



Politechnika  
Śląska

**POLITECHNIKA ŚLĄSKA**  
**WYDZIAŁ AUTOMATYKI, ELEKTRONIKI I INFORMATYKI**

**Projekt inżynierski**

Zamek szyfrowy z czytnikiem linii papilarnych

Autor: Karol Kabulak

Kierujący pracą: Dr inż. Tomasz Rudnicki

Gliwice, styczeń 2021



## Spis treści

1	Wstęp.....	1
1.1	Wprowadzenie .....	1
1.2	Cel pracy.....	1
1.3	Analiza rozwiązań komercyjnych.....	2
2	Przykładowa realizacja.....	2
2.1	Założenia realizacyjne.....	2
2.2	Schemat blokowy .....	3
3	Część projektowa.....	4
3.1	System mikroprocesorowy .....	4
3.1.1	Wybór mikrokontrolera .....	4
3.1.2	Najważniejsze własności wybranego mikrokontrolera.....	5
3.1.3	Opis działania interfejsu USART .....	5
3.1.4	Opis działania interfejsu TWI (I <sup>2</sup> C).....	6
3.2	Czytnik linii papilarnych.....	7
3.2.1	Wybór czytnika linii papilarnych.....	7
3.2.2	Opis działania modułu czytnika linii papilarnych .....	8
3.3	Wyświetlacz .....	9
3.3.1	Wybór wyświetlacza LCD .....	9
3.3.2	Opis działania układu PCF8574A .....	10
3.4	Przełącznik i elektrozamek.....	10
4	Schematy bloków .....	11
4.1	System mikroprocesorowy .....	11
4.2	Wyświetlacz LCD i konwerter I <sup>2</sup> C .....	12
4.3	Przyciski.....	13
4.4	Moduł przełącznika .....	14
4.5	Połączenia między poszczególnymi blokami.....	14
5	Część konstrukcyjna .....	15
5.1	Połączenie modułów .....	15

5.2	Projekt obudowy 3D.....	16
6	Opis oprogramowania.....	17
6.1	Program mikrokontrolera .....	17
6.2	Obsługa wyświetlacza LCD oraz inicjalizacja interfejsu TWI .....	17
6.2.1	Inicjalizacja interfejsu TWI.....	17
6.2.2	Wyświetlanie informacji na wyświetlaczu LCD .....	18
6.3	Obsługa czytnika linii papilarnych wraz z inicjalizacją interfejsu UART ...	20
6.3.1	Inicjalizacja interfejsu UART .....	20
6.3.2	Obsługa czytnika linii papilarnych .....	21
6.4	Główny program urządzenia .....	23
7	Badania układu .....	27
7.1	Badanie poboru prądu urządzenia .....	27
8	Obsługa .....	27
9	Podsumowanie .....	34
10	Bibliografia .....	35
11	Spis rysunków .....	36
12	Spis tabel .....	37
13	Spis wzorów .....	37

## Spis akronimów

A/C	– (ang.) Analog to digital (analogowo cyfrowy)
CTC	– (ang.) Clear Timer on Compare (wyczyść licznik przy porównaniu)
EEPROM	– (ang.) Electrical Erasable Programmable Read-Only Memory (nieulotna pamięć EEPROM)
FLASH	– (ang.) Flash memory (pamięć flash)
I/O	– (ang.) Input/Output (wejście/wyjście)
LCD	– (ang.) Liquid Cristal Display (wyświetlacz ciekłokrystaliczny)
PWM	– (ang.) Pulse-Width Modulation (modulacja szerokości impulsów)
RFID	– (ang.) Radio-frequency Identification (system zdalnej identyfikacji radiowej)
SCL	– (ang.) Serial Clock Line (linia zegara)
SDA	– (ang.) Serial Data Line (linia danych)
SPI	– (ang.) Serial Peripheral Interface (szeregowy interfejs urządzeń peryferyjnych)
SRAM	– (ang.) Static Random Access Memory (statyczna pamięć o dostępie swobodnym)
TWI	– (ang.) Two Wire Interface (dosł. interfejs dwóch linii)
USART	– (ang.) Universal Synchronous Asynchronous Receiver-Transmitter (dosł. uniwersalny synchroniczno-asynchroniczny nadajnik- odbiornik)



# 1 Wstęp

## 1.1 Wprowadzenie

Od zarania dziejów człowiek poszukiwał sposobów na ochronę swoich dóbr materialnych i osobistych. W tym celu konstruowane były różnego rodzaju urządzenia, które pozwalały na zabezpieczenie przedmiotów wartościowych, które mogłyby zostać skradzione. Pierwsza myśl, która się nasuwa, to kłódka – posiada unikatowy wzór wewnątrz swojej konstrukcji, aby do otwarcia potrzebny był identyczny klucz ze wzorem pasującym do konkretnego mechanizmu. Wraz z rozwojem elektroniki możliwe stało się projektowanie oraz konstruowanie urządzeń, które zastępowały tradycyjny klucz na rzecz np. kombinacji cyfr z klawiatury numerycznej, chipów RFID lub wykorzystując ciało ludzkie i jego unikatowe cechy takie jak np. linie papilarne palca, tęczówkę oka lub też kształt twarzy. Urządzenia wykorzystujące wspomniane wcześniej unikatowe cechy ludzkie są coraz bardziej popularne, powszechne i chętnie stosowane, ze względu na wygodę i bezpieczeństwo. Użytkownik nie musi się martwić o to czy ma ze sobą klucz lub też czy pamięta kombinację cyfr, jaką należy wprowadzić w celu otwarcia zamka. Użytkownik również ma poczucie bezpieczeństwa, ze względu na unikalność – wzór linii papilarnych lub też zbiór punktów charakterystycznych dla tęczówki oka jest inny dla każdego człowieka.

## 1.2 Cel pracy

Celem pracy jest zaprojektowanie, montaż oraz uruchomienie zamka szyfrowego, który wykorzystuje gotowy czytnik linii papilarnych w oparciu o mikrokontroler. Aby urządzenie spełniało wymagania, powinno posiadać funkcję zapisu nowego obrazu palca do bazy, usuwanie istniejącego obrazu palca z bazy oraz, co najważniejsze, miało możliwość skanowania przyłożonego palca i wyszukanie w bazie istniejącego wzoru. Wybór odpowiedniej akcji jest dokonywany za pomocą trzech przycisków.

W przypadku gdy moduł zwróci istnienie zgodności skanowanego palca z obrazem zapisanym w bazie, ma zostać uruchomiony elektrozamek, który pozwoli na otwarcie drzwi, do których urządzenie jest zamontowane. W celu ułatwienia nawigacji po funkcjonalnościach urządzenia, użytkownik powiadamiany jest, za pomocą wyświetlacza, o statusie urządzenia oraz ewentualnych błędach jakie mogą się pojawić. Urządzenie powinno również korzystać z zasilania bateryjnego.

### **1.3 Analiza rozwiązań komercyjnych**

Na rynku znaleźć można wiele rozwiązań zamków szyfrowych, które wykorzystują czytniki linii papilarnych w celu otwarcia elektrozamka. Istnieje wiele rodzajów urządzeń, które mają różne zastosowania w życiu codziennym. Mogą to być np. sejfy, klamki do drzwi lub też mechanizmy do zabezpieczania szuflad.

W znacznej większości urządzenia te posiadają dodatkowe funkcjonalności, takie jak klawiatura numeryczna, otwór na klucz (fizyczne otwarcie elektrozamka) bądź też czytnik chipów RFID. Zauważyć można jednak, że brakuje w tych systemach prostego mechanizmu komunikacji z użytkownikiem (np. wyświetlacz), przez co operowanie na takich urządzeniach opiera się na znajomości instrukcji obsługi danego mechanizmu. Dodatkową wadą takich rozwiązań jest sposób zaprojektowania – urządzenia te nie posiadają osoby odpowiedzialnej, tzw. administratora, którego rolą jest możliwość zarządzania bazą dodanych odcisków palców. Dlatego też w dużej mierze oferowane urządzenia nadają się jedynie do użytku prywatnego.

## **2 Przykładowa realizacja**

### **2.1 Założenia realizacyjne**

Pierwszym etapem budowy zamka szyfrowego, który wykorzystuje czytnik linii papilarnych jest projektowanie, które powinno być poprzedzone przemyśleniami dotyczącymi możliwych realizacji i wyborem tych, które są optymalne i spełniają przyjęte założenia. Założenia dodatkowe nie są bezwzględnie wymagane, ale mogą się przyczynić do poprawy pracy zamka szyfrowego.

Założenia podstawowe:

- urządzenie posiada mikrokontroler, który jest odpowiedzialny za pracę pozostałych podzespołów;
- urządzenie posiada gotowy czytnik linii papilarnych, który posiada własną pamięć do przechowywania zapisanych odcisków palców;
- do sterowania funkcjonalnościami urządzenia służą trzy przyciski - odpowiedzialne za wybór przez użytkownika funkcji, które są możliwe do wykonania;
- urządzenie posiada elektrozamek, który zasilany jest napięciem 12 V;
- do sterowania elektrozamkiem urządzenie wykorzystuje przekaźnik, który po otrzymaniu sygnału od układu mikrokontrolera pozwala na uruchomienie elektrozamka;



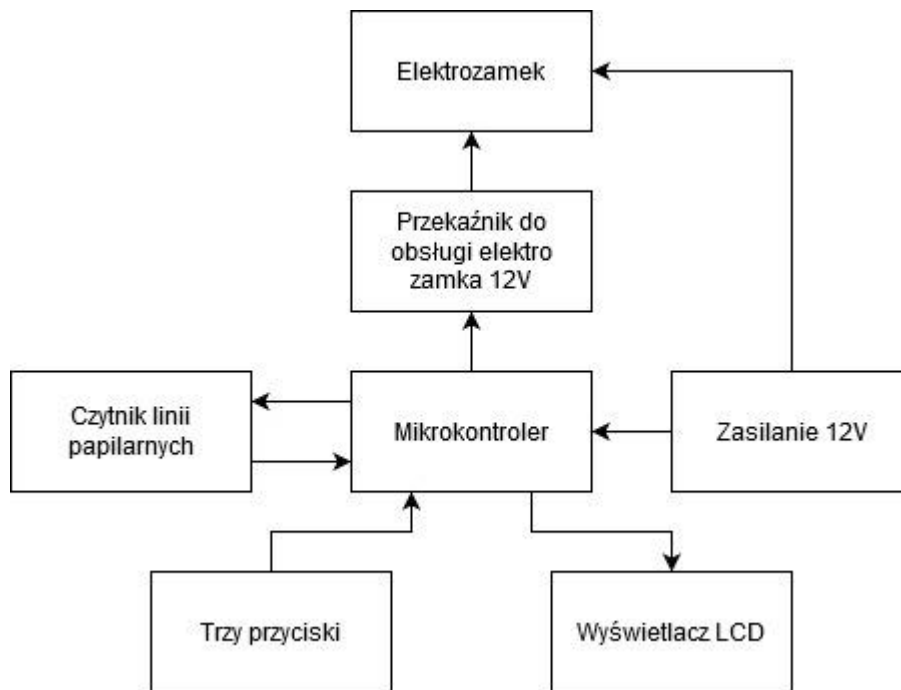
- urządzenie ma możliwość wprowadzenia nowego odcisku palca do bazy, jak i możliwość usunięcia istniejącego obrazu palca z bazy;
- urządzenie ma możliwość odczytania umieszczonego na nim odcisku palca, w celu przeszukania bazy z zapisanymi obrazami odcisków palców, zwracając układowi mikrokontrolera informacje czy dany palec istnieje w bazie;
- jeżeli obraz wprowadzonego palca istnieje w bazie czytnika linii papilarnych, urządzenie ma możliwość otwarcia elektrozaworu;
- interakcja między urządzeniem a użytkownikiem odbywa się za pomocą wyświetlacza LCD, który informuje użytkownika o stanie urządzenia (drzwi otwarte/zamknięte), dostępnych funkcjonalnościach jak i o statusach urządzenia przy wykonywaniu poszczególnych akcji przez użytkownika (brak odcisku w bazie, błędy przy generacji obrazu palca).

