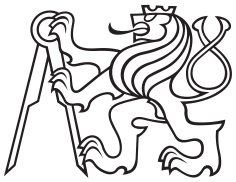

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Telecommunications Engineering

Access control system

Tomáš Hyhlík

Supervisor: Ing. Bc. Marek Neruda, Ph.D
Supervisor–specialist: Ing. Bc. Lukáš Vojtěch, Ph.D
Field of study: Communications, Multimedia, Electronics
Subfield: Multimedial technique
May 2017

Acknowledgements

I would like to thank Supervisor Ing. Bc. Marek Neruda Ph.D and Supervisor-specialist Ing. Bc. Lukáš Vojtěch Ph.D for helping me with this project. Also I would like to thank IMA s.r.o. company, which helped me to get compatible cards to the HID Prox Point plus reader, which is used for the second system design.

Declaration

I declare that I have developed the presented work independently and that I have listed all information sources used in accordance with the Methodical Guidelines on Maintaining Ethical Principles During the Preparation of Higher Education Theses.

In Prague, 20. May 2017

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Hyhlík** Jméno: **Tomáš** Osobní číslo: **434754**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra telekomunikační techniky**

Studijní program: **Komunikace, multimédia a elektronika**

Studijní obor: **Multimediální technika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Přístupový systém

Název bakalářské práce anglicky:

Access Control System

Pokyny pro vypracování:

Navrhněte a realizujte přístupový systém v koncepci embedded s využitím technologie RFID. Zaměřte se na bezpečnost a spolehlivost řešení. Navrhněte a analyzujte možné scénáře útoků na systém.

Seznam doporučené literatury:

[1] K. Finkenzeller, RFID Handbook, 2nd ed., John Wiley & Sons, 2003.
[2] N. Navnath, V. Pande, and N. Bhandari, RFID in Access Control System: A Microcontroller based System, 2011.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Marek Neruda Ph.D., katedra telekomunikační techniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **17.02.2017** Termín odevzdání bakalářské práce: **26.05.2017**

Platnost zadání bakalářské práce: **30.09.2018**

Podpis vedoucí(ho) práce	Podpis vedoucí(ho) ústavu/katedry	Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

. | Datum převzetí zadání

Podpis studenta

Abstract

The purpose of this work is to design, assemble and program the access control system in embedded conception for controlling access into house, room or hall. Further this work focuses on the security of the designed system. In the theoretical part is described how these systems works nowodays, introduction to RFID technology, RFID components and development boards, which is used as a central unit for the designed system.

Keywords: Arduino, card, reader, RFID, tag, transponder

Supervisor: Ing. Bc. Marek Neruda, Ph.D

Abstrakt

Účelem této práce je navrhnout, sestavit v koncepci embedded a naprogramovat systém pro řízení přístupu do domu, místnosti či objektu s použitím RFID technologie. Dále se tato práce zaměřuje na bezpečnost navrženého systemu. V teoreticke části je popsáno, jak v dnešní době tyto systémy fungují, úvod do RFID technologie, součásti RFID technologie a vývojové kity, které pro tento návrh systému slouží jako centrální jednotka.

Klíčová slova: Arduino, čtečka, karta, RFID, tag, transponder

Překlad názvu: Přístupový systém

Contents

Figures

Tables

0.1 List of Abbreviations

AID ..... Application Identifier

APDU ..... Application protocol data unit

API ..... Application Programming Interface

ASK ..... Amplitude Shift key

CCC ..... Chaos Computer Club

EEPROM ..... Erasable programmable read-only memory

FRAM ..... Ferroelectric Random Access Memory

FSK ..... Frequency Shift keying

GND ..... Ground

I2C ..... Inter-Integrated Circuit

IC ..... Integrated Circuit

IDE ..... Integrated development environment

IEC ..... International Electrotechnical Commission

ISO ..... International Organization for Standardization

LCD ..... Liquid Crystal Display

LED ..... Light emitting diode

LFSR ..... Linear feedback shift register

PCB ..... Printed circuit board

PICC ..... Proximity Integrated Circuit Card

RAM ..... Random access memory

RFID ..... Radio-frequency identification

RNG ..... Random number generator

ROM ..... Read only memory

SDK ..... Software development kit

SPI ..... Serial peripheral interface

SRAM ..... Static random-access memory

UART ..... Universal asynchronous receiver/transmitter

UID ..... Unique Identifier

USB ..... Universal Serial Bus

Chapter 1

Introduction

This chapter presents background, purpose and objectives to make clear the goal of this thesis.

1.1 Background

RFID technology is happening to be very popular these days for various applications such as industrial automation, access control, animal identification, public transport, event ticketing, parking, electronic wallet, goods identification and many more. The question is how secure this technology is. The answer is, that there are various manufacturers providing RFID (Radio Frequency Identification) devices with various security. Information delivered by the manufacturers about the security should be clear.

1.2 Why electronic door lock?

However, there are very secure door locks, commonly used mechanical door lock has a lot of disadvantages. It's easy to clone keys and it's even possible to open the door without the key. If a key is lost, changing the lock is needed, which could be even impossible in some cases. Somebody who finds the lost key would be able to open the door. Electronic door lock might came out with solutions of these problems, however every lock is possible to hack somehow. A key of electronic door lock is usually RFID tag or card, but it can be also mobile phone communicating via bluetooth interface etc. The user's card may be also used in other applications, like electronic purse. In case that user loses his card, the card can be deleted from the system. There is also so many advantages like user info can be stored on the RFID card and part of the system might be a server where scanned cards are monitored.

1.3 Objectives

The purpose of this thesis is to design and assembly an RFID based access control system composed of inexpensive RFID devices used today and examine its security and possibility of hacking it. In the conclusions, it's assessed if the information given by the manufacturer gives a clear picture of the security of his RFID technology.

Part I

Theoretical part

Chapter 2

Access control systems

This chapter includes an introduction to electromagnetic door locks and access control systems.

2.1 Electromagnetic door lock

Electromagnetic door lock is the same like ordinary mechanical door lock, but the iron bar blocking the door is controlled by a coil. Current flowing through the coil creates an electromagnetic field, which acts on the bar by force and also there is a spring with a force in the opposite direction than the coil. If the coil is powered on, this force pulls the bar to the coil and the door are released. In quiescent state the coil is powered off and the iron bar is pulled into the door so it blocks the door [1].

2.2 Access control systems

Access control systems can be divided into two main groups: online and offline systems. However, it's possible to combine these two variants [2].

2.2.1 Online Systems

Online access control systems have all the readers and door locks connected to a central computer with a database whereby all the door locks and readers are being controlled and monitored. The history of all the activities is usually being stored in the central computer and all authorized cards are stored in the central computer database, so all the changes like adding and removing a card are easily done by editing the database. This is advantageous, because when a card is lost, it's removed from the database and protected area stays safe. Informations about card's user can be stored in the central computer database, so the card only have to be able to store a small amount of data,

like unique card identification number. Online systems are best suited for access control systems where is many people to enter the building with only a few entrances to the building, for example office buildings or universities [2].

2.2.2 Offline Systems

Offline access control systems have for each door one system which consists of reader, central unit and door lock and these systems of all doors are not interconnected. Each system have saved a list of all authorized cards in its memory. If there are more rooms to access and one card shall provide access only to some of them, information regarding the rooms to which the card can provide access can be stored on the card. Other option is to have every system with different list of authorized cards stored in the memory of the system's central unit.
Buildings with many rooms like hotels use one system for every room with a few card numbers stored in its memory. Receptionist gives a card to guest, which is authorized only for particular room. When authorized card is lost, the card number needs to be deleted from all certain systems with use of a suitable programming device. Offline systems are best suited for the cases where is many individual rooms to access and only a few people have access to these rooms [2].

Chapter 3

Contactless smart cards and proximity cards

This chapter includes introduction to the RFID and NFC technology with whom we are going to deal with. Further this chapter describes fundamentals, communication protocols, data structures, security and usage of contactless smart cards and proximity cards.

3.1 Introduction to RFID

RFID is a technology using electromagnetic field to identify objects. The identification can be done only for a short distance from a few centimeters to a hundreds of meters [3] [4]. RFID communication is always between two sides, which is usually reader/writer device and an RFID tag or card (also called transponder).

The RFID reader acts as a master and the card acts as a slave during the RF communication, which means the reader sends commands to the tag and the card sends answers back to the reader [3].

RFID tags can be divided into passive, semi-passive and active. Passive tags have no power supply, semi-passive tags have a battery but used only for powering its microchip and not to power the RF interface. Active tags have power supply (usually battery) to power the microchip and the RF interface. To establish the wireless communication between the passive or semi-passive tag and reader, the reader must create powerful electromagnetic field in order to power the tag through its own antenna. Passive tags are used only at short ranges up to a few meters. Semi-passive tags don't need such power supplied by the reader, because the chip in the semi-passive tag is powered externally, so it can be used on a longer distance around tens of meters. For even greater reading distance, about hundreds of meters, are being used active tags, but the reading distance depends on many other factors [2].

"True physical tag maximum read distance is determined by the individual RFID reader and antenna power, the actual Integrated Circuit used in the RFID tag, the material and thickness of material the tag is coated or covered with, the type of antenna the tag uses, the material the tag is attached to and more!" [4]
In most of the cases are used passive tags, because it's more comfortable with no need of maintenance.
"Many RFID standards have been released by the industry. Most of them have as operating frequency 125 KHz or 13.56 MHz." [3]
Table 3.1 shows the mostly used RFID bands.

RF band	frequency
LF (low frequencies)	125 kHz and 134,3 kHz
HF (high frequencies)	13.56 MHz
UHF (ultra high frequencies)	860 MHz - 960 MHz and 433 MHz
SHF (super high frequencies)	2.45 GHz

Table 3.1: The mostly used RFID frequency bands [4]

3.2 Introduction to NFC

The NFC technology is a subset within RFID technology developed jointly by companies NXP Semiconductors (at that time named Philips semiconductors) and Sony in frequency range 13.56 MHz with maximum communication range up to 20 cm. It's called near-field communication, because during the wireless communication antennas of receiver and transmitter are in near-field.
The NFC interface is being mounted into electronic devices such as mobile phones and it enables to these devices a card emulation, reader emulation and peer-to-peer communication. So such NFC device can be used for electronic payment instead of credit card[2].
"NFC is specified according to standards ISO/IEC 18092 NFCIP-1 (ECMA 340) and ISO/IEC 21481 NFCIP-2 (ECMA 352) (Philips Semiconductors, 2006)" [2]
NFC is also compatible with all ISO/IEC 14443 type A based smart cards and readers such as NXP's Mifare and Sony's Felica smart cards [2].

3.3 Contactless smart cards and proximity cards - General

RFID cards and tags for access control systems are passive and have various construction formats, the most common are coin and card. Both shapes are practical for users, coin for carrying on a keychain and card to carrying it in a wallet. The transponder's circuit is integrated in a very small chip and antenna is outside of the chip. Figure 3.1 shows a typical physical structure

of a card.

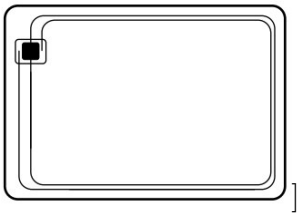

Figure 3.1: Sample structure of a card ISO/IEC 14443 [5]

It's possible to say that RFID card or tag is a memory unit which can be accessed without physical contact, because of the RF interface and microcontroller or other logic which manages the communication and memory access or cryptography [3] [6].

3.3.1 RF interface

The card's RF interface consists of the antenna coil, and modulator circuit. It has two following functions: transmit the data between reader and the card and gain the power from the reader to power the integrated circuit [2]. *"To achieve this, the RF interface draws current from the transponder antenna, which is rectified and supplied to the chip as a regulated supply voltage."* [2] When the card is receiving data, the modulated RF signal sent by the reader is represented in the RF interface and demodulator creates a digital serial data stream of it. For the opposite direction when the card transmits data to the reader, the data stream is modulated via modulator and sent via antenna coil. Majority RFID interfaces uses ASK modulation such as ISO/IEC 14443, but there are many other modulations used by various manufacturers [2].

3.3.2 Memory

The card's memory is being accessed by the microcontroller or addresses and security logic as shown in example of a card block diagram in figure 3.3 [2].

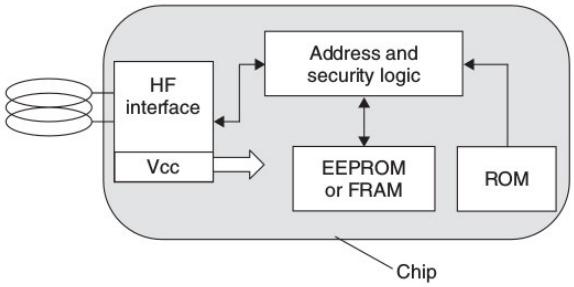

Figure 3.2: Sample of a card block diagram [2]

The read-only transponders contain only ROM (Read-only memory), which is programmed by manufacturer. Transponder with ability to write data on it may have EEPROM (Erasable programmable read-only memory), RAM (Random access memory) or FRAM (Ferroelectric RAM).

3.4 HID Prox Card II

The Prox Card II made by HID Global communicates at 125 kHz and it is inductively-coupled passive proximity card compliant with ISO/IEC 7810. For the communication with a reader it uses wiegand binary interface FSK (Frequency Shift Keying) modulation and data on the card are stored in wiegand format.
It has size of memory from 26 to 40 bits. In this project is used 26 bit format with ordering code number H10301. Bits of the card can be divided into three categories which is facility code, card number, and parity bits. The card number should be unique for one system where it's used. The facility code is programmable and should be the same for one system, company or region. It enables to use the same card number for different system companies or regions. Parity bits are used for directly detecting errors during the communication.

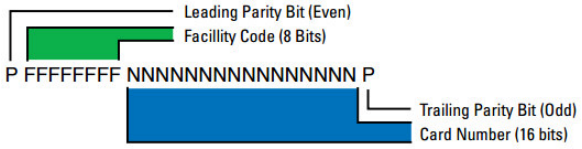

Figure 3.3: Prox Card II 26-bit format data structure [7]

As shown in figure 3.3 the 26 bit array starts with even parity bit covering facility code, then follows 8 bits of facility code, 16 bits of card code and finally the odd parity bit covering the card number. So it has 255 options of facility code and 65535 options of card number. The HID Global has no limitations on use of this format, so if all the bits of the card (without parity bits) are used as one card ID, there is 16 777 215 options of the card ID [8] [7] [9].

