Analiza securității și implementarea

sistemelor de acces fără cheie

Săliștean Andrei

|  |
| --- |
| UNIVERSITATEA DE MEDICINĂ, FARMACIE, ȘTIINȚE ȘI TEHNOLOGIE "GEORGE  EMIL PALADE" DIN TÂRGU-MUREŞ  FACULTATEA DE INGINERIE ȘI TEHNOLOGIA INFORMAȚIEI    Specializarea: Informatică Viza facultății  TEMĂ LUCRARE DE LICENȚĂ      Coordonator științific: Candidat (a): Săliștean Andrei  Conf. dr. Haller Piroska Anul absolvirii: 2024 |
| **Tema lucrării de licență: Analiza securității și implementarea sistemelor de acces fără cheie** |
| **Problemele principale tratate :**   * Analiza vulnerabilităților în automobile. * Arhitectura și funcționalitățile sistemelor de acces fără cheie.   Proiectarea sistemului.   * Implementarea, testarea și validarea sistemului. * Testarea și validarea funcționalităților de securitate.   **Bibliografia recomandată :**   * Oswald, David F. "Wireless attacks on automotive remote keyless entry systems." Proceedings of the 6th international workshop on trustworthy embedded devices. 2016. * Csikor, Levente, et al. "Rollback: A new time-agnostic replay attack against the automotive remote keyless entry systems." ACM Transactions on Cyber-Physical Systems 8.1, 2024. * Zheng, Yong, et al. "Automotive Security in the Digital Era: A Comprehensive Survey of Attacks and Defenses for Keyless Entry System." International Conference on Computational & Experimental Engineering and Sciences.   Cham: Springer Nature Switzerland, 2024. |
| **Termene obligatorii de consultații : Săptămânal** |
| **Locul și durata practicii : Laboratoarele Universității** |
| **Primit tema la data de : 01 mai 2023** |
| **Termen de predare : 05 iulie 2024** |
| **Semnătura directorului de departament Semnătura conducătorului**  A black and white logo  Description automatically generated with medium confidence |
| A black line drawing of a letter  Description automatically generated **Semnătura candidatului** |

**Cuprins**

1. **Introducere**

**1.1 Evoluția și perspectivele sistemelor de securitate in industria închiderii centralizate**

* 1. **Ce este un sistem keyless entry keyless go**

**1.3 Cum funcționează**

1. **Analiza și securitatea sistemelor “Keyless Entry”**

**2.1 Analiza sistemelor**

**2.2 Posibile vulnerabilități**

2.2.1 Atacuri de tip relay

2.2.2 Atacuri de tip “jamming”

2.2.3 Atacuri de tip “man-in-the-middle”

2.2.4 Atacuri de tip replay

2.2.5 Vulnerabilități software

2.2.6 Vulnerabilități fizice

**2.3 Modele existente pe piața**

2.3.1 Modele bazate pe unde radio

2.3.2 Sisteme bazate pe BLE (Bluetooth Low Energy)

2.3.3 Sisteme bazate pe UWB (Ultra Wide Band)

**2.4 Componente**

2.4.1 Cheia electronica ( Smart Key)

2.4.2 Antenele vehiculului

2.4.3 Modulul de control (Body Control Module -BCM)

2.4.4 Sistemul de criptare și autentificare/TPM

1. **Proiectare și structurare**

3.1 Proiectare și cerințele sistemului

3.2 Securitatea sistemului și protocol de comunicații

3.3 Bluetooth Low Energy si Bluetooth 6.0

3.4 Algoritm si diagrame de funcționare

1. **Implementare**

4.1 Proiectare și implementare hardware , PCB

4.2 Programarea modulului ESP32

4.3 Programarea Modulului Arduino Mega

4.4 Programarea clientului BLE

4.4.1 ESP32

4.4.2 Android

1. **Rezultate și concluzii**
2. **Bibliografie**

**Abstract**

Istoria securității și a sistemelor antifurt a mașinilor este una destul de scurtă și la obiect, cu scopul constant de a preveni persoanele nedorite de a avea acces la autovehiculele care nu le aparțin, totodată fără a fi o piedică pentru proprietari când doresc să își folosească mașina. Aceasta istorie , pe cât de simplă ,se întinde pe o perioadă lungă de timp, de mai bine de 130 de ani .

Pornind de la simpla nevoie a unor cunoștințe avansate pentru a porni o mașină , fără sisteme fizice adiționale care să prevină pornirea și operarea, la nevoia unei chei fizice care prin introducerea și rotirea acesteia într-un cilindru se reușea pornirea mașinii, la simpla apăsare de la distanță a unui buton pentru a debloca portierele, iar in cele din urmă la simpla nevoie de a avea o cartela in buzunar pentru a debloca și porni mașina cu o simpla apăsare de buton.

In prezenta lucrare s-au încercat cu succes realizarea unu sistem acces fără cheie și pornire fără cheie (“ Keyless entry , Keyless Go”) pentru orice mașină care nu a beneficiat la vremea apariției acesteia de asemenea sisteme de comfort. Folosind pe post de cheie obiectul folosit zilnic de fiecare persoana, telefonul mobil , a microcontroller-elor care mulțumită avansului tehnologic au ajuns astăzi sa se comercializeze la prețuri extrem de reduse si a câtorva componente hardware pentru a interacționa cu interfața mașinii s-a realizat un sistem keyless entry, keyless go, universal , low cost care aduce câteva îmbunătățiri fata de majoritatea sistemelor de genul care se regăsesc pe mașinile din comerț, cum ar fi deblocarea/ blocarea automata in funcție de distanța fără a trebui atingerea mânerului înainte , aprinderea farurilor automata pe timp de noapte pentru localizare mai ușoară a mașinii și control total asupra mașinii din aplicație pentru pornirea / oprirea motorului , blocare/deblocare și pornire oprire faruri, cu posibilitatea extinderii acestui set de acțiuni prin simpla adăugare a componentelor hardware care să interacționeze cu interfața noului sistem pe care dorim sa îl controlam, toate prin implementarea unui protocol de comunicații ce ne oferă protecție împotriva atacurilor cibernetice.

**Capitolul 1**

**Introducere**

* 1. **Evoluția și perspectivele sistemelor de securitate in industria închiderii centralizate**

Apariția primului automobil in 1885 proiectat si construit de Carl Benz ,patentat în 1886 , patent ce poate fi considerat certificatul de naștere a automobilului[1], a reprezentat o viziune spre viitor , un viitor ce sa profite la maxim de puterea pusa la dispoziție de motorul cu combustie interna , viitor însă de care mulți erau sceptici.

Prima călătorie pe o distanță lunga cu un asemenea autovehicul a fost realizata de Bertha Benz in 1888 cu scopul de a ajunge in Pforzhein, locul ei de naștere , pornind din Mannheim , urmând ca invenția soțului acesteia sa parcurgă cu succes 180 Km , însă nu fără câteva probleme care necesitau atenție constantă , cum ar fi monitorizarea nivelului apei folosite pentru răcirea propulsorului si a combustibilului, oprind in sate și farmacii pentru reaprovizionare cu apa și benzină. Efortul depus de Bertha a arătat lumii utilitatea unui asemenea vehicul , urmând ca in următorii ani industria auto să ia amploare .

Automobilele ieșite pe porțile fabricilor în următorii ani au ajuns să se desprindă de conceptele primitive a autovehiculului original cu privire la întreținerea acestora și comfortul pasagerilor , însă chiar și așa pornirea și manipularea unei asemenea mașinării nu era tocmai “floare la ureche”, având nevoie de o pregătire speciala minima pe care nu toata lumea era dispusa sa o facă , din pricina faptului ca autovehiculele erau considerate încă un lucru pe care doar boierii și-l permiteau. Prin urmare , furtul unui autovehicul nu făcea parte din lista de îngrijorări a producătorilor si a companiilor terțe ce furnizau componente[2] .

Siguranța sau invulnerabilitatea autovehiculelor la furt avea să se schimbe cu creșterea în popularitatea acestora , cauza pentru numărul crescut al persoanelor ce aveau cunoștințele necesare de a porni și in urma , fura un autovehicul. Sistemul de pornire “clasic “ pe care îl cunoaștem , ce necesita o cheie pentru a roti butucul ce închide circuitul electric, necesar pentru funcționarea sistemului de aprindere urma sa apară in 1911 cu brandul Bosch ștanțat pe acesta, iar până atunci existau alternative pentru a bloca rotirea butucului de persoanele nedorite , prin plasare unui lacăt cilindric peste butuc ce putea fi dat jos doar cu o cheie specifică. De atunci și până acum, dorința crescută a consumatorilor de a integra mașinile în viața lor de zi cu zi, totodată confortul si ușurința de a conduce dorindu-se a fi cât mai mari , au dus la portiere ce pot fi blocate cu ușurință cu aceeași cheie cu care se pornește motorul, cheie cu telecomanda pentru blocarea și deblocarea de la distanta iar cea mai recentă evoluție a sistemului de blocare centralizată fiind accesul mâini libere și plecare fără chei (“keyless entry, keyless go”).

