návrat parametrů snímače k jejich základním hodnotám. Další funkcí, kterou modul ovladače nabízí je `FPCInitialize`. Tato funkce provede prvotní nastavení dedikovaného SPI modulu mikrokontroléru a zajistí jeho správnou funkci. Je nezbytné, aby po spuštění mikrokontroléru byla tato funkce volána předtím, než bude docházet k jakékoliv interakci se snímačem. Explicitní volání funkce `FPCReset` před funkcí `FPCInitialize` není nutné, protože funkce `FPCInitialize` provede automaticky funkci RESET připojeného senzoru.

Podívejme se nyní blíže na funkci `FPCReadImage`, která, jak už název napovídá, slouží k vyčtení snímku otisku prstu ze senzoru. Nový snímek je po návratu z této funkce uložen do operační paměti mikrokontroléru.

Vzhledem k relativně velké velikosti snímku k běžným velikostem operační paměti mikrokontrolérů dsPIC je nezbytné, aby snímek byl uložen v tzv. EDS (Extended Data Space) segmentu operační paměti mikrokontroléru. Tato část paměti je dostupná pomocí tzv. stránkování, kdy samostatný 16 bitový odkaz do paměti mikrokontroléru nemusí disponovat potřebným rozsahem adres a je tak potřebné u těchto odkazů do paměti přepínat paměťové stránky EDS [15]. Celková definice takové proměnné pro kompilátor MPLAB XC16 vypadá následovně:

```
__eds__ volatile unsigned char FPCImg[30400] __attribute__((far
,space(eds )));
```

Povšimněme si zejména klíčového atributu `__eds__`, který je zde použit právě proto, aby kompilátor automaticky zajišťoval při přístupu k této proměnné přepínání stránek EDS segmentu. Velikost tohoto pole bajtů je záměrně definována na hodnotu 30400, což je celkový počet obrazových bodu, které generuje použitý snímač otisku prstu.

Výčet snímku ve funkci `FPCReadImage` je realizován dle doporučeného postupu v datovém listu ke snímači. Nejdříve je tedy snímač inicializován a provedeno nastavení základních registrů nutných pro provedení výčtu:

```
//aktivace signálu Chip Select snímače otisků
FPC_SPI_CS_LAT = 0;

//nastavení základních hodnot do registrů potřebných k výčtu snímku
SPIPerformClock(FPC_wr_drivc);//wr_drivc instrukce
SPIPerformClock(0xff);//zápis základní hodnoty 255
SPIPerformClock(FPC_wr_adcref);//wr_adcref instrukce
SPIPerformClock(0x3);//zápis základní hodnoty 3
SPIPerformClock(FPC_wr_sensem);//wr_sensem instrukce
SPIPerformClock(0x0);/zápis základní hodnoty 0
```

Poté je možné přistoupit k sejmutí snímku z aktivní plochy snímače a jeho zaslání do fronty SPI rozhranní:

```
SPIPerformClock(FPC_rd_sensor);//rd_sensor instrukce
```

Nyní je nutné vyčkat definovaný počet cyklů hodinového signálu SPI rozhranní. Vzhledem k tomu, že celý interní systém snímače je řízen hodinovým signálem SPI rozhranní, bude nutné vykonat několik cyklů zaslání "prázdných" bajtů po sběrnici:

```
//zaslání 51 bajtů po sběrnici SPI, což je ekvivalentní výkonu 408
//hodinových cyklů sběrnice
for(i =0; i<51; i++)
//pro zamezení chybného vyhodnocení je prázdný bajt roven hodnotě 0
        SPIPerformClock(0);
```

Po vykonání nezbytného počtu cyklů je možné začít číst data, která se nyní nalézají ve frontě SPI rozhranní:

```
//příjem 30400 bajtu z SPI
for(i=0; i<30400; i++)
{
    //pro příjem dat je vzhledem k plně duplexní sběrnici nutné
    //zasílat prázdné bajty
    FPCImg[i] = SPIPerformClock(0x0);
}

//po vyčtení snímku je možno snímač deaktivovat signálem Chip Select
FPC_SPI_CS_LAT = 1;
```

Poslední funkcí, kterou ovladač senzoru otisku disponuje je funkce `FPCCompare`. Jde o funkci, která počítá průměrnou hodnotu obrazových bodů ve vyčteném snímku. Na základě této hodnoty je pak možné provést ve vhodný okamžik zaslání snímku do nadřazeného systému k identifikaci.

Výrobce snímače otisků doporučuje pro zvýšení životnosti zajištění jeho selektivního vypínání při nečinnosti. To bude zajištěno v kombinaci s externím vstupem, který bude detekovat přítomnost prstu osoby v blízkosti aktivní plochy snímače. Nebude-li externí senzor (např. infračervená

optozávora) detekovat přítomnost prstu dojde k deaktivaci připojeného senzoru dle procedury popsané v datovém listu.

4.2.3 Čtení čipových RFID karet

Modul čtečky RFID karet komunikuje s mikrokontrolérem přes rozhranní UART (Universal Asynchronous Receiver / Transmitter). UART je plně duplexní sběrnice využívající dva vodiče. Jeden pro příjem dat (Rx vodič) a druhý pro zasílání dat (Tx vodič). Sběrnice na rozdíl od SPI nebo I2C nemá separátní vodič definující hodinový signál sběrnice, ale využívá přednastavenou symbolovou rychlost (baud rate) [16]. Předpokladem správné funkce sběrnice je nastavení stejné symbolové rychlosti na obou komunikujících periféríích. Modul čtečky RFID karet je od výrobce přednastaven na pevnou symbolovou rychlost. Ta má hodnotu 9600 Baud/s a modul dále předpokládá, že komunikace bude vázána na 8 datových bitů, přičemž komunikace je pouze jednosměrná (Tx vodič od mikrokontroléru není vůbec využit).