Założenia dodatkowe:

- urządzenie posiada jednego administratora, który jest jedynym użytkownikiem autoryzowanym do dodawania nowego odcisku palca i usuwania obrazu palca z bazy;
- przy pierwszym użyciu (baza obrazów w czytniku linii papilarnych jest pusta) urządzenie będzie wymagało wprowadzenia do bazy nowego użytkownika, który zarazem będzie pełnił funkcję administratora;
- urządzenie ma możliwość zmiany administratora, lecz do dokonania zmiany potrzebny jest wcześniej przypisany administrator.

## 2.2 Schemat blokowy

Na rysunku (Rysunek 2.1) został przedstawiony przykładowy schemat blokowy układu zamka szyfrowego wykorzystującego czytnik linii papilarnych. Schemat ten uwzględnia zarówno bloki spełniające założenia podstawowe, jak i dodatkowe. Głównym elementem układu jest mikrokontroler, który steruje wszystkimi niezbędnymi modułami, takimi jak czytnik linii papilarnych oraz przełącznik. Połączenie czytnika linii papilarnych oraz mikrokontrolera odbywa się w obu kierunkach, ponieważ mikrokontroler zarówno wysyła instrukcje, które mają zostać wykonane, jak i otrzymuje odpowiedź zwrotną od czujnika.



*Rysunek 2.1 Schemat blokowy przykładowej realizacji urządzenia.*

### 3 Część projektowa

#### 3.1 System mikroprocesorowy

##### 3.1.1 Wybór mikrokontrolera

Ze względu na dostępność, popularność i niewygórowaną cenę, zdecydowano się na użycie mikrokontrolera Atmega328p firmy Atmel. Skorzystano z rozwiązania Arduino Uno, który oferuje gotowe moduły pracujące na wyżej wymienionym mikrokontrolerze. Głównymi zaletami takiego rozwiązania jest przede wszystkim wbudowany regulator napięcia, dzięki któremu możliwe staje się podłączenie baterii lub ładowarki, których napięcie jest w przedziale od 7-12 V (złącze koncentryczne) [1]. Dodatkowo Arduino Uno posiada złącze USB, które pozwala zarówno na zasilanie układu (5 V), jak i na programowanie mikrokontrolera poprzez podłączenie układu do komputera osobistego, co eliminuje konieczność posiadania zewnętrznego programatora.

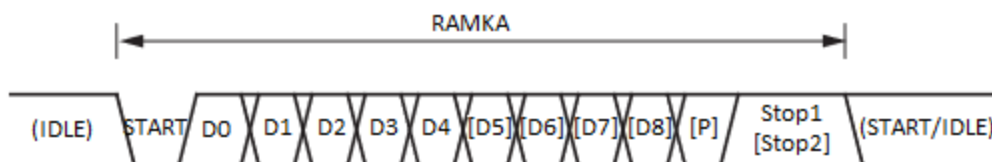
### 3.1.2 Najważniejsze własności wybranego mikrokontrolera

Do najważniejszych własności mikrokontrolera Atmega328p należą [1][2]:

- 32 KB pamięci FLASH (0.5 KB zarezerwowane jest na program rozruchowy);
- 2 KB pamięci SRAM;
- 1 KB pamięci EEPROM;
- taktowanie do 20 MHz (Arduino Uno posiada zewnętrzny rezonator kwarcowy którego wartości wynosi 16MHz);
- 14 cyfrowych wyprowadzeń I/O z których 6 posiada funkcję PWM (6 kanałów PWM);
- 6 analogowych wyprowadzeń wejściowych (6 kanałów 10 bitowego przetwornika A/C);
- sprzętowe interfejsy komunikacyjne: USART, SPI, TWI (I<sup>2</sup>C);
- dwa liczniki 8 bitowe;
- jeden licznik 16 bitowy.

### 3.1.3 Opis działania interfejsu USART

Interfejs USART jest interfejsem komunikacji pomiędzy układami scalonymi. Do zapewnienia komunikacji służą dwie linie RX (odbiór danych) i TX (transmisja danych). USART jest synchronicznym interfejsem komunikacji, co oznacza, że dane mogą być jednocześnie odbierane oraz wysyłane. Możliwa jest również praca w trybie asynchronicznym (UART). Prędkość transmisji jest z góry ustalana poprzez obliczenie wartości i ustawienie jej do odpowiedniego rejestru w mikrokontrolerze. Komunikacja odbywa się za pomocą wysyłania oraz odbierania ramek danych. Przykładowa ramka danych została przedstawiona na rysunku (Rysunek 3.1).



Rysunek 3.1 Ramka danych w interfejsie USART, tłum. własne [2].

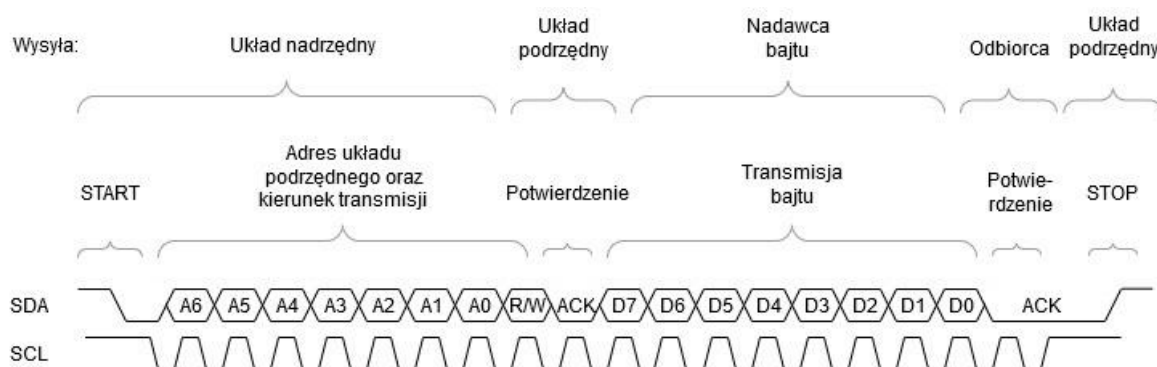
IDLE określa stan, w którym nie są przesyłane żadne dane.

Ramka w interfejsie USART zawiera:

- START - 1 bit startowy;
- D0 - D8 – do wyboru 5, 6, 7, 8 lub 9 bitów przeznaczonych na dane;
- P – do wyboru 0, 1 lub 2 bity parzystości;
- Stop1, [Stop2] – do wyboru 1 lub 2 bity końca ramki.

### 3.1.4 Opis działania interfejsu TWI (I<sup>2</sup>C)

Interfejs TWI służy do komunikacji pomiędzy układami scalonymi. Wykorzystuje dwie linie, które zapewniają komunikację pomiędzy wszystkimi urządzeniami. Linia SCL jest linią zegara, która uaktywniana jest przez układ nadrzędny (master). Linia SDA jest linią danych, którą uaktywnia również układ nadrzędny w celu wysyłania określonych danych do urządzenia podrzędnego. Transmisje realizowane są asynchronicznie, co oznacza, że w danej chwili czasu można przesyłać dane tylko w jednym kierunku. Dodatkowo, niezależnie od kierunku, transmisja danych jest taktowana sygnałem zegara wytwarzanego przez układ nadrzędny [3]. Przebieg przykładowej transmisji został przedstawiony na rysunku poniżej (Rysunek 3.2).



Rysunek 3.2 Przebieg transmisji w interfejsie TWI, tłum. własne [3].

Komunikacja między układem nadrzędnym a podrzędnym może zostać określona w kilku krokach [3]:

1. Układ nadrzędny wysyła bit startu, co jest równoznaczne z przejściem linii SDA w stan wysoki (logiczna jedynka).
2. Układ nadrzędny wysyła siedem bitów informacji, które zawierają adres urządzenia podrzędnego z którym należy się komunikować, oraz jeden bit (R/W) który określa kierunek transmisji.

3. Po prawidłowym rozpoznaniu własnego adresu przez układ podrzędny, wysyła ono jeden bit informacji (ACK), co jest równoznaczne z ustawieniem linii SDA na stan niski (logiczne zero). W przypadku gdy adres okaże się nieprawidłowy (NACK), na linii SDA pozostaje stan wysoki.
4. Po prawidłowym rozpoznaniu adresu oraz ustawieniu przez układ podrzędny linii SDA na stan logicznego zera, układ nadrzędny wysyła 8 bitów z adresem do wewnętrznego rejestru układu podrzędnego, które określa funkcję, jaka ma zostać wykonana.
5. Po otrzymaniu prawidłowego adresu, układ podrzędny wysyła bit potwierdzenia ACK. Następnie urządzenie nadrzędne wysyła bit stopu, co jest równoznaczne z ustawieniem linii SDA na stan wysoki.

## **3.2 Czytnik linii papilarnych**

### **3.2.1 Wybór czytnika linii papilarnych**

Jednym z najważniejszych kryteriów podczas wyboru odpowiedniego czytnika linii papilarnych jest możliwość komunikacji z mikrokontrolerem. Spora część gotowych modułów posiada możliwość komunikacji z urządzeniem nadrzędnym (mikrokontroler) za pomocą interfejsu UART. Kolejnym istotnym parametrem, którym kierowano się przy wyborze czytnika linii papilarnych, był jego fizyczny rozmiar oraz posiadana obudowa na skaner palca. Dodatkowo istotnym elementem jest pamięć do przechowywania odcisków palców, prędkość odczytu obrazu palca oraz prędkość przeszukiwania bazy danych w celu znalezienia pasującego obrazu palca.

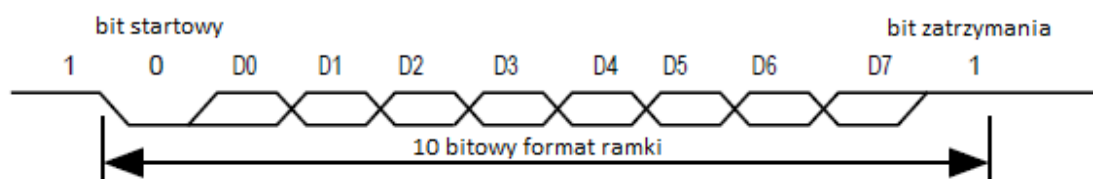
Po analizie rynkowej dostępnych modułów zdecydowano się na moduł Z70. Spełnia on wszystkie wymagania konieczne, a zarazem oferuje imponujące parametry dodatkowe.