* 1. **Ce este un sistem keyless entry, keyless go**

Sistemul de acces mâini libere și pornire fără chei sau cunoscut in engleza sub denumire de *Keyless entry & keyless go* își face apariția pe piața auto sub umbrela celor de la Siemens ce in 1995 scot la iveala acest sistem revoluționar , iar Mercedes-Benz ce îl utilizează pentru prima data in 1998 pe w220 S-Class[3]. Acest sistem vine ca o îmbunătățire a sistemului dezvoltat de Paul Lipschultz in 1981 utilizat prima oara pe Renault Fuego ce utiliza o cheie codata în pulsuri inflarosii pentru a comunica cu mașina. Technologie revoluționară la acea vreme dar deschiderea și închiderea mașinii presupunea interacțiunea cu cheia ce integra transmiterul infraroșu , o acțiune voita și activa din partea utilizatorului. Sistemul creat de Siemens elimina acest pas in plus , acesta blocând sau deblocând mașina în funcție de starea curenta de blocare în care se afla mașina prin simpla acțiune de atingere a mânerului portierei . Pasul următor fiind deschiderea portierei și apăsarea unui simplu buton pentru pornirea motorului, care firește era posibila doar daca cheia se află în interiorul mașinii.

* 1. **Cum functioneaza**

În sistemul celor de la Siemens, comunicarea se realiza prin unde radio de o frecvență cuprinsă între 125 KHz și 300 MHz[3], unde radio captate și trimise de cheie și antenele prezente în ușa șoferului, cu antene opționale prezente în restul ușilor pentru diferite funcționalități (ex. Deschiderea doar a portbagajului când cheia este în spatele mașinii). În momentul în care utilizatorul dorește să intre în mașină, involuntar atinge un senzor plasat pe mânerul ușii, moment în care mașina începe comunicarea cu cheia. Cheia recepționează semnalul și transmite mașinii cheia, antenele urmează să transmită această cheie modulului de confort care verifică dacă cheia este cea corectă, în urma căreia blochează sau deblochează mașina.  
Această metodă inițiază comunicarea cu cheia doar în momentul în care utilizatorul este deja lângă mașină, ceea ce nu permite, spre exemplu, pornirea farurilor pe timp de noapte pentru a localiza mai ușor mașina. Prin urmare, în această lucrare vom implementa un sistem asemănător cu cel de pe automobile Renault, în care comunicarea este începută cu mult înainte ca utilizatorul să fie apropiat de mașină, și care deschide/închide mașina și farurile în funcție de intensitatea pachetelor transmise între mașină și cheie.

**Capitolul 2**

**Analiza si securitatea sistemelor “Keyless Entry”**

**2.1 Analiza sistemelor**

Pe măsură ce numărul de autovehicule prezente pe străzi a crescut, iar prețul acestora a devenit din ce în ce mai mic, anii ’70 au marcat un adevărat boom în industria auto, înregistrându-se cele mai mari vânzări de automobile din istorie [5]. Acest fenomen a fost favorizat de un boom economic înregistrat în țările dezvoltate, ceea ce a dus la creșterea accesibilității automobilelor pentru clasa de mijloc.

A graph showing a line

Description automatically generated

Graficul 2.1. Numărul de mașini vândute în ultimii 70 de ani

Sursa : https://www.statista.com/statistics/199974/us-car-sales-since-1951/

În această perioadă de creștere exponențială a numărului de autovehicule pe străzi, producătorii au început să acorde atenție principiilor de confort și accesibilitate a autovehiculelor, care erau în mare parte absente. Astfel, începând din anii ’70, au apărut primele mașini cu închidere centralizată, cu un buton amplasat în interior ce deschidea sau închidea toate portierele cu ajutorul unor solenoizi amplasați în fiecare ușă. Cu această funcție în plus, accesibilitatea a crescut, dar până a ajunge la acel buton tot trebuia introdusă cheia în broasca ușii șoferului pentru a deschide portiera acestuia, acțiune și mai dificilă pe timp de noapte. Astfel, în anii ’80 au început să apară primele mașini cu deschidere/închidere centralizată din telecomandă.

Prima mașină de acest fel a fost lansată în 1982 sub umbrela celor de la Renault, sub denumirea de Fuego. Aceasta venea cu o telecomandă cu infraroșu ce transmitea o cheie unică către receptorul din mașină. Deși revoluționară în acea perioadă, telecomanda avea nevoie de o linie directă și fără blocaje optice către mașină pentru a funcționa corespunzător.

A red car with black text

Description automatically generated

Fig. 2.1 Renault Fuego

Sursa: <https://jalopnik.com/gateway/michael-schumacher-made-history-in-this-ferrari-and-now-1851700802>

Observând potențialul masiv de confort al sistemelor IR producătorii au trecut pe tehnologii ce aveau la baza undele radio (RF), permițând transmiterea codurilor de acces de la o distanta mai mare , si fără a fi nevoit de a avea o linie directa de vedere către mașina[3] , iar in cele din urmă la primul sistem keyless entry în 1995.

Gama tehnologiilor utilizate pentru a ușura accesul utilizatorilor în mașini a avut un parcurs vast, începând de la sisteme inexistente, trecând prin infraroșu și culminând cu tehnologii wireless precum Bluetooth Low Energy [20], RF sau Ultra-Wideband [19].

Sistemele cu unde radio (RF) sunt utilizate sî în ziua de astăzi pentru mașinile cu închidere centralizata simplă, fără a încerca deschiderea automată pe baza distanței deoarece integritatea semnalului (RSSI) depinde foarte mult de factorii externi , rezultând într-un comportament nepredictibil , iar în sistemele ce implementau funcționalitățile keyless entry, cheia începea comunicare cu mașina , făcând un schimb de chei , ușa urmând sa se deschidă doar în momentul în care utilizatorul atingea portiere în care se afla un senzor pentru detectarea mașini.

Cea mai nouă tehnologie însă este Ulta-Wideband (UWB) , utilizată în trackerele pentru lucruri personale si detectabile cu telefonul mobil, produse de Samsung și Apple, folosita în mașini din 2019 permite și deschiderea portierelor în funcție de distanța ce poate fi măsurata cu precizie de acest sistem , însă acesta implica un cost mai mare din cauza senzorilor și modulelor in plus ce trebuie instalate . De cele mai multe ori aceasta tehnologie nu este utilizată de una singură , având și Bluetooth Low Energy la baza pentru o comunicare rapida, eficienta, cu un consum redus de energie ce este prezentă pe majoritatea dispozitivelor cu capabilități Bluetooth . Profitând la maximi de acesta răspândire masivă a Bluetooth Low Energy fiind Tesla , ce nu necesită de vreo cheie proprietară pentru acces , ci doar de un telefon mobil cu aceste două tehnologi ( Bluetooth Low Energy si UWB ) iar dacă UWB este absent , se poate utiliza doar BLE cu un grad mai mic de precizie.

Avantajele acestor sisteme încep de la confortul crescut din lipsa nevoii de a scoate din buzunar cheia pentru deblocare și pornire, la o securitate avansată, față de cheile cu blocare normală cu unde radio ce nu încep un handshake pentru a asigura autenticitatea partenerului de comunicații , de cele mai multe ori utilizând doar un rolling code , care și acesta poate fi atacat cu succs , iar in cele mai multe cazuri nici măcar atât.

Prezența tehnologiilor wireless prezente pe dispozitivele mobile aduc un plus la capitolul confort acestea putând comunica cu mașina fără a fi nevoie de o cheie fizica pentru transmiterea comenzilor de acces, acestea fiind trimise de telefon alături de alte comenzi pentru pornirea motorului pentru încălzire sau vizualizarea stării autovehiculului .

Totul sună bine și frumos, nu ? Ei bine adevărul este ca în realitate lucrurile nu sunt așa roz precum par . Aceste sisteme pot fi un lucru bun pentru oameni însă trebuie să ne gândim și la minusurile care vin cu acestea . Să începem cu începutul, toate aceste module în plus consuma o cantitate mica de energie, dar totuși consumă, ceea ce se poate aduna la o cantitate semnificativa dacă vehiculul stă o perioada mai lungă de timp fără a fi pornit , în timp descărcând bateria complet . Totodată prezența acestor module aduc costuri suplimentare care pot creste prețul automobilului cu până la 1500 de euro la mărcile premium și 750 la cele axate pe mașini de budget.

Aceste minusuri nu sună atât de rău daca dispui de banii necesari și circuli regulat se poate trăi confortabil cu această opțiune însă cel mai mare minus este cel legat de securitate . O mașina cu un astfel de sistem poate fi furată destul de ușor dacă dispui de un echipament mai specializat, care cu o căutare pe internet poate fi replicat destul de ușor . Deși mașinile produse de tesla vin echipate cu sisteme keyless entry ce folosesc UWB , nu utilizează în mod eficient tehnologia pentru verificarea distanței, care ar putea preveni atacurile de tip relay. Acest lucru se datorează faptului că sistemele Tesla keyless entry folosesc în principal Bluetooth pentru a debloca mașina și a controla imobilizatorul [12].

**2.2 Posibile vulnerabilități**

Din 2014, când sistemele keyless entry au devenit din ce mai populare printre oameni și disponibile pe tot mai multe autovehicule , și până în ziua de azi numărul de mașini furate a început sa crească [6] din cauza echipamentelor specializate care au devenit din ce in ce mai accesibile , care pot fi folosite cu scopuri malițioase . Cu un modul ESP32 ,care poate fi găsit pe piață de la 10 euro , și un script micuț se pot bruia sistemele ce folosesc la baza Bluetooth Low Energy.

Aceste vulnerabilități au apărut din pricina naturii comunicațiilor wireless, limitările acestora si complexitatea componentelor electronice. In acest grup nefavorabil există 6 tipuri distincte de atacuri: Atacuri de tip relay, jamming , man-in-the-middle, replay , vulnerabilități software și fizice.