3.5 Mifare

"MIFARE is NXP's well-known brand for a wide range of contactless IC products with a typical read/write distance of 10 cm (4 inch) used in more than 40 different applications worldwide." [10]
Mifare cards appeared on the market in 1994 and more than 10 billions was sold till now. The price of Mifare cards is low, because it's very widespread. Mifare cards belong to category PICC (Proximity Integrated Circuit Card)

and there is a number of variants of mifare cards with different features. Today the most frequently used are MIFARE Ultralight, MIFARE Classic, MIFARE Plus, MIFARE DESFire and SmartMX [10] [11].
Mifare cards are fully compliant with international standard ISO/IEC 14443 type A which consists of four parts. Table 3.2 shows list of those parts.

ISO/IEC 14443-1	Physical characteristics
ISO/IEC 14443-2	Radio frequency power and signal interface
ISO/IEC 14443-3	Initialization and anti-collision
ISO/IEC 14443-4	Transmission protocols

Table 3.2: ISO/IEC 14443 parts [2]

There is also standard ISO/IEC 14443 type B and differs from type A in parts 2 and 3. These two types were created because manufacturers were unable to agree on a single communication interface. This leads to complications in communication, because to make a compatible reader with both types of standards, it must detect what type of standard uses a presented card, before the communication starts.
It uses encryption to secure data stored in its memory. To get access to the memory is needed cryptographic key which makes the data on the card more secure. Every card has UID (Unique Identifier), which is programmed by vendor and it's not possible to rewrite it. The UID is primarily used to establish communication between the reader and the card.
The type A standard defines that the communication is at 13.56 MHz and for data transmission from reader to card is used 100% ASK (Amplitude Shift Keying) modulation Miller encoding. For the opposite direction is used ASK Active Load Modulation.
The data transfer is a half-duplex, so only one party can transmit at a time. This standard also defines anti-collision, which enables the reader to communicate with one card in case where more than one card is attached to the reader.
The fourth part of this standard describes communication protocol between the reader and the Mifare card, which supports the transmission of APDU. The card and the reader communicate by sending a commands and responses in a form of hexadecimal number [2].

3.6 Types of mifare cards

As it's written above, there are many types of Mifare cards and in this section is discussed only Mifare Classic, because this kind of card is used in this project, and Mifare Desfire family, because it's the successor of the Mifare Classic card.

3.6.1 Mifare Classic

The Mifare Classic appeared on the market at first in 1994 and in the next few years it became the most widespread smart card in the world. Security is proven by CRYPTO1 stream cipher, but it has been broken in 2007 as explained further in this chapter. The card's memory is available in three sizes, 1, 2 and 4 kilobytes [11] [5].

Memory organization

Figure 3.4 shows memory organization of 1k variant which has 1024 bytes of data storage, but only 752 bytes can be used for data storage. It has EEPROM is organized in 16 sectors within 4 blocks and every block has 16 bytes. All the sectors are securely separated. Every sector's fourth block is called Trailer block and it's not for user's data, but it's used for two access keys Key A and Key B and access bits [5].

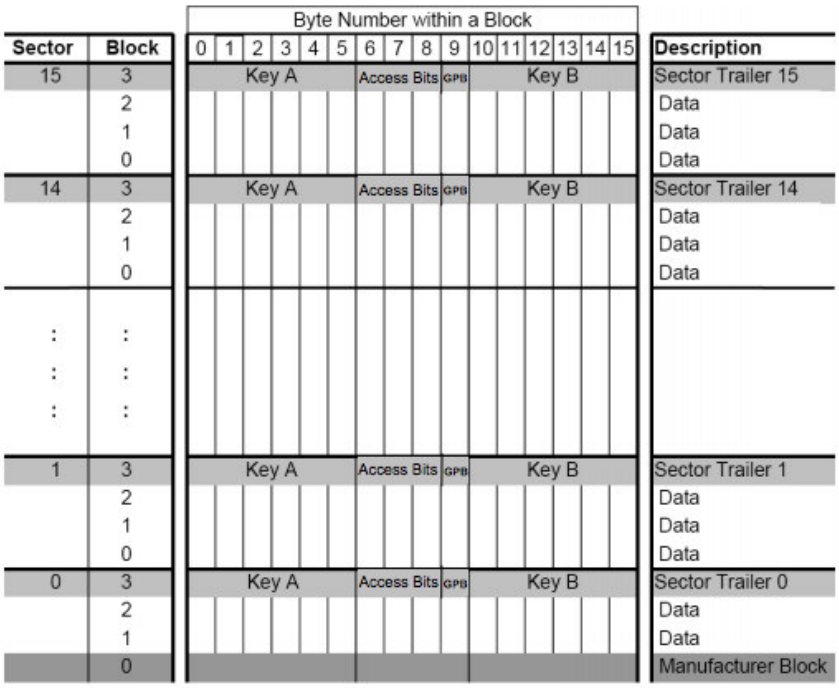

Figure 3.4: Mifare Classic memory organization [5]

Both keys are 6 bytes long, so it has 10^{12} options. If the card is completely new, these keys are set to FF FF FF FF FF FF which is called factory format. When you try to dump these keys, key A return zeros, because it's not possible to read it. Further the trailer block contains one unused byte for user and access bits on whose state depends sector's access conditions and usage of data blocks in the sector. The access conditions can be set to a few states telling us which key is used for authentication, which key is

readable and options of another blocks. With these access bits other blocks of the sector can be set to value blocks which allows the additional value operations like increment, decrement, transfer and restore. This is very useful for electronic purse.
The manufacturer block is stored in Sector 0 block 0 and it contains the manufacturer data within card's UID, so the whole block is read-only. The manufacturer block is not affected by the sector's access bits setting. The card's UID is 4 or 7 bytes long [5].

Memory security

Mifare Classic uses Crypto-1 stream cipher implemented in the chip hardware for fast processing. The NXP Semiconductors used to keep the Crypto-1 algorithm in secret, but it has been revealed and published by CCC in 2007 [12] [11]. More about this story is in section Mifare Classic Hack.

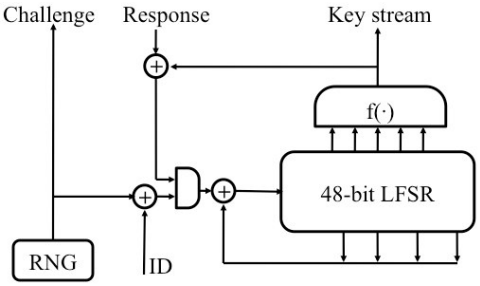

Figure 3.5: Crypto-1 stream cipher [13]

Figure 3.5 shows the Crypto1 diagram. The cipher algorithm consists of a LFSR (Linear Feedback Shift Register) with a linear feedback function f(x). The Crypto1 is used with three pass authentication as discussed bellow in following steps:

1) In the beginning of a card's sector authentication the reader sends a request for authentication of selected sector and type of selected authentication key.

2) The card finds out if it's possible depending on access conditions. If not, it sends error response back. If so, it loads actual value 32-bit number from RNG (Random Generator) and send it to the reader as a challenge n_T. At this point both sides know which key is chosen for the ciphering of challenge and UID.

3) The secret 48-bit key is loaded into the LFSR. The generated 32-bit challenge R_b XORs with the UID and this output string is shifted into the state of the shift register. The filter function generates the key stream using the generating polynomial as follows.
$g(x) = x^{48} + x^{43} + x^{39} + x^{38} + x^{36} + x^{34} + x^{33} + x^{31} + x^{29} + x^{24} + x^{23} +$

$x^{21} + x^{19} + x^{13} + x^9 + x^7 + x^6 + x^5 + 1$
At this point, the encryption is activated and all incoming and outgoing bits are XORed with the keystream. Each cycle, the function f(x) computes one bit for the key stream out of 20 bits it gets from the LFSR. The 18 taps of the LFSR are used to fill the first register bit on each shift. For this, the taps are connected linearly.

4) The reader generates its own key stream and challenge n_R and sends it to the card.

5) The card deciphers the response from the reader and verifies if the authentication was successful. If so, the card generates a new respond and sends it to the reader [14] [5].

3.6.2 Mifare Desfire

The NXP Semiconductors introduced Mifare DESFire in 2002 and in 2006 was introduced successor Mifare DESFire EV1. In 2010 was announced discontinuation of the MIFARE DESFire and in 2016 was introduced MIFARE DESFire EV2. So now the DESFire family consists of Mifare DESFire EV1 and EV2 [11] [15].
The MIFARE DESFire family targets for transport or event ticketing, loyalty and micropayment, access control, parking, student ID, multiple applications and more [15].
"Certified with Common Criteria EAL4+ on both hardware and software implementation." [15]
"The "DES" in the name refers to the use of DES, 2K3DES, 3K3DES and AES hardware cryptographic engine for securing transmission data." [15]
Now there'll be described the evolution one and further the differences against the newer version evolution two.

Memory organization

Mifare desfire cards are available with 2, 4 and 8 kB non-volatile memory with typical write endurance up to 500 000 cycles. This memory is organized using flexible file system. On one card can be up to 28 applications simultaneously and in each application can be up to 32 files. Each application is represented by its 3 bytes AID (Application Identifier) and it has its own access requirements. File size is determined during creation and there is a 5 file types: standard data file, back-up data file, value file, linear record file and cyclic record file [16].

Memory security

MIFARE DESFIRE EV1 has UID 7 bytes long in the manufacturer part of memory, which is write-protected after being programmed by the

manufacturer. The UID contributes to prevent cloning.
"Prior to data transmission a mutual three-pass authentication can be done between MIFARE DESFire EV1 and PCD depending on the configuration employing either 56-bit DES (single DES, DES), 112-bit 3DES (triple DES, 2K3DES), 168-bit 3DES (3 key triple DES, 3K3DES) or AES." [16]
So the cipher algorithm is selectable by modifying access conditions [16].

Differences between Mifare Desfire EV1 and Mifare Desfire EV2

Evolution two is newer and it has following advantages against evolution one:

- Unlimited number of applications
- 6 file types - the new is Transaction MAC file
- Multiple Key Sets per application with fast key rolling mechanism (up to 16 sets)
- Multiple keys assignment for each file access rights (up to 8)
- Proximity Check for protection against Relay Attacks
- It's Common Criteria EAL5+ security certified which is the same security certification level as demanded for smart card IC (Integrated Circuit) products used e.g. for banking cards or electronic passports [15] [16] [17].

3.7 Mifare Classic hack

In this project is used Mifare Classic card as a key for the door lock and also there is demonstration of the hack with the use of mfoc tool so here will be discussed how the cipher was revealed and its vulnerabilities.

3.7.1 History

NXP held the algorithm of Crypto1 in secret, but researchers Henryk Plotz and Karsten Nohl at the University of Virginia revealed and then published the Crypto1 stream ciher and demonstrated a way to crack this encryption in December 2007. NXP tried to stop the full disclosure of Crypto1 cipher by judicial process, but court decided to allow the publication. Then were created some tools for hacking the Mifare Classic card. At this time this kind of cards was used worldwide in various applications such as a payment cards, public transportation and access control systems. Mifare Classic card is still being produced and sold today [12] [18] [13] [11].

3.8 How the cryptography was revealed

Crypto1 is implemented in chip hardware for faster ciphering process. So at first was needed to get access to the chip embedded in the plastic card. This was done with use of acetone [13].
"Once we had isolated the silicon chips, we removed each successive layer through mechanical polishing, which we found easier to control than chemical etching." [13]
The chip is formed by six layers and the lowest contains the transistor gates. They took pictures of it, example is shown in figure 3.6. Each type of transistor gate has a different shape, so they made templates for all kinds of all transistor gates founded on the chip (about 70 types of gates). With use of MATLAB they implemented a template matching of templates with pictures of the chip and finally they found components of the cipher in the corner of the chip. After reconstruction of the connection between all the transistor gates, they made the diagram of the Crypto1 stream cipher as its shown above in figure 3.5 [13].

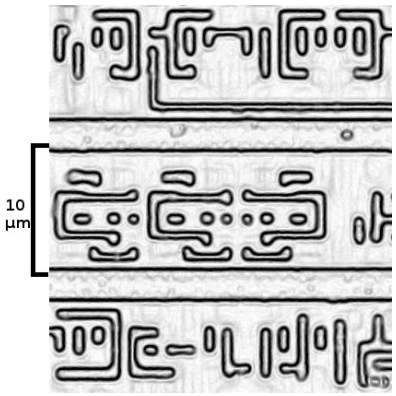

Figure 3.6: Picture of the transistor layer [13]

3.8.1 Weakness of Crypto1 algorithm

The 6-byte long ciphering key makes possible brute-force attack. Karsten Nohl and David Evans discuss how it's possible to try all possible keys within fifty minutes in [13].
Further weakness causes the random number generator (RNG designation in the Crypto1 diagram) which generates only 65536 numbers with a period 0.6 second and it generates every time the same numbers in the same order and always starts at the beginning. It allows an attacker to pre-calculate a generated number with this constant initial condition. Each number generated by RNG depends on the quantity of clock cycles between the time when the reader was turned on and the time when the random number is requested. Therefore, it's possible to control the generated number and by generating repeatedly the same numbers can be recovered keys from communication [13].

Chapter 4

RFID readers

This chapter describes the RFID reader principle, interfaces and usage. Further there are described readers, which where used for this project.

4.1 General

"A software application that is designed to read data from a contactless data carrier (transponder) or write data to a contactless data carrier, requires a contactless reader as an interface." [2]
"The reader's main functions are therefore to activate the data carrier (transponder), structure the communication sequence with the data carrier, and transfer data between the application software and a contactless data carrier. All features of the contactless communication, i.e. making the connection, and performing anticollision and authentication procedures, are handled entirely by the reader." [2]

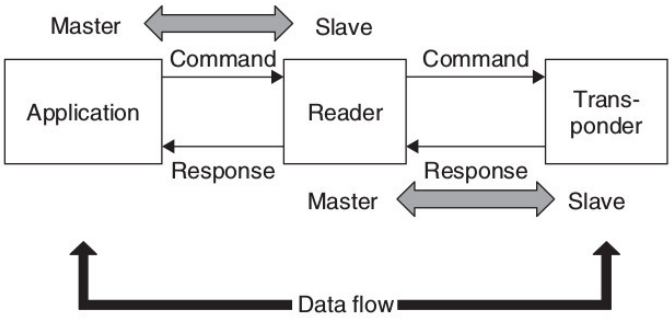

Figure 4.1: Diagram of master-slave communication between application, reader and transponder [2]

Figure 4.1 shows communication diagram between application, reader and transponder. In the wireless communication between the reader and the transponder, the reader is master and the transponder slave. In the communication between the application and the reader, the application is

master and the reader is slave. The application with its software can be for example in a computer with a program or development board with a firmware which handles an access control system. In next section is described a number of reader interfaces for the wired communication between the application and reader [2].