Główne cechy modułu czytnika linii papilarnych Z70 [4]:

- napięcie zasilania – od 3.8 V do 7 V;
- komunikacja za pomocą interfejsu UART (napięcie na linii TX/RX – 5 V);
- czas odczytu obrazu – poniżej 1 s;
- możliwość zapisu do 1000 odcisków palców w bazie;
- średni czas przeszukiwania bazy: 2 ms;
- 5 poziomów czułości modułu (standardowo moduł pracuje na poziomie 3);
- dodatkowe zabezpieczenie modułu w postaci hasła (2 bajty - standardowe hasło to 0x0000).

### 3.2.2 Opis działania modułu czytnika linii papilarnych

Obsługa czytnika linii papilarnych polega na wysyłaniu oraz odbieraniu za pomocą interfejsu UART pakietów danych, które określone zostały w nacie katalogowej producenta [4]. Standardowa prędkość transmisji jest równa 57600 b/s (może ona zostać zmieniona w zakresie od 9600 b/s do 115200 b/s). Ramka danych jest w formacie 10 bitowym – jeden startowy bit, 8 bitów danych (D0 – D7) oraz jeden bit stopu, co zostało przedstawione na rysunku (Rysunek 3.3).



*Rysunek 3.3 Wysyłanie jeden ramki danych do czytnika linii papilarnych za pomocą interfejsu UART, tłum. własne [4].*

Aby sterować pracą i funkcjonalnościami czytnika linii papilarnych należy wysyłać pakiety danych. Format pakietu, który należy wysłać do modułu czytnika linii papilarnych, został przedstawiony w tabeli (Tabela 1).

*Tabela 1 Format ramki danych dla czytnika linii papilarnych, tłum. własne [4].*

Nazwa	Długość	Opis	
Nagłówek	2 bajty	Stała wartość 0xEF01. Starszy bit wysyłany jest jako pierwszy.	
Adres	4 bajty	Domyślna wartość adresu jest równa 0xFFFFFFFF, która może zostać zmodyfikowana używając komendy. Starszy bit wysyłany jest jako pierwszy - w przypadku, gdy adres jest błędny moduł odrzuci nie przyjmie pakietu.	
Identyfikator ramki	1 bajt	0x01	Ramka poleceń.
		0x02	Ramka danych. Musi zostać poprzedzony ramką poleceń lub ramką potwierdzenia.
		0x07	Ramka potwierdzenia (ACK).
		0x08	Koniec ramki danych.

Długość ramki	2 bajty	Długość ramki – odnosi się do zawartości (ramki poleceń lub ramki danych) oraz długości sumy kontrolnej (2 bajty). Maksymalna długość to 256 bajtów. Starszy bajt przesyłany jest jako pierwszy.
Zawartość danych	-	Dane rozumiane są jako polecenia, dane, parametry poleceń, potwierdzenia, wyniki itp.
Suma kontrolna	2 bajty	Suma arytmetyczna identyfikatora ramki, długości pakietu oraz całej zawartości danych. Przepelnione bajty zostają pominięte.

Czytnik linii papilarnych posiada w swojej pamięci jeden bufor na obraz palca (Image buffer) oraz dwa bufony znakowe (Char buffer 1, Char buffer 2) o wielkości 512 bajtów każdy. Użytkownik może odczytywać i zapisywać do wszystkich trzech buforów za pomocą odpowiednich poleceń.

### 3.3 Wyświetlacz

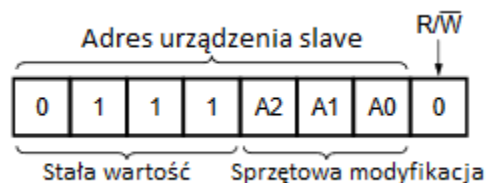
#### 3.3.1 Wybór wyświetlacza LCD

Głównymi cechami przy wyborze odpowiedniego wyświetlacza była przede wszystkim dostępna ilość informacji, jaka może zostać wyświetlona, cena oraz łatwość programowania. Rozważano dwa rodzaje wyświetlaczy - LCD oraz OLED. Po przeprowadzeniu analizy rynkowej dostępnych modułów, wybór padł na wyświetlacz LCD. Głównym aspektem, który zaważył o wyborze danego wyświetlacza była różnica w cenie. Ponadto wyświetlacz LCD posiada wystarczającą ilość miejsca, potrzebną do wyświetlenia koniecznych informacji użytkownikowi, a zarazem wyświetlanie informacji jest mało skomplikowane do zaprogramowania.

Zdecydowano się na użycie wyświetlacza LCD ze sterownikiem zgodnym z HD44780, który oferuje wyświetlanie 32 znaków w 2 liniach (16 znaków na jedną linię). Skorzystano również z konwertera I<sup>2</sup>C, którego działanie opiera się na układzie PCF8574A. Pozwala to w znaczny sposób ograniczyć ilość wyprowadzeń, koniecznych do podłączenia wyświetlacza z układem mikrokontrolera, redukując końcową ilość do czterech połączeń (napięcie zasilania, masa, linia SDA oraz linia SCL).

### 3.3.2 Opis działania układu PCF8574A

Układ PCF8574A na którym zbudowany jest konwerter I<sup>2</sup>C korzysta z 7-bitowego adresowania, które zostało przestawione na rysunku (Rysunek 3.4).



Rysunek 3.4 Adresowanie dla układu PCF8574A, tłum. własne [5].

Bity A0 - A2 w domyślnej konfiguracji mają logiczną jedynkę dlatego, że ich wyprowadzenia w układzie PCF8574A doprowadzone są do zasilania poprzez rezystor podciągający. Oznacza to, że domyślny adres urządzenia podrzędnego jest równy „0111111” (binarnie). Najmniej znaczący bit (R/W) określa, czy urządzenie nadrzędne chce dokonać odczytu (jedyńka logiczna) czy zapisu (zero logiczne).

Układ PCF8574A posiada wewnątrz 8 bitowy rejestr (P0 – P7), który pozwala na pracę każdego z wyprowadzeń jako wejście lub wyjście. Ustawienie wyprowadzeń w wewnętrznym rejestrze na pracę w trybie wejścia lub wyjścia odbywa się, gdy urządzenie nadrzędne, chcące nawiązać komunikację, wyśle zero logiczne na bicie R/W. Ustawienie jedynki logicznej oznacza, że dane wyprowadzenie ma działać w trybie wejścia (odczytu), natomiast ustawienie na odpowiednim wyprowadzeniu logicznego zera oznacza, że ma on działać w trybie wyjścia (zapisu). W przypadku gdy układ nadrzędny chce wysłać dane (np. do wyświetlenia informacji na wyświetlaczu LCD), podczas nawiązywania połączenia (rozsyłanie adresu urządzenia podrzędnego wraz z bitem R/W), należy ustawić bit R/W jako jedynkę logiczną. Pozwoli to na wysyłanie danych na wyprowadzeniach, które pracują w trybie zapisu.

### 3.4 Przekaznik i elektrozamek

Przy wyborze elektrozamka kierowano się jedynie napięciem jakie należy dostarczyć, aby praca elektrozamka była możliwa. Zdecydowano się na prosty mechanizm, który pracuje na napięciu 12V, mający rozmiary odpowiednie do zastosowania w praktycznych rozwiązaniach.

Wybór przekaznika opierał się na analizie rynkowej gotowych modułów, które posiadają w swoim układzie zabezpieczenia konieczne do prawidłowej i bezpiecznej pracy



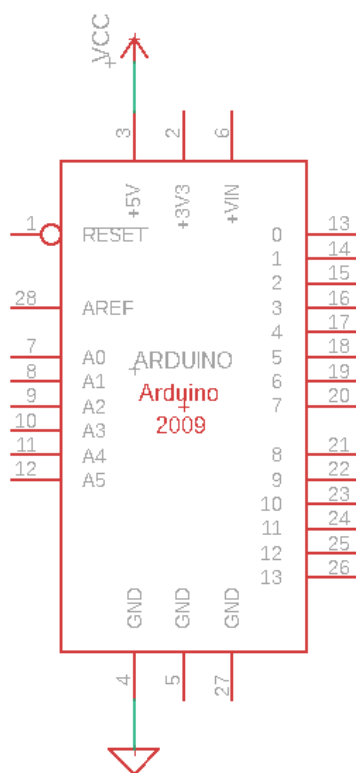
całego układu. Do najważniejszych elementów, które były konieczne, aby spełnić powyższe wymagania, było posiadanie przez moduł przekaźnika między innymi:

- optoizolacji – która odseparowuje obwód mikrokontrolera od obwodu przekaźnika;
- diody prostowniczej – zapobiega przed przedostaniem się wysokiego napięcia do układu, które jest wytwarzane po odłączeniu napięcia od przekaźnika.

## 4 Schematy bloków

### 4.1 System mikroprocesorowy

Na rysunku (Rysunek 4.1) przedstawiony został symbol użytego modułu Arduino Uno, wykorzystującego mikrokontroler Atmega328p. Ze względu na dużą ilość dodatkowych elementów, które wchodzi w skład całego układu całkowity schemat blokowy dostępny jest na oficjalnej stronie producenta [6].



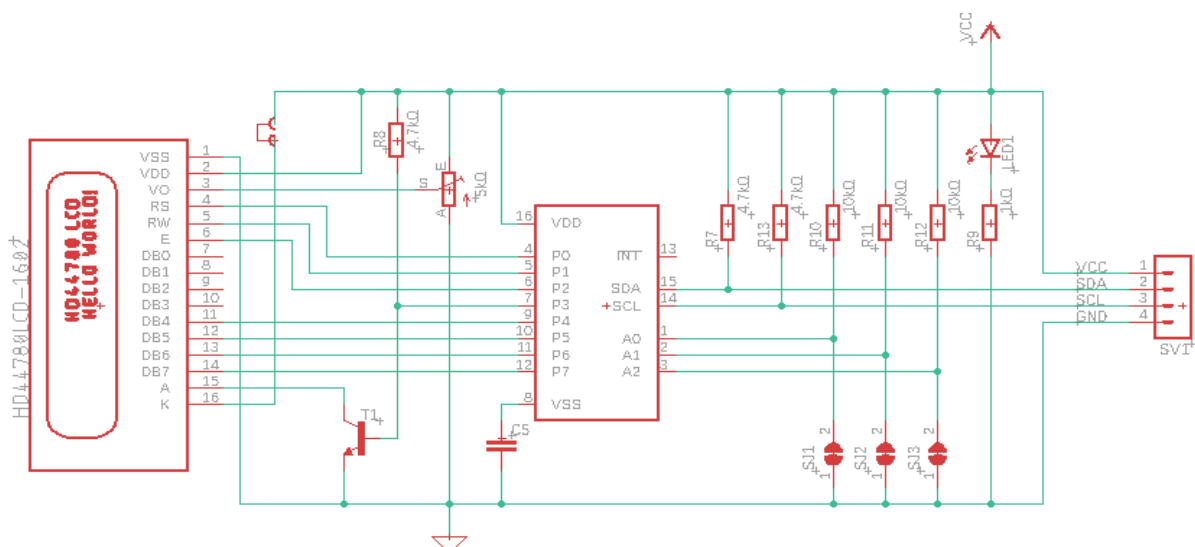
*Rysunek 4.1 Symbol modułu Arduino Uno wykorzystujący mikrokontroler Atmega328p.*

Układ posiada szereg wyprowadzeń, które przy odpowiednim zaprogramowaniu pozwalają na pracę pozostałych części projektu. Najważniejszymi wyprowadzeniami, które należy nadmienić i które znajdują się na rysunku (Rysunek 4.1) są:

- wyprowadzenia A4 i A5 – przy odpowiednim zaprogramowaniu pełnią rolę magistrali w interfejsie TWI, gdzie jedno wyprowadzenie służy za przesyłanie sygnału zegarowego (SCL - A5), natomiast drugie wyprowadzenie odpowiada za przesyłanie danych (SDA - A4);
- wyprowadzenia 0 i 13 – cyfrowe wyprowadzenia wejścia/wyjścia, które posiadają możliwość sterowania wejściami 0V i 5V (tryb wyjścia) oraz mają możliwość odczytu napięcia około 0V i napięcia między 2.5V – 5V (tryb wejścia). Przy odpowiednim zaprogramowaniu wyprowadzenia 0 oraz 1 pełnią rolę linii TX (nadawanie danych) oraz RX (odbieranie danych) w interfejsie szeregowym UART;
- wyprowadzenia 5V, 3.3V, GND – układ posiada wyprowadzenia pozwalające na zasilanie dodatkowych modułów i urządzeń bez konieczności prowadzenia osobnych połączeń z głównego punktu zasilania.