Mikrokontrolér obsahuje dedikovaný modul pro rozhranní UART, díky kterému je možné vynechat z firmwaru klientské jednotky samostatnou obsluhu UART rozhranní. Pojďme se nyní zaměřit na konfiguraci UART modulu v použitém mikrokontroléru dsPIC pro komunikaci s modulem čtečky RFID karet.

Nejprve je nutné správně nastavit generátor symbolové rychlosti pro dedikovaný modul UART rozhranní. K tomuto účelu slouží registr `UxBRG`, kde `x` je číslo UART modulu, kterého se nastavení týká. Správná hodnota je definována vzorcem [16]:

$$UxBRG = \frac{F_p}{16\ x\ Symbolová\ rychlost} - 1$$

Kde Fp je instrukční rychlost mikrokontroléru a "Symbolová rychlost" je požadovaná symbolová rychlost, která musí být shodná s periférií. Pro instrukční rychlost 70Mhz a požadovanou symbolovou rychlost modulu 9600 Baud/s bude vypadat nastavení následovně:

$$UxBRG = \frac{70000000}{16\ x\ 9600} - 1 \cong 455$$

Nastavení modulu UART 1 pro příjem dat z modulu čtečky bude vypadat následovně:

```
U1BRG = 455;
U1MODE = 0x8800;// povolení UART 1 modulu bez řízení toku dat
IFS0bits.U1RXIF = 0;// smazání příznaku přerušení
```

```
IEC0bits.U1RXIE = 1;// povolení přerušení modulu UART 1
```

Obsluha přerušení řeší stav, kdy UART modul obdržel z připojené periférie data. Je to situace, kdy je požadováno, aby mikrokontrolér data zpracoval. Ukázková metoda obsluhy přerušení mikrokontroléru, který právě obdržel data z čtečky RFID karet může vypadat např. následovně:

```
//signatura metody pro obsluhu přerušení UART1 v XC16 kompilátoru
void __attribute__((__interrupt__, auto_psv)) _U1RXInterrupt(void)
{
    char byteRecv;//dočasná proměnná pro právě přijatý bajt

    //je-li modul aktivní a nastalo-li přerušení na UART1
    if((IFS0bits.U1RXIF==1) && (IEC0bits.U1RXIE==1))
    {
        //dokud jsou dostupné ve vyrovnávací paměti přijaté data
        while(U1STAbits.URXDA)
        {
            byteRecv = U1RXREG;//přečti registr přijatého bajtu

            //nenastala-li chyba v přenosu
            if ((U1STAbits.PERR == 0) && (U1STAbits.FERR == 0))
            {
                //zapiš data na právě aktivní pozici v poli
                uartRx[uartIndex++] = byteRecv;

                //je-li obdrženo všech 12 bajtů z RFID karty
                if(uartIndex==12)
                {
                   uartIndex=0;//vynuluj index aktivní buňky pole
                   //deaktivuj modul UART1 pro zamezení přepsání dat
                   U1MODEbits.UARTEN = 0;
                   //zde může být nastaven příznak pro zpracování
                   //dat v hlavní smyčce programu
                   break;
                }
            }
            else
                uartIndex=0;//nastala-li chyba, vynuluj pole
        }

        //po dokončení obsluhy je nutné smazat příznak přerušení
        IFS0bits.U1RXIF = 0;
    }
 }
```

Mikrokontrolér by neměl řešit hlubší zpracování dat přímo v metodě obsluhy přerušení, protože tím dojde k zablokování možnosti obsluhy přerušení jiných periférií. Časově náročné zpracování dat by mělo být řešeno v hlavní smyčce programu, aby se zablokování dalších přerušení zamezilo. To je možné zajistit nastavením příznaku, který bude hlavní smyčku programu informovat o dokončení příjmu dat a požadavku na jejich zpracování.

4.3 Implementace aplikace nadřazeného systému

Nadřazený systém, v našem případě serverová aplikace, musí řešit celou řadu úloh spojených s obsluhou a správou klientských jednotek, zajišťováním identifikace uživatelů systému a logováním dat přístupu k jednotlivým prostředkům. Pro splnění těchto požadavků bylo nezbytné serverovou aplikaci realizovat.

Serverová aplikace byla řešena jako standardní desktopová aplikace pro architekturu x86. Aplikace využívá samostatný aplikační rámec pro zajištění funkce rozpoznávání otisků prstů. Databázová vrstva aplikace je řešena DBMS Microsoft SQL Server. Všechny tři vrstvy aplikace, tedy jak databázová, tak logická i prezentační, jsou od sebe odděleny pro zajištění větší přehlednosti a modulárnosti kódu.