4.2 Reader interfaces

In this section are described common reader interfaces for communication with a master, including interfaces of readers which were used in this project.

4.2.1 Wiegand

Wiegand interface is very commonly used for RFID readers. Data are buffered into two separated signal lines before transmission over the wiegand interface. One line carries "1" and the other one "0", so these lines are typically called DATA1 and DATA0. For this operation of separating "1" and "0" into two lines is used a digital buffering circuit. Both data lines are held high and are pulled low when the current logic state is presented. So the logical "0" is represented by lowering the DATA0 signal, while the signal DATA1 remains high [19].

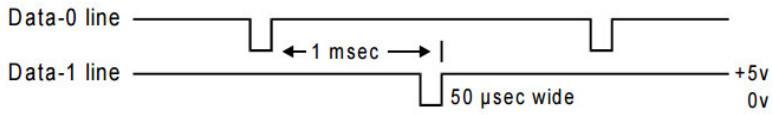

Figure 4.2: Wiegand signal [19]

Figure 4.2 shows how binary number 010 is presented, with a typical impulse width 50 microseconds and distance between two closest pulses is 1 millisecond. The actual values of timing, transmission rate, tolerance and voltage levels are determined by the device manufacturer.A short pause usually about 500 milliseconds follows after every data packet [19].

4.2.2 SPI

The SPI (Serial Peripheral Interface) is a very common serial interface for interfacing various electrical devices. The SPI enables a full duplex master-slave communication typically used at short distances. In one SPI connection is one master and one or multiple slaves and the whole communication is controlled by the master. The SPI uses four signal lines as listed in table 4.1 with a brief description [20] [21].

SS (Slave Select)	signal sent by master to enable or disable specific slave.
SCK (Serial Clock)	clock signal generated by master for synchronized communication
MOSI (Master Output, Slave Input)	line is used to send data from the master to the slave
MISO (Master Input, Slave Output)	line is used to send data from the slave to the master

Table 4.1: Definition of signal lines used for SPI

The master defines the transmission rate by setting frequency of the clock signal. Different devices support different maximal transmission rates. In one SPI communication the frequency of the clock signal must be suited to the slowest device [20].
For example the MFRC522 reader used in this project supports transmission speed up to 10 MBit/s [22].
The master keeps SS signal pulled high and when it's pulled low, the communication with the slave, to which is connected, is activated. In case there is only one master and one slave, all these signal lines are connected directly from the master to the slave. In case there are more slaves, the master needs more SS signal lines for each slave to be able to activate slaves individually [20] [21].

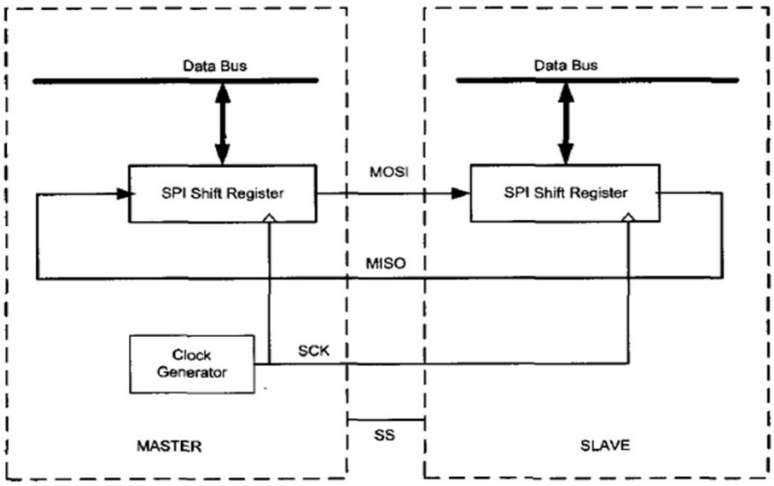

Figure 4.3: SPI diagram [20]

Devices with enabled SPI must have 3 exact registers as follows:

- SPI Control Register - contains eight bits that tell us about actual SPI configuration and can be rewritten by the master. This configuration contain: Interrupt enable, SPI enable, data order, Master/Slave select, clock polarity, clock phase and clock rate.

- SPI Status Register - also contains eight bits, but only three are used. First is SPI interrupt flag and is set when SS pin is an input and is driven low by master. Write Collision Flag is set if the slave writes on SPI Data Register during data transfer. Last used bit is set to double the SPI speed.
- SPI Data Shift Register. In figure 4.3 is shown how the data are shifted in both directions simultaneously. Master shifts out data from its register to slave and shifts in data from the slave and that's the way how the data are transmitted [20].

4.2.3 I2C

Inter Integrated Circuit (I2C) is a master-slave, half duplex serial bus invented by Philips Semiconductor (now NXP Semiconductors) widely used by almost all IC manufacturers usually to connect devices at short distances like EEPROMs, signal converters, sensors or other peripheral devices. The I2C protocol supports four transmission rates as shown in table 4.2 [23] [24].

Standard-mode (Sm)	100 kbit/s
Fast-mode (Fm)	400 kbit/s
Fast-mode Plus (Fm+)	1 Mbit/s
High-speed mode (Hs-mode)	3.4 Mbit/s

Table 4.2: I2C transmission rates

The I2C interface uses only two wires: SCL (clock signal line) and SDA (data signal line). Each clock cycle one bit is transferred, so the bit rate equals to the frequency of clock signal. Transferred data are split into 8-bit packets. These features make it very simple to implement, so there is no need for special I2C hardware controller like for SPI. In one I2C connection can be one or multiple masters and slaves [23] [24]. Figure 4.4 shows how multiple masters and slaves is connected.

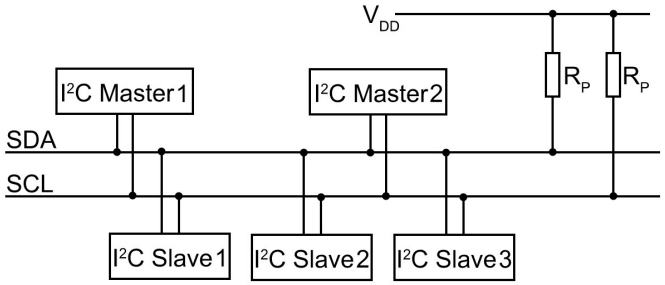

Figure 4.4: I2C diagram

"All devices are connected to the bus via an open collector or open drain. With this connection, the device may output either a logic zero or nothing at

all (the output is in the high impedance state)." [24]
That's why both lines SCL and SCK are connected via pull-up resistors to a positive source of power supply. When a device outputs "0", the line voltage will be set to low. If the bus is free, both lines are set to high. Pull-up resistors prevent short-circuit current when "0" is presented and it's resistance is typically in range from 1k Ohms to 10k Ohms [24].

Each slave device has a 7-bit address, which needs to be unique on the bus and it is set bet by master. When the master is to communicate with a slave, it starts to generate clock signal and it sends a start condition, which is one packet containing the slave's address and one bit which indicates data direction, "0" indicates write and "1" indicates a request for data. Depending on this one bit the next packet will send the master or the slave on SDA line. It's also possible to have a 10-bit addresses for a larger number of slaves. In this case the master's call command is separated into two packets.
During the communication the master can resend the start condition, that changes communication mode.
Once all the data were sent through the SDA, the master sends stop condition and stops generating clock signal, so other devices know that the communication has ended. At this point the I2C bus is free so other master device may use it [23] [24].

4.2.4 Serial UART

The Universal Asynchronous Receiver/Transmitter (UART) peripheral in accordance with TL16C550 standard is a block of circuitry implementing communication between parallel and serial interfaces. Therefore, it implements serial-to-parallel conversion of data received from a peripheral device and parallel-to-serial conversion of data received from a parallel device [25] [26].
"On one end of the UART is a bus of eight-or-so data lines (plus some control pins), on the other is the two serial wires - RX and TX." [26]

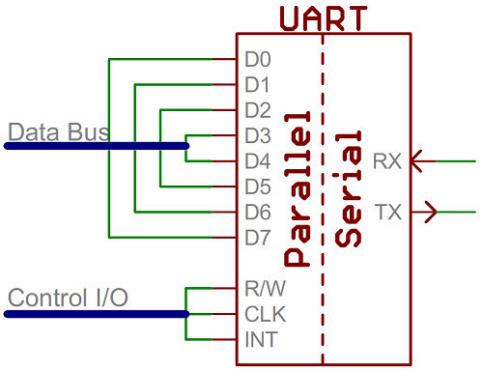

Figure 4.5: Simplified UART interface [26]

So with use of UART the data received from peripheral device are converted to parallel expression and the data received from the parallel device are converted to peripheral expression. The communication is synchronized by clock signal (CLK). Various devices supports various CLK speeds [25] [26].

4.3 HID Prox Point Plus

The Prox Point Plus reader, made by American manufacturer HID Global, is compatible with HID proximity cards with transmit frequency 125 kHz FSK modulation and card formats up to 85 bits.
There are two types of this reader: 6005B and 6008B, which differ in supported interfaces for communication with a host. Type 6005B supports wiegand and type 6008B supports clock-and-data [27] [28].

Figure 4.6: Photo of ProxPoint Plus reader [28]

It has integrated a beeper and multi color LED which can be controlled by master. Typical maximal read range is 7.5 cm. Power supply can be 5-16 VDC with peak current consumption 75 mA.
In this work is used type 6005B, which has two wires DATA-0 and DATA-1 for the wiegand interface, power supply wires VDC and GND and wires for controlling the LED [27] [28].

4.4 NXP MFRC522 reader

The MFRC522 reader made by NXP Semiconductors with communication operating frequency 13.56 MHz supports 14443 Type A (Mirare) and NTAG at range up to 40 mm.

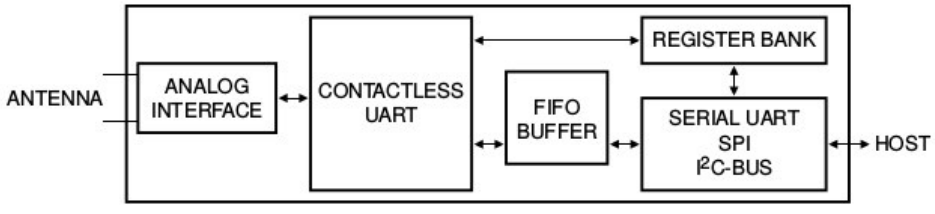

Figure 4.7: MFRC522 reader block diagram [22]

Figure 4.7 shows the reader diagram. Antenna with the analog interface

forms RF interface for wireless communication with a card. The contactless UART controls the communication protocol requirements for the host which is a serial port [22].
"The contactless UART manages the protocol requirements for the communication protocols in cooperation with the host. The FIFO buffer ensures fast and convenient data transfer to and from the host and the contactless UART and vice versa." [22]
The MFRC522 reader supports SPI, I2C and serial UART for interfacing a host. The reader identifies the host's interface automatically and then the reader's interface is automatically set. This is done when it's connected to power supply or when the host sends to the reader command hard reset.
In this project was chosen SPI which enables transmission speed up to 10 Mbit/s [22].

4.5 ACS ACR122U reader

The ACR122U reader is made by Advanced Card Systems (ACS) company from Hong Kong developing smart cards and smart card readers.
The ACR122U reader communicates at 13.56 MHz with all smart cards compliant with ISO/IEC 14443 Type A and B, MIFARE, FeliCa, and all 4 types of NFC ISO/IEC 18092) tags with transmission rate up to 424 kbps in typical communication range up to 50 mm between the reader and tag.
It has integrated bi-color LED and buzzer, both can be controlled by a host. The ACR122U has USB 2.0 full speed of up to 12 Mbps with supported CCID protocol (Chip Card Interface Device), which allows a smart card to be directly connected to a computer or other similar device. It supports application programming interfaces PC/SC and CT-API on operating systems: Windows, Win CE 5.0 and 6.0, Linux, Mac OS, and Android [29] [30].

Figure 4.8: ACR122U reader [30]

Chapter 5

Microcontroller development boards

Offline access control systems can use microcontroller development board as a central unit as it is in this work. This chapter discusses basic properties of microcontroller development boards and also this chapter describes the boards used in this work.

5.1 General

Microcontroller development board is a PCB (Printed Circuit Board) with a microcontroller. Basically it's a processor connected to circuity and hardware designed to facilitate working with the microcontroller. The board circuits usually contain power circuit, LEDs, pushbutton and interface like USB to connect the board to a computer for easy way to program the microcontroller from a computer. Additionally, the board may contain some hardware like bluetooth or WiFi module, LCD display etc. On the PCB are I/O pins with various abilities for usage [31] [32]. Table 5.1 shows list of pin abbreviations sufficient for boards used in this work.

Shortcut	Explanation
D	Digital I/O
A	Analog input
PWM	Pulse Width Modulation output
INT	Interrupt
USB+/-	It's connected to USB port
AREF	Analog Reference - configures the reference voltage used for analog input
Vin	Input power supply
RXD, TXD	pins for serial UART
MISO, MOSI, SCK, SS	pins for SPI
SDA, SCK	pins for I2C

Table 5.1: Pin abbreviations [31]

5.2 Arduino

Arduino is a company which manufactures family of open-source microcontroller development boards, hardware and software [33].

5.2.1 Arduino IDE

The Arduino company provides a programming interface Arduino IDE (integrated development environment) for easy to program Arduino boards, but many other boards are also compatible. It aims to write code, compile it and directly upload the code to the bard. Therefore, it contains text editor for writing the code, a message area for responses about compilation to the programmer, a text console, a toolbar with buttons for common functions and a series of menus. The programming language in Arduino IDE for programming microcontroller board is C and C++ with use of special rules to organize code with supplied library "Wiring" which includes many input/output functions [34] [33]. To make a program with this library are needed at least two functions. First is function setup() which runs only once and contains initial environment settings. The second function is loop() which is called repeatedly while the board is powered on [35].
The Arduino IDE itself is written in java programming language and it's available on platforms Linux, Windows and Mac OS X [36].

5.2.2 Memory

The Arduino and many other development boards are based on microcontroller with three kinds of memory.

Flash memory

This is the place, where programmer uploads his compiled program. This memory is non-volatile, so the inviolable when the power is switched off.

SRAM

In this memory the uploaded program stores variables and manipulates it. This memory is volatile, so the content will be lost when the power is switched off.

EEPROM

This non-volatile memory is empty until the programmer uses it for storing some information [37].