## 4.2 Wyświetlacz LCD i konwerter I<sup>2</sup>C

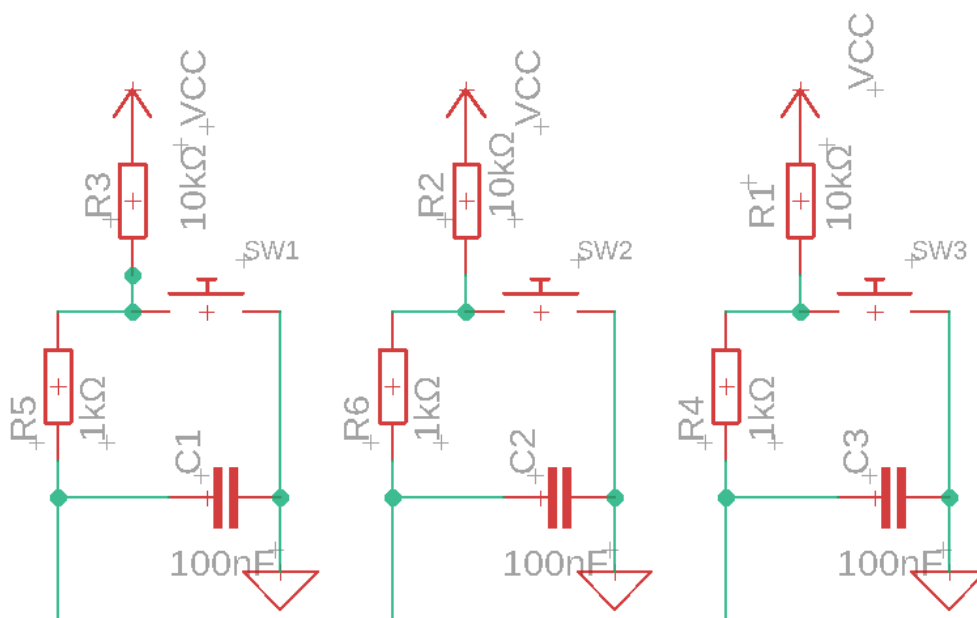
Na rysunku (Rysunek 4.2) został przedstawiony schemat blokowy podłączenia wyświetlacza LCD do układu konwertera I<sup>2</sup>C wykorzystującego układ PCF8574A. Układ ten działa w trybie 4 bitowym, dlatego też wyprowadzenia DB0 – DB3 nie zostały podłączone i nie pełnią żadnej roli w procesie wyświetlania informacji na wyświetlaczu. Ponadto układ posiada trzy wyprowadzenia A0 – A2 które odpowiadają za zmianę adresu urządzenia (w podstawowej konfiguracji zworki A0 – A2 nie są połączone do masy). Konwerter posiada również potencjometr do regulacji kontrastu wyświetlacza LCD oraz diodę LED informującą o tym, czy konwerter jest zasilony.



Rysunek 4.2 Schemat blokowy wyświetlacza LCD i konwertera I<sup>2</sup>C.

### 4.3 Przyciski

Rysunek (Rysunek 4.3) przedstawia schemat blokowy trzech użytych przycisków wraz z dodatkowymi elementami, które pozwalają na eliminację drgań styków.



Rysunek 4.3 Schemat blokowy trzech przycisków.

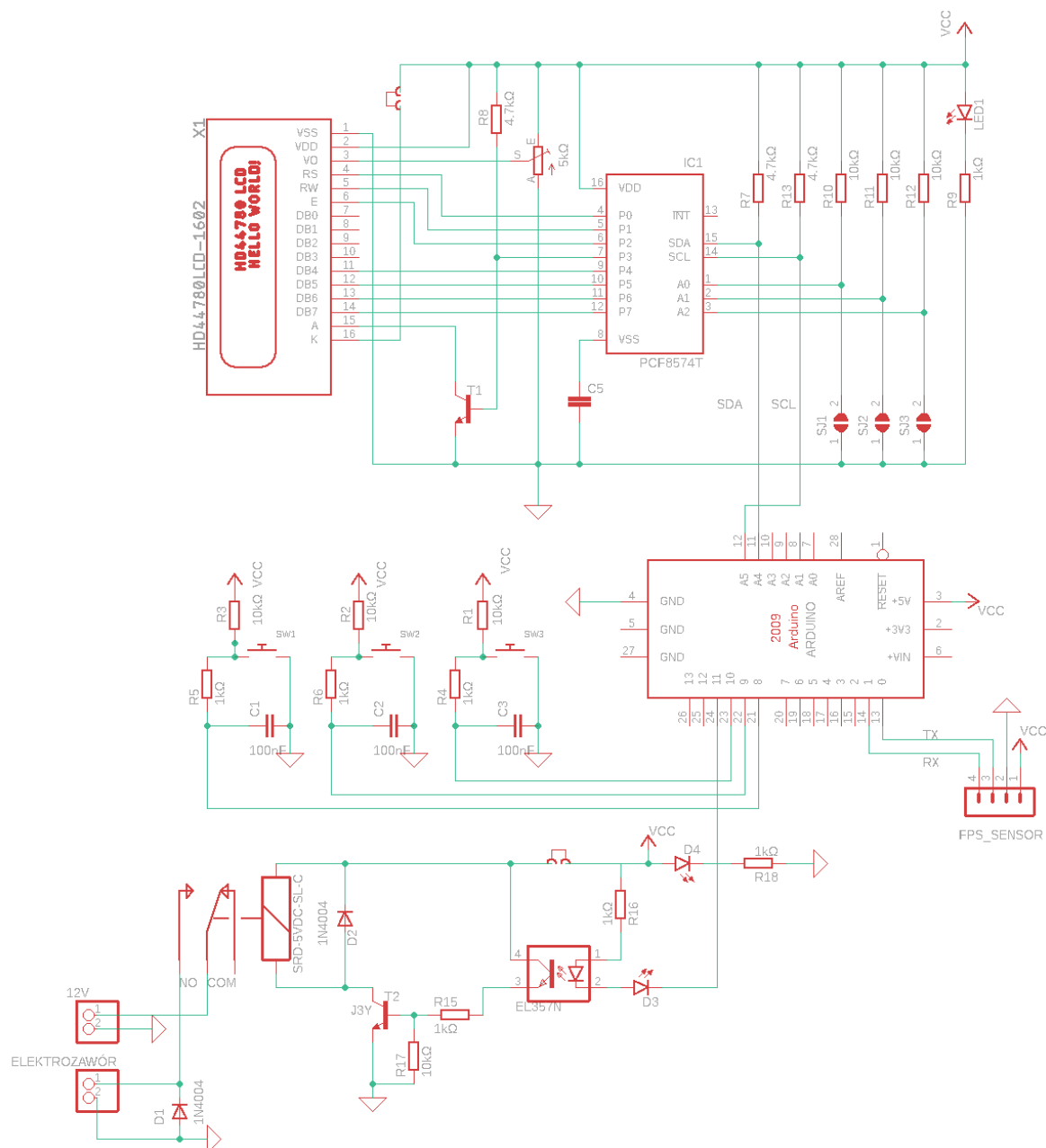
Drganie styków określa się bardzo szybkie zmiany stanu napięcia logicznego wykrywane przez mikrokontroler przy naciskaniu lub rozłączaniu przycisku. Spowodowane jest to nierównomiernym zwieraniem styków, co może przyczynić się do nieprawidłowej pracy układu. Mikrokontroler może odczytać nawet do kilkunastu zmian stanów napięcia podczas jednego naciśnięcia lub rozłączenia przycisku.

Aby wyeliminować drgania styków i zapewnić poprawną pracę układu, zastosowano filtr dolnoprzepustowy RC. Rezystor i kondensator został dobrany w taki sposób, aby częstotliwość graniczna filtra była jak najmniejsza (Wzór 1). Dobrane wartości filtra (rezystor 1 kΩ, kondensator 100 nF) pozwalają na uzyskanie na wyjściu układu jednolitego sygnału, dzięki czemu mikrokontroler może w sposób prawidłowy odczytywać stan przycisków. Ponadto układ posiada rezystor o wartości 10 kΩ, który służy jako rezystor podciągający, co powoduje odczyt stanów przycisków przez mikrokontroler logicznym zerem.

$$f[Hz] = \frac{1}{2 * \pi * R[\Omega] * C[F]}$$

Wzór 1 Wzór na częstotliwość graniczną filtra RC.





Rysunek 4.5 Schemat blokowy układu zamka szyfrowego wykorzystującego czytnik linii papilarnych.

