POLITECHNIKA GDAŃSKA
WYDZIAŁ ELEKTRONIKI, TELEKOMUNIKACJI I INFORMATYKI

Imię i nazwisko studenta: Adrianna Piekarska
Nr albumu: 165152
Studia pierwszego stopnia
Forma studiów: stacjonarne
Kierunek studiów: Informatyka
Profil: Architektura systemów komputerowych

Imię i nazwisko studenta: Grzegorz Wąs
Nr albumu: 165464
Studia pierwszego stopnia
Forma studiów: stacjonarne
Kierunek studiów: Informatyka
Profil: Inteligentne systemy interaktywne

PROJEKT DYPLOMOWY INŻYNIERSKI

Tytuł projektu w języku polskim: Bezprzewodowy system dostępu do pomieszczeń

Tytuł projektu w języku angielskim: Wireless access control system

Potwierdzenie przyjęcia projektu	
Opiekun projektu *podpis*	Kierownik Katedry/Zakładu (pozostawić właściwe) *podpis*
dr inż. Tomasz Dziubich	

Data oddania projektu do dziekanatu:

Streszczenie

Postępująca miniaturyzacja systemów wbudowanych wynikająca z wykładniczego rozwoju technologii półprzewodnikowych oraz rosnąca ich dostępność sprawiają, że systemy informatyczne znajdują coraz szersze zastosowanie w wielu dziedzinach. Nowoczesne i wydajne systemy stopniowo zastępują tradycyjne rozwiązania sprzed ery informatyzacji. Nowe możliwości niosą ze sobą jednak równie wiele wyzwań, szczególnie z zakresu bezpieczeństwa. Jedną z dziedzin o szerokich perspektywach rozwoju jest bezpieczeństwo przestrzeni wraz z ich uzytkownikami. Niniejsza praca opisuje projekt i implementację bezprzewodowego systemu dostępu do pomieszczeń. Omawia architekturę rozwiązania z uwzględnieniem poszczególnych podsystemów, przedstawia ciekawe aspekty realizacji projektu oraz jego rezultaty. Ponadto, prezentuje zagadnienia związane z bezpieczeństwem oraz wydajnością energetyczną bezprzewodowych systemów opartych na mikrokontrolerach.

TBD - rozszerzyć

Słowa kluczowe: zamek elektroniczny, mikrokontroler, kontrola dostępu, WiFi, RFID, sieć bezprzewodowa, autoryzacja

Dziedzina nauki i techniki, zgodnie z wymogami OECD:

Abstract

Postępująca miniaturyzacja systemów wbudowanych wynikająca z wykładniczego rozwoju technologii półprzewodnikowych oraz rosnąca ich dostępność sprawiają, że systemy informatyczne znajdują coraz szersze zastosowanie w wielu dziedzinach. Nowoczesne i wydajne systemy stopniowo zastępują tradycyjne rozwiązania sprzed ery informatyzacji. Nowe możliwości niosą ze sobą jednak równie wiele wyzwań, szczególnie z zakresu bezpieczeństwa. Jedną z dziedzin o szerokich perspektywach rozwoju jest bezpieczeństwo przestrzeni wraz z ich uzytkownikami. Niniejsza praca opisuje projekt i implementację bezprzewodowego systemu dostępu do pomieszczeń. Omawia architekturę rozwiązania z uwzględnieniem poszczególnych podsystemów, przedstawia ciekawe aspekty realizacji projektu oraz jego rezultaty. Ponadto, prezentuje zagadnienia związane z bezpieczeństwem oraz wydajnością energetyczną bezprzewodowych systemów opartych na mikrokontrolerach.

Keywords:

Spis treści

Spis rysunków

Spis tablic

Spis kodów źródłowych

Wykaz ważniejszych oznaczeń i skrótów

Pojęcie	Wyjaśnienie
Punkt dostępu	Fizyczne zabezpieczenie chroniące przed nieuprawnionym dostępem, przykładowo: zamek, bramka
RFID	Technologia wykorzystująca fale radiowe w celu przesyłania danych (ang. Radio-frequency identification)
Identyfikator	Identyfikator RFID. Inne określenia: karta, token
Kontroler	Komponent odpowiedzialny za zarządzanie układem zamka
Break glass device	Urządzenie awaryjne umożliwiające odcięcie zasilania w zamku

Rozdział 1

Wstep i cel pracy

1.1 Wstęp

Zapewnienie bezpieczeństwa przestrzeni użytkowych i osób z nich korzystających stanowi kluczowy aspekt zarządzania obiektami zarówno publicznymi, jak i prywatnymi. Dzięki zastosowaniu odpowiedniej infrstruktury, bezpieczeństwo osób przebywających na terenie obiektu rośnie, a ryzyko kradzieży lub zniszczenia mienia przez niepowołane osoby spada. Podstawową metodą kontroli dostępu do pomieszczeń jest montaż urządzeń ryglujących z układem zapadkowym rozpoznającym fizyczny klucz. Jednak ze względu na postępujący rozwój technologiczny, w obecnych czasach coraz częściej stosowane są systemy oparte na uwierzytelnianiu elektronicznym.

Pod względem celu i ogólnych zasad działania elektroniczny system kontroli dostępu do pomieszczeń nie rózni się od swojego tradycyjnego odpowiednika. Głównym celem jest autoryzacja prób dostępu użytkowników na podstawie kluczy w taki sposób, aby dostęp został przyznany tylko osobie posiadającej powiązany z danym punktem dostępu klucz.

Przewagę systemów opartych na urządzeniach elektronicznych nad systemami czysto mechanicznymi stanowią cechy takie jak łatwość obsługi czy możliwość zdalnego zarządzania oraz zbierania danych i monitorowania prób dostępu w celu późniejszej analizy.

1.2 Cel i zakres pracy

Celem niniejszej pracy jest projekt oraz implementacja sytemu dostępu do pomieszczeń z wykorzystaniem technologii takich jak Wi-Fi oraz RFID (ang. *Radio-frequency identification*), w którym podmiotem odpowiedzialnym za autoryzację prób dostępu jest serwer, a komunikacja pomiędzy układem sterującym zamkiem a podsystemem autoryzacji jest realizowana bezprzewodowo. Może on znaleźć zastosowanie jako łatwy w instalacji i obsłudze, lekki i wydajny system dla małych i średnich obiektów.

Podstawowe założenia dotyczące opisywanego systemu są następujące:

1. Logika uwierzytelniania powinna być zaimplementowana na serwerze. Urządzenie klienckie

(zamek) powinno pełnić jedynie rolę pośrednika w tym procesie.

2. Komunikacja pomiędzy urządzeniami klienckimi (układami zamka) a serwerem powinna odbywać się bezprzewodowo.
3. System powinien być wydajny energetycznie i umożliwiać operację zamków na zasilaniu bateryjnym.
4. System powinien implementować niezbędne mechanizmy bezpieczeństwa.

Pracę nad systemem prowadziły dwie osoby. W ramach tej pracy powstały:

- Protoyp układu zamka,
- Oprogramowanie serwera uwierzytelniania,
- Aplikacja do zarządzania,
- Baza danych.

Implementacja prototypu obejmowała stworzenie pojedynczego układu zamka. Ze względu na prototypowy charakter pracy, nie przetestowano działania systemu z większą liczbą zamków. Nie ma jednak powodów by twierdzić, że po minimalnych modyfikacjach system nie działałby poprawnie z większą liczbą zamków.

Tutaj podział pracy i obowiązków - TBD

1.3 Struktura pracy

Rozdział 2 pokrótce przedstawia dziedzinę problemu, przywołuje najważniejsze definicje związane z tematem oraz ogólny opis działania systemów kontroli dostępu. Opisuje też podstawowe zagrożenia bezpieczeństwa, z którymi spotkali się autorzy podczas pracy nad systemem, a także dokonuje przedstawienia i porównania kilku istniejących rozwiązań. Rozdział 3 prezentuje projekt rozwiązania. Rozdziały 4 oraz 5 przedstawiają proces implementacji tego projektu wraz z prezentacją najciekawszych problemów implementacyjnych oraz wykorzystanych technologii odpowiednio po stronie układu zamka oraz serwera. **?** Ze względu na to, iż w ramach pracy zaimplementowany został prototyp rozwiązania, rozdział ten prezentuje również możliwe modyfikacje i rozszerzenia tego prototypu. **?** Pracę zamyka rozdział 6, który opisuje rezultaty pracy nad projektem i dokonuje podsumowania.