5.3 Arduino UNO

The Adruino UNO is a development board based on microcontroller ATmega328P. It has 20 pins whose description can be found in table 5.2 and the Arduino also contains pins for power supply 5 V, 3.3 V and GND. The I/O pin's operating voltage is 5 volts with maximal supplied DC current per I/O pin 20 mA and for 3.3V pin the current is 50 mA.
The ATmega328P combines 32 kB flash memory, 2 kB SRAM, 1 kB EEPROM with clock speed 16 MHz. The Arduino Uno board can be powered via the USB connection or with an external power supply connected to Vin pin, which can be from 6 to 20 volts, but from 7 to 12 volts is recommended. The Arduino Uno enables interfaces SPI, I2C, UART TTL (5V) and USB-B for communication with a computer, another board, microcontroller or other device [31].

Arduino pins	ATmega328P pins	features
0	PD0	D0 / RXD
1	PD1	D1 / TXD
2	PD2	D2 / INT0
3	PD3	D3 / INT1 / PWM
4	PD4	D4
5	PD5	D5PWM / PWM
6	PD6	D6 / PWM
7	PD7	D7
8	PD8	D8
9	PD9	D9 / PWM
10	PB1	D10 / SS / PWM
11	PB2	D11 / MOSI / PWM
12	PB3	D12 / MISO
13	PB4	D13 / SCK
AREF	-	AREF
A0	PC0	D14 / A0
A1	PC1	D15/ A1
A2	PC2	D16 / A2
A3	PC3	D17 / A3
A4	PC4	D18 / A4 / SDA
A5	PC5	D19 / A5 / SCL

Table 5.2: Arduino UNO pins [31]

5.4 Digispark USB development board

The digispark is a very small and cheap (can be bought for less than $2) development board developed by Digistump based on Attiny85 microcontroller compatible with Arduino IDE. It can be powered via built-in USB or external

source connected to Vin pin, which can be from 7 to 35 volts, but 12 or less is recommended.

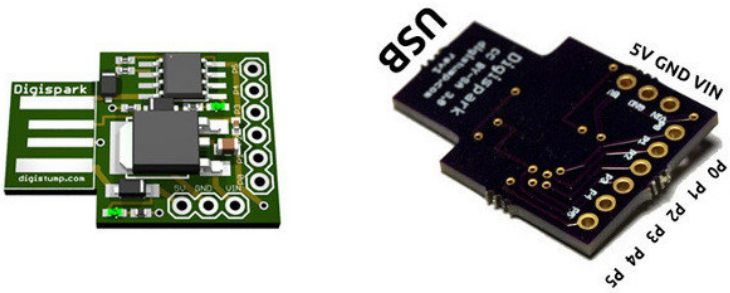

Figure 5.1: Digispark USB development board [38]

The Attiny85 is a microcontroller designed by Atmel with 8 kB flash memory 512 B EEPROM and 512 B SRAM. The CPU speed is 20 MIPS and it supports interfaces SPI and I2C.
The Attiny85 has 8 pins connected to digispark's pins whose features are described in table 5.3 in accordance with 5.1. Pin 3 and 4 are directly connected to USB+/-, so they can be used only if the Digispark is not communicating via USB [40] [38] [39].

Pin	features
P0	D0 / PWM0 / AREF /MOSI / SDA
P1	D1 / PWM1 / MISO
P2	D2 / A1 / SCK /SCL / INT0
P3	D3 / A3 / USB+
P4	D4 / A2 / USB- / PWM4
P5	D5 / A0
5V	Power source - 5V
GND	Power source - ground
Vin	Input power supply (7-35V)

Table 5.3: Digispark pins [39]

Part II

Practical part

Chapter 6

Design of the access control system

This chapter describes design of the access control system with NXP RFID technology at 13.56 MHz.

6.1 Part list

- Arduino Uno development board
- NXP MFRC522 reader
- Mifare Classic 1K
- Relay 1 channel 5VDC
- Solenoid 12V 1A
- LED two color RG antiparallel
- Buzzer
- Press button - Microswitch SMD 1-pole switching ON-OFF
- R - resistor 4k7

This system contains LED and buzzer for the feedback, so user knows what the system actually does. The press button is here for the system editing (as explained further in the text). The relay switches power supply for the solenoid, which controls the door.

The Mifare Classic card (with 4 bytes long UID only) is chosen, because it's very popular in the world for various applications, such as access control systems and it costs about $0.30 per one when it's purchased in amount of 50

pieces. A compatible reader with the Mifare card is chosen MFRC522 which costs about $1.89, so this RFID technology is quite inexpensive.

The development board requirements are: reliability, at least 10 I/O pins available, 3.3 and 5 volts power supply for powering the connected components and memory, for user data, because we need to store UID of every authorized card. Therefore, Arduino Uno based on ATmega328P appears as the best solution.

6.2 Diagram

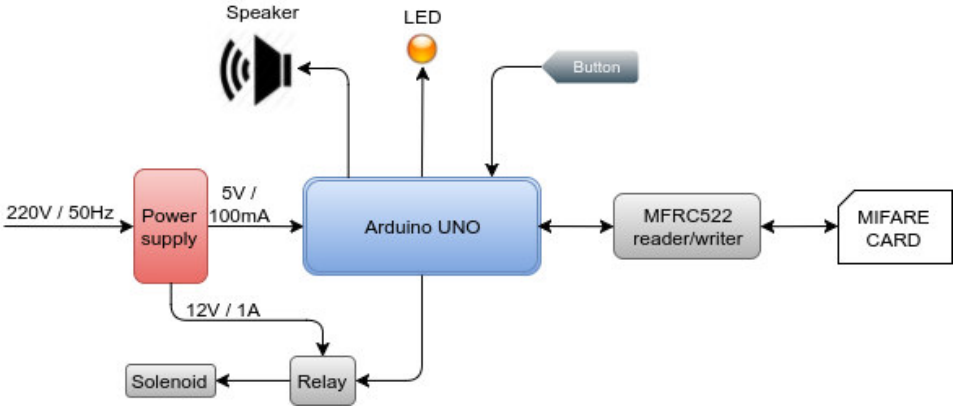

Figure 6.1: Access control system block diagram

As shown in figure 6.1 there is needed power source with two DC outputs. One 5V / 100mA for Arduino and 12V / 1A for the solenoid. The Arduino board can be powered by USB-B and all other parts are powered by the Arduino.

Pin	Connected line
4	relay
5	buzzer
6	press button
7	LED green
8	LED red
9	MFRC522 - RST
10	MFRC522 - SDA(SS)
11	MFRC522 - MOSI
12	MFRC522 - MISO
13	MFRC522 - SCK

Table 6.1: Arduino UNO pin connection

The MFRC522 reader is connected to Arduino via SPI and for the communication is used Arduino SPI default library. For controlling the reader is used this library [41]. More about SPI and MFRC522 reader is in

chapter RFID readers. The Arduino Uno has predefined pins 10-13 for SPI and this library allows changing SS pin and in addition it has RST (reset) pin for reseting the SPI configuration. For wiring the reader is used it's datasheet [22].
The door is controlled by the solenoid which is switched by the relay and the relay is connected to I/O pin of the Arduino Uno. The solenoid needs 12 V, 1A DC power supply.
The buzzer is connected to digital output of Arduino and it generates tones depending on switching frequency between digital low and high.
The antiparallel two color RG LED operates at 2.2 V, 20 mA (max. 25 mA). Arduino's maximal DC Current per I/O Pin is 20 mA so it can be directly connected to digital pins. To set the LED to light red or green color, one pin has to be set to high and the other is set to low. That means on one pin is delivered 5 volts and the other is grounded.

Button connection

In figure 6.2 is shown connection of the button to the Arduino Uno board. Arduino's pin is defined as input and it reports digital high on this input if a voltage there is greater than 3 volts and digital low if a voltage there is lower than 3 volts.
Five volts power supply is connected over the resistor to the pin, where the voltage is actually being sensed. If the button is not pressed, Arduino will report digital high on this pin. When the button is pressed, there is a short circuit over the button to the ground and delivered voltage on the sensing pin is close to zero, so at this point it reports digital low. Short circuit current is limited to approximately 1 mA by resistor R = 4.7 kohm.

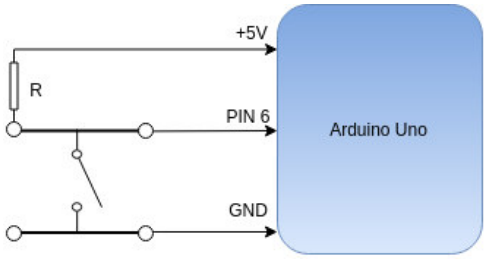

Figure 6.2: Button circuit connection

6.3 Program description

In this section is described how the software is designed and how the system works. Authorized cards are divided into admin and user cards. Both are used as a key to the door and the admin card has in addition ability to add or remove other cards from the system.

6.3.1 Card authentication

To make it as much secure as possible with this kind of card, there are two kinds of verifying.

1) UID of every authorized card is stored in Arduino's EEPROM whose size is 1024 bytes. The card's UID is 4 bytes long, so the system has capacity of 256 authorized cards. The system was not tested with cards, which have UID 7 bytes long, but probably it would also work and in the Arduino's EEPROM would be only saved the first four bytes of the UID. When the authentication starts, the Arduino looks in its EEPROM if the UID of the scanned card is there. That's for the case, that some user would lose his card. Who finds it would be able to pass through the door, while in this design we can wipe out the EEPROM and then add all the cards to the system.

2) The Mifare Classic card provides cryptography to protect the user's data written on the card, so in the next part of the card authentication it reads a number written on the card. This number is stored in sector 2, block 8. Sum of all 16 bytes in this block gives a number and according to this number is recognized admin and user card. User's number is 2137 and admin's is 2343. Those numbers were chosen randomly. The sector 2 is enciphered with key A: 10 20 30 40 50 60 which is needed to get access to this sector.

Figure 6.3 shows the structure of the card authentication. This loop persists, when the system is in normal state. If something goes wrong during the authentication, system evaluates that attached card is invalid and access is denied, speaker beeps a negative tone and the LED lights red for a second. When a card is authenticated as user or admin, the buzzer beeps positive tone, LED lights green for a two seconds and the solenoid releases the door. If the authenticated card is user, it returns back to start. If the authenticated card is admin, additionally the system waits 4 seconds if the button will be pressed and how long. If so, it enters to the system editing loop and it behaves depending on how long was the button pressed as shown in 6.4. To recognize if a presented card was recognized as user or admin, the buzzer beeps a little longer for the admin card than for the user card.

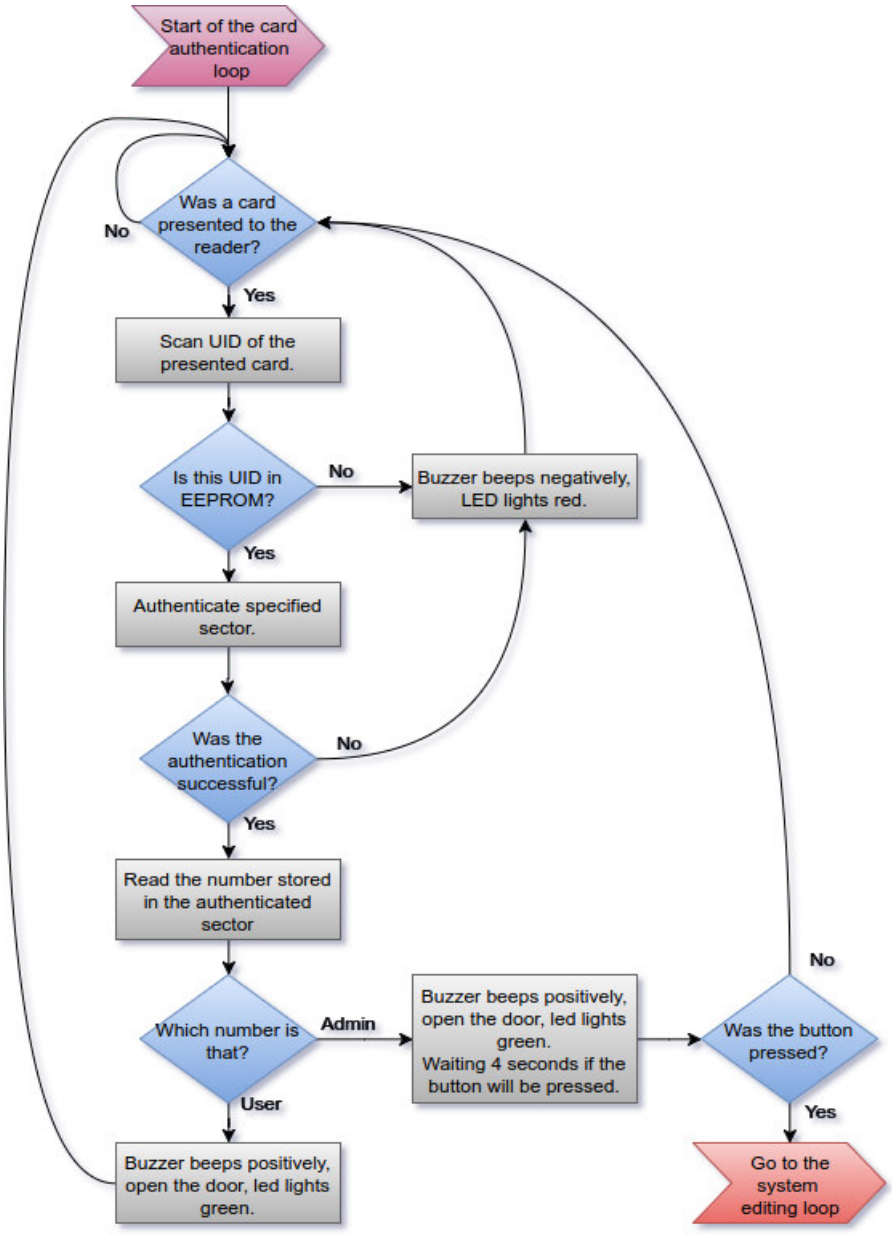

Figure 6.3: Flowchart of the card authentication.

6.3.2 Editing the system

By presenting the admin card and pressing the button can be added a new user or admin card to the system or deleted any card from the system and delete all the cards at once from the system. It's done by pressing the button for a specific duration. Figure 6.4 shows the editing procedure.