A graph with a line going up

Description automatically generated

Grafic 2.2 Numărul de mașini furate

Sursa: <https://www.statista.com/statistics/191216/reported-motor-vehicle-theft-rate-in-the-us-since-1990/>

**2.2.1Atacuri de tip relay**

Aceste tipuri de atacuri sunt printre cele mai cunoscute metode de compromitere a sistemelor keyless entry sau a cardurilor de plată contactless [11], ce au ca principiu de funcționare captarea semnalului emis de cheie și re emiterea acestuia aproape de mașina astfel eronând sistemul în a crede faptul că în apropierea mașinii se află cheia . Acest tip de atac are nevoie de doar 2 componente principale , un receptor , care captează semnalul cheii, și un emițător care îl reproduce aproape de mașină, aceste două componente fiind conectate între ele fie via o tehnologie wireless precum wi-fi sau prin cablu, în 2017 reușindu-se un asemenea atac cu echipamente în valoare de doar 225$ [7] .

**Studiu de caz.**

În 2017 un studiu realizat de ADAC ( Automobil Club German) [8] pe un număr considerabil de mașini, a arătat faptul că marea majoritatea autovehiculelor sunt vulnerabile la un atac de tip relay . Printre nefericiții participanți la acest studiu se numără și mărci premiu precum Volkswage , BMW și Audi, mașini care au putut fi deschise și pornite cu cele doua module menționate mai sus fără a avea acces la cheia autovehiculului, aceasta aflându-se la o distanță considerabilă de mașina, în casa proprietarului.

Printre cele mai vulnerabile sisteme au fost cele ce funcționau la frecvente de 125 kHz sau 433 MHz, iar principalele vulnerabilități fiind lipsa criptării semnalului sau autentificarea bazată pe distanță . În ciuda avansului tehnologic ADAC a arătat vulnerabilitatea sistemelor , sugerând producătorilor să utilizeze tehnologii bazate pe Ultra-Wideband , iar utilizatorilor ce dețin o mașină testată , prezentă în lista completă [9] să își țină cheia în casa într-o cușcă “Faraday” care blochează undele radio.

* + 1. **Atacuri de tip “jamming”**

Acest tip de atac are nu are ca scop încercarea de a comunica in mod direct sau

indirect cu mașina , ci să blocheze unele comenzi trimise de către cheie. În cel mai grav scenariu acest atac poate bloca semnalul ce conține comanda de blocare , lăsând mașina deschisa atacatorului pentru a fura conținutul acesteia , dar in cele mai multe cazuri , când utilizatori așteaptă sa primească confirmarea blocării mașinii prin aprinderea semnalizărilor , nu este nimic mai mult decât o simpla inconveniență . Sistemele ce utilizeaz o singura frecventa pentru comunicare, însă cele mai avansate sau cele bazate pe Bluetooth Low Energy , utilizează mai multe frecvențe pentru comunicare , făcând imposibil scenariul in care comunicarea dintre cheie si mașină să nu poată fi optima .

* + 1. **Atacuri de tip “man-in-the-middle”**

Cel mai periculos si complex atac , man-in-the-middle are ca țintă sistemele ce

utilizează Bluetooth Low Energy sau alte tehnologii fără o criptare adecvata . Un atac de tip „man in the middle” (MITM) este un termen general care descrie situația în care un atacator se poziționează în mijlocul unei discuții între un client și o aplicație, fie pentru a asculta în mod clandestin, fie pentru a imita una dintre părți, făcând să pară că un schimb normal de informații este în desfășurare[13]. Un asemenea atac utilizează un dispozitiv cu capacități de comunicare wireless (un microcontroller sau chiar un router infectat cu software malițios[15]) ce se amplasează între server și client , astfel , clientul crezând că atacatorul este serverul iar serverul crezând atacatorul este de fapt clientul. În această poziție atacatorul preia toate mesajele dintre cei doi parteneri de comunicare , având abilitatea de a modifica pachetele dintre server și client.

Acest tip de atac nu este unul nou , fiind utilizat de mulți ani pentru a ataca persoane care doresc să acceseze internetul prin intermediul rețelelor publice nesigure , cu ajutorul cărora atacatorii pot efectua atacuri acțiuni malițioase precum :

1. ARP Cache Poisoning
2. DNS Spoofing
3. Session Hijacking
4. SSL Hijacking [15]

De atunci dezvoltându-se mecanisme de protecție împotriva „man-in-the-middle” , autentificarea bidirecțională, atât clientul cât și serverul se validează reciproc înainte de a începe schimbul de date, asigurându-se în acest fel de autenticitatea partenerului de comunicații , toate acestea fiind criptate cu un algoritm de criptare simetric , cum ar fi AES, acest tip de criptare fiind mai puternic împotriva atacurilor ca și criptarea asimetrica, utilizându-se o cheie diferită pentru fiecare sesiune de comunicații, astfel încercarea decriptării folosind o metoda de tip „ Brute force” fiind aproape imposibilă .

* + 1. **Atacuri de tip replay**

Un atac de tip replay este un atac în care transmiterea datelor valide este repetată sau întârziată în mod malițios sau fraudos[14]. În cazul nostru un asemenea atac încearcă captarea semnalelor emise de cheie și retransmiterea lor către vehicul când acesta este blocat , simulând o comanda validă de deschidere a portierelor, astfel permițând accesul neautorizat în vehicul sau chiar pornirea motorului fără prezența cheii fizice. Deși acest atac este unul simplu cu efecte grave , metodele de protecție împotriva acestuia sunt simple. Prin adăugarea unui contor al pachetelor trimise , sistemele devin mai rezistente la acest atac , aceasta metoda nefiind singura implementată și folosită în industrie. Cea mai răspândită metodă este cea a „Rolling code”-urilor, utilizându-se un cod diferit pentru fiecare comandă de același fel, acest cod fiind preluat dintr-o listă prestabilită între cheie și mașină , însă această metodă poate fi vulnerabilă la atacuri combinate de tip jam-listen-replay[18] ce , după denumire, blochează semnalul cheii către mașină , îl ascultă și îl re-emit ulterior permițând accesul neautorizat, ghiduri complete fiind prezente online ce pun la dispoziție liste de componente dedicate dezvoltatorilor ce pot fi utilizate cu scop malițios [16]. În mașinile care se pot conecta la o rețea GSM se poate folosi o metoda mai bună de a preveni atacurile replay, atașat fiecărui pachet este un timestamp ce arată când a fost creat prima data pachetul , mașina respingându-l dacă acesta este unul din trecut.

* + 1. **Vulnerabilități software**

Exploatând vulnerabilitățile software prezente în cheie sau în BCM (body Control Module), atacatorii pot avea acces la vehicul fără a fi nevoie de a interacționa în vreun fel cu semnalele dintre cheie și mașină . Aceste vulnerabilități ajung pe dispozitive din cauza lipsei testării sau testarea prea puțina a metodelor esențiale pentru securitatea mașinii, vulnerabilități ce nu sunt remediate pentru o perioadă lungă de timp în care persoanele rău intenționate pot profita de acestea . Astăzi, mașinile cu conexiune la telefonul mobil sau la rețea pot dispune de actualizări OTA (Over the air) prin care producătorul poate remedia vulnerabilitățile detectate după livrarea autovehiculelor[17].

Un astfel de exemplu negativ au fost autovehiculele din grupul Kia Hyundai, care din 2011 și până în 2022 au omis adăugarea unui imobilizator electronic sau cel puțin adăugarea unuia care să funcționeze corespunzător pe unele dintre modelele lor , făcându-le unele dintre cele mai furate vehicule din America [21].



Fig. 2.2 Cele mai des furate mărci auto

Sursa: https://www.nicb.org/media/3329/download

Nu a ajutat la creionarea imaginii acestor vehicule nici creșterea în popularitate a platformelor sociale dedicate videoclipurilor de durata scurtă precum TikTok care arătau cât de ușor se poate fura o mașina din acest grup cu doar o șurubelniță și un cablu USB. În cele din urmă Kia și Hyndai au lansat un update care a redus numărul vehiculelor furate cu 50% conform unui studiu [19].

Cu toate acestea , vulnerabilitățile acestor mașini nu s-au limitat doar la imobilizatoare, în septembrie 2024 cercetătorii din domeniul securității descoperind o noua vulnerabilitate în portalul web Kia [20]. Atacatorii puteau prelua controlul asupra comenzilor de blocare/deblocare sau pornire a motorului, comenzi ce sunt valabile și în aplicația Kia Connect ce oferă utilizatorilor acces de la distantă la autovehicul. Pentru a accesa mașina atacatorii aveau nevoie doar de numărul de înmatriculare al acestea. În cele din urmă Kia a remediat această problemă , arătând cât de importantă este integritatea și rezistența la atacuri a sistemelor .

* + 1. **Vulnerabilități fizice**

Vulnerabilitățile sistemelor de acces moderne nu se limitează doar la atacuri cibernetice, bazate pe comunicații wireless. De cele mai multe ori este mai ușor pentru un atacator să forțeze o ușă pentru a avea acces rapid la bunurile uitate în mașini.

O metoda ușoară , nedestructivă pentru a intra în autovehicul este amplificarea semnalului cheii pentru ca mașina sa creadă că în proximitatea ei se află cheia, astfel deschizându-se pentru o persoană neautorizată. Dacă atacatorul este de moda mai veche poate încerca metoda clasica de descifrare a încuietorii cu o unealta de tip „lishi „ ce se poate găsi online foarte ușor . Chiar dacă mașina are un sistem de alarmă atacatorul nu are nevoie doar de câteva secunde pentru a fura bunurile din mașină , iar dacă lipsește un asemenea sistem de alarmă , mașina este mai vulnerabilă . Cu o diagnoză auto , ce se conectează la modulele mașinii prin interfața OBD-2 ( On Board Diagnostic) introdusa din 1996 pe toate automobilele , se poate coda o cheie nouă , atacatorul furând in cel mai elegant mod mașina si putându-se utiliza de aceasta ca și cum ar fi a lui , fără nici o problemă , până la apariția unui control din partea unui echipaj de poliție, dar să sperăm că nu o să se ajungă până acolo.