Rozdział 2

Dziedzina problemu

Niniejszy rozdział krótko opisuje dziedzinę problemu w oderwaniu od szczegółów technicznych przygotowanego w ramach pracy rozwiązania, nakreśla podstawowe zagadnienia związane z bezpieczeństwem tego typu systemów, a także przedstawia porównanie niektórych z obecnie dostępnych na rynku systemów kontroli dostępu.

2.1 Kontrola dostępu

Kontrola dostępu to środki mające na celu zapewnienie, że do zasobów systemu przetwarzania danych mogą mieć dostęp tylko uprawnione jednostki w uprawniony sposób [1].

British Security Industry Association wyodrębnia kilka komponentów składających się na system kontroli dostępu [2]. Poniżej przedstawiono wybrane komponenty, które mają zastosowanie lub są powiązane z opisywanem systemem.

Poświadczenie tożsamości (ang. *credentials*) to fizyczny lub materialny obiekt, element wiedzy lub cecha biometryczna umożliwiająca uzyskanie dostępu do kontrolowanej strefy. Najczęściej jako poświadczenie tożsamości stosuje się kody, np. PIN (ang. *Personal Identification Number*, osobisty numer identyfikacyjny), tokeny (karty, urządzenia mobilne itp.) oraz dane biometryczne.

British Security Industry Association terminem ”czytniki” (ang. *readers*) nazywa urządzenia odpowiedzialne za kontrolę dostępu. Dla uproszczenia nomenklatura ta została zachowana w niniejszej sekcji. W innych częściach niniejszej pracy termin ”czytnik” używany jest w znaczeniu urządzenia odpowiedzialnego wyłącznie za odczyt danych z nośnika.

Czytniki mogą pracować samodzielnie; wyposażone są wówczas w urządzenia wejścia/wyjścia niezbędne do zarządzania zamkiem oraz pamięć i moc obliczeniową niezbędne do autonomicznego podejmowania decyzji. Zazwyczaj wyposażone są w uniwersalny kod umożliwiający uzyskanie dostępu każdemu kto wejdzie w jego posiadanie.

Czytniki mogą też pracować pod kontrolą innego urządzenia. Odczytane z nośnika dane poświadczające tożsamość przekazują do nadrzędnego urządzenia zwanego kontrolerem.

Istnieją również czytniki łączące funkcjonalność zarówno czytnika jak i kontrolera w jednym

urządzeniu. Posiadają one lokalną kopię bazy danych, na podstawie której podejmowana jest decyzja o przyznaniu lub odmowie dostępu.

Tzw. czytniki offline (ang. *offline readers*) różnią się od zwykłych czytników łączących funkcjonalności czytnika i kontrolera tym, iż nie przechwoują kopii bazy danych. W tym przypadku to nośnik danych(karta) zawiera informacje o tym, które zamki może otworzyć. Czytnik offline analizuje te dane i na ich podstawie podejmuje stosowną decyzję o podjęciu lub odmowie dostępu.

Czytniki online (ang. *online readers*) także nie przechowują kopii bazy danych. Decyzja o przyznaniu lub odmowie dostępu jest w ich przypadku podejmowana przez podłączony komputer, który przesyła odpowiednią komendę po udanej autentykacji.

W rozwiązaniach sieciowych czytniki mogą być podłączone do kontrolera, który przechowuje informacje niezbędne do podjęcia decyzji o przyznaniu bądź odmowie dostępu.

Urządzenia wyjściowe (ang. *egress devices*) umożliwiają użytkownikowi opusczenie strefy chronionej od wewnątrz. Jako urządzenia wyjściowe najczęściej używa się przełączników, czujników ruchu lub czytników. Według British Security Industry Association urządzenia wyjściowe można podzielić je na zwykłe oraz awaryjne (ang. *emergency egress*), przy czym, ze względu na krytyczne znaczenie w wypadku awarii, działanie tych drugich nie powinno zależeć od komponentów systemu (kontrolera systemu, oprogramowania itp.). Jako urządzenie awaryjne często stosuje się tzw. *break glass device*, którego uaktywnienie powoduje odcięcie zasilania w zamku, a tym samym wstrzymanie kontroli dostępu w danym punkcie. Dostęp uzyskany za pomocą tego urządzenia powinien wygenerować stosowne powiadomienie bądź alarm.

W zależności od potrzeb, oprogramowanie w systemie może być samodzielnym programem zainstalowanym na komputerze osobistym bądź złożonym i bezpiecznym oprogramowaniem zainstalowanym na serwerze. Często opiera się na rozwiązaniach webowych lub mobilnych, umożliwiając dostęp z dowolnego urządzenia.

British Security Industry Association w następujący sposób przedstawia sposób działania systemów kontroli dostępu:

W systemie on-line, w momencie gdy poświadczenie tożsamości zostaje przedstawione czytnikowi, informacja przesyłana jest do kontrolera. Kontroler porównuje otrzymane dane z listą autoryzowanych użytkowników w bazie danych. Jeżeli przedstawione dane znajdują się w bazie, kontroler wysyła sygnał otworzenia zamka. Sygnał wysyłany jest do czytnika w celu wizualnego lub dźwiękowego powiadomienia użytkownika o podjętej decyzji.

W systemie off-line, w momencie gdy poświadczenie tożsamości zostaje przedstawione czytnikowi, czytnik dokonuje sprawdzenia, czy dostęp powinien zostać przyznany. Jeżeli tak, czytnik zezwala na dostęp i aktualizuje przedstawiony nośnik danych poświadczających tożsamość. W momencie zaprezentowania tego samego nośnika w czytniku wyposażonym w kontroler dane na temat dostępów zostaną zanotowane w systemie, a sam nośnik może zostać zaktualizowany o zmiany w prawach dostępu, jeśli takie nastąpiły.

W większości przypadków tylko wejście podlega kontroli. Aby możliwa była również kontrola wyjścia z chronionego terenu, potrzebny jest drugi czytnik umieszczony po drugiej stronie drzwi.

Jeżeli obustronna kontrola nie jest wymagana, stosuje się zazwyczaj przycisk umożliwiający otworzenie zamka od środka.

W przypadku, gdy system kontroli dostępu nie funkcjonuje odpowiednio (np. z powodu braku zasilania), stosuje się tzw. *break glass device*.

2.2 Bezpieczeństwo

Przegląd zagrożeń związanych z bezpieczeństwem - TBD

Projektując system o tak krytycznym znaczeniu jak system kontroli dostępu, należy poświęcić znaczną uwagę zagadnieniom związanym z bezpieczeństwem. Bezpieczeństwo całego systemu nie może zostać osiągnięte, jeśli chociaż jeden z jego elementów nie zostanie odpowiednio zabezpieczony. Należy więc dołożyć wszelkich starań, aby zapewnić odpowiednie mechanizmy bezpieczeństwa we wszystkich warstwach systemu: sprzętowej oraz programowej oraz odpowiednią ochronę danych we wszystkich fazach działania systemu.

2.3 Przegląd dostępnych rozwiązań

Ten podrozdział wymaga dopracowania i rozszerzenia - TBD

Obecnie stosowane rozwiązania różnią się od siebie pod wieloma względami. W mniej wymagających systemach często stosuje się rozwiązania oparte na architekturze rozproszonej. W rozwiązaniach tego typu urządzenia kontrolujące zamki pracują w sposób autonomiczny. Oznacza to, iż cały proces uwierzytelniania dokonywany jest przez oprogramowanie mikroprocesora obsługującego zamek.

Na rynku dostępne są także rozwiązania sieciowe, bądź też takie, które umożliwiają konfigurację urządzeń w tryb zarówno autonomiczny, jak i sieciowy. Rozwiązania sieciowe charakteryzują się znacznym stopniem skomplikowania, zarówno pod względem architektonicznym (ilość i rodzaj potrzebnych komponentów sprzętowych) [3], jak i konfiguracyjnym (trudność instalacji, konieczność modyfikacji istniejącej infrastruktury). Mogą oferować oddzielenie funkcjonalności czytnika dostępu od kontrolera, umożliwiając obsługę do kilkunastu czytników za pomocą jednego urządzenia kontrolującego [3]. Mimo możliwości dołączenia do kontrolera zamków bezprzewodowych, działanie całego systemu wciąż pozostaje w pewnym stopniu uzależnione od komunikacji przewodowej.

Ze względu na ilość komponentów sprzętowych, istniejące rozwiązania bywają drogie.