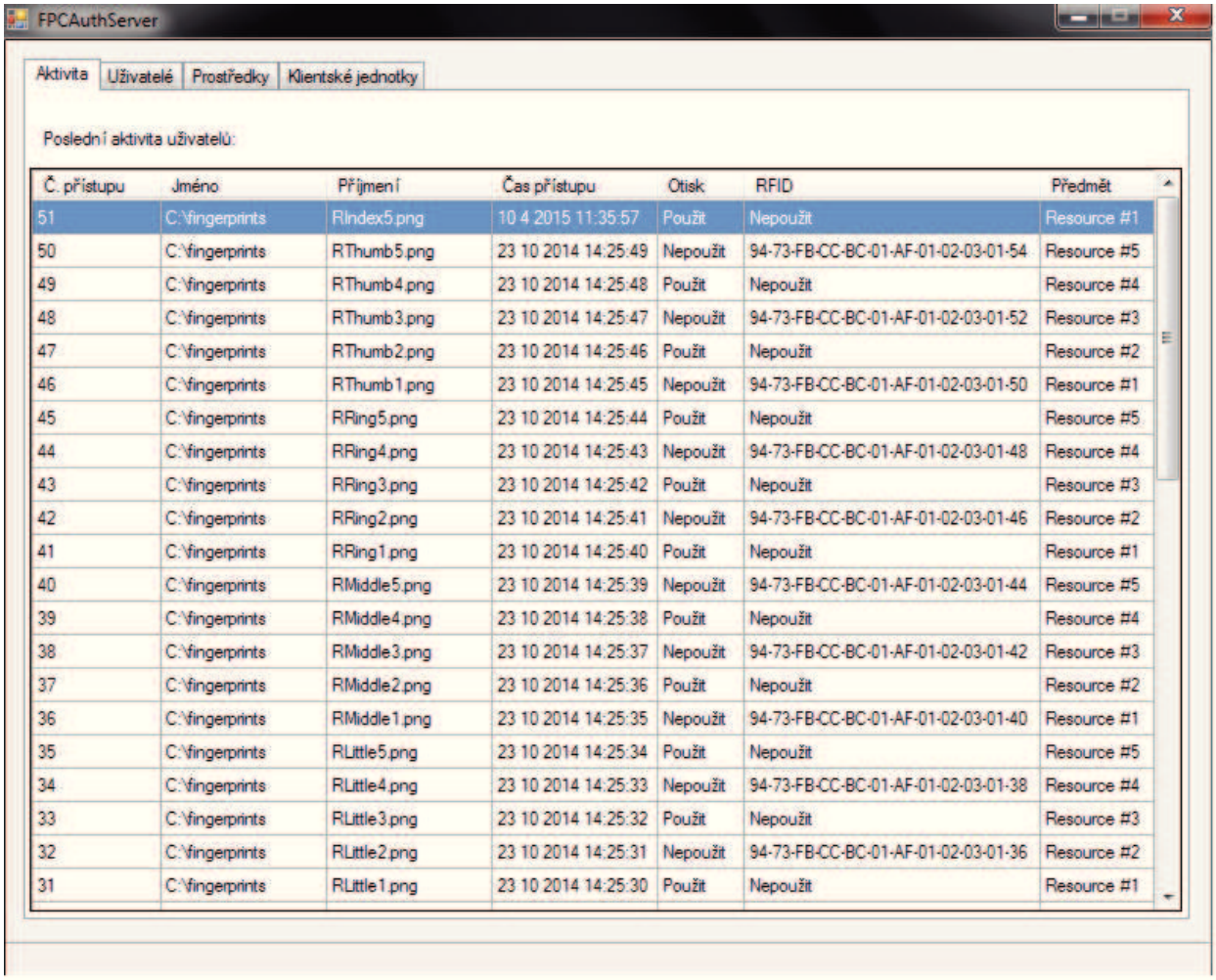
Serverová aplikace využívá služeb aplikačního rámce SourceAFIS pro zajištění úloh spojených s rozpoznáváním sejmutých otisků prstů z klientských jednotek. Tento aplikační rámec byl zvolen z důvodů relativně vysoké rychlosti rozpoznávání otisků při jejich identifikaci a také nízkým počtem špatně vyhodnocených otisků [17]. V této disciplíně dosahuje SourceAFIS hodnot 0.01% pro parametr FAR (False Acceptance Rate), tedy počet otisků chybně vyhodnocených jako správných a 10.9% FRR (False Rejection Rate), tedy počet otisků chybně vyhodnocených jako nesprávných.

SourceAFIS využívá pro rozpoznávání otisků algoritmus pracující na principu extrakce jedinečností. Díky této funkci nepotřebuje algoritmus mít pro svou činnost uloženy původní snímky z klientských jednotek, ale ukládají se pouze extrahované jedinečnosti, které nejsou zpětně reverzibilní. Není tedy možné zpětně zrekonstruovat původní snímky otisků prstů. To je důležitý bezpečnostní faktor prakticky znemožňující zneužití těchto uložených dat.

Použitý aplikační rámec je nezávislý na rozlišení vstupního snímku. Lze jej tedy aplikovat na prakticky libovolný snímač otisku prstu. Aplikační rámec je primárně vyvíjen v jazyce C# a obsahuje rozhranní jak pro verifikaci (porovnávání 1:1), tak pro identifikaci osoby (porovnávání 1:N otisků).

Serverová aplikace byla proto naimplementována také v jazyce C#, čímž by měly být eliminovány problémy s kompatibilitou jednotlivých architektur. Databázová vrstva aplikace používá manuální mapování entit na tabulky databáze. Kompletní databázové schéma je k dispozici v příloze

této diplomové práce. Grafické uživatelské rozhranní využívá pro vykreslování vestavěné API Microsoft GDI+. Ukázka uživatelského rozhranní je k dispozici na následujícím obrázku (Obr. 12).



Obr. 12: Serverová aplikace

Z obrázku je patrné, že aplikace obsahuje několik přepínatelných záložek. Jmenovitě jde o :

1. Aktivita - zobrazuje obsah logovaných dat. Ukazuje kdo, kdy a kde prošel přístupovým systémem
2. Uživatelé - zobrazuje registrované uživatele systému a umožňuje jejich editaci včetně editace otisků prstů a RFID čipových karet, kterými daná osoba bude disponovat. Pro editaci karet nebo přidávání, případně i změny, otisků je zapotřebí se spojit s klientskou jednotkou.

3. Prostředky - umožňuje správu registrovaných prostředků, a to včetně povolení přístupu k nim
4. Klientské jednotky - spravuje klientské jednotky, umožňuje jejich registraci do systému a přidělení k jednotlivým prostředkům

Kompletní zdrojový kód serverové aplikace je k dispozici v elektronické příloze této diplomové práce.