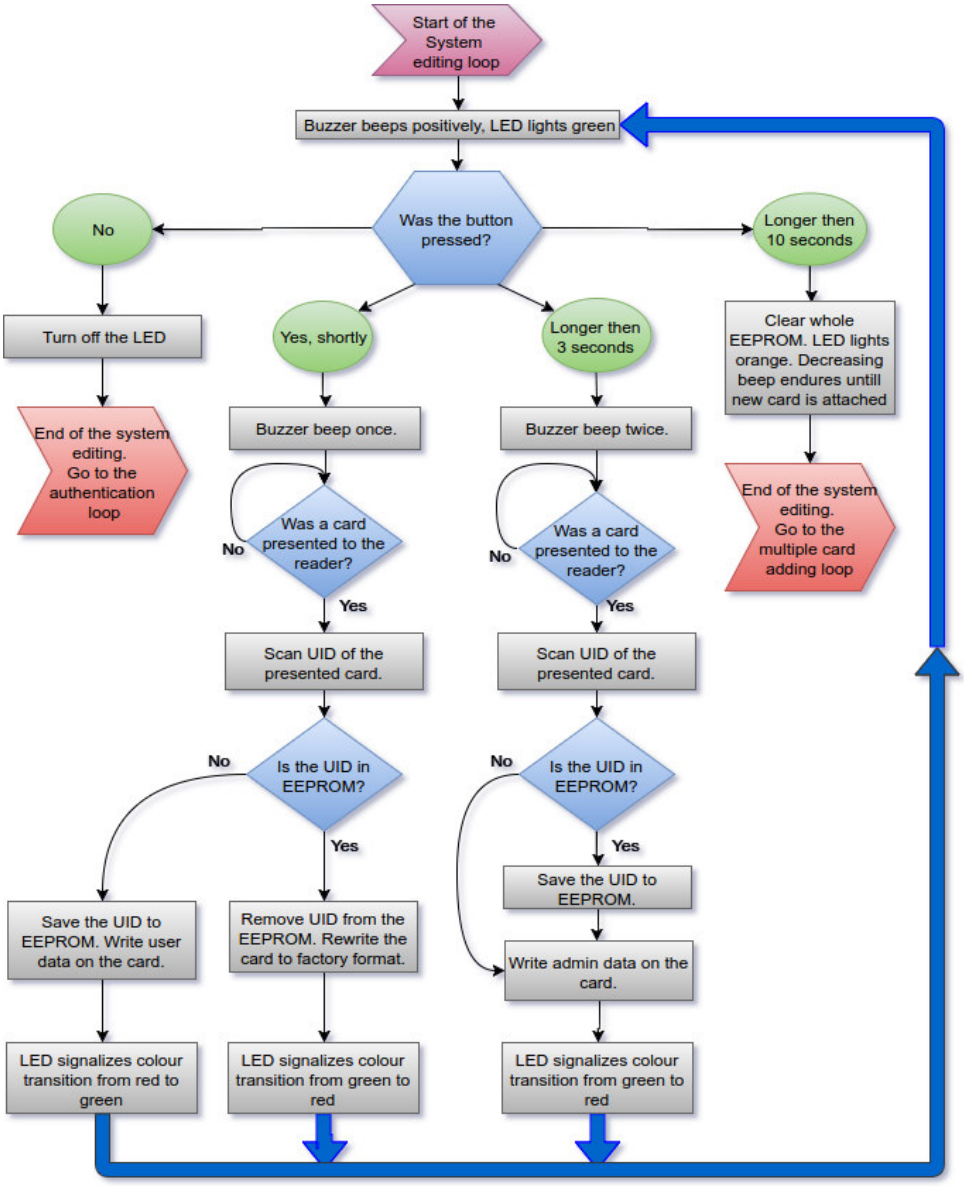

Figure 6.4: The system editing flowchart.

When the admin card is authenticated, the system gives feedback by the buzzer and LED as written upwards and then it waits for four seconds if the button will be pressed. If not, no edits will happen and it returns back to authentication loop. After every operation of system editing it returns back to the editing loop, so it gives feedback by speaker and LED and waits again four seconds if the button will be pressed.

Add or delete card from the system

If the button is pressed shortly (no longer than three seconds), the system waits 30 seconds for any card to be presented. If no card is presented in this duration the editing loop ends, LED turns off and it goes to the authentication loop, but if a card is presented, it looks if the card is already in the system. If so, the card will be removed from the system. If the presented card is not already in the system, the card will be added to the system and as user.
In authentication loop when an admin card is presented and the button is pressed shortly, the next presented card will be removed or added as user to the system depending on if the UID of the card is in EEPROM.
If the system evaluates that it's the same admin card, which was presented in the authentication loop to enter to the system editing loop, it won't remove this card from the system and it jumps at the start of the system editing loop. This ability is not shown in the diagram because it doesn't fit there. This ability is implemented here because irresponsible removing cards would may cause, that there is no admin card therefore there would be no possibility of adding or removing cards anymore.

Add admin card to the system

If the button is pressed longer than three seconds and not longer than ten seconds, he system waits 30 seconds for any card to be presented. If no card is presented in this duration the editing loop ends. But if a card is presented, it will be added to the system and set as admin.

Remove all the cards from the system at once and multiple card adding to the system

If the button is pressed longer than ten seconds, the system's memory is gonna be wiped out. The LED lights orange and buzzer beeps falling tone and the tone remains until any new card is presented to add the card to the system and set as admin. At this point the system enters the multiple card adding loop.

6.3.3 Multiple card adding

In case when the system detects that the EEPROM is clear, so no card is authorized, it enters to the multiple card adding loop as shown in figure 6.5. It simply sets as admin all the presented cards until the button is pressed, then all the next presented cards are set as user and when the button is presented again, it quits this loop end enters to the authentication loop.

Every time when the system is powered on, it checks if the EEPROM is clear. If so, it enters to this loop, thus user of this system has an easy way to add multiple cards to the system, when it's brand new.

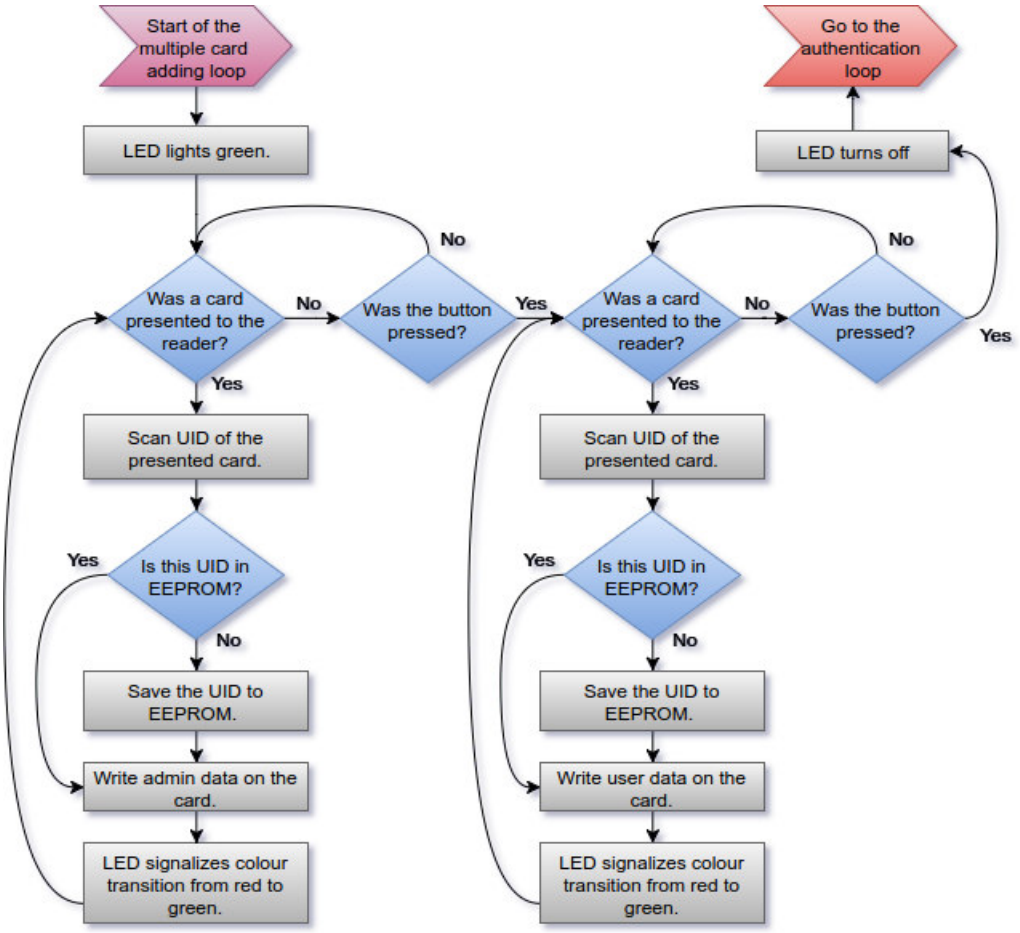

Figure 6.5: The multiple card adding flowchart.

Chapter 7

Design of the access control system - reduced and smaller solution

This chapter describes the design of smaller solution than the solution in previous chapter. The main requirement for this system is minimum size for the best practical use.

7.1 Part list

- Digispark USB Development Board
- HID ProxPoint Plus 6005B reader
- HID ProxCard II - 26 bits version
- Relay 1 channel
- Solenoid 12V 1A

The RFID technology HID reader with the HID card is chosen, because it's directly made for access control systems and it's widely used for this application. The reader can be found on the market for about $90 and one Prox Card cost about $3 when it's purchased in amount of 50 pieces. So it's much more expensive than the NXP RFID technology used for the system design in the previous chapter.

The HID reader has integrated buzzer and multicolor LED, so there is no need of these components in this system design anymore. It's preprogrammed, that when a card is presented, the reader's integrated buzzer beeps and it sends bits of the presented card via two wiegand wires.

The door is controlled by the solenoid which is switched by the relay and the relay is connected to I/O pin of the Digispark development board, so it's the same as it is in the previous system design in the previous chapter.

The requirements for the development board are: reliability, small dimensions, at least 5 I/O pins, 5 VDC power supply for the reader and the relay, memory for user data, because we need to store numbers of all authorized cards. The Digispark development board based on Attiny85 appears as the best solution, because it fulfills all the conditions and it's very small and cheap (It can be bought for less than $2).

7.2 Diagram

This system uses the same relay and solenoid as it is in the previous system in the previous chapter. The Digispark board is powered via USB, the HID reader and relay is powered by the Digispark and the solenoid needs power supply 12 V, 1 A.

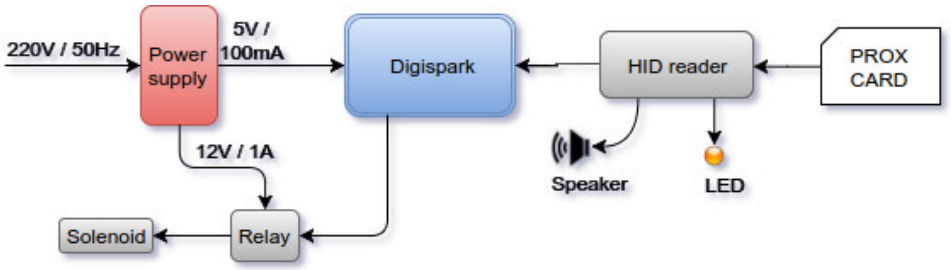

Figure 7.1: Access control system - smaller solution

P0	HID DATA-0
P1	LED red
P2	HID DATA-1
P3	LED green
P4	relay

Table 7.1: Digispark pin connection

The wiegand signal on the reader's wires DATA1 and DATA0 has a very short pulses, so the best solution to sense these pulses is to use pin change interrupt function available for interrupt pins but the Digispark provides only one interrupt pin and there are needed two. For this reason is used external library PinChangeInterrupt [42] which provides interrupt function for all pins

of the Digispark.

The integrated LED in the HID reader has three colors and for every color it has it's own controlling wire. In this project is needed only red and green color, so the led is controlled by only two wires. To set the LED to light one of these colors, wire of the color must be grounded. Thus, it is solved in that both wires for red and green color are set to HIGH in idle state and when the LED should glow in some color, wire of the color is set to LOW.
For wiring the HID reader is used source [28].

7.3 Program description

Authorized cards in this system are divided into admin and user cards, which is the same as in previous chapter, but only the user card is used as the key for accessing the door. The admin card is here for the system editing.

Numbers of all authorized cards are stored in the EEPROM of the Attiny85. These numbers are there stored as unsigned inteegers covering two bytes and the EEPROM is size 512 bytes. Numbers of user cards are stored on addresses from 0 to 399, and admin card numbers are stored on addresses from 400 till the end, so it's up to 512. It follows from this that this device has capacity of 199 user cards and 56 admin cards.

7.3.1 First turned on

When it's the first time turned on, the system evaluate that its EEPROM is clear, so it enters to the admin card adding loop. The LED lights green and all the presented cards are added to EEPROM as admin. This admin card adding ends by presenting the first presented card, then it enters to the authentication loop. To add some user cards, some admin card must be presented as shown in following diagrams.

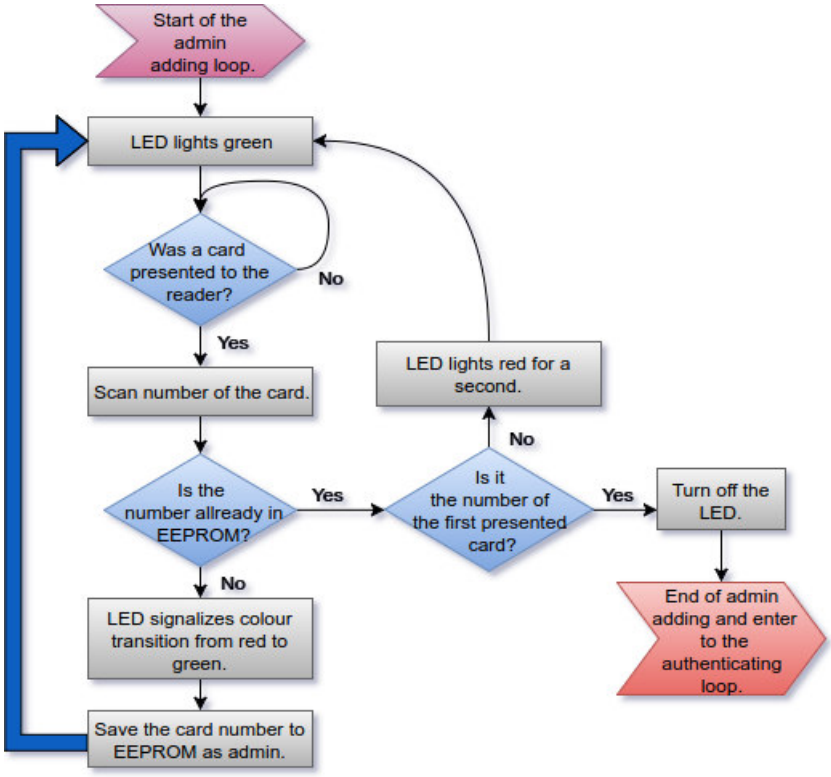

Figure 7.2: This flowchart runs when the system is first time turned on, so it has clean EEPROM.