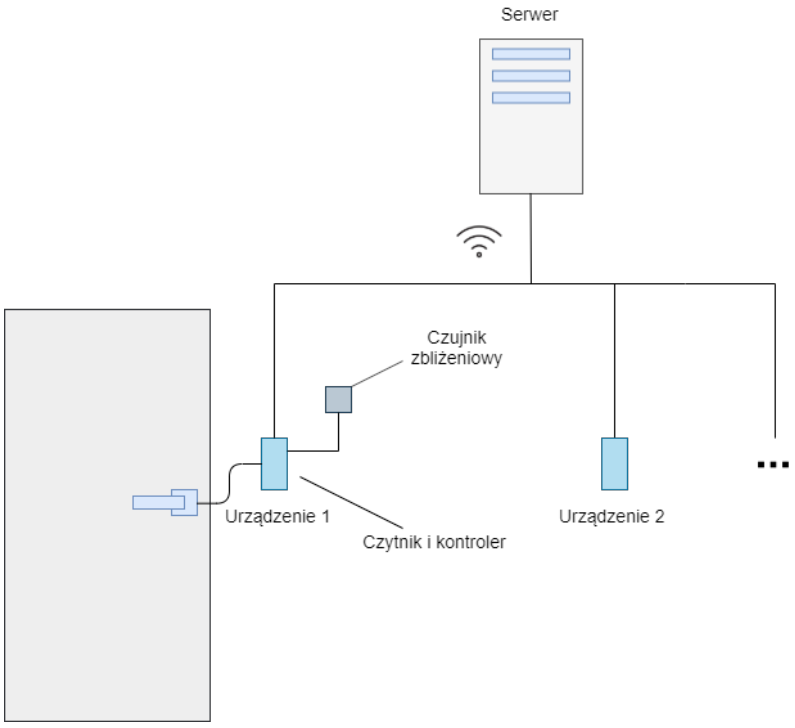
Rozdział 3

Projekt rozwiązania

Niniejszy rozdział przedstawia koncepcję systemu. Definiuje użytkowników oraz wymagania funkcjonalne i pozafunkcjonalne. Przedstawia budowę systemu, opisuje kolejno tworzące system komponenty sprzętowe oraz podsystemy, a także relacje, które między nimi zachodzą. Ponadto, prezentuje zakładany sposób działania systemu w kilku najbardziej prawdopodobnych sytuacjach.

3.1 Idea

Idea systemu przedstawiona została na rysnku 3.1.



Rysunek 3.1: Idea systemu

3.2 Użytkownicy

Na potrzeby projektu rozwiązania zidentyfikowano dwóch użytkowników. Ich specyfikacja przedstawiona została w tabeli 3.1.

Tablica 3.1: Użytkownicy systemu

USER_1	**Użytkownik**
Opis	Osoba posiadająca identyfikator RFID, której celem jest uzyskanie dostępu do chronionego zamkiem pomieszczenia
Źródło	Placeholder

USER_2	**Administrator**
Opis	Osoba posiadająca uprawnienia administracyjne w systemie, mająca dostęp do oprogramowania zarządzającego systemem
Źródło	Placeholder

3.3 Wymagania funkcjonalne

W tabeli 3.2 przedstawiono wymagania funkcjonalne systemu.

Tablica 3.2: Wymagania funkcjonalne

FNRQ_1	**Kontrola dostępu do pomieszczeń**
Opis	Dopuszczanie do pomieszczeń użytkowników posiadających odpowiedni identyfikator i niedopuszczanie użytkowników nieposiadających odpowiedniego identyfikatora
Źródło	Placeholder

FNRQ_2	**Przeglądanie historii prób dostępu do pomieszczeń**
Opis	Dostęp do listy dokonanych w przeszłości prób dostępu zakończonych zarówno sukcesem, jak i porażką
Źródło	Placeholder

FNRQ_3	**Dodawanie identyfikatorów**
Opis	Dodawanie identyfikatorów wraz z przyznaniem dostępu do wybranej grupy zamków
Źródło	Placeholder

FNRQ_4	**Przeglądanie identyfikatorów powiązanych z danym zamkiem**
Opis	Dostęp do listy powiązań między identyfikatorami a zamkami
Źródło	Placeholder

FNRQ_5	**Blokowanie dostępu do pomieszczeń dla wybranego identyfikatora**
Opis	Manualny wybór opcji czasowego usunięcia powiązania wybranego identyfikatora z wybraną grupą zamków
Źródło	Placeholder

3.4 Wymagania pozafunkcjonalne

W tabeli 3.3 przedstawiono wymagania pozafunkcjonalne systemu.

3.5 Komponenty systemu

System podzielony został na podsystemy realizujące określone funkcjonalności i istniejące w ramach fizycznych komponentów sprzętowych.

Tablica 3.2: Wymagania funkcjonalne, c.d.

FNRQ_6	**Dostęp do grupy pomieszczeń za pomocą jednego identyfikatora**
Opis	Ustawienie powiązań identyfikatorów i zamków w taki sposób, aby możliwy był dostęp do grupy zamków za pomocą jednego identyfikatora
Źródło	Placeholder

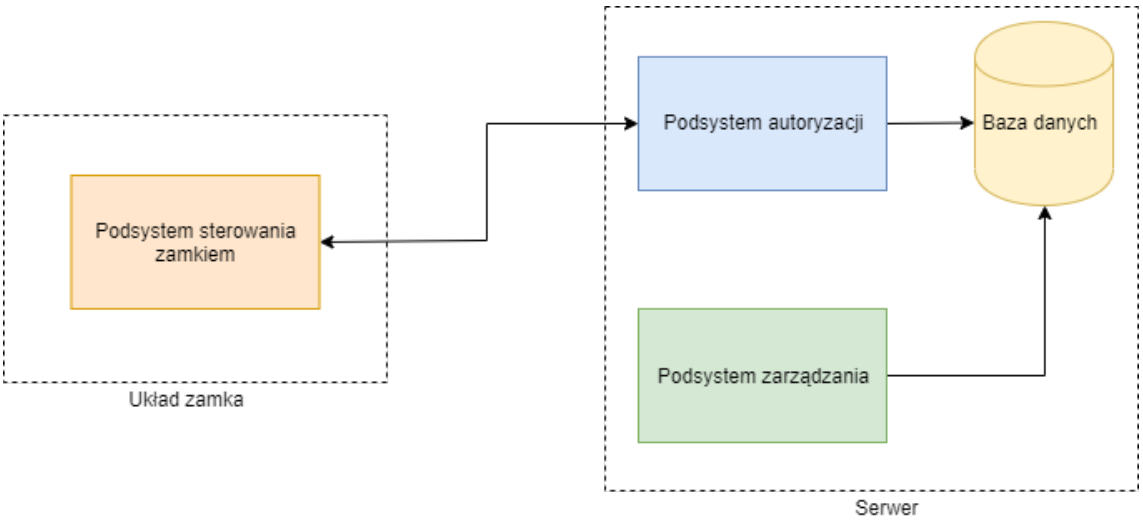
FNRQ_7	**Dostęp do pomieszczenia za pomocą grupy identyfikatorów**
Opis	Ustawienie powiązań identyfikatorów i zamków w taki sposób, aby możliwy był dostęp do jednego zamka za pomocą grupy identyfikatorów
Źródło	Placeholder

FNRQ_8	**Sygnalizacja przyznania lub odmowy dostępu**
Opis	Powiadamianie użytkownika o podjętej przez system decyzji
Źródło	Placeholder

FNRQ_9	**Sygnalizacja stanu naładowania baterii w układzie zamka**
Opis	Okresowe powiadamianie administratora systemu o bieżącym stanie naładowania baterii
Źródło	Placeholder

Tablica 3.3: Wymagania pozafunkcjonalne

XXRQ_1	**Długość czasu pracy na baterii równa minimum 1 rok**
Opis	Placeholder
Źródło	Placeholder



Rysunek 3.2: Architektura systemu

3.5.1 Podsystemy

W tabeli 3.4 przedstawiono podsystemy wraz z lokalizacją oraz przynależnymi im komponentami programowymi.

3.5.2 Komponenty sprzętowe

W tabeli 3.5 przedstawiono komponenty sprzętowe systemu.

3.5.3 Komponenty programowe

W tabeli 3.6 przedstawiono komponenty programowe systemu.

3.6 Projekt architektury

Niniejszy punkt prezentuje budowę systemu.

Na rysunku 3.2 przedstawiono architekturę systemu z podziałem na komponenty sprzętowe oraz podsystemy.

Na rysunku 3.3 przedstawiono szczegółową budowę układu zamka.

TBD - budowa podsystemu autoryzacyjnego

Na rysunku 3.4 przedstawiono szczegółową budowę podsystemu zarządzającego.