4.3.1 Popis komunikace s klientskou jednotkou

Serverová aplikace musí obsahovat metodu pro komunikaci s klientskými jednotkami, aby mohla vyhodnocovat identifikační data jednotlivých uživatelů přístupového systému. Komunikace s klientskými jednotkami musí být dostatečně "robustní" (odolná i vůči např. katastrofálnímu selhání fyzické vrstvy komunikace) a spolehlivá současně. Serverová aplikace proto obsahuje samostatný modul pro komunikaci s klientskými jednotkami.

Modul používá pro komunikaci vlastní protokol zpráv. Komunikace je vedena pomocí TCP soketového spojení, přičemž klientské jednotky komunikují se serverem formou zasílání zpráv. Tyto zprávy umožňují vytvořit flexibilní a zároveň odolnou komunikační metodu se serverem. Každá zpráva obsahuje signaturu, kterou zpráva začíná a definuje její celkovou délku. Je-li přijímána zpráva bez této signatury, dojde ihned k jejímu zahození bez dalšího zpracování. Typickou sekvenci bajtů představující zprávu jednotky charakterizuje Tabulka 2.

Tabulka 2: Sekvence bajtů zprávy

Pozice bajtu	*0*	*1*	*2*	*3*	*4*	*5 - X*
Přijatý bajt	*'F'*	*Délka*	*Délka*	*'L'*	*Typ*	*Data zprávy*

Je viditelné, že zpráva je rozdělena do několika sekcí, nejdříve je požadováno, aby každá zpráva začínala povinným znakem "F", po kterém následují dva bajty definující celkovou délku zprávy. Maximální teoretická velikost zprávy tedy může být 65536 bajtů. To je hodnota v daném systému víc než dostačující a nepředpokládá se ani, že by jí bylo nutno v budoucnu měnit. Po bajtech definující velikost zprávy následuje povinný znak "L". Neobsahuje-li přijímaná zpráva jeden z těchto dvou povinných znaků, nebo je-li udávaná velikost zprávy delší než přednastavená maximální velikost zpráv pro zpracování softwarem, pak nedochází k jejímu dalšímu zpracování a zpráva je zahozena. Po

druhém povinném znaku přichází bajt definující jaký význam daná zpráva má a co je jejím obsahem. Opět je maximální počet významů limitován paměťovým rozsahem (v tomto případě jeden bajt). Maximální teoretický počet významů zpráv je tedy 256, přičemž se opět nepředpokládá, že by bylo nutné hranici počtu významů zpráv posunout, protože je volena s relativně velkou rezervou pro daný systém. Po bajtu definujícím význam zprávy mohou, dle typu zprávy, následovat další data (např. jméno autentizované osoby, token pro odblokování zámku apod.). Po dosažení definovaného počtu bajtu získaného z udávané délky zprávy je čtení ukončeno a zpráva je předána k dekódování, kde dojde také k vyvolání zamýšlené reakce, kterou má zpráva vyvolat.

Speciálním případem typu zprávy je zpráva typu "Keepalive", která testuje, zda nedošlo mezi serverem a klientskou jednotkou k (často katastrofálnímu) selhání spojení. Zpráva typu "Keepalive" je periodicky vysílána serverem a klientská stanice na ní musí v určitém definovaném časovém úseku odpověď. Nedojde-li k odpovědi na tuto zprávu, pak lze předpokládat, že došlo ke ztrátě spojeni klientské jednotky se serverem. Klientská jednotka podobně jako serverová aplikace kontroluje, zda byla do určitého časového úseku získána nová zpráva typu "Keepalive". Dojde-li ke ztrátě spojení, pak se v relativně krátkém časovém úseku pokusí klientská jednotka navázat spojení se serverem znovu a serverová aplikace ukončí veškeré činnosti spojené s předchozím soketovým spojením, které již není platné. Popsaným typem zprávy lze relativně rychle obnovit spojení v případě katastrofálního selhání, které samotná rodina TCP/IP protokolů neřeší.

Úplný seznam všech typů(významů) zpráv by nebylo účelné zde vypisovat a je k dispozici ve zdrojovém kódu k nahlédnutí.

4.3.1.1 Navázání spojení s jednotkou

Při návrhu způsobu komunikace klientské jednotky se serverem byl kladen důraz na co největší škálovatelnost firmwaru klientské jednotky tak, aby zdrojový kód nemusel být mezi jednotlivými jednotkami vůbec měněn. Nebylo tedy chtěné, aby každá jednotka byla naprogramována s nějakým unikátním identifikátorem zvlášť. Systém předpokládá, že všechny jednotky budou z pohledu HW uspořádání jednotky i SW uspořádání mikrokontroléru naprosto totožné. K zajištění základní funkčnosti a rozlišení jednotlivých jednotek v systému bylo však potřebné, aby jednotlivé jednotky byly od sebe nějak odlišitelné. Odlišení pramení z použitého ethernetového řadiče, který obsahuje od výrobce pevnou MAC adresu, která je unikátní napříč spektrem všech ethernetových řadičů jednoho výrobce. Díky tomu je možné použít stejný firmware u všech jednotek a serverová aplikace přesto bude schopná rozlišit jednotlivé klientské jednotky.

Dalším úskalím spojeným s unifikovaností firmwaru všech jednotek je zajištění spojení v počítačové síti. Každá klientská stanice je v dané sítí (nebo podsíti) identifikována na základě (v dané síti) jedinečné IP adresy. Bylo tedy zapotřebí zajistit, aby každá klientská jednotka měla v síti jedinečnou IP adresu. V zásadě jsou dvě možnosti jakými je možné toto v dané situaci zajistit:

- Statická IP adresa zapsaná ve firmwaru zařízení
- Přidělení IP adresy DHCP serverem

První možnost řeší situaci zápisem IP adresy do firmwaru zařízení, kde každá jednotka bude mít unikátní IP adresu v dané síti. Toto řešení však nabourává požadavek na unifikovaný firmware všech jednotek. Druhé řešení je naopak podstatně flexibilnější a umožňuje přidělit IP adresu dané jednotce. K tomu přijde jako vysoce účelná integrovaná a pevně nastavená MAC adresa v ethernetovém řadiči, díky které může DHCP server přidělit v dané sítí unikátní IP adresy všem klientským jednotkám.