7.3.2 Card authentication

This is the normal state of the system. It simply reads the number of the presented card and then it looks if the number is already stored in the system's memory. If so, it opens the door or enters to the user editing loop depending on if the presented card is user or admin. If some admin card is presented twice in a row, it enters to the admin card list editing and if the same admin card is presented again, it returns back to the authentication loop.

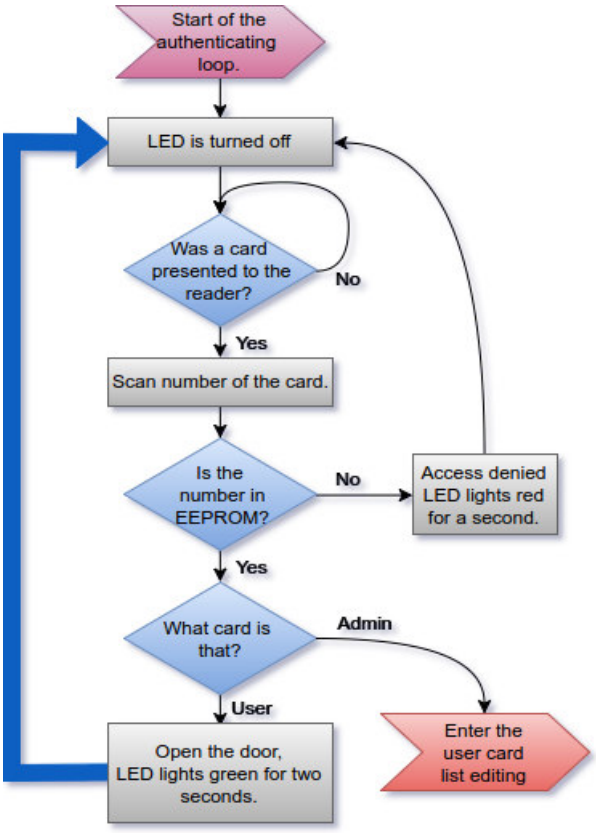

Figure 7.3: This flowchart shows authentication loop which runs when the system is not in editing state and any card is presented to the reader.

7.3.3 User card list editing

When some admin card is presented in the authentication loop, It enters the user card list editing. The next presented cards are added to EEPROM as user if they already are not so or removed from the EEPROM if they already are there as user. This loop ends and it jumps to authentication loop, when the same admin card, which was presented at the beginning is presented again.

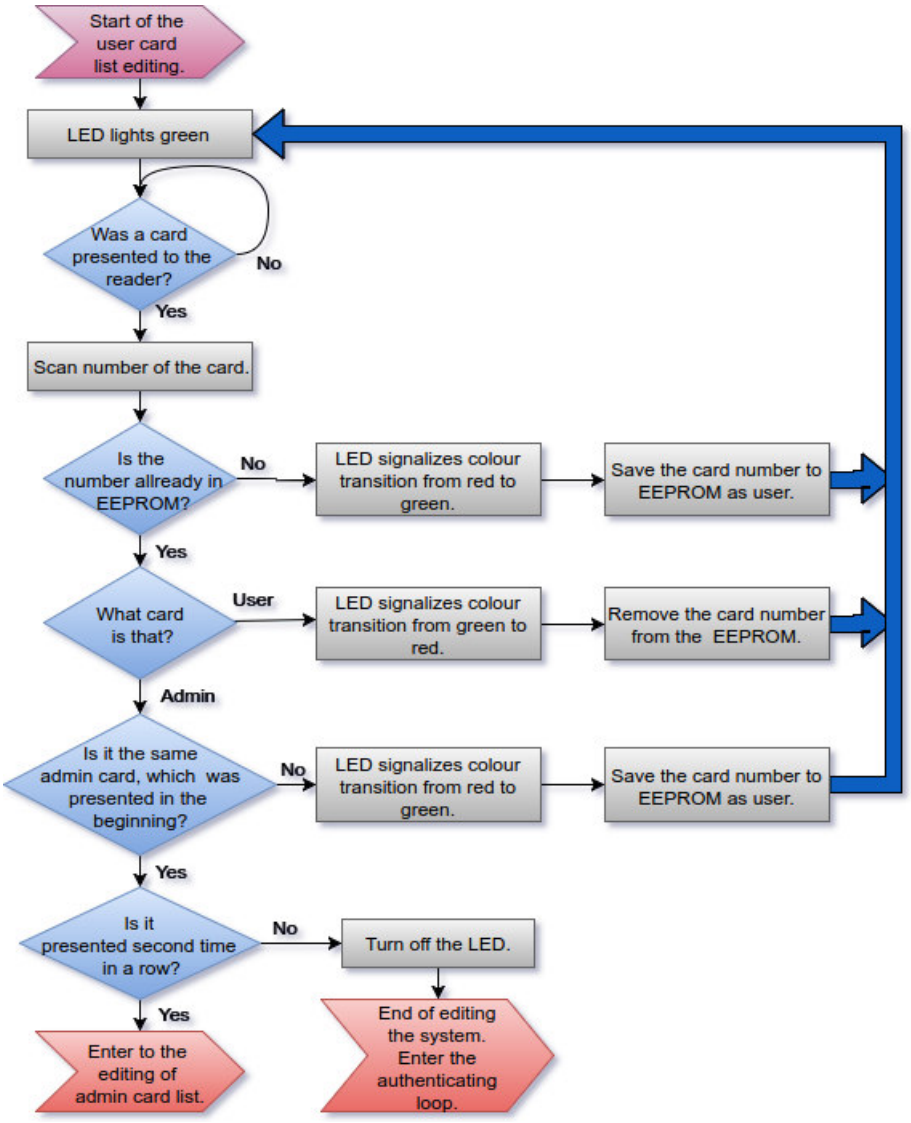

Figure 7.4: This flowchart shows the user card list editing loop.

If no card is presented longer than 30 seconds, the editing loop ends and it jumps to card authentication loop. When it enters to the user card list editing, the LED lights green and when it enters back to the authentication loop the led turns off, but when some cards are added or removed from the system, the LED signalizes color transition as shown in the diagram.

7.3.4 Admin card list editing

When some admin card is presented twice in a row in the authentication loop, it enters to the admin card list editing. The next presented cards are added as admin if they already are not so or removed from the EEPROM if they already are there as admin. This loop ends and it jumps to authentication loop, when the same admin card, which was presented at the beginning, is presented again.

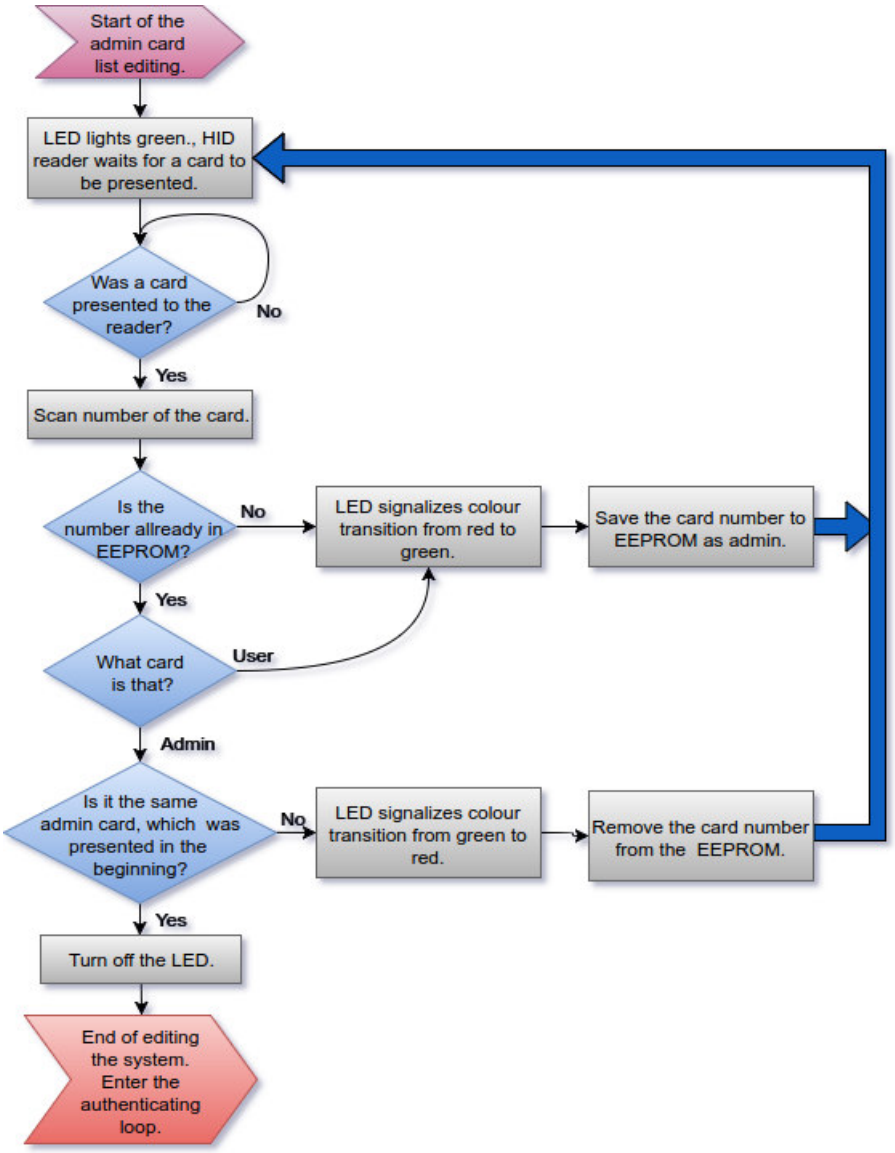

Figure 7.5: This flowchart shows the admin card list editing loop.

Following is the same as in the user card list editing. If no card is presented longer than 30 seconds, the editing loop ends and it jumps to card authentication loop. When it enters to the admin card list editing, the LED lights green and when it enters back to the authentication loop the led turns

off, but when some cards are added or removed from the system, the LED signalizes color transition as shown in the diagram.

Chapter 8

Hacking the electronic door lock system

This chapter includes discussion of security of the designed access control systems from two previous chapters.

8.1 Possibilities of hacking access control systems

One way to pass through access control system is to make a clone of the card which is used as a key. To execute this, the hacker needs to get access to the card used in the system, to be able to scan data from that card. Some RFID cards provide encryption like Mifare cards based on ISO/IEC 14443 to prevent unauthorized reading data from the card. However, the cryptography of some cards such as Mifare Classic has been broken by researchers who are analyzing vulnerabilities of the cryptography.

The other way to pass through the access control system is to break into the case where is the central unit. The hacker in this case has an opportunity to program the central unit and control the electronic door lock by its controlling wire.

8.2 Mifare Classic hack documentation

In this section is documentation of the Mifare Classic hack to prove that it's possible to reveal enciphering keys and data on the card.
For this hack was used Linux distribution Ubuntu, version 16.04 LTS, reader ACS ACR122U from the chapter RFID readers. Sample card for the hack was chosen the card used in the electronic door lock in chapter 6.
To set up a "hacking lab" were installed following tools:

1. pcsc-tools [43] which is the driver for the reader.

2. libnfc [44] which is a free open source library, SDK and API for RFID and NFC applications supporting ISO/IEC 14443 Type A and Type B, FeliCa, Jewel/Topaz tags and Data Exchange Protocol (P2P) as target and as initiator.

3. mfoc [45], Mifare Classic Offine Cracker, which is an open source tool depending on libnfc for cracking Mifare Classic cards, written by Nethemba Szetei and Pavol Luptak with contribution by others.

At this point is important to successfully install compatible versions of mfoc and libnfc. For this hack was used mfoc version 0.10.7 and libnfc version libnfc-1.7.1-107-g8e5ec4a. When these tools works properly, the ACR122U reader is plugged in the PC, the card is attached to the reader, than everything is ready, so in terminal is send command which starts the hacking.
mfoc -P 500 -O dump.mfd
Where -P 500 means probe each sector up to 500 times and -O dump.mfd means create a mfd file and seve here output. If the attempt is successful, it will reveal keys of each sector and it prints out whole content of the card and in addition it saves the card content to the dump.mfd file.