## 5 Część konstrukcyjna

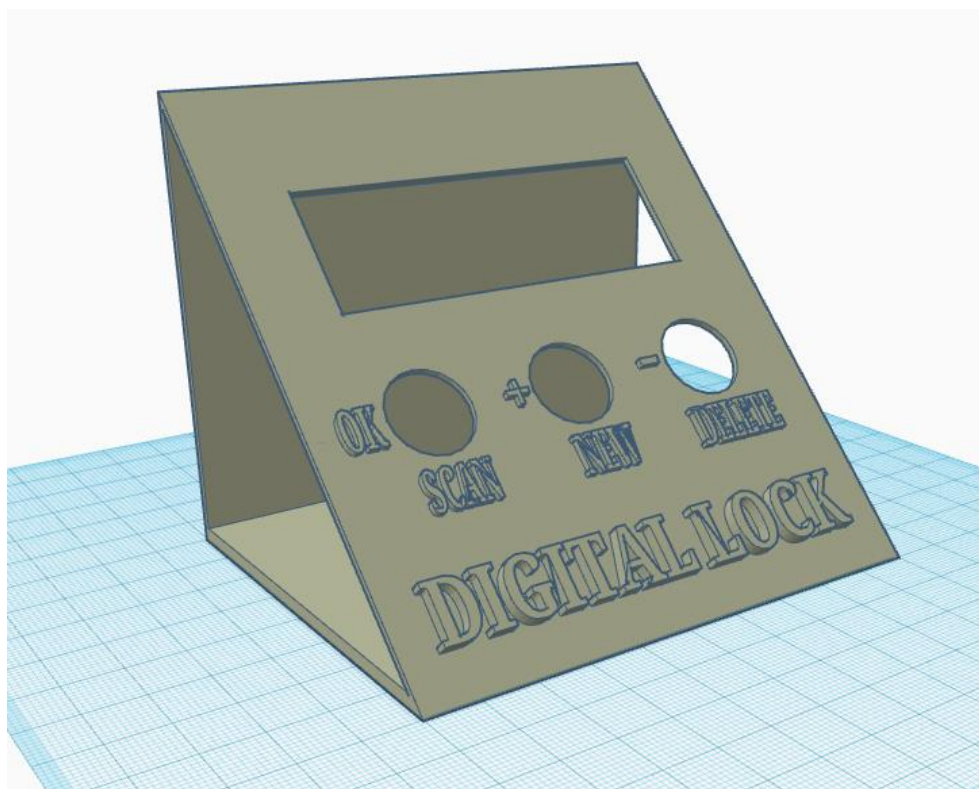
### 5.1 Połączenie modułów

Do połączenia wszystkich modułów z układem Arduino Uno pracującym na mikrokontrolerze Atmega328p, została użyta dodatkowa płytki rozszerzeń Arduino Uno Prototype Shield. Oferuje ona specjalnie wykonanie ścieżki wraz z wyprowadzeniami dla zasilania 5V oraz masy GND, do których można podłączyć wszystkie moduły potrzebujące

zasilenia. Budowa płytki rozszerzeń opiera się na budowie płytki uniwersalnej, na której mogą zostać przylutowane dodatkowe części elektroniczne. Przylutowane zostały elementy odpowiedzialne za obsługę trzech przycisków (filtr RC i rezystor podciągający) oraz dodatkowe złącza, które ułatwiają wpięcie kabli z innych modułów do mikrokontrolera. Zaproponowane rozwiązanie pozwala na szybką diagnozę i wymianę w razie uszkodzenia jednego z elementów układu, jak i na dodawanie dodatkowych funkcjonalności w przyszłości.

## 5.2 Projekt obudowy 3D

Projekt obudowy (Rysunek 5.1) pozwala na umieszczenie wyświetlacza LCD oraz trzech przełączników chwilowych. Dodatkowo nadrukowane zostały nazwy akcji możliwych do wykonania przez poszczególny przycisk. Rozwiązanie to przyczynia się do poprawy użytkowania urządzenia. Projekt obudowy został wykonany w programie Tinkercad, na licencji CC (Creative Commons) [9]. Plik z rozszerzeniem \*.STL zawierająca projekt obudowy znajduje się na płycie CD dołączonej do pracy dyplomowej.



*Rysunek 5.1 Zrzut ekranu z zaprojektowaną obudową na wyświetlacz LCD oraz trzy przełączniki chwilowe.*

## 6 Opis oprogramowania

### 6.1 Program mikrokontrolera

Program mikrokontrolera Atmega328p został napisany w języku C za pomocą aplikacji Atmel Studio 7.0. Kod wynikowy zajmuje około 6.04 KB. Wydruk całego programu mikrokontrolera znajduje się na płycie CD dołączonej do pracy dyplomowej.

Program mikrokontrolera składa się z czterech części:

- inicjalizacja oraz implementacja funkcji zarządzających pracą wyświetlacza LCD, wraz z obsługą interfejsu TWI;
- inicjalizacja oraz implementacja funkcji do obsługi interfejsu UART;
- obsługa pracy czytnika linii papilarnych za pośrednictwem funkcji zaimplementowanych dla interfejsu UART;
- główna część programowa, która składa się z inicjalizacji zmiennych, głównej pętli programu oraz zaimplementowanych dodatkowych funkcji do wykonywania konkretnych akcji.

### 6.2 Obsługa wyświetlacza LCD oraz inicjalizacja interfejsu TWI

Do obsługi wyświetlacza i inicjalizacji interfejsu TWI zmodyfikowano gotową bibliotekę, która posiada licencję GNU GPL [10].

#### 6.2.1 Inicjalizacja interfejsu TWI

Inicjalizację interfejsu TWI należy zacząć od ustalenia częstotliwości z jaką będzie pracować zegar na linii SCL, który określony jest wzorem (Wzór 2):

$$\text{Taktowanie zegara na linii SCL [Hz]} = \frac{\text{Częstotliwość zegaru mikroprocesora [Hz]}}{16 + 2(\text{TWBR}) * (\text{Wartość preskalera})}$$

*Wzór 2 Wzór na częstotliwość zegara linii SCL w interfejsie TWI [2].*

TWBR jest rejestrem odpowiedzialnym za szybkość transmisji w interfejsie TWI, natomiast wartość preskalera jest określana w dwóch najmniej znaczących bitach w rejestrze TWSR, który jest rejestrem statusów. Do prawidłowej pracy konwertera I<sup>2</sup>C z układem mikrokontrolera, wszystkie bity w rejestrze TWBR zostały ustawione jako jedynka logiczna

(wartość TWBR – 255), natomiast dwa najmniej znaczące bity w rejestrze TWSR zostały ustawione na zero logiczne (wartość preskalera jest równa 1). Obliczona wartość częstotliwości zegara na linii SCL wynosi około 30kHz.

Aby rozpocząć pracę z interfejsem TWI, należy w pierwszej kolejności wysłać bit startu. TWCR jest rejestrem w mikrokontrolerze odpowiedzialnym za obsługę interfejsu TWI. W rejestrze TWCR ustawiana jest jedynka logiczna na bicie TWINT – służący jako flaga przerwań. Ustawienie jedynki logicznej jest równoznaczne z wyczyszczeniem flagi, co pozwala na rozpoczęcie pracy interfejsu TWI. Wprowadzana jest również jedynka logiczna na bicie TWSTA, który jest bitem warunku początkowego. Ustawienie tego bitu jako jedynki logicznej oznacza, że operujący mikrokontroler chce zostać urządzeniem nadrzędnym w interfejsie TWI. Na samym końcu ustawiana jest jedynka logiczna na bicie TWEN, którego zadaniem jest uruchomienie interfejsu TWI. W dalszej kolejności, w nieskończonej pętli należy czekać na ustawienie się zera logicznego na bicie TWINT – oznacza to, że warunek startu został pomyślnie wysłany.

Za wysyłanie danych odpowiedzialny jest rejestr TWDR, który jest rejestrem danych dla interfejsu TWI. Pozwala na wprowadzenie jednego bajta danych, który ma zostać wysłany. Proces wysyłania danych rozpoczyna się od ustawienia flagi przerwań TWINT oraz bitu aktywującego pracę interfejsu TWI (TWEN) na stan jedynki logicznej. Należy również w nieskończonej pętli czekać na ustawienie się bitu TWINT na stan logicznego zera, który oznacza, że dany bajt został pomyślnie wysłany.

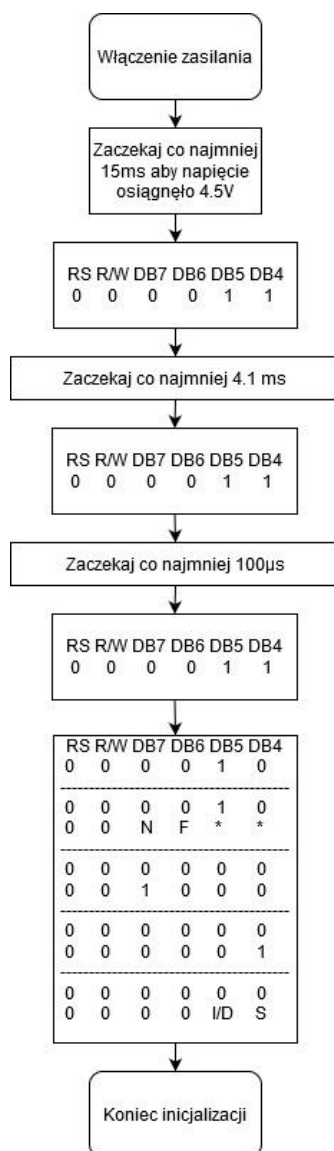
Aby zakończyć komunikację interfejsu TWI należy wysłać bit stopu. Proces jest analogiczny do wysyłania bitu startu, z tą różnicą, że ustawiana jest jedynka logiczna na bicie TWSTO, która jest bitem warunku końcowego. Nie ma również potrzeby, aby czekać na ustawienie się flagi TWINT do stanu zera logicznego.

### **6.2.2 Wyświetlanie informacji na wyświetlaczu LCD**

Aby móc wyświetlać informacje na wyświetlaczu LCD, należy w pierwszej kolejności zainicjalizować wyświetlacz LCD, aby działał w trybie 4 bitowym, zgodnie z założeniem konwertera I<sup>2</sup>C działającego na układzie PCF8574A.

Inicjalizacja odbywa się poprzez wysyłanie odpowiednich adresów do wewnętrznego rejestru układu konwertera I<sup>2</sup>C, który następnie ustawia wyprowadzenia na wyjściu do odpowiedniego stanu logicznego.





Rysunek 6.1 Inicjalizacja wyświetlacza w trybie 4 bitowym, tłum. własne [11].

Rysunek (Rysunek 6.1) przedstawia proces inicjalizacji wyświetlacza LCD w trybie 4 bitowym. Inicjalizacja wyświetlacza LCD odbywa się za pomocą wewnętrznego układu reset (Internal reset circuit). W tym czasie ustawiana jest flaga zajętości (BF), której stan utrzymuje się przez 10 ms po osiągnięciu przez wyświetlacz wymaganych 4.5 V zasilania, oraz jej stan jest ustawiany na logiczne zero ( $BF = 0$ ), gdy zakończy się proces inicjalizacji.

W trakcie inicjalizacji ustawiane są takie parametry jak:

- tryb pracy – 4 bitowy lub 8 bitowy;
- ilość linii wyświetlacza – parametr N = 1 (oznacza, że wyświetlacz posiada 2 linie na znaki);

- format znaków – parametr F = 0 (oznacza, że wyświetlacz będzie obsługiwany z czcionką znaków 5x8 kropek);
- włączenie wyświetlacza, wyłączenie kursora oraz wyłączenie migania kursora;
- inkrementacja znaków o 1 – parametr I/D = 1;
- kierunek inkrementacji znaków – parametr S = 1;

Po ukończeniu inicjalizacji, rozpoczyna się normalna praca wyświetlacza LCD. Aby wysyłać dane należy pamiętać, aby najmniej znaczący bit (R/W) ustawić na wartość zera logicznego, co pozwala na wysyłanie danych do wyświetlacza LCD. Z racji ustawienia wyświetlacza w tryb 4 bitowy, wysyłając dane należy dwukrotnie przesłać każdy znak. W pierwszej kolejności wysyłane są cztery najbardziej znaczące bity informacji, a następnie cztery najmniej znaczące.