Posledním problémem spojeným se zajištěním spojení v počítačové síti je získání IP adresy vzdálené stanice, ke které je zapotřebí se připojit. V našem případě se jedná o IP adresu serveru. Pro tento účel je vhodným prvkem v počítačové síti typ paketu zvaný broadcast. Paket typu broadcast řeší situaci, kdy je zapotřebí zaslat jednu informaci všem stanicím v dané síti nebo podsíti. Ze zaslaného paketu je také možné získat IP adresu klientské stanice, která paket vyslala. Předpokladem tedy bude, že serverová aplikace bude periodicky vysílat informace o své aktuální adrese a všechny připojené klientské jednotky ji přijmou, zpracují a budou schopny se připojit k serveru. Tímto způsobem bude možné řešit i nefrekventované změny IP adresy serveru, protože všechny připojené klientské jednotky budou mít stalý přehled o aktuální IP adrese serveru a budou tak schopny se z tohoto katastrofálního selhání zotavit.

Nastíněný způsob navázání komunikace klientské jednotky se serverem byl využit a naimplementován jak na straně serverové aplikace, která musí periodicky vysílat pakety o své aktuální IP adrese, tak na straně klientské jednotky, která musí tyto vyslané pakety přijímat a dekódovat.

4.3.1.2 Zabezpečení přenosu dat

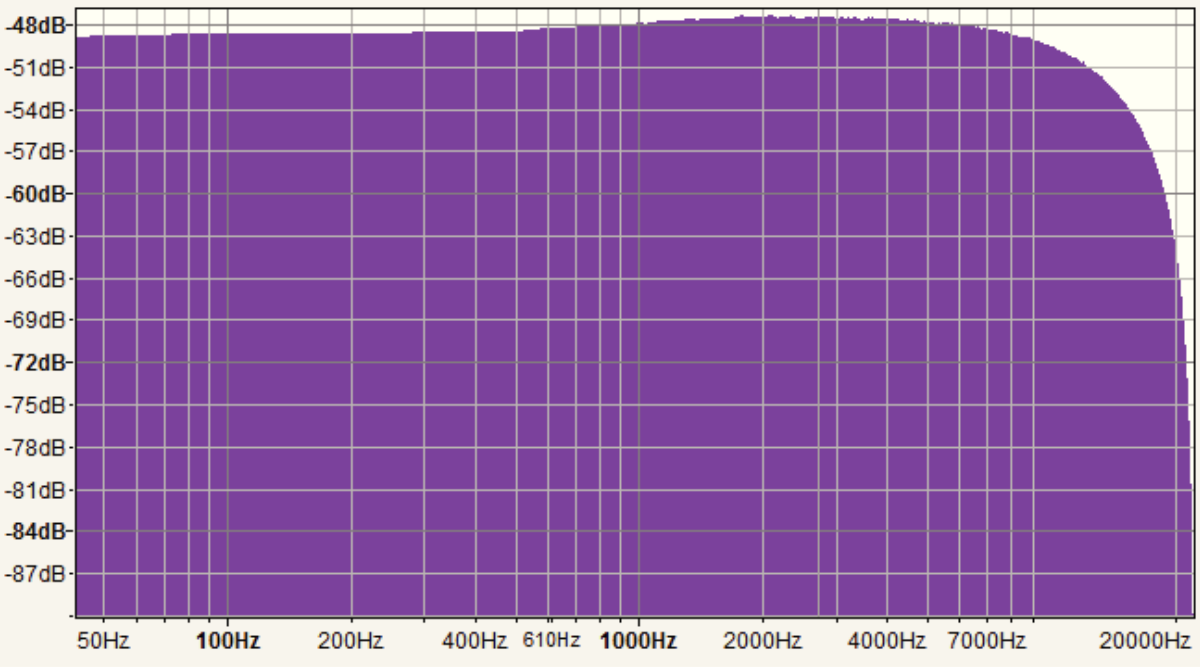
Vzhledem k tomu, že se otisk prstu považuje za citlivý osobní údaj není možné, aby byl přenášen nebo uchováván v původní, nezměněné podobě. Popsaný přístupový systém využívá nereverzibilní způsob extrakce jedinečností z otisků, a proto lze považovat požadavek na uchovávání v pozměněné podobě za irelevantní . Zbývá tedy zajistit, aby na všech exponovaných místech systému, kudy dochází k přenosu snímku otisku z klientské jednotky do serveru, bylo zajištěno patřičné

šifrování dat tak, aby nebylo možné případnou komunikaci odchytit a přenášený snímek vyčíst. Pokud nebudeme brát v úvahu zanedbatelná místa s relativně nízkým stupněm nebezpečí vzhledem k viditelnému narušení celistvosti zařízení klientské jednotky (např. pokus o zachycení snímku přímo na vývodech senzoru otisku, což by nepochybně vyvolalo u uživatele podezření, že dané zařízení bylo kompromitováno) pak platí, že v našem systému je prakticky jediným exponovaným místem počítačová síť. Cesta počítačovou sítí k serveru může vést přes několik různých fyzických médií včetně např. bezdrátové komunikace. Nebude-li na této fyzické vrstvě implementováno patřičné šifrování tak hrozí, že k zaslaným informacím z klientské jednotky bude mít přístup prakticky kdokoliv. Je tedy nezbytné, aby v klientské jednotce i serverové aplikaci byly učiněny patřičné kroky, které výčet přenášených dat prostým odposlechem zamezí. Vhodným zabráněním odposlechu přenášených dat je jejich šifrování.