3.7 Sposób działania

Działanie systemu opiera się na współpracy układu zamka z serwerem w celu zapewnienia poprawnej autoryzacji identyfikatorów w zamkach. Funkcjonalność autoryzacji realizowana jest

Tablica 3.4: Komponenty programowe

SSYS_1	**Podsystem sterowania zamkiem**
Opis	Odpowiedzialny za odczyt danych identyfikatora użytkownika oraz przekazanie ich do podsystemu autoryzacji, zarządzanie zasilaniem elementów układu zamka oraz zarządzanie samym zamkiem.
Lokalizacja	HCMP_1 Układ zamka
Komponenty	SCMP_1 Oprogramowanie mikrokontrolera w układzie zamka
Źródło	Placeholder

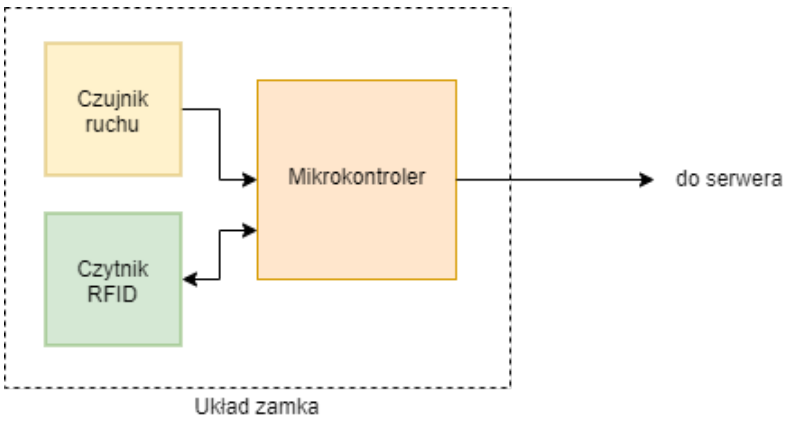
SSYS_2	**Podsystem autoryzacji**
Opis	Odpowiedzialny za podjęcie decyzji o przyznaniu bądź odmowie dostępu na podstawie otrzymanych od podsystemu sterowania zamkiem danych. Komunikuje się z podsystemem sterowania zamkiem oraz bazą danych.
Lokalizacja	HCMP_2 Serwer
Komponenty	SCMP_2 Oprogramowanie autoryzujące
Źródło	Placeholder

SSYS_3	**Podsystem zarządzający**
Opis	Odpowiedzialny za umożliwienie administratorowi systemu wglądu do danych takich jak historia prób dostępu, zbiór identyfikatorów, zamków, oraz powiązań między nimi, a także stan poszczególnych zamków. Dzięki niemu możliwa jest konfiguracja rozpoznawanych przez system identyfikatorów i zamków oraz manualne przyznawanie dostępu poszczególnym identyfikatorom.
Lokalizacja	HCMP_2 Serwer
Komponenty	SCMP_3 Oprogramowanie zarządzające, SCMP_4 Baza danych
Źródło	Placeholder

Tablica 3.5: Komponenty sprzętowe

HCMP_1	**Układ zamka**
Opis	Złożony z następujących subkomponentów: • Mikrokontroler Odpowiada za sterowanie peryferiami, zarządzaniem ich zasilaniem, inicjację i przeprowadzenie bezprzewodowej komunikacji z serwerem i sterowanie samym zamkiem. • Czujnik ruchu Jego jedynym zadaniem jest wykrycie zbliżającego się do zamka użytkownika. • Czytnik RFID Stanowi interfejs pomiędzy użytkownikiem a systemem.
Powiązania	SCMP_1 Oprogramowanie mikrokontrolera w układzie zamka
Źródło	Placeholder

HCMP_2	**Serwer**
Opis	Placeholder
Powiązania	SCMP_2 Oprogramowanie autoryzujące, SCMP_3 Oprogramowanie zarządzające, SCMP_4 Baza danych
Źródło	Placeholder



Rysunek 3.3: Budowa układu zamka

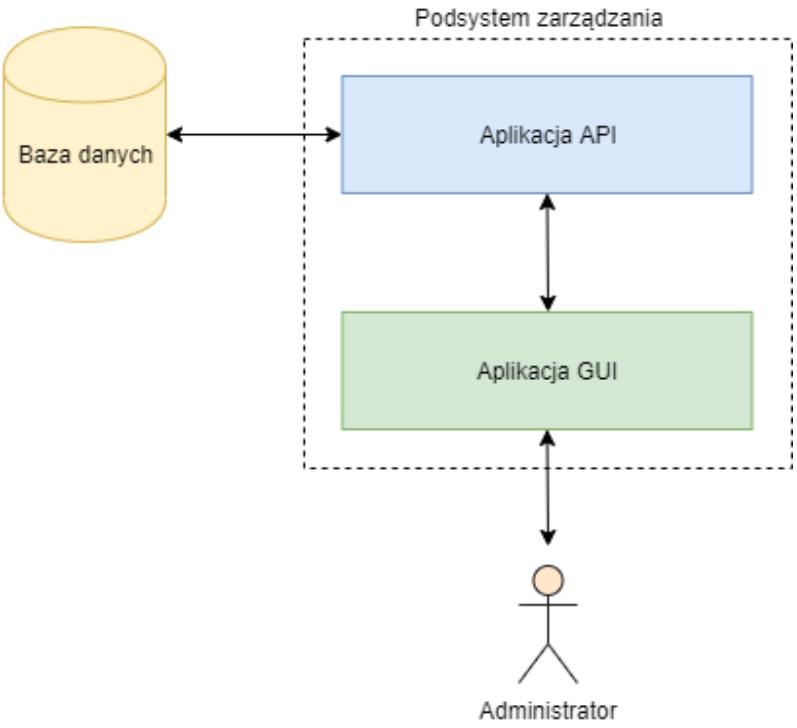
Tablica 3.6: Komponenty programowe

SCMP_1	**Oprogramowanie mikrokontrolera w układzie zamka**
Opis	Odpowiedzialne za zarządzanie peryferiami układu, komunikację z serwerem oraz kontrolę zamka.
Powiązania	HCMP_1 Układ zamka
Źródło	Placeholder

SCMP_2	**Oprogramowanie autoryzujące**
Opis	Zawiera logikę autoryzacyjną. Odbiera dane od układu zamka, komunikuje się z bazą danych w celu podjęcia decyzji o autoryzacji.
Powiązania	HCMP_2 Serwer
Źródło	Placeholder

SCMP_3	**Oprogramowanie zarządzające**
Opis	Umożliwia zarządzanie istniejącymi identyfikatorami i zamkami oraz dodawanie nowych, przeglądanie historii prób dostępu.
Powiązania	HCMP_2 Serwer
Źródło	Placeholder

SCMP_4	**Baza danych**
Opis	Przechowuje dane dotyczące poszczególnych identyfikatorów i zamków zarejestrowanych w systemie, powiązań pomiędzy nimi oraz dokonanych w przeszłości prób dostępu zakończonych zarówno sukcesem jak i porażką.
Powiązania	HCMP_2 Serwer
Źródło	Placeholder