## 6.3 Obsługa czytnika linii papilarnych wraz z inicjalizacją interfejsu UART

### 6.3.1 Inicjalizacja interfejsu UART

Inicjalizacja interfejsu UART została przeprowadzona zgodnie z instrukcją zamieszczoną w nocie katalogowej mikrokontrolera Atmega328p [2]. W pierwszej kolejności należało ustalić jaka jest szybkość transmisji (baud rate), na której działa czytnik linii papilarnych. Standardowa szybkość transmisji wynosi 57600 b/s.

Skorzystano ze wzoru (Wzór 3) na obliczenie wartości dla rejestru UBRR0H oraz UBRR0L (rejestry w interfejsie UART określające szybkość transmisji).

$$UBRR = \frac{\text{Częstotliwość zegaru mikroprocesora [Hz]}}{16 * \text{prędkość bodowa [b/s]}} - 1$$

*Wzór 3 Wzór na określenie wartości UBRR dla interfejsu UART [2].*

Pierwszym krokiem przy inicjalizacji interfejsu UART było wpisanie wcześniej ustalonej wartości UBRR do rejestrów BRR0H oraz UBRR0L. Następnie należy włączyć linie RX oraz TX – w tym celu do rejestru UCSR0B należy ustawić jedynkę logiczną na bicie TXEN0 (włączenie linii TX) oraz jedynkę logiczną na bicie RXEN0 (włączenie linii RX). Podczas inicjalizacji ustalony zostaje również format ramki danych na 8 bitów, poprzez ustawienie w rejestrze UCSR0C bitów UCSZ00 oraz UCSZ01 na wartość jedynki logicznej (8 bitowy format ramki danych).

Dane w interfejsie UART są umieszczane w rejestrze UDR0 – zarówno zawiera on ostatni odebrany bajt, jak również można w nim umieścić jeden bajt informacji, który należy wysłać. Do poprawnej pracy przy wysyłaniu danych, należy sprawdzić czy UDRE0 w rejestrze UCSR0A jest ustawiony na wartość logicznego zera. Flaga na bicie UDRE0 określa czy bufor danych jest pusty i gotowy do wysłania kolejnych danych. Analogiczna sytuacja występuje przy odbieraniu danych - na bicie RXC0 w rejestrze UCSR0A należy sprawdzić czy ustawiona jest wartość zera logicznego. Flaga RXC0 informuje o tym, czy bufor otrzymał dane.

### 6.3.2 Obsługa czytnika linii papilarnych

Program mikrokontrolera używa jedynie ramek poleceń (identyfikator ramki = 0x01), dlatego w miejscu przeznaczonym na zawartość danych należy przesłać odpowiedni jednobajtowy kod odpowiedzialny za wykonanie konkretnej akcji. Użyte w programie mikrokontrolera polecenia oraz ich zastosowania zostały zaprezentowane w tabeli poniżej (Tabela 2), wraz z możliwymi odpowiedziami zwrotnymi od czytnika.

*Tabela 2 Kody poleceń wraz z ich zastosowaniem i możliwą odpowiedzią zwrotną, tłum. własne [4].*

Kod	Zastosowanie	Odpowiedź zwrotna	
0x01	Wykrywanie palca na skanerze czytnika linii papilarnych oraz zapis odczytanego obrazu w buforze obrazu palca (Image buffer).	0x00	Pomyślne odczytanie obrazu palca.
		0x01	Błąd podczas otrzymywania ramki.
		0x02	Błąd wykrywania palca.
		0x03	Błąd zapisu obrazu palca.
0x02	Generowanie pliku znakowego z plik obrazu zapisanego w buforze obrazu palca (Image buffer) oraz zapis wygenerowanego pliku do bufora znaków (Char buffer 1 lub Char Buffer 2). Należy również wybrać, do którego z buforów znakowych należy zapisać plik (1 plik bajt).	0x00	Pomyślny zapis do bufora znakowego.
		0x01	Błąd podczas otrzymywania ramki.
		0x06	Błąd podczas generacji pliku znakowego z powodu zbyt nieuporządkowanego obrazu palca.
		0x07	Błąd podczas generacji pliku znakowego z powodu małej ilości punktów charakterystycznych lub zbyt małego obrazu palca.

		0x15	Błąd podczas generacji pliku znakowego z powodu braku zapisanego obrazu w buforze obrazu palca (Image Buffer).
0x04	Przeszukiwanie bazy zapisanych wzorów palców w poszukiwaniu odpowiadającego wzoru zapisanego w buforze znaków (Char buffer 1 lub Char buffer 2). Należy również podać numer bufora znakowego (1 bajt), początkową stronę (2 bajty) oraz numery stron do przeszukania (2 bajty).	0x00	Znaleziono pasujący palec w bazie. Zwracana jest również informacja z lokalizacją znalezionej obrazu palca (2 bajty) oraz wynik zgodności (2 bajty).
		0x01	Błąd podczas otrzymywania ramki.
		0x09	Brak zapisanego palca w bazie odcisków palców. Zwracana jest również lokalizacja oraz wynik zgodności, które wynoszą 0.
0x05	Łączenie ze sobą plików znakowych z dwóch buforów – Char buffer 1 oraz Char buffer 2). Nowy plik znakowy, składający się z dwóch plików znakowych umieszczany jest z powrotem do obu buforów (Char buffer 1 i Char buffer 2).	0x00	Operacja została zakończona pomyślnie.
		0x01	Błąd podczas otrzymywania ramki.
		0x0A	Błąd podczas łączenia ze sobą plików znakowych – pliki nie należą do jednego palca.
0x06	Zapis pliku znakowego do wskazanej przez użytkownika lokalizacji. Należy podać z jakiego bufora ma zostać pobrany plik znakowy – Char buffer 1 lub Char buffer 2 (1 bajt) oraz lokalizację w jakiej ma zostać zapisany obraz palca (2 bajty).	0x00	Pomyślny zapis obrazu palca do bazy.
		0x01	Błąd podczas otrzymywania ramki.
		0x0B	Podana przez użytkownika lokalizacja wykracza poza zakres posiadanego miejsca w czytniku linii papilarnych.
		0x18	Błąd zapisu do pamięci FLASH.
0x0C	Usunięcie obrazu palca z czytnika. Należy dodatkowo podać lokalizację obrazu	0x00	Pomyślne usunięcie obrazu palca z pamięci czytnika.
		0x01	Błąd podczas otrzymywania ramki.

	palca, który ma zostać usunięty (2 bajty) oraz liczbę zapisanych obrazów palca, które mają zostać usunięte (2 bajty).	0x10	Błąd podczas usuwania obrazu palca z czytnika.
0x1D	Odczyt liczby obrazów palców zapisanych w pamięci czytnika linii papilarnych.	0x00	Pomyślny odczyt. Zwracana jest również liczba zapisanych wzorów (2 bajty, starszy bajt przesyłany jest jako pierwszy).
		0x01	Błąd podczas otrzymywania ramki.

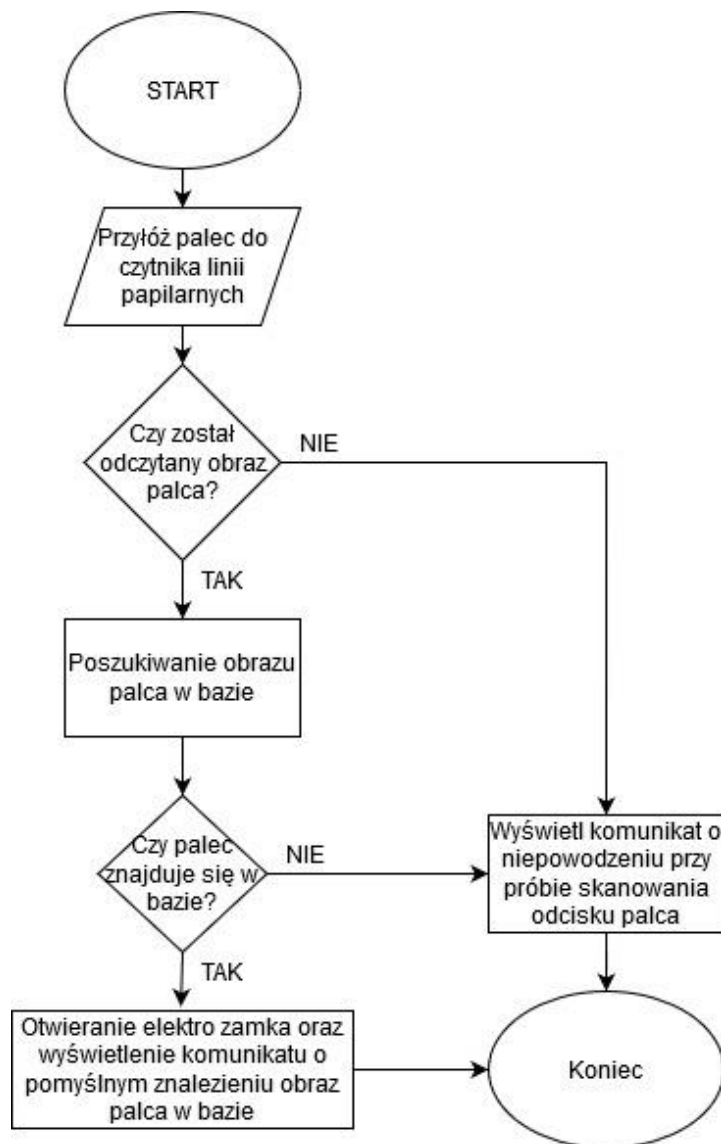
## 6.4 Główny program urządzenia

Główny program urządzenia zawiera pętlę nieskończoną, w której sprawdzany jest warunek czy elektrozamek jest wyłączony (drzwi są zamknięte) – jeżeli tak to program czeka na wykonanie akcji przez użytkownika w postaci naciśnięcia odpowiedniego przycisku. Każdy z przycisków wywołuje odpowiednią funkcję - odpowiadające każdemu z przycisków diagramy sekwencji zostały przedstawione na rysunkach poniżej (Rysunek 6.2 – 6.4).