K zajištění šifrovaného zasílání zpráv je zapotřebí nejdříve zvolit vhodný algoritmický způsob šifrování. Žádaná zde bude zejména z důvodu efektivnosti symetrická bloková šifra. Vhodným algoritmem s poměrně nízkými nároky na výpočetní výkon a jednoduchou implementovatelností je algoritmus AES. Jedná se o symetrickou blokovou šifru s šířkou šifrovacího klíče 128, 192 nebo 256 bitů. Použití samotné blokové šifry by však nebylo výhodné, protože opakované použití stejného šifrovacího klíče by vedlo vždy ke stejnému výstupu. Této vlastnosti by potenciální útočník mohl využít a relativně snadno provést útok hrubou silou na šifrovací klíč po odposlechu jediné zprávy vyslané z klientské jednotky. Pro eliminaci této zranitelnosti bude zapotřebí využít pro šifrování inicializační vektor a vhodný operační mód použitého blokového šifrovacího algoritmu. Operační mód využije inicializační vektor, který bude obsahovat náhodné hodnoty a použije jej k zašifrování prvního bloku dat pomocí funkce nonekvivalence (XOR) předtím, než dojde k použití algoritmu AES. Každý další blok bude navíc funkcí XOR šifrován s blokem předchozím a až poté dojde k použití algoritmu AES u daného bloku. Tento operační mód je u blokových šifer nazýván CBC (Cipher Block Chaining) a eliminuje zranitelnost při použití stejného šifrovacího klíče.

Popsaný operační mód symetrického šifrovacího algoritmu byl ve firmwaru klientské jednotky naimplementován. Základním předpokladem funkčnosti operačního módu CBC šifrovacího algoritmu je využití náhodných hodnot v inicializačním vektoru. Takový požadavek je však na deterministických počítačích, do kterých monolitický počítač nepochybně patří, obtížně realizovatelný, protože deterministický algoritmus je plně předvídatelný a umožňuje pouze vytváření čísel pseudonáhodných. Pro šifrování citlivých osobních údajů (biometrický údaj), které jsou potenciálně zneužitelné, je použití pseudonáhodného generátoru nevhodné. Bylo tedy nutné zavést do systému možnost generování "skutečně náhodných" čísel pro využití v inicializačním vektoru. K tomuto účelu byl ke klientské jednotce připojen generátor bílého šumu. Generátor používá jako zdroj šumu nedestruktivní průraz PN přechodu tranzistoru. Jedná se o modifikované zapojení generátoru převzaté z [18].

Sestavený generátor byl pro ověření funkčnosti připojen ke zvukové kartě počítače a programem Audacity došlo k sejmutí frekvenčního spektra výstupu generátoru (Obr. 13).

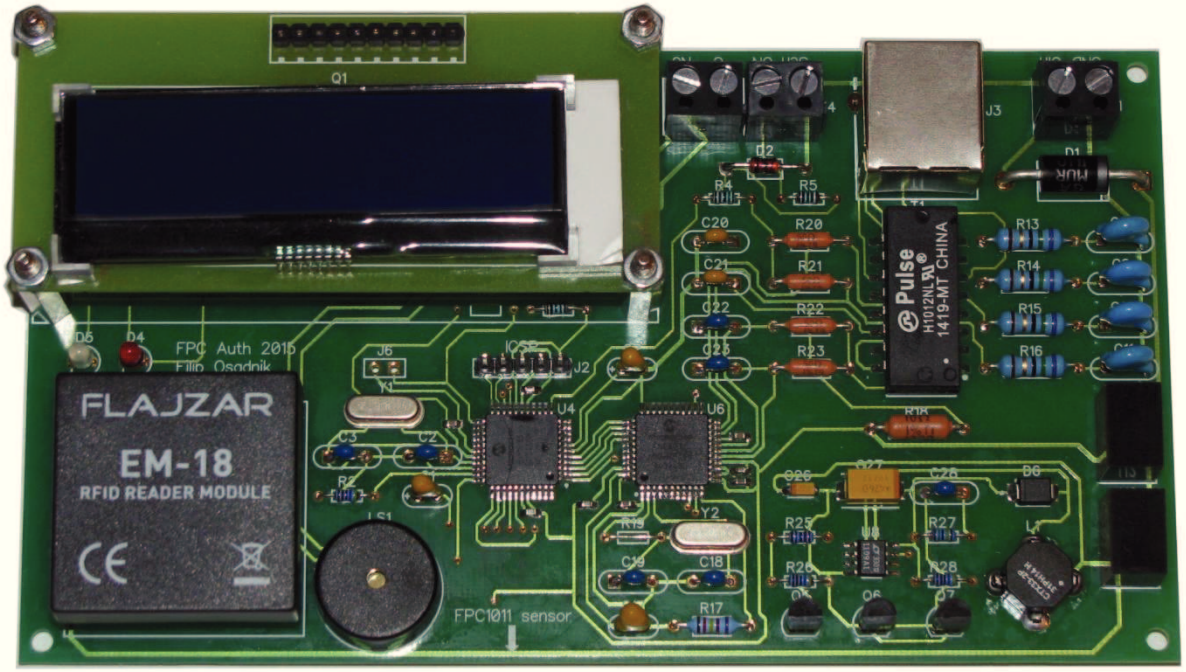


Obr. 13: Frekvenční spektrum generátoru šumu

Z obrázku je patrné relativně rovnoměrné rozložení hodnot jednotlivých frekvencí v rámci celého frekvenčního spektra, které je schopna zvuková karta zachytit. Na základě snímku generovaného frekvenčního spektra se lze domnívat, že vytvořený generátor bílého šumu je pro danou aplikaci vyhovující a nároky na generování náhodných čísel pro inicializační vektor splní.