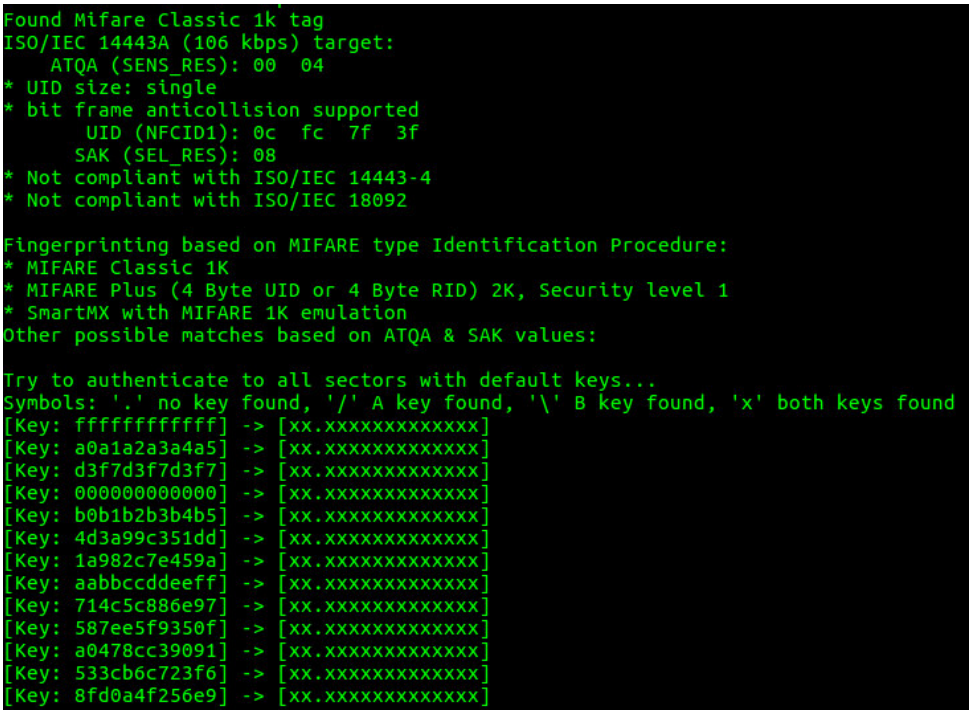

Figure 8.1: Mifare Classic Hack - terminal output part 1

1. Figure 8.1 shows, that some details about the card are announced and then a few common keys including factory key are tried to unlock every single sector as a key A and key B.

```
Sector 00 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 01 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 02 - Unknown Key A               Unknown Key B
Sector 03 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 04 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 05 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 06 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 07 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 08 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 09 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 10 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 11 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 12 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 13 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 14 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
Sector 15 - Found   Key A: ffffffffffff Found   Key B: ffffffffffff
```

Figure 8.2: Mifare Classic Hack - terminal output part 2

2. Figure 8.2 shows, that it lists keys with which it managed to authenticate individual sectors. Be noticed, that all the sectors are using factory key as a Key A and B except sector 02 so this sector will be probed 500 times.

```
Using sector 00 as an exploit sector
Sector: 2, type A, probe 0, distance 14951 .....
Sector: 2, type A, probe 1, distance 14953 .....
Sector: 2, type A, probe 2, distance 14901 .....
  Found Key: A [102030405060]
  Data read with Key A revealed Key B: [95f4fa4172cd] - checking Auth: OK
Auth with all sectors succeeded, dumping keys to a file!
```

Figure 8.3: Mifare Classic Hack - terminal output part 3

3. Figure 8.3 shows, that the sector with unknown keys is being probed by counting the RNG cycles to predict the generated number, so distance means order of generated number. At this time it took only three probes and only a few seconds, but in other cases this step can take even a few minutes. Untill when the key A is known, it also reads and prints key B of this sector.

```
Block 15, type A, key ffffffffffff :00 00 00 00 00 00 ff 07 80 69 ff ff ff ff ff ff
Block 14, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 13, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 12, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 11, type A, key 102030405060 :00 00 00 00 00 00 ff 07 80 69 95 f4 fa 41 72 cd
Block 10, type A, key 102030405060 :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 09, type A, key 102030405060 :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 08, type A, key 102030405060 :0b 8a f4 ac 1d 92 ff 07 80 69 4a 31 fa cd 60 e4
Block 07, type A, key ffffffffffff :00 00 00 00 00 00 ff 07 80 69 ff ff ff ff ff ff
Block 06, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 05, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 04, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 8.4: Mifare Classic Hack - terminal output part 4

4. And finally it dumps whole card with keys. Figure 8.4 shows part of this message including sector 02, because only this one contains useful data and all other sectors are in factory format. So the sector 02 uses key A: 102030405060, key B: 95f4fa4172cd and the useful data are in the block 08.

8.2.1 Attempt of hack a strange Mifare Classic card

To verify the functionality of these tools for dracking cards, the hack was also implemented on ISIC (International Student Identity Card) shown in figure 8.5. It's 1k variant and the card is already being used as electronic wallet for canteen and cafeteria, access control system and as a key for electronic door lock of the school buildings.

Figure 8.5: Sample ISIC card used for hack

The hack was also useful. All sectors are in factory format accept sector 01 and useful data are stored only in block 04. In figure 8.6 is shown part of the final message including the sector 01 and as you can see it uses key A: bd493a3962b6 and the secret data are at the begining of block 04 which is 54 54 58 4d 57 51 4d.