Fig. 2.3 Decoder Lishi

Sursa : https://www.originallishi.com

**2.3 Modele existente pe piață și comparație**

În momentul de față , pe piață există sisteme ce folosesc o combinație dintre tehnoligiile menționate anterior , pentru a se alinia cu cerințele li bugetul fiecărui model .

**2.3.1 Modele bazate pe unde radio**

Cele mai des întâlnite sunt cele ce se bazează pe unde radio (RF) din pricina costului redus și a disponibilității pieselor , folosite de producători precum:

* Renault (Hands-Free Key Card)
* VolksWagen (Keyless Access)
* BMW (Comfort Access)

Aceasta nu este singura tehnologie utilizată de acești producători, utilizând pe

modelele mai premium și BLE sau UWB dar acest sistem bazat pe unde radio este mai accesibil din punct de vedere financiar făcându-l o alegere bună pentru mașinile de buget , dar care vine cu minusurile lui când vine vorba de atacurile de tip relay sau replay.

**2.3.2 Sisteme bazate pe BLE (Bluetooth Low Energy)**

Sistemele ce utilizează BLE sunt mai diversificate din punct de vedere al dispozitivelor ce pot funcționa pe post de cheie, acestea având posibilitatea de a comunica cu telefonul mobil pentru blocarea/deblocarea mașinii , pornirea motorului sau transmiterea de date referitoare la parametrii mașinii. Producătorii precum :

* Tesla (Passive Key)
* Hyundai (Digital Key)
* Merceds-Benz (Keyless Go)

implementând aplicații care să poată comunica cu mașinile acestora nefiind nevoie de o cheie fizica , în cazul Tesla mașina nevenind cu o cheie din fabrică , doar cu o cartelă rf pentru deblocări in caz de urgență sau pentru prima conectare cu telefonul.

Un asemenea sistem vine cu un mare plus când vine vorba de confort însă dacă implementarea nu este una corectă, care sa respecte niște minime în securitate, poate fi vulnerabil la atacuri de tip „ man-in-the-middle”.

**2.3.3 Sisteme bazate pe UWB (Ultra Wide Band)**

Ultima tehnologie utilizată fiind UWB ( Ultra WIde Band) , tehnologie avansată ce poate determina distanța dintre mașină și cheie pentru funcțiile de blocare/deblocare eliminând posibilitatea unui atac de tip relay ,dar care aduce un cost în plus, aceasta fiind valabilă doar pe modelele de top a producătorilor premium precum Audi, BMW (modele recente) sau Wolkswagen (modele recente) și lipsa integrării cu telefonul mobil deoarece aceasta lipsește pe majoritatea telefoanelor .

**2.4 Componente**

Deși sistemele keyless entry folosesc o gamă larga de tehnologii de comunicare, componentele de bază ale acestora sunt asemănătoare între ele cât și cu cele ale unui vehicul cu deblocare manuală din telecomandă și pornire din butuc.

**2.4.1 Cheia electronică ( Smart Key)**

Cheile electronice reprezintă un avans tehnologic față de cheile mecanice tradiționale, acestea oferind un plus din punct de vedere al securității și un salt înainte când vine vorba de confort sau accesibilitate, cheia electronică oferind mașinilor posibilitatea de a crea o experiență mai plăcută utilizatorilor prin întâmpinarea de la distanță folosind farurile pentru a ilumina împrejurările mașinii sau chiar pornirea motorului pentru a încălzi habitaclul sau a se dezgheța pe timp de iarnă.

Cheile inteligente au parcurs un drum îndelungat , de la sistemele la care era nevoită atingerea mânerului pentru deblocare la cele ce se deschid automat când detectează cheia în apropiere , trecând de la tehnologii ce foloseau Unde radio la cele de ultima generație , ce folosesc tehnologii de comunicare wireless avansate precum BLE , UWB sau o combinație .

Pe cât de avansate sunt aceste chei , în interior sunt pe atât de simple , asemănându-se cu o plăcuță de dezvoltare Arduino sau ESP .

A green circuit board with black circles and small round objects

Description automatically generated

Fig. 2.4 Smartcard Renault Megane

Sursa: <https://www.diagecu.com/index.php?route=product/product&product_id=920>

A green circuit board with many small holes

Description automatically generated

Fig. 2.5 Arduino Nano V3

Sursa : https://store.arduino.cc/en-ro/products/arduino-nano

Ca și componente ale ambelor dispozitive sunt microcontreller-ele ce executa un set prestabilit de instrucțiuni, cristalul oscilator ce rezonează la o frecventa de 433 MHz in cheile din ultimii ani si pinii pentru programarea microcontroller-ului propriu zis . Cheile smart vin cu niște componente în plus , cum ar fi un tpm ce are rolul de a cripta si decripta pachetele dintre cheie și mașină și de a stoca într-un mod sigur cheile criptografice , un modul radio , antene ce realizează transferul propriu zis de date dintre mașină și cheie și un chip NFC (Near Field Communication ) pentru a permite pornirea motorului când bateria cheii este descărcată și comunicarea dintre acestea nu mai este posibilă.

**2.4.2 Antenele vehiculului**

Antenele sunt elementele cele mai simple dar și cele mai importante , fără ele comunicarea dintre mașină și cheie fiind imposibilă . Acestea au capacitatea de a prelua și emite semnale radio ce permit transferul de date dintre cheie și mașină , totodată prin simpla prezență a mai multe antene , plasate strategic în mașină , fiind posibilă aproximarea poziției cheii , nu doar distanța , astfel făcând posibile funcții precum deschiderea portbagajului (dacă mașina dispune de un portbagaj cu deschidere electrică) dacă semnalul din acea zona este mai puternic fata de cel detectat de antenele din portiere.



Fig. 2.6 Antene integrate în plăcuțe PCB

Sursa: https://pcbantennas.com/

Aceste antene nu sunt forate complicate când vine vorba de procesul de fabricație, fiind doar niște trasee de cupru dar partea mai grea apare în faza de proiectare când se decide forma , lungimea și distanța dintre trasee, acești factori determinând frecvențele pe care acea antena le poate recepționa sau emite.

Antele prezente pe piață utilizează frecvențe ce variază ântre 13, 315 MHZ (pentru modulele NFC din interiorul mașinii) până la 10,6 GHz pentru sistemele ce utilizează UWB .

Pentru a determina lungimea antenei avem nevoie să știm la ce frecvență avem nevoie ca antena sa poată emite si detecta undele radio primite de la cheie. Pentru un sistem modern ce are la baza BLE pentru comunicare între cheie și mașină , trebuie să știm frecventa pe care se transfera datele care in cazul nostru este o plaja intre 2,4GHz si 2,48 GHz [25]. Pentru a detecta unde, antena trebuie sa fie cel puțin jumătate din lungimea de undă a frecvenței pentru a putea detecta un vârf și un minim al undei .