Rysunek 3.4: Budowa podsystemu zarządzania

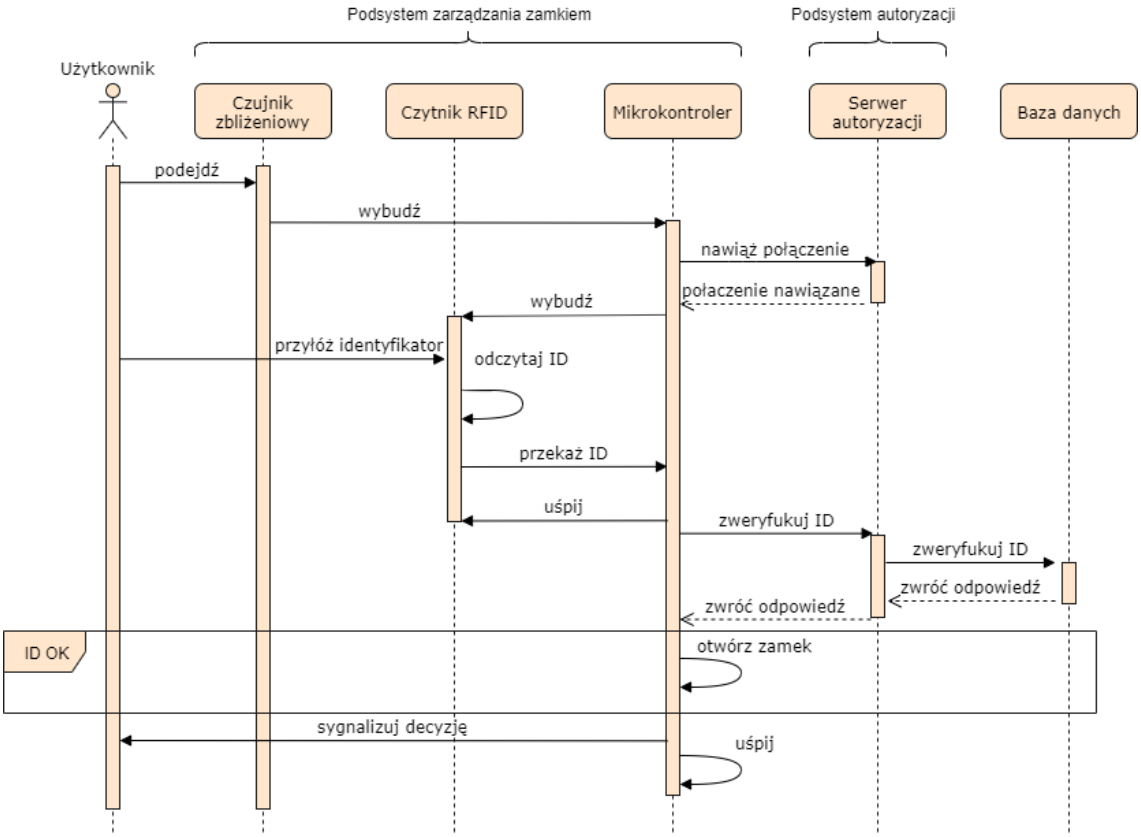
wyłącznie po stronie serwera. Układ zamka, będący urządzeniem klienckim, jest jedynie pośrednikiem przekazującym dane od użytkownika do serwera.

W celu osiągnięcia największej możliwej wydajności energetycznej czytnik RFID oraz mikrokontroler przez większą część czasu pozostają w stanie uśpienia. Zadaniem czujnika ruchu, zasilanego przez cały czas, jest odpowiednio wczesne wykrycie zbliżającego się użytkownika i wybudzenie mikrokontrolera, który z kolei jest odpowiedzialny za zasilenie czytnika RFID oraz nawiązanie połączenia z serwerem. Jeśli operacja nawiązania połączenia przebiegnie pomyślnie, a do czytnika przyłożony zostanie identyfikator, rozpoczyna się proces przekazywania danych odczytanych z identyfikatora do serwera w celu autoryzacji identyfikatora. Należy zwrócić uwagę na niekorzystne z punktu widzenia systemu warunki, które mogą zajść w trakcie procesu:

1. Jeżeli identyfikator nie zostanie przyłożony w przeciągu 10 sek. od momentu wybudzenia czytnika, mikrokontroler ponownie wprowadza czytnik oraz samego siebie w stan uśpienia.

2. Jeżeli połączenie z serwerem nie może zostać nawiązane w czasie $t + 5\ sek.$, gdzie t jest zmiennym czasem upływającym od momentu podjęcia próby nawiązania połączenia z serwerem do momentu zakończenia odczytu danych z identyfikatora, mikrokontroler sygnalizuje użytkownikowi błąd połączenia, po czym wprowadza się w stan uśpienia.

Jeżeli żadna z wymienionych wyżej niekorzystnych sytuacji nie wystąpi i dane zostaną pomyślnie przesłane do serwera, serwer podejmuje decyzję o przyznaniu bądź odmowie dostępu. Dokonuje tego po wysłaniu stosownego zapytania do bazy danych, a następnie wysyła potwierdzenie lub odmowę do mikrokontrolera. Mikrokontroler w sposób wizualny sygnalizuje decyzję



Rysunek 3.5: Przepływ sterowania w procesie autoryzacji identyfikatora, pomyślny scenariusz

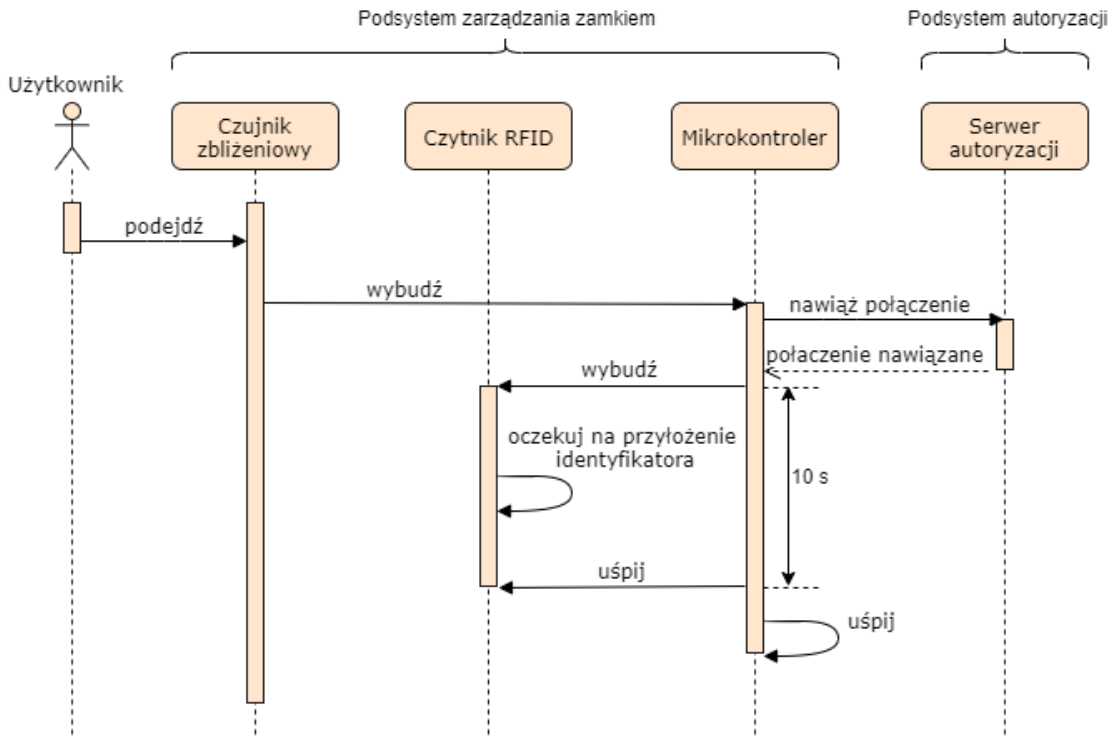
użytkownikowi, a jeżeli była ona pomyślna, dodatkowo wysyła sygnał otwierający zamek. Niezależnie od decyzji serwera, wpis o próbie dostępu zostaje zapisany w bazie danych, skąd może być pobrany przez podsystem zarządzający w celu prezentacji danych administratorowi systemu.

Opisane wyżej mechanizmy zostały szerzej ukazane na diagramach sekwencji. Diagramy 3.5-3.7 dotyczą podsystemu sterowania zamkiem oraz podsystemu autoryzacji, natomiast diagram 3.8 dotyczy podsystemu zarządzającego. Diagram 3.5 przedstawia proces autoryzacji identyfikatora użytkownika w przypadku najbardziej pomyślnego scenariusza. Diagram 3.6 ukazuje przepływ sterowania pomiędzy elementami systemu w sytuacji, gdy użytkownik zostanie wykryty, ale identyfikator nie zostanie przyłożony do czytnika w zadanym przedziale czasu. Diagram 3.7 przedstawia przepływ sterowania pomiędzy elementami systemu w sytuacji, gdy niemożliwe jest nawiązanie połączenia z serwerem. Diagram 3.8 ukazuje przepływ sterowania pomiędzy warstwami podsystemu zarządzający w sytuacji żądania dostępu do historii prób dostępu przez administratora systemu.