4.4 Hardwarové uspořádání klientské jednotky

Navržené schéma zapojení klientské jednotky bylo využito pro návrh a realizaci fyzické implementace zapojení. Došlo k vytvoření desky plošného spoje a sestavení klientské jednotky z jednotlivých součástí. Klišé desky je k dispozici ve formátu DipTrace PCB v elektronické příloze této diplomové práce. Sestavená deska plošného spoje je zachycena na obrázku (Obr. 14):



Obr. 14: Sestavený prototyp klientské jednotky

Z ukázky sestavené klientské jednotky je patrno několik připojovacích konektorů. Význam těchto konektorů je následující:

1. Konektor J1 slouží pro připojení napájecího napětí, které musí být v rozsahu 6.5 - 32VDC.
2. Konektor J3 slouží pro připojení klientské jednotky k počítačové síti.
3. Konektor J4 slouží pro připojení externího čidla detekce přítomnosti prstu poblíž aktivní plochy snímače (např. infračervená optozávora). Dojde-li ke spojení těchto dvou svorek konektoru je snímání aktivní. V opačném případě dojde k odpojení napájecího napětí ze snímače a uvedení do pohotovostního stavu.
4. Konektor J5 slouží pro připojení externí zátěže. Jedná se o výstup z relé, kde v pohotovostním stavu je kontakt rozpojen.

4.5 Testování systému

Sestavená klientská jednotka byla podrobena důkladnému testování funkčnosti. Bylo zejména zkoušeno, zda splňuje základní předpokládané parametry a zda nedochází v průběhu testování k nežádoucím činnostem jednotky nebo dokonce k jejímu fatálnímu selhání.

Testovaná klientská jednotka nejevila v průběhu testování žádné zvláštní odchylky od předpokládané funkce. Došlo k otestování spolehlivosti rozpoznávání otisků prstů a čipových RFID karet.

Pro účely testování rozpoznávání otisků byla testovací databáze složena z 50 snímků od 5 různých osob a následovalo 10 pokusů o identifikaci každé osoby přístupovým systémem. Celkem z 50 pokusů bylo vyhodnoceno 6 identifikací chybně vyhodnocených jako špatných a 0 identifikací chybně vyhodnocených jako správných. Z uskutečněného testování tedy plyne, že parametr FAR dosáhl hodnoty 0% a parametr FRR dosáhl hodnoty 12%. Uvedené testovací výsledky je však nutno brát s patřičnou rezervou, protože skupina testovacích subjektů byla pro praktické využití výsledků, zejména z legislativních důvodů, příliš malá. Z pohledu testování spolehlivosti rozpoznávacího aplikačního rámce jsou výsledky poněkud irelevantní, protože zvýšenou chybu FRR je možné vysvětlit artefakty při používání snímače otisků prstů a to zejména u začínajících uživatelů systému. Vzhledem k relativní podobnosti hodnot získaných testováním se lze důvodně domnívat, že uváděné parametry FAR a FRR v datovém listu použitého aplikačního rámce SourceAFIS jsou správné.

Dále byla ověřena funkčnost rozpoznávání bezdrátových RFID čipových karet. K testování bylo použito 5 ks RFID karet od 2 výrobců a došlo k 10 pokusům o identifikaci. Ve všech testovacích případech byla úspěšnost rozpoznávání 100%. Toto chování systému ve srovnání s metodou rozpoznávání otisků si lze vysvětlit v exaktnosti vstupních dat. Zatímco použité RFID čipové karty zasílají do systému vždy stále stejný 12-ti bajtový unikátní identifikátor, tak senzor pro snímání otisků generuje prakticky při každém výčtu data rozdílná.

Poslední testovací disciplínou bylo měření celkové energetické náročnosti navrženého zařízení. Při použití napájecího napětí 12V odebíral sestavený modul klientské jednotky proud 84 mA. To odpovídá příkonu celé klientské jednotky ve výši 1W. Provedené měření platí pro modul s aktivním podsvícením displeje a navázaným spojením ethernetového řadiče se vzdálenou stanicí. Při pokusné deaktivaci spojení ethernetového řadiče klesl odběr klientské jednotky na hodnotu přibližně 50mA. Během testování bylo zjištěno, že zvolený ethernetový řadič vykazuje značně odlišnou provozní teplotu od zbytku použitých součástí. Infračerveným bezdrátovým teploměrem bylo provedeno měření pouzdra čipu ethernetového řadiče a zjištěna provozní teplota dosahující 48°C. Tato hodnota je však v rozsahu pracovních teplot specifikovaných v datovém listu řadiče a souvisí s

množstvím energie, které je nutno absorbovat plochou čipu v relativně malém pouzdru QFP při využívání fyzické vrstvy ethernetu. Lze tedy předpokládat, že toto chování je při provozu normální a nevyžaduje zvláštní pozornost.

5 Závěr

Diplomová práce poskytuje čtenáři ucelený náhled na problematiku návrhu a realizace centralizovaného přístupového systému. Autentifikace na bázi rozpoznávání otisků prstů je metodou, která se do těchto systémů implementuje čím dál častěji a pro mnoho uživatelů je takový způsob autentizace každodenní rutinou. Diplomová práce toto reflektuje a poskytuje detailní návod na realizaci přístupového systému založeného na detekci otisků prstů, přičemž zachovává zpětnou kompatibilitu s existujícími čipovými RFID kartami, čímž řeší také uživatele, pro které je autentifikace pomocí otisku prstu nepřijatelnou formou.