![A diagram of a waveform

Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD6RXhpZgAATU0AKgAAAAgABAE7AAIAAAAQAAAISodpAAQAAAABAAAIWpydAAEAAAAgAAAQ0uocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAENoYW5kYSBNYWd1aWdhbgAABZADAAIAAAAUAAAQqJAEAAIAAAAUAAAQvJKRAAIAAAADMjUAAJKSAAIAAAADMjUAAOocAAcAAAgMAAAInAAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAADIwMjE6MDc6MTUgMDk6MzA6MDkAMjAyMTowNzoxNSAwOTozMDowOQAAAEMAaABhAG4AZABhACAATQBhAGcAdQBpAGcAYQBuAAAA/+ELImh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjEtMDctMTVUMDk6MzA6MDkuMjQ3PC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPkNoYW5kYSBNYWd1aWdhbjwvcmRmOmxpPjwvcmRmOlNlcT4NCgkJCTwvZGM6Y3JlYXRvcj48L3JkZjpEZXNjcmlwdGlvbj48L3JkZjpSREY+PC94OnhtcG1ldGE+DQogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgIDw/eHBhY2tldCBlbmQ9J3cnPz7/2wBDAAcFBQYFBAcGBQYIBwcIChELCgkJChUPEAwRGBUaGRgVGBcbHichGx0lHRcYIi4iJSgpKywrGiAvMy8qMicqKyr/2wBDAQcICAoJChQLCxQqHBgcKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKir/wAARCAE0AVwDASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6/9oADAMBAAIRAxEAPwD6RooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiqmmanb6taPcWZYxpcT2x3Lg74pWif8NyNQBbooooAKKKKACiiigArP17Wbfw9oN5q96kr21nGZZRCoLBB1OCR0HJ9hWhUN7aQ39hcWdyu6G4iaKRfVWGCPyNAGWfFWmjxpH4X3SHUJLE3ykKPLMe/bjdn73fGOgzUdh410G9stNuJtQt7BtVYrYwXs8cUtz8+wFF3fNu4IxkkMvc4rz9fCXjSLwmuoraxyeKLa6WGFfPQCS3W1NruLZxgljPtznt14qx4z8KeI2tJNE8NWVx/ZqabaWlqbM2iK/lM2UuHm/e4UY2CPjLNkjOQAemJqunyQrKl9bNG85t1cTKQ0oYqYwc/eDAjb1yCKyNT8a6Ra+H9f1DSr6y1afQ7Sa4ubS2u0ZkaNGbY+3JQkoRyOx44rAfwjqsvjy7QJ5Wg75dUtrgSLlL2WEQFNg5wAZZMnjc4xyOOes/A+tnwfqGn3Nprbalb+GLrSbRbiXT1s3aSNV2RGELIQWRSplAwM5wSaAPTv+Ej0iK8s7G71OyttQvY1kgspblFlkB/uoSC3QjgdjSjxHojaumlLrGnnUZCwSzF0nnNtzuwmdxxtbPHGD6V59d+DdWbxJfrdprk+m6pPZTn+zHsBFGYUiUCUzgSgq8W8eWSMHj5s50k8JagiXLrp8Ynk8WJqQcMm5oA6/vM5/uBhjrjjFAHSX3jfwvpt1cWl74i0qG7t1ZpbV76JZV2ruOULZHHPPasm3+KHh+7t9Dvoby1TTNVt5JpLue7jQWRRY28uTkqHzIFI3DB45zXH+EvNbXrEaiJhZ3d1qsWjXEVpGscrzySSuGfzmlYBY2wTFGOOSflrd8NeHdWC+AhquitbN4dsJbS4M0kL7ZBBFGsibXbhtrYPBAzkDNAHoVvcQ3dtFcWsqTQTIHjljYMrqRkMCOCCOc1JWD4J0u40bwfZafeQC3lgMg8oEEIpkYqBjjG0it6gAooooAKKKKACiiigAooooAKKgS9t31CWxSVTcwxJNJF3VHLhW+hMbj/gJqegAooooAKKKKACiiigAqpe6nb2F3p9vcFhJqFwbaDC5BcRSSnPoNsTfpVuub8Uf8jF4M/7DUn/AKb7ygDpKKKKACiiigAooooAK8/8J+NfCukaXfWOq+JdHsbuLWtU8y3ub+KORM385GVZgRkEH6GvQK5vwH/yLt1/2GtV/wDThcUAH/Cx/BH/AEOXh/8A8GkH/wAVR/wsfwR/0OXh/wD8GkH/AMVXSUUAc3/wsfwR/wBDl4f/APBpB/8AFUf8LH8Ef9Dl4f8A/BpB/wDFV0lFAHN/8LH8Ef8AQ5eH/wDwaQf/ABVH/Cx/BH/Q5eH/APwaQf8AxVdJRQBzf/Cx/BH/AEOXh/8A8GkH/wAVR/wsfwR/0OXh/wD8GkH/AMVXSUUAc3/wsfwR/wBDl4f/APBpB/8AFUf8LH8Ef9Dl4f8A/BpB/wDFV0lFAHN/8LH8Ef8AQ5eH/wDwaQf/ABVH/Cx/BH/Q5eH/APwaQf8AxVdJRQBzf/Cx/BH/AEOXh/8A8GkH/wAVR/wsfwR/0OXh/wD8GkH/AMVXSUUAcPaeIfhVp+rTapYav4OtdQnLGa7hubVJZCxy25wcnJ5OTya1P+Fj+CP+hy8P/wDg0g/+KrpKKAOb/wCFj+CP+hy8P/8Ag0g/+Ko/4WP4I/6HLw//AODSD/4qukooA5v/AIWP4I/6HLw//wCDSD/4qj/hY/gj/ocvD/8A4NIP/iq6SigDm/8AhY/gj/ocvD//AINIP/iqP+Fj+CP+hy8P/wDg0g/+KrpKKAOb/wCFj+CP+hy8P/8Ag0g/+Ko/4WP4I/6HLw//AODSD/4qukooA5v/AIWP4I/6HLw//wCDSD/4qj/hY/gj/ocvD/8A4NIP/iq6SigDjfDmt6Vr3xJ1+60PU7PUrdNJ06NpbO4WZFYTXpKkqSM4IOPcV2Vc3Y/8lT13/sC6b/6Pvq6SgAooooAKKKKACiiigArm/FH/ACMXgz/sNSf+m+8rpK5vxR/yMXgz/sNSf+m+8oA6SiiigAooooAKq6hqdlpVsbjUbmO3j7FzyfYDqT9Kr65rUWi2auyNPcTN5dvbp96V/Qf1NUNL8OyS3S6r4jZbvUCPkiIzFbD+6o9ff/8AXXRClHl9pUdl+L9P8zjq158/sqKvLrfZevn2X5EY8VXt3htI8O31zEekkzLArD1Geopq+J5tNX/ia+Hbywh3FmkhCyopJJLEr6kkk+prp6KftaW3s9PV3/y/An2GI39s7+it91r/AIlaw1G01S1Fxp9wk8R/iQ9D6Edj9as1zOsaHNp9y2t+G08u7T5p7VeEul7gj+96GtvStTt9X0yG9tGzHKucHqp7g+4NTUpxUeem7r8U+z/z6l0a0nN0qqtJa6bNd1+q6FuiiisDrCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiq2oahbaXYy3l7II4Yhlif5D1NNJydkTKSjFyk7JEtxcQ2tu89zKkUSDLO7YAH1rn/wDhMPtjMug6TeamoOPOUCOI/Rm/wqj5I1eJte8YSx2OkQDzIbS4kCRov/PSUnA/A8V0+l6jpuq6dHdaLeWt5ZtlY5rSVZIzg4IBUkcEYrqcaVHSS5pfgvu3OGMq+J96D5I9NPefnrol6pv0OfGs3llfzahfeEruGWaJIpZ7eRZ2MaFyoIHYGRz/AMCNbela5p2tRs2nXCyMn34yCrp9VPNaFc74m0i08v8AtaG6j0zUIiqxXbSCNXZmCqjk8HcxCjvkge1JSo1NGuV91e3zTv8AmOUcTRXNGXOuzST+TVl8mvmjoqKytA1n+2bFmliMF3A/lXMB6xuOv4HtWrWE4ShJxlujrp1I1YKcNmFFFFQaBRRRQAVzfij/AJGLwZ/2GpP/AE33ldJXN+KP+Ri8Gf8AYak/9N95QB0lFFFABTZHWKNpJGCooLMx6ADvTq5zxdNJdR2mhWjFZ9Tk2yMOqQjlz+XH51rSp+0mo/1bqYYit7Gk52v2XdvRL5si8OxNrmpzeJLxTsYmLT42/wCWcQ4L49WrqKjt4I7W2jggUJFEoRFHYAYAqSitU9pO626egsNR9jT5Xq3q33b3/rsFFFFZHQFcxYgaF40nsB8tnqqG4gXssq/fUfUc/lXT1heLrGW40cXlmP8ATNPcXUJ9dv3l/EZ4rpw7XN7OW0tP8n95xYyLUFVjvDX5dV81+Njdoqtp19Fqem297bnMc6Bx7Z7fh0qzXO04uzOuMlKKlHZhRRRSKCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAK5aRf+Em8VtC3zaZpDAuv8M1x6H1C/z+taviLVf7H0Oe6QbpyPLgXGd0jcKP6/hS+HtK/sbQ4LVjumxvmfOd0h5Y/nx+FdVP93TdTq9F+r/T5nBX/f1lQ6LWX6L5vX5eZheO5IbTUfDGoaoMaRZ6mZLyRh+7gPkyCKWTsFWQr8x4UlScYyMzV7zS/EviPQIvD+qyjT9Qvrj7dcaRctCt40dsCP30ZBfG1F3K3GzbkbSB6HRXKd55Fo2p6rL8Q3hvdasbO9i1i4SSzuvEMomls13iNFsWj2YMflyCQHccbieStY2oz2WpeD9YtJ9Xl1c2F7p15cazYeILie2ljN7hiwDhLd1QM7RplVyjA/KpX3aigDj3+yaLrWj6tpc/naXqMKWsk4nMwlG0eTKZCSXyP4yTkc5Oa7Cs/XdLXWdDubFsBpE/dt/dccqfzAqHwxqjatoMM0+Rcx5huFPVZF4Of5/jXVP95SU+q0f6f5fJHBS/c4iVPpL3l6/aX5P5s1qKKK5TvCiiigArm/FH/IxeDP8AsNSf+m+8rpK5vxR/yMXgz/sNSf8ApvvKAOkooooAK5nw7/xONd1HX3+aLd9ksyf+ean5mH1P9ateLb+Sy0F4rTm7vXFrbgddz8Z/AZrS0uwj0vSrayh+5BGEz6nufxOTXVH93RcustPkt/0/E4Z/vsSodIav1ei+5Xf3FqiiiuU7gooooAKOtFFAHM+Gf+JVrGpaA3EcT/arQf8ATJ+oHsG4/GumrmfE/wDxLdW0rXV4WGX7Ncn/AKZPxk/Q/wA66aunEe9y1f5t/Vb/AOfzOHB/u+ah/K9PR6r7tV8gooormO4KKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiq9/eRadp895OcRwRl298DpTSbdkKUlFOT2RgXX/E78cQWo+a10hBPL6GZvuD8Bz+ddPWD4Ps5INF+2XY/0vUZDdTE9t33R+AxxW9XRiGubkW0dP8AP8TjwcW6bqy3nr8ui+St87hRRRXMdoUUUUAFczB/xJvHksH3bXWI/OT0EyfeH4jmumrnvGdvJ/YqajbDNzpsq3Se4B+YfTGfyrpwzTn7N7S0/wAvxscWNTVP2q3g7/Lr96udDRUVtcR3dpFcwHdHMgdD6gjIqWudpp2Z2Jpq6CiiikMK5vxR/wAjF4M/7DUn/pvvK6Sub8Uf8jF4M/7DUn/pvvKAOkooqtqV9HpmmXF7N9yCMuR64HT8elOKcnZEykoxcpbIwv8AkM+Pv71to0X4GZ/8F/I101YfhCxktNASe65u75zdTnuWfkfpityt8Q1z8i2jp/n97ucmDi/Z+0lvN3fz2XyVkFFFFc52hRRRQAUUUUAUtZ09dV0W7sWx++iKqT2bqD+BxVTwrqDal4btJZf9ci+VMD1Dr8pz+WfxrYrmtG/4lnjDVtMPEV0BfQD68P8Ariumn79GUO2v6P8AT7jhq/u8TCp0l7r/ADX6r5nS0UUVzHcFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABXNeKydRvNO0CM8Xkvm3OO0Kcn8z/KulrmfDn/E117VNdbmMv9jtT/0zT7xHsW5rpw/ut1f5fz6f5/I4cZ76jQX23r6LV/ft8zpgAqgKMAcADtRRRXMdwUUUUAFFFFABTZY0mheKVdyOpVge4PWnUUA9dGc54MkeHTbnSZ2Jl0y4aDJ6smcofyP6V0dc1L/xK/iFDJ0h1e3Mbf8AXWPkH/vniulrpxOs/aL7Sv8A5/jc4cF7tN0nvB8vy6fg0FFFFcx3BXN+KP8AkYvBn/Yak/8ATfeV0lc34o/5GLwZ/wBhqT/033lAHSVzPin/AImWoaZoCcrdS+dcgdok5wfqf5V01cz4b/4mmu6prrcoz/ZLU/8ATNOpHsW5/CunD+7er/L+