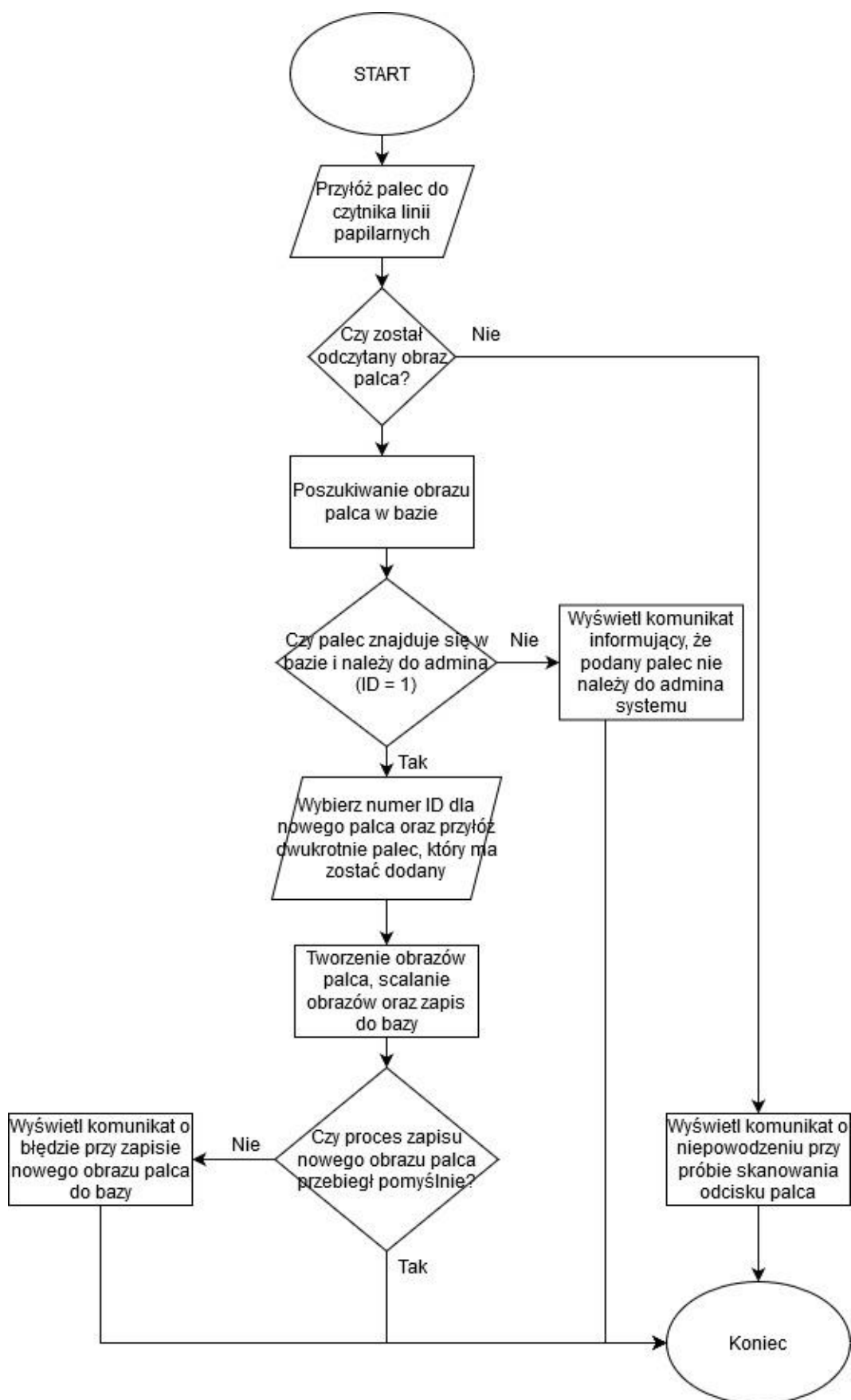
W programie głównym mikrokontrolera wykorzystano przerwania do obsługi głównej pętli. Przy wykorzystaniu wewnętrznego licznika 16 bitowego, ustawiana jest globalna flaga, która pozwala na sprawdzenie warunków znajdujących się w nieskończonej pętli. Inicjalizacja obsługi przerwań rozpoczyna się od ustalenia częstotliwości z jaką ma być wykonywane przerwanie. Ustalona wartość wynosi 200 Hz (5 ms). W rejestrze TCCR1B, który jest rejestrem kontrolnym licznika 16 bitowego, należało ustawić bity CS11 oraz CS10 na wartość jedynki logicznej. Dane bity odpowiedzialne są za ustalenie preskalera dla licznika, których wartość po ustawieniu jest równa 64. Korzystanie z takiej wartości preskalera pozwala na uzyskanie dokładniejszych wyników. Następnie w rejestrze TCCR1B należało również ustawić bit WG12 na wartość jedynki logicznej - jest on jednym z czterech bitów odpowiedzialnych za wybór trybu pracy licznika. W danej konfiguracji licznik pracuje w trybie CTC - wartość licznika jest porównywana z ustaloną wartością w rejestrze OCR1A, który jest 16 bitowym rejestrem porównania. Wartość rejestru OCR1A została ustalona na wartość 1249 (wartości liczone są od 0). Należy również uruchomić obsługę przerwań za pomocą wbudowanej funkcji sei().

Gdy wartość z licznika (zapisywana w rejestrze TCNT1) osiągnie wartość określoną w rejestrze OCR1A, następuje wywołanie przerwania, w którym ustawiana jest flaga

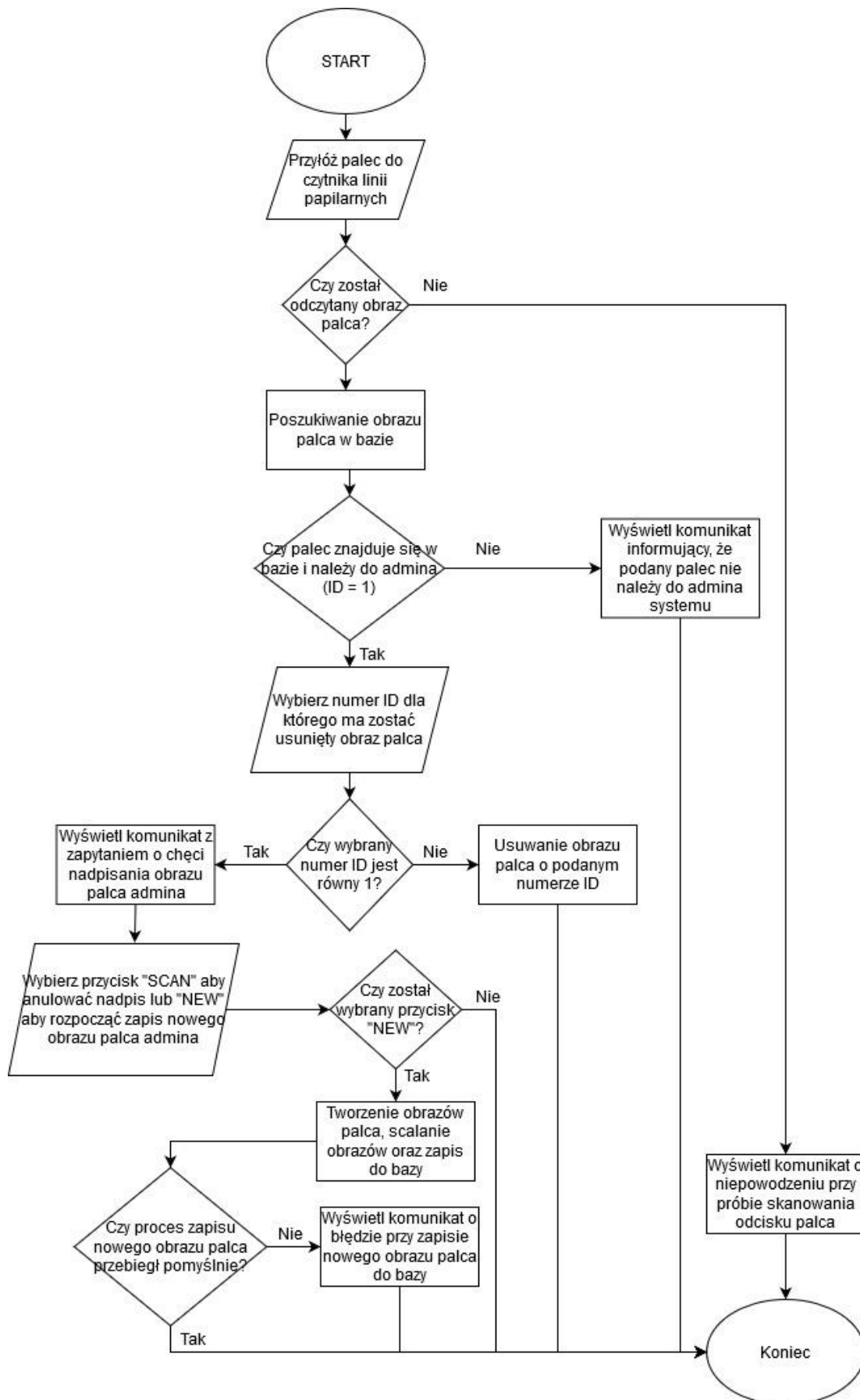
pozwalająca na sprawdzenie warunków w pętli nieskończonej. Flaga jest każdorazowo usuwana po przejściu przez główną pętlę.



Rysunek 6.2 Diagram sekwencji dla przycisku „SCAN”.



Rysunek 6.3 Diagram sekwencji dla przycisku „NEW”.



Rysunek 6.4 Diagram sekwencji dla przycisku „DELETE”.



W przypadku gdy elektrozamek jest włączony (drzwi są otwarte), w głównej pętli programu możliwa będzie jedynie jedna akcja do wykonania. Program będzie czekać na naciśnięcie przez użytkownika przycisku „SCAN”, który po naciśnięciu odcina zasilanie od przekaźnika, powodując wysunięcie się bolca elektrozamka. Program powraca do początkowej części pętli, oczekując na wykonanie akcji przez użytkownika w postaci naciśnięcia jednego z trzech przycisków.

Obrazy palców, zapisane w bazie czytnika linii papilarnych, mogą zostać nadpisane (chyba, że obraz palca należy do administratora, ID = 1), oraz usuwane, nawet gdy nie zawierają zapisanego obrazu palca. Dlatego istnieje administrator systemu, którego zadaniem jest również prowadzenie spisu lub listy dostępnych miejsc na nowe odciski palców.

## **7 Badania układu**

### **7.1 Badanie poboru prądu urządzenia**

Badanie polegało na sprawdzeniu poboru prądu przez urządzenie zamka szyfrowego wykorzystującego czytnik linii papilarnych. Urządzenie zasilane jest stałym napięciem 12V, które jest obniżane do 5 V za pomocą wewnętrznego regulatora napięcia, znajdującego się na układzie płytki Arduino Uno. Pozostałe moduły (wyświetlacz LCD z konwerterem I<sup>2</sup>C, moduł przekaźnika, regulator napięcia oraz moduł czytnika linii papilarnych) zasilane są również za pomocą 5 V, dostarczonych z układu Arduino Uno.

Przebadano prąd dostarczany do układu poprzez zmierzenie wartości natężenia za pomocą multimetru na wejściu złącza zasilania 12 V. Wynik pomiaru wykazał pobór prądu około 0.26 A, dla urządzenia nieaktywnego (żadne działanie nie jest podejmowane) oraz około 0.28 A, gdy urządzenie np. wyświetlało komunikat na wyświetlaczu LCD.

## **8 Obsługa**

Posiadany wyświetlacz LCD oraz obudowa z nadrukowanymi funkcjonalnościami każdego z przycisków ułatwia obsługę urządzenia. Jest ono gotowe do pracy po podłączeniu zasilania 12 V z ładowarki sieciowej lub baterii. Przy pierwszym użyciu na wyświetlaczu LCD pojawi się komunikat z prośbą o wprowadzenie nowego palca (Rysunek 8.1) który będzie pełnił rolę administratora systemu.



*Rysunek 8.1 Pierwsze użycie urządzenia, wyświetlenie się informacji o konieczności wprowadzenia nowego palca do systemu.*

Gdy zostanie wprowadzony nowy palec do systemu, pojawi się komunikat o zamkniętych drzwiach. W tym momencie użytkownik znajduje się w głównej części programu i może zdecydować o akcji, jaka ma zostać przeprowadzona, naciskając odpowiedni przycisk. Komunikaty zawierające błędy lub niepowodzenia przy przeprowadzeniu akcji, kierują użytkownika do głównej części programu, gdzie raz jeszcze może wybrać przycisk do wykonania odpowiedniej akcji. Przyciski „NEW” oraz „DELETE” służą także przy wyborze odpowiedniego numeru ID – przycisk „NEW” służy za inkrementację wartości, natomiast przycisk „DELETE” służy za dekrementację wartości (przedział dostępnych wartości: 1 – 255).