Výsledným produktem této diplomové práce je fyzický prototyp klientské jednotky přístupového systému a serverová aplikace tvořící jeho protějšek. Touto implementací přístupového systému došlo ke splnění všech bodů zadání diplomové práce. Provedené praktické testování systému tuto skutečnost potvrdilo a prokázalo, že navržený systém splňuje nároky na spolehlivost a bezpečnost provozu při zachování jednoduchosti a intuitivnosti jeho ovládání.

Navržený přístupový systém je možné dále rozšiřovat a upravovat do podoby vhodné pro jeho nasazení v ostrém provozu. Jmenovitě jde např. o rozdělení akčního členu klientské jednotky do chráněného prostoru, vytvoření spojovacího rozhranní pro napojení k již existujícím uživatelským databázím a případně zvýšení přesnosti identifikace osoby a současně odolnosti systému proti využití falzifikátů otisků implementováním dalšího způsobu biometrického rozpoznávání.

Použitá literatura

[1] Jain, Anil, Ross, Arun A., Nandakumar, Karthik, "Introduction to Biometrics", Springer, 2011, ISBN 978-0-387-77325-4

[2] Maltoni, D., Maio, D., Jain, A., Prabhakar, S., "Handbook of Fingerprint Recognition (Second edition)", Springer, 2009, ISBN 978-1-84882-253-5

[3] Roman Rak, Václav Matyáš, Zdeněk Říha, "Biometrie a identita člověka", Grada, 2008, ISBN 8024723655

[4] Heeseung Choi; Raechoong Kang; Kyoungtaek Choi; Andrew Teoh Beng Jin; Jai Hie Kim, "Fake-fingerprint detection using multiple static features" [online], 2009 [cit. 3. února 2015]. Dostupné z: http://cherup.yonsei.ac.kr/files/Paper/Fake-fingerprint%20detection%20using%20multiple%20static%20features%20(2009).pdf

[5] Davide Maltoni, "Fingerprint Recognition - Basics and Recent Advances" [online], 2012 [cit. 1 února 2015]. Dostupné z: http://icb12.iiitd.ac.in/Fingerprint-ICB2012.pdf

[6] Priyanka Das, Kannan Karthik, Boul Chandra Garai, "A robust alignment-free fingerprint hashing algorithm based on minimum distance graphs" [online], 2012 [cit. 3. února 2015]. Dostupné z: http://www.iitg.ernet.in/engfac/k.karthik/Research/MDG_Final%20Published%20PDF_May%2029,%202012.pdf

[7] Fingerprint Cards AB, "FPC1011F3 Area Sensor Package Product Specification", 2012 [cit. 12. března 2015].

[8] MIDAS COMPONENTS LTD., "Specification MCCOG22005A6W " [online], 2011 [cit. 12. března 2015]. Dostupné z: http://www.farnell.com/datasheets/1663638.pdf

[9] MICROCHIP TECHNOLOGY INC., "ENC424J600/624J600 Data Sheet" [online], 2010 [cit. 15. března 2015]. Dostupné z: http://ww1.microchip.com/downloads/en/DeviceDoc/39935c.pdf

[10] MICROCHIP TECHNOLOGY INC., " Microchip TCP/IP Stack Help " [online], 2012 [cit. 15. března 2015]. Dostupné z: http://ww1.microchip.com/downloads/en/DeviceDoc/Microchip_Libraries_for_Applications_v2013-06-15_Help_Files..zip

[11] MICROCHIP TECHNOLOGY INC., "dsPIC33EPXXXGM3XX/6XX/7XX Data Sheet" [online], 2014 [cit. 16. března 2015].
Dostupné z: http://ww1.microchip.com/downloads/en/DeviceDoc/70000689d.pdf

[12] eNTesla, "EM-18 RFID Reader" [online], [cit. 16. března 2015].
Dostupné z: http://entesla.com/index.php?route=product/download/download&download_id=57

[13] MICROCHIP TECHNOLOGY INC., "MPLAB® XC16 C Compiler - User's Guide" [online], 2014 [cit. 1. dubna 2015].
Dostupné z: http://ww1.microchip.com/downloads/en/DeviceDoc/50002071E.pdf

[14] PHILIPS SEMICONDUCTORS., "AN10216-01 I2C MANUAL" [online], 2003 [cit. 1. dubna 2015]. Dostupné z: http://www.nxp.com/documents/application_note/AN10216.pdf

[15] MICROCHIP TECHNOLOGY INC., "dsPIC33E/PIC24E Family Reference Manual - Section 3. Data Memory" [online], 2011 [cit. 10. dubna 2015].
Dostupné z: http://ww1.microchip.com/downloads/en/DeviceDoc/DS-70595C.pdf

[16] MICROCHIP TECHNOLOGY INC., "dsPIC33E/PIC24E Family Reference Manual - Universal Asynchronous Receiver Transmitter (UART)" [online], 2013 [cit. 16. dubna 2015].
Dostupné z: http://ww1.microchip.com/downloads/en/DeviceDoc/70000582e.pdf

[17] Robert Važan, "SourceAFIS - Datasheet" [online], 2014 [cit. 16. dubna 2015].
Dostupné z: http://www.sourceafis.org/blog/datasheet/

[18] Aaron Logue, " Hardware Random Number Generator" [online], 2002 [cit. 21.dubna 2015].
Dostupné z: http://www.cryogenius.com/hardware/rng/

Seznam příloh

Příloha 1: Adresářová struktura na přiloženém disku DVD

/FPCAuth	Projekt firmwaru klientské jednotky pro vývojové prostředí MPLAB X
/FPCAuthServer	Solution serverové aplikace pro obsluhu klientských jednotek
/datasheets	Datové listy všech komponent modulu klientské jednotky
/docs	Obrazce DPS a schémata ve formátu PDF
/pcb	Obraz DPS klientské jednotky pro program DipTrace PCB Layout
/schematics	Schéma klientské jednotky pro program DipTrace Schematic
/dip	Diplomová práce ve formátu PDF/A