b2/z+Rw4z31Gh/O9fRav79vmdMBgYHAooormO4KKKKACiiigAooooAK5rxR/xL9U0fWl4W3uPInP/TOTjJ+h/nXS1n6/pw1bQLyyxlpYiE/3hyv6gVvh5qFROW2z9HozlxdOVShJR3Wq9VqvxRoUVleGdROqeG7K6c5kaMLJnrvX5W/UVq1nOLhJxe6NqVSNWnGpHZq/3hRRRUGgUUUUAFFFFABRRRQAUUUUAFFFFABRRRQBjeK9RfTfDtw8GTczYggA6l34GPccn8KuaPpyaTo1rYx9IIwpI7t1J/E5NY17/wATfx3aWg5t9Kj+0yjsZW4QfUDmumrpqe5SjDvq/wBPw1+Zw0f3tedXovdXy3/HT5BRRRXMdwUUUUAFFFFABRRRQBz3jSB/7DXULcZn02dLpPcKfmH0wT+VbsE6XNtFPCd0cqB1PqCMiieFLm3kgmG6OVCjD1BGDWF4Knf+wmsJzmfTp3tX/wCAng/TBH5V0/HQ/wAL/B/8FficP8PF+U1+Mf8ANP8AA6GiiiuY7grm/FH/ACMXgz/sNSf+m+8rpK5vxR/yMXgz/sNSf+m+8oAt+K9RfTfDtw8GTcTYggA6l34GPccn8KuaPpyaTo1rYx4xBGFJHdupP4nJrGvf+Jv47tLMc2+lR/aZfQytwg+oHNdNXTU9ylGHfV/p+GvzOGj+9rzq9F7q+Wr/AB0+QUUUVzHcFFFFABRRRQAUUUUAFFFFAHNeHP8AiX+INa0g8IJhdwD/AGZByB7A8V0tc1rf/Eu8XaNqg4SctYzH/e5T/wAezXS104j3nGp3X4rR/lf5nDg/cU6P8rf3PVfnb5BRRRXMdwUUUUAFFFFABRRRQAUUUUAFFFFABUc88dtbSTzHbHEhdz6ADJqSuc8ZSvPY2uj27FZtUnEOR1WMcufy/nWtGn7Soo/1bqYYir7GlKfVbeb6L5sXwbBI+mz6rcjE+qTNcEH+FOiL9MfzroqZDEkEKQxLtjjUKqjsAMAU+irU9pNyDD0vY0o0+34vq/mwooorI3CiiigAooooAKKKKACuat/+JZ8QrmHpFqtssy/9dE4I/Lmulrm/GINpFp2soOdOulZz/wBM3+Vx/KunDazdP+ZW+fT8bHDjfdpqr/I0/ls/wbOkooBBAIOQehFFcx3BXL+L5ktta8IzzNtji1eV3PoBp14TXUVwPxPia+u/CGmQSFJ7vWWAx2T7JcK5/ANWlOKnNRZlWm6dKU0rtL8eiNzwbBJJps+rXIxPqkzTkH+FOiL9MfzroqZDEkECQxLtjjUKqjsAMAU+nVqe0m5E4el7GlGn2/F9X82FFFFZG4UUUUAFFFFABRRRQAUUUUAYvi+xa+8L3axZE0K+fER1DId3H5EfjWhpd8upaTa3qdJ4lfA7EjkfgatEBgQRkHgg965nwVJ9ng1DRm3btNunRMj/AJZsSVP866o+/Qa/ld/k9H+Njhl+7xcX/OrfNar8G/uOmooorlO4KKKKACiiigAooooAKKKKACiiigArmbD/AIm/jm+vjzBpifZIfTzDy5+o6Vsa3qSaPol1fvz5MeVHqx4UfmRVPwlYGw8M2olyZ51+0TMepd/mOffkD8K6qfuUpVO+i/X8NPmcFb97iIUui95/kvx1+RtUUUVyneFFFFABRRRQAUUUUAFFFFABVTVbFdT0i6sm6TxMgJ7Ejg/nVuinGTi010JlFTi4y2ZjeEb5r/wvZvLnzok8mUHqGQ7efyz+NbNc3oIOn+Kdb0w8RyOt7CPZ+G/UV0lb4iKVVtbPX79TlwUnKhFS3Wj9Vp+NrhXJeONOstW1TwjZarZ297aTay4kt7mJZI3xY3bDKsCDggH6gGutrm/FH/IxeDP+w1J/6b7yuc7A/wCFceCP+hN8P/8Agrh/+Jo/4Vx4I/6E7QP/AAVw/wDxNdJRQBzf/CuPBH/QnaB/4LIf/iaP+FceCP8AoTtB/wDBZD/8TXSUUAc3/wAK58Ef9CfoP/gsh/8AiaP+Fc+Cf+hP0H/wWw//ABNdJRQBzf8AwrnwT/0J+g/+C2H/AOJo/wCFc+Cf+hQ0L/wWw/8AxNdJRQBzf/CufBP/AEKGhf8Agui/+Jo/4Vz4J/6FDQ//AAXRf/E10lFAHN/8K58Ff9Cjof8A4Lov/iaP+Fc+Cv8AoUdD/wDBfF/8TXSUUAc3/wAK58Ff9Cjon/gvi/8Aia57UvAXhHTPGGnTN4X0f7DfqbV0NjHsSXqhA29T0r0WsfxVprap4duIoc/aIgJoCByHXkY9z0/Gt8O4qpaWz0+/r8tzkxcZuk5Q3jqvO2tvnt8yn/wrnwV/0KWif+AEX/xNH/CufBX/AEKei/8AgBF/8TWtouorq2i2l8mP38YZgOzdGH4HIq9WUouMnF7o6ITjUipx2epzf/CufBf/AEKei/8AgBH/AIUf8K58F/8AQp6L/wCAEf8AhXSUVJZzf/CufBf/AEKmjf8AgDH/AIUf8K58F/8AQqaN/wCAMf8AhXSUUAc3/wAK58F/9Cpo/wD4Ax/4Uf8ACufBf/Qq6P8A+AUf+FdJRQBzf/CuvBn/AEKuj/8AgFH/AIUf8K68Gf8AQq6R/wCAUf8AhXSUUAc3/wAK68Gf9CtpH/gGn+FH/CuvBn/QraR/4Bp/hXSUUAedeJ/AfhJrjStLtPDmmRS3t0N5jtUDeUgy/b6V0P8AwrrwZ/0K+k/+Aif4UlpnUviDeXHWHTLdbdD/ANNH+Zj+XFdLXTWSjGMPK7+ev5WOLDNznUq9G7L0jp+dzm/+FdeDf+hX0n/wET/Cj/hXXg3/AKFfSv8AwET/AArpKK5jtOb/AOFdeDf+hY0r/wABE/wo/wCFdeDf+hY0v/wFT/CukooA5v8A4V14N/6FjS//AAFX/Cj/AIV14O/6FnS//AVf8K6SigDm/wDhXXg7/oWdL/8AAZf8KP8AhXXg7/oWdM/8Bl/wrpKKAOb/AOFdeDv+ha0z/wABlo/4V14O/wCha03/AMB1rpKKAOb/AOFdeDv+ha03/wAB1o/4V14P/wChb03/AMB1rpKKAMjSvCmg6FdNc6PpNpZTuhjaSCIKSpIOM+mQPyrXoooAK5vxR/yMXgz/ALDUn/pvvK6Sub8Uf8jF4M/7DUn/AKb7ygDpKKKKACiiigAooooAKKKKACiiigAooooAKKKKAOZ8OD+yde1TQiNsYf7Zaj/pm/UAegbj866aua8Vf8S2/wBM15OFtZvJuT/0xfgk/Q4/E10oORkV1Yj31Gr339Vv+j+Zw4T9250P5Xp6PVfdqvkFFFFcp3BRRRQAUUUUAFFFFABUdxOlray3EpxHEhdj6ADJqSue8azOdDXT7dsT6lOlqn0Y/MfpgH861ow9pUUO5hiKvsaMqnZfj0DwVA40D7dcAifUJnupM/7R4/QD866Go4IUtreOCIYSJAij0AGBUlFWftKjn3DD0vY0Y0+y/wCHCiiisjcKKKKACiiigAooooAKKKKACiiigAooooAK5vxR/wAjF4M/7DUn/pvvK6Sub8Uf8jF4M/7DUn/pvvKAOkooooAKKKKACiiigAooooAKKKKACiiigAooooAq6lYx6npdzZTfcnjKE46ZHB/DrWb4QvpLzw/HFc5+1WTG1nB67k4/lityuZT/AIk3j90Py22sxbl9BNGOR+K8+5NdVL36cqfzXy3/AA/I4a/7qtCt0fuv57fjp8zpqKKK5TuCiiigAooooAKKKKACuanzqfxCgi6w6TbmVv8ArpJwB/3zzXSO6xxs7kKqjJJ7Cuc8GKbmzvNYkBD6lctKue0YO1R+hrpo+7CdTyt83/wLnDif3lSnR7vmfpH/AINjpKKKK5juCiiigAooooAKKKKACiiigAooooAKKKKACiiigArm/FH/ACMXgz/sNSf+m+8rpK5vxR/yMXgz/sNSf+m+8oA6SiiigAooooAKKKKACiiigAooooAKKKKACiiigArB8YWcs+hm7tB/penyLdQ+5Xkj8s8VvUEAggjIPUGtKc3TmproY16SrUpU31K9heR6hp8F5B/q54w6+2R0qxXM+FD/AGbeaj4fkyBaS+bbZ7wvyMfQ8H6101VWgqdRpbdPR7EYWq6tJSlvs/VaP8QooorE6QooooAKKKKAMHxldyW/h2SC3/4+L51tYRnGS5wf0zWvY2iWGn29pD9yCNY198DFYF3/AMTXx/a23WDSoTcSf9dX4UfgOa6aumr7lKMPm/nt+H5nDQ/eV6lXovdXy3/F2+QUUUVzHcFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABXN+KP+Ri8Gf9hqT/ANN95XSVzfij/kYvBn/Yak/9N95QB0lFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQBzPiYf2Vq+m+IE4jhf7Ndn/pk54J9lbn6kV0wORkVW1Gxi1LTbiyn/1c8ZQnHTI6/UdayvCN9NcaObO9P+madIbWb32/db8Rjnvg11S/eUU+sdPk9vxv+BwR/c4lx6T1XqtH96s/kzeooorlO8KKKKACmTzJb28k8zbY41LsfQAZNPrnPF8z3MNpoVqxE+py7GI/giXlz+XH51rRp+0mo/1bqYYir7Gk59enm9kvmxfBsLyadcatOpWbVJ2nweqpnCD8v510VMhiS3gjhhXbHGoRVHYAYAp9FWp7SbkGHpexpRp9vz6v5sKKKKyNwooooAKKKKACiiigAooooAKKKKACiiigAooooAK5vxR/yMXgz/sNSf8ApvvK6Sub8Uf8jF4M/wCw1J/6b7ygDpKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArmNU/4kPiy21cfLZ6hi1u/RX/5Zuf5Z7D6109VdT06DVtMnsboZimXafUHsR7g4NbUaihL3tno/T+tTmxNKVSn7nxLVeq/z2fky1RXO+HdUmimOg6023UbVcRuTxcxjo6+pwOR/9fHRVNSm6cuV/wDD+ZdCtGtDnj812fVMKKKKzNgrmPD/APxOPEOoa6/zQqfslnnpsU/Mw+pqz4q1OW3s003Tvn1HUCYoVB+4P4nPsBWnpWnRaTpVvY2/3IUC5/vHufxPNdUf3dFy6y0Xp1/y+84Jfv8AEqC2hq/Xovkrv7i3RRRXKd4UUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABXN+KP+Ri8Gf9hqT/033ldJXN+KP+Ri8Gf9hqT/ANN95QB0lFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAZutaHb61boJWaG4hbfBcxnDxN6g/0rJi8RX2hYt/FdswjXhdSt0LRP6bgOVNdRQRkYNbwrWjyTV1+K9H/SOSph25+1pS5Zfen6rr66PzMlPFOhPHvXV7MDGcNMAfyPNUZvGMFyzW/hy1m1W56AxoViQ/7TmtZtE0p33vplmzE53G3QnP5VcjjSJAkSKijoqjAFVzUI6qLfq9Pw/4BHJi56SmkvJa/i7L7mY2iaHLaXMupatMt1qlwMPIowsS/wBxB6fzrboorGpUlUlzSOmlShRhyQ/rzfmFFFFQahRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFc34o/5GLwZ/2GpP/TfeV0lZ+o6THqN/pN1JIyNpl210iqOHJglhwfbExP1AoA0KKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA4fxFBrSeONGs7LxZqtnaaobgyQxQWbCLy4wwCF4Gbk9dxb8KSx8d6ndaWtyuhxEXGoSaXp4kv/nupo5ZEZ3xFiNMROxI3HggL0z1N5o1ve61p2pyvKJtO83ylUja3mLtO7jPQcYIrM/4QmwTw9HpMF1eQ+RfSX9vdo6+dBM8zyllypUjMjLhlIKnBzQBUtfG87X1pp2o6UtvqL6udLuo47rzI4ibV7lJEcopdSiqMFVILH+7zXm8f3Ui+TpmjwTXrX19bLHc33kxFLUkO5cRsQxwMLtPU5OBmr0ngS0k09Y/7U1FdRW/GonVg0X2lpwnl7iPL8vHlHy9uzbt7Z5rk/FHw42LpdraWF/rVhHNeXNy4FlcXPnzMjZ23Y8kqSHycFwdoBALZAHxeNfEFxq1vrFjpCTpceF4tTfT31NkhiBd24Yx8yldo+4BwQWAAz6VY3keo6dbXsAYRXMSyoGGDtYAjPvzXOaN4ZubmwivvEssn9sXGkLp12IigULljnAXG/wCbkj5c9Biui0+yj07TbaxgLNFbQpChc5YhQAM478UAWKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigD/9k=)