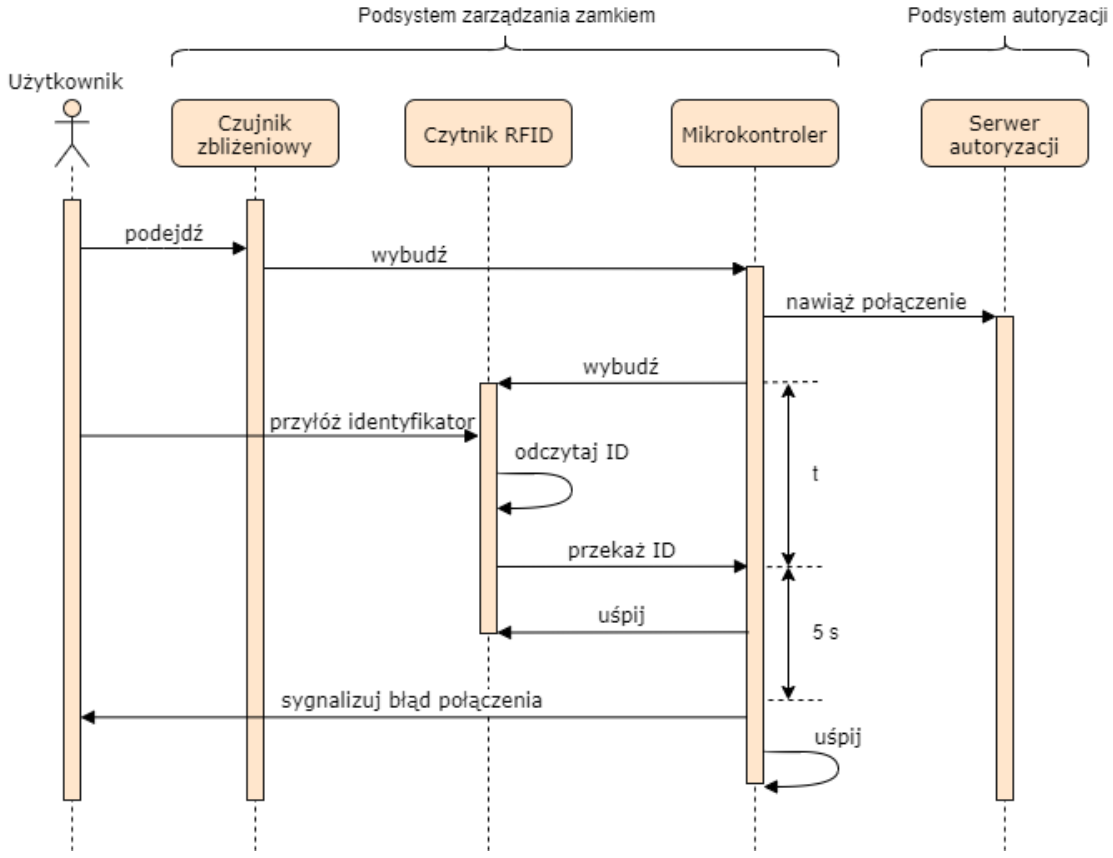
Należy zwrócić uwagę na możliwość wystąpienia również innych sytuacji niekorzystnych, takich jak błąd połączenia z bazą danych lub **co jeszcze?**. Ich obsługa jest pomijalna z punktu widzenia współpracy komponentów systemu, dlatego nie została uwzględniona na diagramach.

dodać scenariusz dodawania nowego identyfikatora

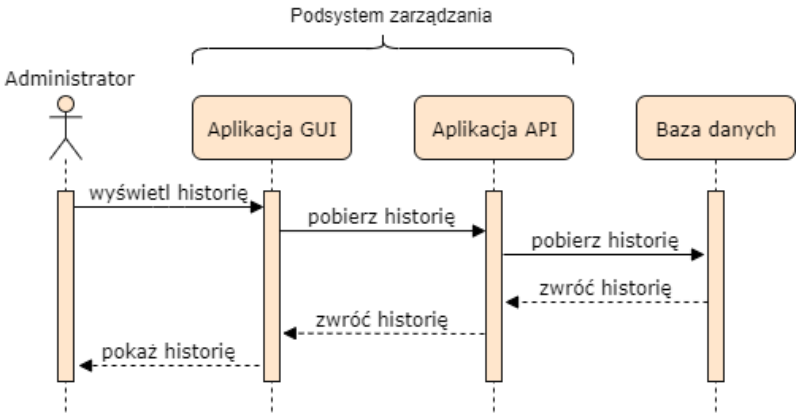
TBD - oddzielić jakoś diagramy sprzętowe od funkcjonalnych



Rysunek 3.6: Przepływ sterowania w sytuacji wykrycia użytkownika nie prezentującego identyfikatora



Rysunek 3.7: Przepływ sterowania w sytuacji braku możliwości nawiązania połączenia z serwerem



Rysunek 3.8: Przepływ sterowania w procesie żądania dostępu do historii prób dostępu

Rozdział 4

Komponent sterowania zamkiem

Niniejszy rozdział zagłębia się w szczegóły implementacyjne projektu. Opisuje sposób konfiguracji środowiska deweloperskiego, przedstawia wybrane zagadnienia związane z implementacją oraz wykorzystane w projekcie technologie sprzętowe i programowe.

Spełnienie założeń dotyczących systemu wymienionych w rozdziale 3 wymagało zastosowania mikrokontrolera integrującego układ komunikacji bezprzewodowej zgodnej ze zbiorem standardów IEEE 802.11, obsługującego interfejs szeregowej komunikacji z peryferiami SPI (ang. *Serial Peripheral Interface*) jednocześnie umożliwiając komunikację za pośrednictwem programowalnych wejść/wyjść GPIO (ang. *General-Purpose Input/Output*). Pośród wielu rozwiązań dostępnych na rynku wybrany został mikrokontroler ESP32, ze względu na możliwość równoległego przetwarzania za pomocą dwóch fizycznych rdzeni, sprzętowe wsparcie dla algorytmów kryptograficznych oraz możliwość zaawansowanej kontroli pracy podzespołów mikrokontrolera, szczególnie w zakresie zasilania.

Oprogramowanie dla mikrokontrolera ESP32 może być wytwarzane za pomocą środowiska Arduino. Jednakże na potrzeby projektu zdecydowano się na wykorzystanie w tym celu oficjalnego frameworka Espressif IoT Development Framework (ESP-IDF) firmy Espressif Systems. Motywacją tej decyzji była chęć uzyskania pełniejszej kontroli nad sprzętem oraz lepszego poznania mechanizmów działania systemu operacyjnego czasu rzeczywistego FreeRTOS, na którym bazuje framework ESP-IDF.

Kryterium wyboru czytnika kart RFID była obsługa kart o częstotliwości nośnej 13,56 MHz, dostępność oprogramowania sterującego czytnikiem dla wybranego mikrokontrolera oraz obsługa interfejsu komunikacji szeregowej SPI w celu prostej integracji czytnika z mikrokontrolerem. Do tego zadania wybrany został układ MFRC522.

Efektywne wykorzystanie energii w układzie zamka wymaga źródła impulsu wybudzającego o jak najniższym poborze mocy w stanie spoczynku. Wybranym rozwiązaniem jest czujnik PIR (ang. *Passive Infra Red*, pasywny czujnik podczerwieni), model HC-SR501.

4.1 Konfiguracja środowiska

TBD

4.2 Oprogramowanie mikrokontrolera

4.2.1 Struktura kodu

Kod źródłowy oprogramowania mikrokontrolera podzielony jest na komponenty o zadanych funkcjonalnościach:

1. Komponent główny (*main*): odpowiedzialny za rozróżnienie rodzajów uruchomienia (pierwsze uruchomienie lub wybudzenie z uśpienia), wywołanie odpowiedniej procedury komponentu sterującego oraz uśpienie układu po jej zakończeniu.
2. Komponent RFID: odpowiedzialny za inicjalizację czytnika MFRC522, wykrywanie i odczyt karty oraz sygnalizację zdarzeń związanych z odczytem kluczy dostępu do pomieszczeń.
3. Komponent WiFi: odpowiedzialny za realizację komumnikacji bezprzewodowej z serwerem. Obsługuje transmisje wychodzące i przychodzące wraz z opcjonalnym zestawieniem bezpiecznego kanału komunikacji z wykorzystaniem protokołu TLS.
4. Komponent sterujący (*flow-controller*): odpowiedzialny za kontrolę przepływu sterowania, aktywowanie poszczególnych komponentów.

4.2.2 Współbieżność

Ze względu na restrykcyjne wymagania dotyczące czasu trwania procesu zestawiania połączenia z serwerem, wykorzystano przetwarzanie współbieżne na dwóch wątkach.

Coś o podziale zadań między taskami - TBD

Diagram pokazujący jak taski między sobą się komunikują itd

Powiadamianie o wszystkich zdarzeniach w oprogramowaniu mikrokontrolera realizowane jest przez mechanizm tzw. *Event Group*. Jest on zapewniany przez system FreeRTOS. W przypadku wystąpienia zdarzenia zostaje ono zapisane we wspomnianym wyżej *Event Group*, co jest równoznaczne z ustawieniem w stan wysoki bitu przypisanego danemu zdarzeniu. Kluczowy z perspektywy przepływu sterowania jest wykorzystany mechanizm oczekiwania na zdarzenia, który przez wywłaszczenie oczekującego wątku pozwala na efektywną implementację synchronizacji wątków.