Gdy użytkownik wciśnie przycisk „SCAN”, należy przyłożyć palec do czytnika linii papilarnych (użytkownik będzie miał około pięć sekund na przyłożenie palca do czytnika). Jeżeli obraz palca istnieje w bazie obrazów czytnika, nastąpi wyświetlenie informacji o znalezieniu obrazu palca wraz z procentem zgodności przyłożonego palca z zapisanym obrazem palca w bazie (Rysunek 8.2). Zostanie również uruchomiony elektrozamek (drzwi zostaną otwarte) oraz wyświetlony zostanie komunikat „DRZWI OTWARTE” (Rysunek 8.3). Aby zamknąć drzwi i powrócić do głównej części programu należy wcisnąć przycisk „SCAN”.



*Rysunek 8.2 Pomyślne znalezienie obrazu palca w bazie obrazów palców czytnika linii papilarnych.*



*Rysunek 8.3 Komunikat informujący użytkownika o tym, że elektrozamek jest otwarty.*

W przypadku gdy nie istnieje zapisany wzór palca, który jest skanowany wyświetlony zostanie komunikat o braku danego palca w systemie (Rysunek 8.4).



*Rysunek 8.4 Komunikat o braku istnienia skanowanego palca w bazie odcisków palców.*

Gdy naciśnięty zostanie przycisk „NEW”, użytkownik zostanie poinformowany o konieczności wprowadzenia palca należącego do administratora systemu (Rysunek 8.5). W przypadku gdy palec należy do administratora, następnym krokiem jest wybór numeru ID (Rysunek 8.6), w którym zostanie zapisany nowy palec (jeżeli numer ID = 1, użytkownik zostanie poinformowany o błędzie przy próbie nadpisu palca należącego do administratora). W dalszej kolejności należy wprowadzić dwukrotnie nowy odcisk palca. Jeżeli próba dodawania nowego palca została zakończona pomyślnie, wyświetlony zostaje stosowny komunikat (Rysunek 8.7) oraz następuje przejście do głównej części programu.



*Rysunek 8.5 Komunikat z prośbą o wprowadzenie odcisku palca należącego do administratora.*



*Rysunek 8.6 Wybór odpowiedniego numeru ID przy dodawaniu lub usuwaniu odcisku palca z bazy.*





*Rysunek 8.7 Komunikat o pomyślnym dodaniu nowego odcisku palca do bazy obrazów palców.*

Gdy naciśnięty zostanie przycisk „DELETE”, użytkownik będzie proszony o wprowadzenie palca należącego do administratora oraz do wyboru odpowiedniego numeru ID, tak jak miało to miejsce przy naciśnięciu przycisku „NEW”. Jeżeli numer ID jest równy 1, pokazuje się komunikat (Rysunek 8.8) informujący o zamiarze nadpisania obecnego administratora. Następnie wyświetlony zostaje komunikat z prośbą o naciśnięcie odpowiedniego przycisku przez użytkownika (Rysunek 8.9) – przycisk „SCAN” powoduje przejście do głównej części programu, tym samym nie nadpisując obecnego administratora, natomiast przycisk „NEW” powoduje przejście do dwukrotnego skanowania nowego odcisku palca oraz nadpisanie obecnego administratora (gdy proces dodawania nowego palca się nie powiedzie, administrator nie zostaje zmieniony).



*Rysunek 8.8 Komunikat informujący użytkownika o zamiarze nadpisania obecnego administratora.*



*Rysunek 8.9 Komunikat z wyborem dostępnych akcji do wykonania podczas próby nadpisu administratora.*

## 9 Podsumowanie

Projektowanie, montaż oraz uruchomienie działającego zamka szyfrowego używającego czytnika linii papilarnych zostało przeprowadzone pomyślnie. Projekt wymagał wiedzy z zakresu elektroniki (wybór części, podłączanie układu, wiedza na temat sposobu działania elementów elektronicznych), jak i programowania (umiejętność programowania mikrokontrolera, umiejętność wyszukiwania oraz rozumienia zagadnień zawartych w notach katalogowych układów mikroprocesorowych).

Pomimo napotkanych przeszkód udało się w pełni zrealizować założenia projektowe. Urządzenie jest funkcjonalne i może zostać zaimplementowane do użytku codziennego np. w domu lub pracy. Nie oznacza to jednak, że projekt osiągnął swoje możliwości funkcjonalne. Istnieje ogromny potencjał, aby przy użyciu dodatkowych modułów usprawnić lub też w pełni zautomatyzować pracę urządzenia. Korzystając z gotowych modułów oferujących nawiązywanie połączenie mikrokontrolera z siecią internetową Wi-Fi, możliwe staje się stworzenie dziennika statusów zamka szyfrowego, które pozwala na zwiększenie bezpieczeństwa poprzez zapis wszystkich statusów z zamka szyfrowego (np. kto otwierał drzwi, kiedy) w lokalnej bazie danych. Pozwala to również na automatyzację dodawania i usuwania obrazów palców z bazy odcisków, ponieważ administrator systemu nie będzie zmuszony do prowadzenia listy z dostępnymi miejscami na odciski palców, oraz możliwe stanie się przypisanie konkretnego ID do jednego użytkownika. Możliwości rozwoju jest wiele, natomiast cele konkretnej pracy dyplomowej zostały w pełni zrealizowane, w efekcie czego otrzymano działające urządzenie.

Do pracy dołączona została płyta CD zawierająca między innymi: kod źródłowy programu mikrokontrolera, notę katalogową czytnika linii papilarnych Z70 oraz plik do druku obudowy na drukarce 3D.



## 10 Bibliografia

- [1] Strona producenta układu Arduino Uno Rev3  
<https://store.arduino.cc/arduino-uno-rev3>
- [2] Nota katalogowa mikrokontrolera Atmega328p  
[https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P\\_Datasheet.pdf](https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf)
- [3] Paul Horowitz, Winfield Hill (2018): *Sztuka elektroniki 2*, Wydawnictwa Komunikacji i Łączności WKŁ
- [4] Nota katalogowa czytnika linii papilarnych Z70
- [5] Nota katalogowa układu PCF8574, PCF8574A  
[https://www.nxp.com/docs/en/data-sheet/PCF8574\\_PCF8574A.pdf](https://www.nxp.com/docs/en/data-sheet/PCF8574_PCF8574A.pdf)
- [6] Schemat blokowy Arduino Uno Rev3  
[https://www.arduino.cc/en/uploads/Main/Arduino\\_Uno\\_Rev3-schematic.pdf](https://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf)
- [7] Schemat blokowy wyświetlacza LCD oraz konwertera I<sup>2</sup>C  
[https://www.microchip.com/forums/download.axd?file=0;1107608&where=&f=PCF8574\\_LCD\\_module.png](https://www.microchip.com/forums/download.axd?file=0;1107608&where=&f=PCF8574_LCD_module.png)
- [8] Objasnienie problemu napięcia zwrotnego występującego w elektromagnesach  
<https://progeny.co.uk/back-emf-suppression/>
- [9] Licencja i warunki użytkowania programu Tinkercad  
<https://www.autodesk.com/company/legal-notices-trademarks/terms-of-service-autodesk360-web-services/terms-of-service-for-tinkercad>
- [10] Biblioteka do obsługi wyświetlacza LCD działającym na module konwertera I<sup>2</sup>C  
<https://github.com/LittleBuster/avr-lcd1602>
- [11] Nota katalogowa wyświetlacza LCD HD44780  
<https://www.sparkfun.com/datasheets/LCD/HD44780.pdf>

## 11 Spis rysunków

Rysunek 2.1 Schemat blokowy przykładowej realizacji urządzenia. ....	4
Rysunek 3.1 Ramka danych w interfejsie USART, tłum. własne [2]. ....	5
Rysunek 3.2 Przebieg transmisji w interfejsie TWI, tłum. własne [3]. ....	6
Rysunek 3.3 Wysyłanie jeden ramki danych do czytnika linii papilarnych za pomocą interfejsu UART, tłum. własne [4]. ....	8
Rysunek 3.4 Adresowanie dla układu PCF8574A, tłum. własne [5]. ....	10
Rysunek 4.1 Symbol modułu Arduino Uno wykorzystujący mikrokontroler Atmega328p. ....	11
Rysunek 4.2 Schemat blokowy wyświetlacza LCD i konwertera I2C. ....	12
Rysunek 4.3 Schemat blokowy trzech przycisków. ....	13
Rysunek 4.4 Schemat blokowy przełącznika wraz podłączonym elektrozamkiem. ....	14
Rysunek 4.5 Schemat blokowy układu zamka szyfrowego wykorzystującego czytnik linii papilarnych. ....	15
Rysunek 5.1 Zrzut ekranu z zaprojektowaną obudową na wyświetlacz LCD oraz trzy przełączniki chwilowe. ....	16
Rysunek 6.1 Inicjalizacja wyświetlacza w trybie 4 bitowym, tłum. własne [11]. ....	19
Rysunek 6.2 Diagram sekwencji dla przycisku „SCAN”. ....	24
Rysunek 6.3 Diagram sekwencji dla przycisku „NEW”. ....	25
Rysunek 6.4 Diagram sekwencji dla przycisku „DELETE”. ....	26
Rysunek 8.1 Pierwsze użycie urządzenia, wyświetlenie się informacji o konieczności wprowadzenia nowego palca do systemu. ....	28
Rysunek 8.2 Pomyślne znalezienie obrazu palca w bazie obrazów palców czytnika linii papilarnych. ....	29
Rysunek 8.3 Komunikat informujący użytkownika o tym, że elektrozamek jest otwarty. ..	29
Rysunek 8.4 Komunikat o braku istnienia skanowanego palca w bazie odcisków palców. ....	30
Rysunek 8.5 Komunikat z prośbą o wprowadzenie odcisku palca należącego do administratora. ....	31
Rysunek 8.6 Wybór odpowiedniego numeru ID przy dodawaniu lub usuwaniu odcisku palca z bazy. ....	31
Rysunek 8.7 Komunikat o pomyślnym dodaniu nowego odcisku palca do bazy obrazów palców. ....	32
Rysunek 8.8 Komunikat informujący użytkownika o zamiarze nadpisania obecnego administratora. ....	33
Rysunek 8.9 Komunikat z wyborem dostępnych akcji do wykonania podczas próby nadpisu administratora. ....	33

## 12 Spis tabel

Tabela 1 Format ramki danych dla czytnika linii papilarnych, tłum. własne [4].....	8
Tabela 2 Kody poleceń wraz z ich zastosowaniem i możliwą odpowiedzią zwrotną, tłum. własne [4]. .....	21

## 13 Spis wzorów

Wzór 1 Wzór na częstotliwość graniczną filtra RC. ....	13
Wzór 2 Wzór na częstotliwość zegara linii SCL w interfejsie TWI [2].....	17
Wzór 3 Wzór na określenie wartości UBRR dla interfejsu UART [2].....	20