Fig. 2.7 Lungime de unda

Sursa: https://study.com/learn/lesson/wavelength-formula-calculate.html

Lungimea dorită se poate calcula conform formulei λ=v/f[24],unde λ – lungimea de undă , v- viteza undelor electromagnetice in vid ( aprox. 3 × 10 ^8 m/s) și f – frecvența dorită. Dacă dorim să detectăm o plajă de unde iar antena trebuie să fie cel puțin jumătate din undă ajungem la concluzia că lungimea antenei trebuie proiectată în funcție de frecvența mai micuță, aceasta având o lungime de unde mai mare. Înlocuim datele în formulă obținem λ=3x10^8/2.4x10^9, iar λ=0.125 M, adică 12,5 cm. Pentru a determina lungimea antenei putem folosi două tipuri de antene Monopol sau dipol[26]. Antenele dipol au o eficiență sporită față de cele monopol și pot detecta o gamă mai largă de lungime de unde, dar ocupă o suprafață mai mare , de aceea în dispozitive portabile precum telefoane sau module radio, antenele monopol sunt folosite chiar daca nu sunt la fel de eficiente în recepția undelor, dimensiunea lor redusă fiind atuul lor , astfel o antena dedicată comunicațiilor Bluetooth de tip monopol v-a aveam un sfert din lungimea de unda a semnalului , adică 12,5/4=3.125cm .

Adunând toate segmentele din Fig. 2.8 putem observa ca lungimea antenei este aproximativ 3,2cm , ideală pentru comunicațiile Wireless Bluetooth.

A close-up of a computer chip

Description automatically generated

Fig 2.8 Modul ESP32

**2.4.3 Modulul de control (Body Control Module -BCM)**

Modulul de control BCM ( Body control Module) este elementul central al mașinii, acesta fiind responsabil cu gestionarea funcțiilor generale care țin de confort, accesibilitate sau siguranță, de la deschiderea și închiderea mașinii până la aprinderea luminilor și deschiderea geamurilor. Acest modul comunică cu restul modulelor ce țin de mobilitatea mașinii, cum ar fi ECU/TCU ( Engine/Transmission Control Unit) pentru pornirea și manevrarea mașinii doar dacă cheia se află în interiorul habitaclului.

A close-up of a circuit board

Description automatically generated

Fig. 2.9 Modul BCM

Sursa: <https://upload.wikimedia.org/wikipedia/commons/e/ed/KeylessEntryGoCircuitUnits.jpg>

Un modul BCM este alcătuit din mai multe componente ce au rolul de a gestiona anumite părți și funcționalități cum ar fi microcontroller-ul care se ocupa cu comunicarea cu restul modulelor mașinii și execuția comenzilor pentru diferite funcții, componente ce țin de comunicarea wireless cu cheia pe diferite frecvențe , interfața CAN Bus ce se ocupă cu comunicarea cu restul modulelor și în unele cazuri un modul TPM (Trusted Platfor Module) pentru a stoca în mod sigur cheile digitale și de criptare pentru comunicarea cu cheia fizică.

**2.4.4 Sistemul de criptare și autentificare /TPM**

Sistemul de criptare și autentificare se poate împărții în două categorii, cele ce au toate funcțiile realizate de unitatea centrală de procesare a BCM-ului sau se pot prelua de un TPM pentru o securitate crescută și eliberarea procesorului de sarcinile preluate de TPM.

În prima instanță în sistemele ce utilizează BLE , BCM-ul este programat din fabrică cu codurile pentru cheile sale originale, adresele MAC ale acestora și cheia privată și publică necesare criptării asimetrice, acestea fiind stocate în memoria flash a modulului sau în TPM, urmând a fi mutate în RAM în timpul funcționării , modulul urmând să aștepte de la sistemele de comunicații datele primite de la cheie pentru a fi decriptate, analizate și ulterior validate.