Dodać źródło dokumentacja FreeRTOS

TBD - Przenieść część poniższych informacji na Schematy blokowe

4.2.3 Pierwsze uruchomienie

Przy pierwszym uruchomienu zamka, które następuje automatycznie po podłączeniu zasilania do układu, wykonywana jest procedura przejścia w stan głębokiego uśpienia (Deep-sleep mode). W tym celu jako sposób wybudzenia konfigurowany jest tryb EXT0 (External Wakeup 0). Tryb ten wymusza aby po przejściu w stan uśpienia podtrzymane zostało zasilanie peryferiów RTC (ang. *Real-Time Clock*, zegar czasu rzeczywistego) [4], co z kolei pozwala na konfigurację źródła wybudzającego przerwania zewnętrznego jako wybranego wejścia RTC GPIO. W projekcie w tym celu wykorzystany został pin nr 26. Ze względu na charakterystykę wykorzystanego źródła przerwania (pasywny czujnik zbliżeniowy), konieczne było zastosowanie trybu pulldown dla wspomnianego wyżej wejścia aby zapobiec występowaniu na nim stanu nieokreślonego. Po konfiguracji źródła przerwania układ zostaje wprowadzony w stan uśpienia.

4.2.4 Wybudzenie z głębokiego uśpienia

Podzielić blok tekstu na części bardziej skupiające się na poszczególnych komponentach - TBD

Po wykryciu ruchu, czujnik PIR generuje stan wysoki na linii wybudzającej mikrokontrolera, co wywołuje procedurę wyjścia z głębokiego uśpienia. Po wybudzeniu następuje inicjalizacja systemu obsługi zdarzeń, wykorzystywany przez wszystkie wątki jako główne narzędzie sygnalizacji postępu i synchronizacji dostępu do danych. Po przygotowaniu mechanizmu sygnalizacji zdarzeń komponent sterujący inicjuje współbieżną inicjalizację komponentów RFID oraz WiFi.

Inicjalizacja komponentu WiFi może przebiec na dwa sposoby. Jeśli następuje po raz pierwszy od czasu podłączenia zasilania do układu, wymagane jest wykonanie konfiguracji sterownika WiFi dostarczanego przez ESP-IDF. W tym celu ustawiane są dane dostępu do sieci (SSID i hasło, osadzone w bezpośrednio w oprogramowaniu) a sterownik przestawiany jest w tryb station, pozwalający mikrokontrolerowi na nawiązanie połączenia z punktem dostępu sieci bezprzewodowej. Następnie dane konfiguracji zapisywane są w pamięci nieulotnej. Alternatywny scenariusz następuje przy każdej następnej inicjalizacji. Wykorzystuje on dane zapisane podczas poprzedniej inicjalizacji w celu odtworzenia konfiguracji sterownika WiFi.

Wspomniana wyżej konfiguracja zostaje następnie wykorzystana do nawiązania połączenia z siecią bezprzewodową. Następnie następuje próba zestawienia bezpiecznego połączenia z serwerem, opartego na mechanizmach TLS i wzajemnego uwierzytelnienia (ang. *Mutual authentication*). Po udanym zestawieniu połączenia z wykorzystaniem uścisku dłoni TLS (ang. *TLS Handshake*) program przechodzi do pętli obsługi żądań transmisji pochodzących od pozostałych komponentów.

Na listingu 4.1 przedstawiono pseudokod pętli obsługi żądań w komponencie WiFi.

Zmodyfikowac ten pseudokod, zeby byl bardziej czytelny i zrozumialy dla kogos nieznajacego wewnetrznej struktury i specyfiki kodu

Listing 4.1: Pseudokod pętli obsługi żądań w komponencie WiFi

```
def start_transmission_request_handling():
    # Klient WiFi powinien zaczekac na polaczenie z siecia
    wait_event(WIFI_CONNECTED)

    while (TRUE):
        status = wifi_socket_connect()
        if (status != SUCCESS):
            break
        # Glowna petla obslugi zadan
        while (TRUE):
            # Czekaj na zadanie transmisji przychodzacej lub wychodzacej
            wait_event(SEND_PENDING or RECEIVE_PENDING)
            if (event_pending(SEND_PENDING)):
                # Przyjeto zadanie obslugi transmisji wychodzacej
                status = wifi_socket_send(data_to_send)
                if (status != SUCCESS):
                    # Zasygnalizuj blad transmisji
                    notify_event(TRANSMISSION_FAIL)
                    break
            else:
                # Przyjeto zadanie obslugi transmisji przychodzacej
                status, received_data = wifi_socket_receive_data()
                if (status != SUCCESS):
                    # Zasygnalizuj blad transmisji
                    notify_event(TRANSMISSION_FAIL)
                    break
            # Zasygnalizuj sukces transmisji
            notify_event(TRANSMISSION_SUCCESS)
        wifi_socket_shutdown()
```

Komponent RFID bazuje na zewnętrznej bibliotece [5]. Inicjalizacja rozpoczyna się od inicjalizacji interfejsu SPI na określonych z góry wyjściach mikrokontrolera. Następnie rozpoczyna się cykliczne odpytywanie czytnika rc522 za pomocą wspomnianego interfejsu [5]. W przypadku sygnalizacji wykrycia karty przez czytnik, pozyskiwany jest numer karty, odpytywanie zostaje zakończone a numer karty zostaje przekazany do komponentu sterującego. **TBD**

sterowanie po przylozeniu karty TBD Po stworzeniu wątków odpowiedzialnych za wykonanie kodu komponentów WiFi i RFID komponent sterujący rozpoczyna oczekiwanie na sygnalizację wykrycia karty przez komponent RFID. Gdy to nastąpi, odczytany zostaje pozyskay numer karty. W celu wymiany danych z serwerem, wykorzystywany interfejs udostępniany przez komponent WiFi w postaci funkcji do transmisji wychodzącej i przychodzącej **dodac pseudokod funkcji żądania transmisji, uszczegółowić**. Zależnie od informacji zwrotnej od serwera, następuje przyznanie

dostępu, lub jego odmowa i ponowne przejście w stan uśpienia.

4.3 Wykorzystane technologie

4.3.1 ESP32

ESP32-DevKitC jest produkowaną przez firmę Espressif platformą deweloperską bazującą na module ESP32-WROOM-32D. Sercem modułu jest układ z rodziny ESP32 (ESP32-D0WD) wyposażony w CPU (ang. *Central Processing Unit*, centralna jednostka obliczeniowa) o dwóch rdzeniach, z których każdy może być kontrolowany niezależnie [6]. Moduł integruje Bluetooth, Bluetooth Low Energy oraz WiFi, a także szeroki zakres peryferiów: czujniki dotyku, czujniki pola magnetycznego, interfejs karty SD, Ethernet, SPI, UART (ang. *Universal Asynchronous Receiver-Transmitter*), I^2S (ang. *Inter-IC Sound*) i I^2C (ang. *Inter-Integrated Circuit*) [6]. Dodatkowo umożliwia korzystanie z niskoenergetycznego koprocesora Ultra-Low-Power (ang. *ULP co-processor*), podczas gdy główne jednostki pozostają w trybie głębokiego uśpienia [7].

ESP32 oferuje efektywną i elastyczną technologię zarządzania energią. Dokument *ESP32 Series Datasheet* wymienia pięć predefiniowanychstanów energetycznych [8]:

1. Active mode: Aktywne CPU wraz z układem radiowym, możliwa bezprzewodowa transmisja.
2. Modem-sleep mode: Aktywne CPU z konfigurowalnym zegarem. Chip radiowy w tym trybie pozostaje wyłączony.
3. Light-sleep mode: Uśpione CPU. Pamięć i peryferia RTC wraz z koprocesorem ULP pozostają aktywne. Jakiekolwiek zdarzenia wybudzające (MAC, host, timer RTC i zewnętrzne przerwania) doprowadzą do wybudzenia układu.
4. Deep-sleep mode: Tylko pamięć RTC i peryferia RTC pozostają zasilone. Dane dotyczące połączeń WiFi i Bluetooth zostają przechowane w pamięci RTC. Opcjonalnie dostępny jest koprocesor ULP.
5. Hibernation mode: Wewnętrzny rezonator kwarcowy o częstotliwośći 8 MHz wraz z koprocesorem ULP zostają wyłączone. Również pamięć RTC jest wyłączona. Wybodzenie możliwe jest tylko poprzez timer RTC lub predefiniowane wejścia RTC GPIO.