```
Block 11, type A, key ffffffffffff :00 00 00 00 00 00 ff 07 80 69 ff ff ff ff ff ff
Block 10, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 09, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 08, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 07, type A, key bd493a3962b6 :00 00 00 00 00 00 78 77 88 00 00 00 00 00 00 00
Block 06, type A, key bd493a3962b6 :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 05, type A, key bd493a3962b6 :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 04, type A, key bd493a3962b6 :54 54 58 4d 57 51 4d 00 00 00 00 00 00 00 00 00
Block 03, type A, key ffffffffffff :00 00 00 00 00 00 ff 07 80 69 ff ff ff ff ff ff
Block 02, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 01, type A, key ffffffffffff :00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Block 00, type A, key ffffffffffff :c3 c9 5a ef bf 88 04 00 48 ba 14 50 61 50 11 10
```

Figure 8.6: ISIC card hack - important part of terminal output

According to standard ASCII (American Standard Code for Information Interchange) these bytes means TTXMWQM which is written on the card below the barcode. All the data about individual student must be stored on server and the card is only used to provide the 7 bytes long code.

8.3 Security of the HID RFID technology used in the second system design

The HID Prox Card II is read only, but also on the Chinese market is available writable Prox Card II and writing device [46], so it's possible to clone these cards.

If the hacker does not have the card and he doesn't know the correct card number, there is an option how to get the number, but it's applicable only for the case when it's possible to remove the reader from the wall and hide behind the reader some sensing device. So the hacker needs to tear the reader out of the wall and connect to the wiegand wires of the reader some device which can sense the wiegand signal from the reader's wires and store card numbers in memory, for example some small development board like the Digispark or Arduino Micro. When somebody presents a card to the reader to open the door, the hacker's sensing device catches the card number and saves it in it's memory. Then the hacker takes his device then and he has the correct card numbers.

If the hacker has access to the wiegand wires of the reader, he can also bypass the RF interface and directly emulate the reader by sending the card number by his device to the readers's wiegand signal lines.

8.4 Conclussions of the card cloning

So it's possible to read whole content of any Mifare Classic card and write the data to another card to make a clone. But there is one more thing which makes it difficult to make a perfect clone. The manufacturer data stored in sector 0 block 0 including UID are not writeable. However, there are cards on the Chinese market with possibility to overwrite the manufacturer block thus UID too with price less than 1$ per one piece.

It follows from that both types of cards, Mifare Classic and Prox Card II can be cloned and in some cases where the HID reader can be removed from the wall, it's possible catch the correct card numbers and bypass the RF interface. However, to clone a card we need to make a scan of the card and to scan the Mifare Classic card can take even about a few minutes because of cracking the cryptography while scanning the Prox Card II can be done in less than a second, because there is no protection, thus it can be done through the purse in public transport and the card user would not notice that somebody scanned his card. To prevent it, on the market are available RFID signal blocking wallets, which is not more expensive then common wallet.

8.5 Other options of hacking the designed systems

The electromechanical door lock system or solenoid, which controls the door is switched by relay, which has wire connected to the central unit and if the signal on this wire is set to HIGH state, there is delivered 5 volts power supply and it releases the door and when the signal on this line is set to LOW

state, its grounded and the door is locked.
If the hacker get access to the microcontroller central unit, he would be able to program it. And set the signal for the door lock system to HIGH state so the door would be released.
If the hacker get access to the wire which controls the electromechanical door lock system or solenoid, he would be able to open the door by connecting required power supply to these wires, which is 12 V, 1 A for the solenoid used in this work.
It follows that to make secure system, there must be no possibility of getting access to the central unit, wires of the reader or wires of the electromagnetic door lock. If these rules are upheld and the hacker don't know anything about the data stored on authorized cards in the system, he is unable to pass through the door.

Chapter 9

Conclusions

In this work were designed two access control systems with different RFID technologies and features.

The first designed system has central unit Arduino Uno based on ATmega328P, and uses Mifare Classic card as a key to the door. The MFRC522 reader made by NXP was chosen as an interface to the data on the card. This reader is inexpensive (it costs less then $2), however, it doesn't contain any components like a LED or buzzer to get feedback to the user, so these components are added externally. For easy to edit the system, it has a press button, which is active four seconds after presenting the admin card, so both types of cards, admin and user are used as a key to the door and admin card has moreover ability to edit the system. That means that with use of admin card can be added or removed any admin or user card.

Requirements for the second system design are simplicity and small dimensions. So it uses HID reader, which has integrated buzzer and LED, so there is no need for components for the feedback to the user. So there can be used board with only a few pins for interfacing the HID reader and one pin for controlling the door. That's why the Digispark development board based on Attiny85 is used as a central unit of this system. There is no button for easy to edit the list of authorized cards, so admin cards are used only for the system editing and user cards are used as a key to the door.
A great benefit of this device is that it's very small, which makes it easy to install and possibility to use it for applications with a bad space condition.

Both system works on the purpose, that they store list of all authorized cards in the memory of the central unit microcontroller, but the device with Mifare cards uses moreover memory authentication and reading a number stored in the card's memory for greater security. The device with Mifare cards stores 4 bytes long UID of all authorized cards and the ATmega328P

provides EEPROM with size of 1024 bytes, therefore this device has capacity of 256 authorized cards. The device with the HID Prox Card stores number of every authorized card as unsigned 2-byte integer and the Attiny85 provides EEPROM with size of 512 bytes. This memory is divided into space for user cards and spece for admin cards, hence the capacity of 256 cards is divided into capacity of 199 user cards and 56 admin cards.

Both systems were constructed and programmed according to the design. All the programming was implemented in Arduino IDE. Programming the Arduino Uno was easier, because this board is fully supported by the Arduino IDE.
Programming the Digispark was more complicated, because it has following disadvantages. It doesn't support the console of the Arduino IDE, so in case it doesn't work properly, the programmer might don't know what the system does or what went wrong. The Digispark must be always unplugged and then plugged in to the PC when a program is being uploaded on it. Another complication is that two Digispark's I/O pins are connected to USB, so these pins must be disconnected to be able to upload a program on it.
These complications are make it harder to program, but on the other side, it works properly and reliably like the Arduino Uno.
Eventually both systems were programmed well and it works properly. Functionality of both systems has been verified for all possible cases what these systems can do, including adding or removing admin and user cards.

Reliability of these designed systems depends on the used hardware, firmware, complexity of the design and the RFID technology.
The firmware of these systems is programmed as simply as possible to make it reliable.
The system with the NXP MFRC522 reader sometimes fails during the encrypted communication. The communication is based on the Arduino libraries MFRC522 for the communication between the card and the reader and SPI for the communication between the reader and the Arduino Uno, so this problem is probably related to them or to one of them. This failure is the most common in case, when a card is attached to the reader not close enough. So for the most reliable communication the center of the card must be attached to the center of the reader antenna coil as close as possible.
The HID reader doesn't have such problem because its communication is much more simple.

There is not done power supply for both systems. For both cases in the system design is written that the development board is powered via USB and the solenoid needs external power supply 12 V, 1 A. It's also possible to power Arduino Uno and Digispark by the 12 V, 1 A power supply via "Vin" pin. Selection of the way power supply depends on conditions and options in

exact situation where the system is mounted for its application.
The maximum load of the relay is AC 250V/10A or DC 30V/10A. So the solenoid can be replaced with any electromagnetic door lock, but there is also need to change the power supply for the exact chosen door lock instead of the 12 V, 1 A for the solenoid.
If user of the designed systems wants to replace the solenoid with electromagnetic door lock and use the same power supply 12 V, 1 A, he could choose this one [47] made by SDC (Security Door Controls) or many more made by this manufacturer and also there is many compatible electromagnetic door locks at the market from many manufacturers, because very popular power supply for these locks is 12 V and 24 V with current less than 1 A.

The chapter 8 discuss possibilities of hacking the designed systems and also there is demonstrated hack of the Mifare Classic card used in the first design of the access control system and additionally this hack was successfully tested on ISIC card.
So it's possible to clone both types of cards HID Prox Card II and Mifare Classic. But scanning the Mifare card takes such longer (it can be even e few minutes) because of cracking the cryptography meanwhile scanning the Prox Card takes no longer than a second. However, NXP semiconductors (Manufacturer of the Mifare Classic) supplies information that this card is not recommended for "security relevant applications" [48].
The HID Global (manufacturer of the Prox Card II) supplies no information about the card's security on its official page [49], but there is written that it's "access control card". In the card datasheet [8] is written under title security, that the card has available over 137 billion unique codes. This doesn't rally mean anything about the security. The hacker can easily scan the code of the card and make a clone.
However, for some applications it's possible to use RFID cards that we know that it's possible to clone it. For the Mifare Classic is dangerous when some secret data are stored in the card's memory such as storing credit on the card, which is used as electronic wallet. In this case a hacker would be able to change the value of the credit. To prevent cloning and possibility of reading and manipulating with the data on the card by hacker, there is needed card with no broken cryptography, such as the Mifare Desfire EV1.

Both used technologies use cloneable cards whereas clonning the Mifare Classic card takes longer. To clone both types of cards we need a special card. For clonning Mifare Classic is needed a card with a changeable UID. For clonning the Prox Card II is needed a writable type of this card and a special reader which can read and write to Prox Card II while Mifare Classic can be cracked with a common reader which supports the ISO/IEC 14443 Type A, but there are needed specified tools like the mfoc installed on the hacking unit connected to the reader.

In chapters of the system design are listed prices of actually used RFID technologies. The HID technology is more expensive although it's less secure. The Mifare Classic offers more features, such as over 700 bytes of usable memory protected by encryption while the Prox Card II stores only one pre-programmed 26-bit code and no security, so everybody has access to read this code. Communicating ranges of both readers are nearly the same.

Bibliography

[1] *The Electromagnetic Door Lock*. A Star Maths and Physics. [Online]. Available: https://astarmathsandphysics.com/gcse-physics-notes/849-the-electromagnetic-door-lock.html [Accessed: 05-January-2017].

[2] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*. 3rd Edition, John Wiley & Sons, 2010

[3] Libelium Comunicaciones Distribuidas S.L. *RFID 125KHz Networking Guide*. Document version: v0.2 - 06/2012. [Online]. Available: https://www.libelium.com/v11-files/documentation/waspmote/rfid_125-networking_guide.pdf [Accessed: 22-December-2016].

[4] *RFID Tag Maximum Read Distance*. Sky RFID Inc. [Online]. Available: http://skyrfid.com/RFID_Tag_Read_Ranges.php [Accessed: 08-January-2017].

[5] NXP Semiconductors. *MF1S50yyX/V1 - MIFARE Classic EV1 1K - Mainstream contactless smart card IC*. Product data sheet. 2014. [Online]. Available: http://cache.nxp.com/documents/data_sheet/MF1S50YYX_V1.pdf [Accessed: 05-January-2017].

[6] RANKL W. *Smart Card Applications: Design Models for using and programming smart cards*. John Wiley & Sons, 2007. [Online]. Available: http://www.e-reading.club/bookreader.php/142137/Rankl_-_Smart_Card_applications_design_models_for_using_and_programming_Smart_Cards.pdf [Accessed: 16-January-2017].

[7] HID Corporation. *Understanding Card Data Formats*. 2006. [Online]. Available: https://www.hidglobal.com/sites/

default/files/hid-understanding_card_data_formats-wp-en.pdf [Accessed: 21-January-2017].

[8] HID Global. *ProxCard II Card*. [Online]. Available: https://www.hidglobal.com/sites/default/files/resource_files/prox-proxcard-ii-card-ds-en.pdf [Accessed: 22-January-2017].

[9] *How to Program HID Proximity Cards*. ID Card Group. 2017. [Online]. Available: http://www.idcardgroup.com/blog/post/how-to-program-hid-proximity-cards.aspx [Accessed: 25-January-2017].

[10] *MIFARE:The leading brand of contactless IC products*. Mifare. 2017. [Online]. Available: https://www.mifare.net/en/ [Accessed: 28-January-2017].

[11] *MIFARE*. Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/MIFARE [Accessed: 05-February-2017].

[12] *Karsten Nohl:Former Graduate Student, Computer Science Department, University of Virginia*. virginia.edu. [Online]. Available: https://www.cs.virginia.edu/~kn5f/index.html [Accessed: 02-February-2017].

[13] NOHL, K and EVANS, D from University of Virginia, SARBUG and PLOTZ, H from CCC. *Reverse-Engineering a Cryptographic RFID Tag*. virginia.edu. [Online]. Available: http://www.cs.virginia.edu/~evans/pubs/usenix08/mifare.html [Accessed: 02-February-2017].

[14] GARCIA, F. D., ROSSUM, P., VERDULT, R., SCHREUR, R. *Wirelessly pickpocketing a MIFARE Classic card*. [Online]. Available: https://blog.mmn-o.se/wp-content/uploads/2011/01/Pickpocketing.Mifare.pdf [Accessed: 05-February-2017].

[15] *MIFARE® DESFire*. NXP. [Online]. Available: http://www.nxp.com/products/identification-and-security/mifare-ics/mifare-desfire:MC_53450 [Accessed: 06-February-2017].

[16] NXP Semiconductors. *MF3ICDx21_41_81: MIFARE DESFire EV1 contactless multi-application IC*. 2015. [Online]. Available: http://www.nxp.com/documents/short_data_sheet/MF3ICDX21_41_81_SDS.pdf [Accessed: 09-February-2017].

[17] NXP Semiconductors. *MF3D(H)x2:MIFARE DESFire EV2 contactless multi-application IC*. 2016. [Online]. Available: http://www.nxp.com/documents/short_data_sheet/MF3DX2_MF3DHX2_SDS.pdf [Accessed: 10-February-2017].

[18] NOHL K. *Cryptanalysis of Crypto-1*. University of Virginia. [Online]. Available: https://www.cs.virginia.edu/~kn5f/Mifare.Cryptanalysis.htm [Accessed: 25-February-2017].

[19] Northen Computers Inc. *Wiegand Interface Definition*. 1996. [Online]. Available: https://www.honeywellaccess.com/documents/Td2058.pdf [Accessed: 30-February-2017].

[20] *SPI Serial Port (in AVR Microcontrollers)*. 2015. [Online]. Available: http://ce.sharif.edu/courses/93-94/2/ce513-1/resources/root/Slides/Micro-L7-SPI.pdf [Accessed: 31-February-2017].

[21] *Communication*. washington.edu. 2011. [Online]. Available: https://courses.cs.washington.edu/courses/cse466/11au/calendar/07-comms-posted2.pdf [Accessed: 02-March-2017].

[22] NXP Semiconductors. *MFRC522: Standard performance MIFARE and NTAG frontend*. 2016. [Online]. Available: https://www.nxp.com/documents/data_sheet/MFRC522.pdf [Accessed: 27-January-2017].

[23] *I2C Info – I2C Bus, Interface and Protocol*. I2C. 2017. [Online]. Available: http://i2c.info/ [Accessed: 04-March-2017].

[24] *I2C Bus*. DLNWARE. DLN WARE. 2016. [Online]. Available: http://dlnware.com/i2c [Accessed: 05-March-2017].

[25] TEXAS INSTRUMENTS. *KeyStone Architecture Universal Asynchronous Receiver/Transmitter (UART)*. 2010. [Online]. Available: http://www.ti.com/lit/ug/sprugp1/sprugp1.pdf [Accessed: 08-March-2017].

[26] *Serial Communication*. Sparkfun. [Online]. Available: https://learn.sparkfun.com/tutorials/serial-communication [Accessed: 08-March-2017].

[27] HID corporation. *ProxPoint Plus datasheet*. U.S.A. 2001. [Online]. Available: http://www.securitex.com.sg/securitex_proxpoint_plus.pdf [Accessed: 12-March-2017].

[28] *Paxton:Connecting a HID Proxpoint Plus reader to Net2*. 2015. [Online]. Available: https://www.paxton.co.uk/docs/oem%20readers/19.pdf [Accessed: 16-March-2017].

[29] Advanced Card Systems. *ACR122U USB NFC Reader: Application Programming interface*. version 2.03. [Online]. Available: https://www.acs.com.hk/download-manual/419/API-ACR122U-2.03.pdf [Accessed: 19-March-2017].

[30] *ACR122U USB NFC Reader*. Advanced Card Systems. [Online]. Available: http://www.acs.com.hk/en/products/3/acr122u-usb-nfc-reader/ [Accessed: 19-March-2017].

[31] *ARDUINO & GENUINO PRODUCTS: Arduino/Genuino UNO*. ARDUINO. 2017. [Online]. Available: https://www.arduino.cc/en/main/arduinoBoardUno [Accessed: 15-March-2017].

[32] *Microcontroller Development Boards*. electronics forum. 2014. [Online]. Available: http://electronicsforu.com/buyers-guides/hardware-buyers-guide/microcontroller-development-boards [Accessed: 16-March-2017].

[33] *Arduino*. Wikipedia. 2017. [Online]. Available: https://en.wikipedia.org/wiki/Arduino [Accessed: 28-March-2017].

[34] *Arduino Software (IDE)*. ARDUINO. 2017. [Online]. Available: https://www.arduino.cc/en/guide/environment [Accessed: 24-March-2017].

[35] *Wiring (development platform)*. Wikipedia. 2017. [Online]. Available: https://en.wikipedia.org/wiki/Wiring_(development_platform) [Accessed: 25-March-2017].

[36] *SOFTWARE*, ARDUINO. 2017. [Online]. Available: https://www.arduino.cc/en/Main/Software [Accessed: 18-March-2017].

[37] *ARDUINO: Memory*. ARDUINO. 2017. [Online]. Available: https://www.arduino.cc/en/tutorial/memory [Accessed: 29-March-2017].

[38] *Lacrosse TX2/TX3 Sensors and Digispark*. Guillier. 2014. [Online]. Available: http://www.guillier.org/blog/2014/09/lacrosse-tx2tx3-sensors-and-digispark/ [Accessed: 10-April-2017].

[39] *First Impressions of the Digispark*. sp.io. 2014. [Online]. Available: https://5p.io/first-impressions-of-the-digispark/ [Accessed: 11-April-2017].

[40] Atmel. *Atmel: ATtiny25, ATtiny45, ATtiny85 Datasheet*. 2013. [Online]. Available: http://www.atmel.com/images/atmel-2586-avr-8-bit-microcontroller-attiny25-attiny45-attiny85_datasheet.pdf [Accessed: 12-April-2017].

[41] *rfid: Arduino RFID Library for MFRC522*. Github. 2014. [Online]. Available: https://github.com/miguelbalboa/rfid [Accessed: 14-April-2017].

[42] *PinChangeInterrupt: A simple & compact PinChangeInterrupt library for Arduino*. Github. 2016. [Online]. Available: https://github.com/NicoHood/PinChangeInterrupt [Accessed: 18-April-2017].

[43] *pcsc-tools: Some tools to be used with smart cards and PC/SC*. Github. 2016. [Online]. Available: https://github.com/LudovicRousseau/pcsc-tools [Accessed: 01-April-2017].

[44] *Libnfc:API*. NFC Tools. 2017. [Online]. Available: http://nfc-tools.org/index.php?title=Libnfc [Accessed: 02-April-2017].

[45] *Mfoc: Mifare Classic Offline Cracker*. Github. 2017. [Online]. Available: https://github.com/nfc-tools/mfoc [Accessed: 02-April-2017].

[46] *125KHz RFID/ID Card Reader Writer Copier Duplicator*. ebay. 2017. [Online]. Available: http://www.ebay.com/itm/Handheld-125KHz-RFID-HID-Prox-Card-II-Copier-\Writer-5pcs-Writable-T5577-Cards-/162436298328?hash=item25d1f53658:g:N5MAAOSwsW9YwsLy [Accessed: 09-April-2017].

[47] *1510: Electromagnetic Door Lock*. Security Door Controls. 2017. [Online]. Available: http://www.sdcsecurity.com/1510-Series-Electromagnetic-Door-Lock.htm [Accessed: 11-April-2017].

[48] *MIFARE® Classic*. NXP. 2017. [Online]. Available: http://www.nxp.com/products/identification-and-security/mifare-ics/mifare-classic:MC_41863 [Accessed: 11-April-2017].

[49] *HID 1326 ProxCard II Clamshell Card*. HID Global. 2017. [Online]. Available: https://www.hidglobal.com/products/cards-and-credentials/hid-proximity/1326 [Accessed: 12-April-2017].

Appendices

Appendix A

Photo documentation of the created access control systems

This appendix includes photos of the created access control systems as described in chapter 6 and chapter 7.

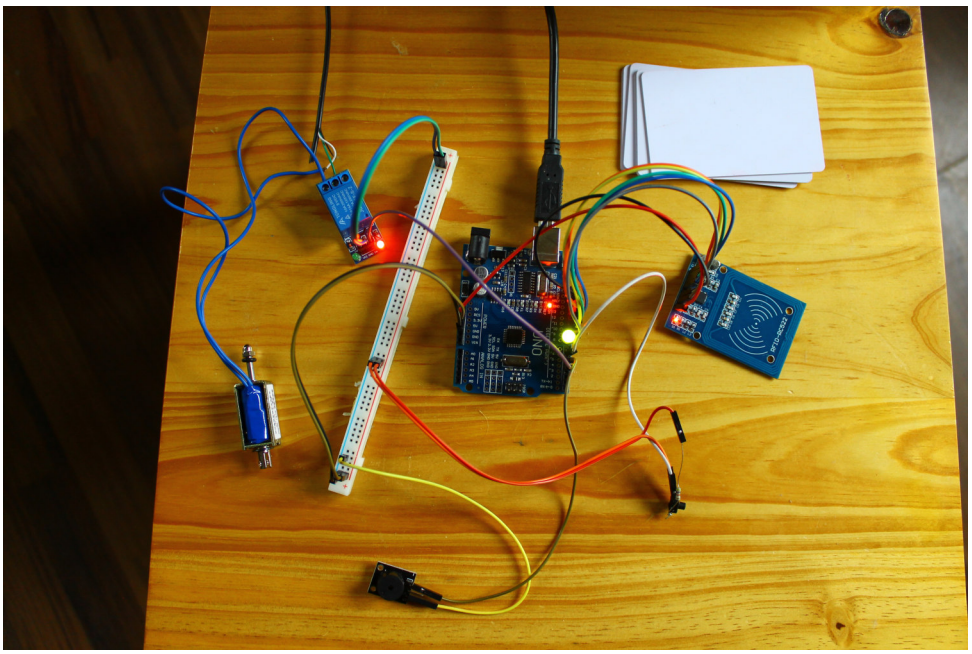

Figure A.1: Photo of the final working device - Access control system

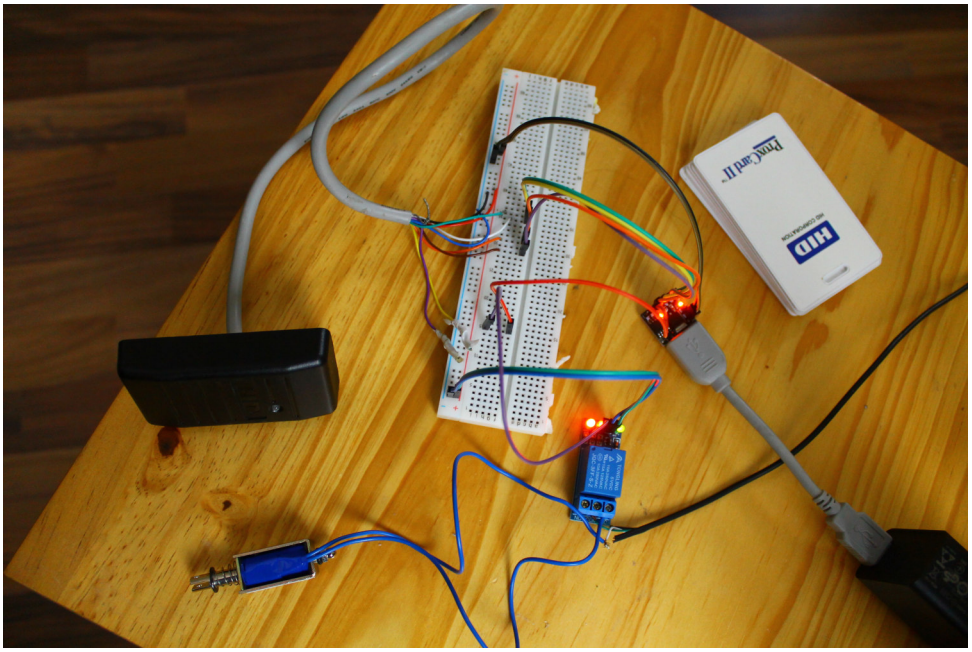

Figure A.2: Photo of the final working device - Access control system, smaller and reduced solution

Appendix B

Video documentation of the created access control systems

This thesis includes CD with videos of demonstration of using both devices designed in this work. For each device is made one short video. These systems are in factory format, so they have clean memory and no authorized cards, so adding cards follows. In both videos, cards are divided into three clusters: admin cards, user cards and unauthorized cards. Then there is shown some removing and adding user and admin cards to the system. For the system with Mifare cards is also shown erasing whole memory and adding cards to the system again.
Further there is shown that it works properly after the system is turned off and on. The solenoid is powered by an external power supply and it is switched by the relay. When the solenoid is powered on, it reveals the door, otherwise it blocks the door. In the video it reveals the door when the system is powered off, because the relay is not powered, while the external power supply of the solenoid remains on. If there is a power failure, these systems would not stop blocking the door.
The system with Mifare cars once do not recognize an authorized card, however when the card is presented again, it works. This error in communication is discussed in chapter Conclusions.