Jest to szczególnie istotne z punktu widzenia wymagań dotyczących poboru energii.

4.3.2 Czytnik MFRC522

Do realizacji komunikacji w standardzie RFID High Frequency (13,56 MHz) wykorzystany został zintegrowany odbiornik/nadajnik MFRC522 produkowany przez firmę NXP Semiconductors, umożliwiający bezprzewodową komunikację z kartami zgodnymi ze standardem ISO/IEC 14443 A/MIFARE. Układ wspiera komunikację poprzez interfejsy SPI, UART oraz I^2C [9].

4.3.3 Czujnik zbliżeniowy

TBD

4.4 Bezpieczeństwo

Tutaj opisać mechanizmy wykorzystane w celu zapewnienia bezpieczeństwa systemu - TBD

Ważnym aspektem opracowanego systemu jest bezpieczeństwo. Bezpieczeństwo systemu tworzą następujące składowe:

1. Bezpieczeństwo zastosowanego mikrokontrolera
2. Bezpieczeństwo komunikacji bezprzewodowej
3. Bezpieczeństwo serwera

4.4.1 Bezpieczeństwo mikrokontrolerów

TBD

4.4.2 Bezpieczeństwo komunikacji bezprzewodowej

TBD

4.4.3 Bezpieczeństwo serwera

TBD

4.5 Problemy

Konfiguracja WiFi? Sygnalizacja stanu baterii? zarządzanie stanami energetycznymi Enkrypcja flash

4.5.1 Zarządzanie energią

Szacunki długości życia na baterii - TBD Teoretyczne zużycie podawane w specyfikacji komponentów Pomiary poboru mocy w różnych fazach działania - TBD

Rozdział 5

Serwer

Rozdział 6

Podsumowanie

Niniejszy rozdział dokonuje podsumowania rezultatów prowadzonej pracy. Porównuje szacunki dotyczące poboru mocy z rzeczywistym poborem. Ponadto dokonuje oceny bezpieczeństwa oraz responsywności systemu.

W ramach niniejszej pracy stworzono funkcjonalny prototyp systemu, który po odpowiednich modyfikacjach miałby szansę efektywnej pracy w odpowiednim środowisku.

Dzięki wykorzystaniu zdalnego serwera do przeprowadzenia procesu uwierzytelniania system zapewnia większą elastyczność i łatwość zarządzania niż alternatywne systemy wykorzystujące zamki pracujące w sposób autonomiczny.

Rozwiązanie cechuje się wygodą montażu, ponieważ nie wymaga przewodów zasilających i komunikacyjnych prowadzonych w ścianach budynków. Przy wdrażaniu rozwiązania nie jest konieczna modyfikacja istniejącej infrastruktury budynku, z wyjątkiem wymiany samych zamków. System nie wymaga żadnych dodatkowych komponentów sprzętowych poza zamkami i serwerem.

Wydajność energetyczna podsystemu sterowania zamkiem została osiągnięta przez zarządzanie zasilaniem jego peryferiów oraz kontrolę stanu zasilania mikrokontrolera w celu minimalizacji poboru mocy i maksymalizacji czasu pracy na zasilaniu bateryjnym.

Bezpieczeństwo systemu na wielu poziomach zapewnia wykorzystanie mechanizmów takich jak TLS w warstwie komunikacji pomiędzy zamkiem a serwerem czy szyfrowanie pamięci Flash w warstwie operacji na danych w mikroprocesorze w układzie zamka.

Wydajność energetyczna - TBD

Responsywność - TBD

Bezpieczeństwo - TBD

6.1 Możliwe rozszerzenia

W ramach niniejszej pracy stworzony został prototyp rozwiązania. Niektóre planowane funkcjonalności nie zostały zaimplementowane ze względu na ograniczenia czasowe i budżetowe.

Pozostawiono jednak możliwość rozbudowy systemu. Poniżej przedstawiono kilka problemów, których system w obecnym stanie nie adresuje, wraz z możliwymi rozwiązaniami.

6.1.1 Mechanizm wyjścia

Opisywane rozwiązanie nie obejmuje implementacji mechanizmu opuszczenia strefy chronionej systemem kontroli dostępu. W zależności od potrzeb końcowego użytkownika, możliwe rozwiązanie to montaż dodatkowego czytnika po przeciwnej stronie drzwi i połączenie go z kontrolerem wejścia w przypadku gdy wymagana jest obustronna kontrola dostępu bądź zastosowanie przycisku którego naciśnięcie powoduje zwolnienie zamka w przypadku gdy wymagana jest tylko kontrola wejścia do chronionego obszaru.

6.1.2 Obsługa większej liczby zamków

W celu umożliwienia obsługi przez system liczby zamków przekraczającej 1, wystarczająca byłaby modyfikacja podsystemu autoryzacji w taki sposób, aby mógł on obsługiwać równoległe żądania od klientów.

6.1.3 Wygodna konfiguracja parametrów sieci

W obecnej implementacji dane dostępu do sieci (nazwa sieci oraz hasło) zostały zagnieżdżone w oprogramowaniu konrolera. Zmniejsza to elastyczność konfiguracji urządzenia, wymagając jego przeprogramowania za każdym razem gdy zmianie ulegnie nazwa lub klucz dostępu do sieci.

Możliwym rozwiązaniem tego problemu byłaby implementacja trybu konfiguracji. Tryb ten powodowałby przejście kontrolera w tryb Access Point przy zachowaniu dwóch warunków: (1) nastąpiło uruchomienie, a nie wybudzenie z trybu głębokiego uśpienia oraz (2) na określonym wejściu pojawiło się napięcie. Przejście kontrolera w tryb Access Point umożliwiłoby udostępnienie prostego interfejsu webowego, za pomocą którego administrator systemu mógłby wprowadzić niezbędne do działania dane, takie jak nazwa sieci, hasło, a także adres IP i numer portu serwera autoryzacji. Aby zachować wysoki poziom bezpieczeństwa, komunikacja pomiędzy urządzeniem administratora i kontrolerem powinna odbywać się przy wykorzystaniu protokołu TLS.

6.1.4 Sygnalizacja stanu baterii

W obecnym stanie system nie implementuje mechanizmów informowania serwera o swoim stanie energetycznym. Sygnalizacja stanu baterii umożliwiłaby administratorowi systemu bieżące monitorowanie wszystkich zamków objętych systemem oraz szybką reakcję w przypadku, gdy baterie wymagałyby wymiany. Modyfikacja ta wymagałaby uzyskania dostępu do danych na temat naładowania baterii przez mikrokontroler oraz przesyłania ich okresowo do serwera, najlepiej w momentach, gdy jest on już wybudzony z powodu wykrycia ruchu w jego otoczeniu.

6.1.5 Obsługa kont użytkowników w podsystemie zarządzania

Podsystem zarządzania nie implementuje mechanizmu dostępu do zasobów, co czyni system mniej bezpiecznym. W końcowym produkcie należałoby rozszerzyć go o możliwość tworzenia kont użytkowników i przypisywania im określonych uprawnień co do odczytu i zapisu danych.

Bibliografia

[1] Polski Komitet Normalizacyjny, *Technika informatyczna. Zabezpieczenia w systemach informatycznych. Terminologia*, Marzec 2002.

[2] British Security Industry Association, *A specifier's guide to access control systems*, Kwiecień 2016.

[3] Roger sp. z o.o. sp. k., *Przewodnik po systemie RACS 5.* v5.3 (dostęp 20.10.1019).

[4] Espressif Systems, *ESP32 Api Reference, Sleep Modes*, 2019. (rewizja a45e9985).

[5] A. Bobija, "C library for interfacing esp32 with mfrc522 rfid card reader." Github, Październik 2019. https://github.com/abobija/esp-idf-rc522 (rewizja 557af67).

[6] Espressif Systems, *ESP32-WROOM-32D & ESP32-WROOM-32U Datasheet*, 2018. Version 1.7.

[7] Espressif Systems, *ESP32 Technical Reference Manual*, 2018. Version 4.0.

[8] Espressif Systems, *ESP32 Series Datasheet*, 2019. Version 3.2.

[9] NXP Semiconductors, *MDRC522 Product Datasheet*, 2016. Version 3.9.

Dodatki

Dodatek A

Uruchomienie projektu

TBD