Periodic, BCM-ul scanează pentru dispozitive BLE autorizate ,cheia digitală sau telefonul mobil, urmând ca acesta să emită un răspuns cu adresa MAC. Daca această adresă MAC este una deja prezenta în memoria BCM-ului acesta începe procesul de „handshake”.

Handshake-ul nu este specificat cu exactitate pentru fiecare model și producător, fiecare având un proces de împerechere puțin diferit, ținut pe cât se poate secret pentru a creste securitatea sistemelor, dar în general se efectuează un schimb de chei pentru a verifica autenticitatea cheii sau în modelele cu securitate crescuta verificarea reciprocă a autenticității partenerului de comunicare, urmând să se stabilească o cheie de criptare temporală, utilizată doar pe parcursul sesiunii curente de comunicare.

Sisteme ce folosesc un modul TPM au ca și avantaj eliberarea procesorului de executarea comenzilor cu scop criptografic și securitatea cheilor. Trusted Platform Module (TPM) este un standard internațional pentru un criptoprocesor securizat, un microcontroler dedicat conceput pentru a securiza hardware-ul prin utilizarea cheilor criptografice integrate[28].TPM este un chip specializat pentru a executa operațiuni criptografice precum criptare/decriptare , hashing , generare de numere complet random și stocarea sigură a cheilor criptografice. Importanța funcțiilor de criptare și decriptare prezente pe TPM este ușor de înțeles când se ia în calcul durata și complexitatea ce poate apărea când se utilizează chei mai lungi , de 512 bit sau blocuri mari de date , când acesta are scopul de a ușura munca depusa de un procesor ce are ca scop doar procesarea de comenzi într-un sistem de tip IoT și nu de funcții criptografice, iar generarea de numere random , non-deterministice , ajută la eliminarea procesului de „reverse engeneering” a numerelor folosite pe post de token-uri de autentificare . Într-un sistem normal generarea unor numere 100% aleatoare este imposibilă, cele mai multe ori aplicându-se un algoritm pe un set de numere inițiale cum ar fi , data , ora sau temperatura procesorului, valori ce ar putea fi aflate , iar ulterior , după aplicarea algoritmului aflarea numărului „aleator”.

Un alt plus adus de modulele TPM este rezistența la atacurile de tip „memory dump”[29]atacuri ce au ca țintă copierea datelor ce se află în memorie , date ce pot include chei critice pentru integritatea și securitatea datelor utilizatorului. TPM-ul având capacitatea de a stoca în siguranță chei elimină riscul de furt al acestora din memoria calculatorului .

**Capitolul 3**

**Proiectare și structurare**

**3.1 Proiectare și cerințele sistemului**

A black background with white rectangles

Description automatically generated

Fig. 3.1 Diagrama block a sistemului

În timpul proiectării sistemului trebuie să ținem cont de niște principii pe care ne v-om construi toată structura :

* Securitatea și integritatea sistemului
* Bugetul pe cât se poate de redus
* Scalabilitate
* Compatibilitate
* Funcționalitate failsafe

V-om începe cu securitatea deoarece aceasta este cea mai importantă parte a întregului sistem fără de care restul principiilor nu mai au sens . Pentru această aparte trebuie să alegem un algoritm de criptare pentru a ne proteja pachetele, cum ar fi AES sau RSA . Algoritmii asimetrici precum RSA sunt ușor de implementat, dar pentru o cheie cu aceiași lungime, algoritmii simetrici precum AES au o rezistență sporită împotriva atacurilor, iar mulțumită faptului că AES este un algoritm simetric( are nevoie de aceeași cheie pentru a cripta și decripta mesajul) putem folosi cheia master pe post de „ shared secret” , cheie ce v-a fi folosită doar pentru a cripta cheia de sesiune, astfel asigurându-ne de faptul că avem o conexiune cu o cheie autorizata. Tot odată criptarea datelor nu este suficientă , fiind nevoiți să folosim și un protocol de comunicare ce ne va asigura faptul că partenerul de comunicații este cu adevărat unul autorizat și ce v-a împiedica unele atacuri asupra sistemului.

Sistemul trebuie să aibă două componente principale, o cheie și un modul în mașină, cheia poate fi implementată cu ajutorul unui microcontroller sau cu o aplicație ce să ruleze pe telefon. Varianta chei prezente pe telefon sub forma unei aplicații este cea mai la îndemână și ieftină, dar nu v-a permite împrumutarea mașinii sau să fie lăsată la un mecanic fără a renunța la telefon în acea perioadă. Deși o chei în plus , de forma unei cartele aduce un cost în plus poate fi mai practică. Pentru modulul din interiorul mașinii v-om utiliza un modul ESP32 datorită popularității acestuia , a disponibilității pe o mulțime de site-uri , ceea ce duce la un preț redus , capabilități bluetooth și performanță sporită, modul ce poate fi folosit și pentru cheia digitală separată de telefon. Pe lângă acest ESP care se va ocupa de sarcinile mai grele, cum ar fi rularea serverului pentru comunicare și criptarea sau decriptarea pachetelor, acesta nu dispune de suficienți pini GPIO pentru a interacționa cu toate interfețele mașinii pe care dorim să le controlam și pentru sistemul failsafe, așadar îi v-om adăuga un modul Arduino Mega cu microporcesorul ATMega 2560, modul ce are la dispoziție 54 de pini digitali, suficienți pentru implementarea noastră , asigurând și scalabilitatea pe viitor dacă se dorește controlul unei alte componente a mașinii, cum ar fi geamurile sau clima.

Pentru funcționalitatea de failsafe în cazul proiectării unei chei digitale separate , se v-a încerca integrarea lamei originale a mașinii cu care se pot deschide portierele manual in carcasa cheii digitale, astfel oferind o cale de acces în cazul în care modulul rămâne fără curent, iar pentru a porni motorul se v-a folosi un modul pentru citirea cartelelor RFID și un chip RFID amplasat pe spatele cheii digitale pentru a putea fi citit si a permite utilizatorului sa pornească cu mașina.

La capitolul compatibilitate trebuie aleasă o tehnologie de comunicare wireless, des întâlnită, de mare viteză și ce este disponibilă pe telefoane. Luând aceste cerințe în considerare, RF nu poate fi un candidat, rămânând cu Wi-Fi și Bluetooth ca și tehnologii de comunicare wireless întâlnite pe telefoane. Pentru aplicarea noastră v-om folosi Bluetooth, mai exact Bluetooth Low Energy deoarece are un consum redus de energie față de Wi-Fi sau Bluetooth Classic care consumă curent constant pentru menținerea canalului de comunicații, Bluetooth Low Energy organizând datele în Profile ,Service-uri si Characteristici în care sunt plasate date.

**3.2 Securitatea sistemului și protocolul de comunicații**

**3.2.1 Securitatea sistemului**

Securitatea este un element esențial în sistemele informatice, mai ales în cele ce

protejează date sensibile sau bunuri fizice ale utilizatorului, cum ar fi în cazul nostru un sistem „keyless entry” . În acest proiect securitatea sistemului v-a fi privită din trei puncte de vedere distincte:

* Protecția datelor transmise
* Securitatea fizică a sistemului
* Mecanisme pentru prevenirea atacurilor

Din punct de vedere a protecției datelor transmise, avem de abordat trei principii de

bază ale securității, cunoscute și sub denumirea de „triada CIA”[48]:

* Integritatea
* Confidelițialitatea
* Disponibilitatea

Pe lângă aceste principii de baza persoanele din domeniul securității consideră ca se mai pot lua în considerare concepte adiționale precum anonimitatea și autentificarea pentru a prezenta o imagine completă[48].

Integritatea datelor presupune asigurarea faptului că datele transmise nu au fost

manipulate de vreo parte terță a conexiunii, neautorizată cum ar fi în cazul unui atac de tip man-in-the-middle. Pentru asigurarea acestui principiu se pot folosi mai multe mecanisme, cum ar fi calcularea hash-ului datelor și adăugarea acestuia în pachetul trimis, acesta fiind verificat de receptor, determinând integritatea datelor. Un alt mecanism, similar cu cel al hash-ului ar fi adăugarea unui checksum, ce numără biții pachetului de date ce sunt setați pe 1, această abordare nu este la fel de sigură ca cea a hash-ului, aceasta ne putând determina dacă biții doar au fost mutați între ei, depistând doar diferențe, dar aceasta varianta este mult mai ușor de implementat și simplă pentru sistemele IoT, problema schimbării biților intre ei ne fiind o problemă mulțumită algoritmului de criptare pe car îl v-om folosi și la care v-om reveni , deci pentru a asigura integritatea datelor ne v-om folosi de un checksum.

Confidelițialitatea implică protejarea datelor transmise, astfel incât acestea să poată fi accesate doar de părțile autorizate, aceasta prevenind accesarea datelor de către atacatori.

Pentru îndeplinirea acestui criteriu criptarea datelor devine necesara pentru a preveni accesibilitatea datelor de către părți neautorizate, criptare ce poate avea o cheie specifică pentru fiecare sesiune de comunicare, cheie ce poate fi transmisă intre părți la începutului comunicării, criptată de o cheie master ce are rol de element „ shared secret” și de criptare a cheilor de sesiune. Astfel în cazul nefavorabil în care un atacator adună date dintr-o partidă de comunicare și reușește decriptarea lor, și în consecință aflarea cheii, acesta ne putând să descifreze următoarele pachete, acestea având o cheie diferită pentru fiecare conexiune realizată. Pentru un atac brute force asupra algoritmului AES128, ales la capitolul 3.1 Proiectare și cerințele sistemului, atacatorul are nevoie de până la 2,158,000,000,000 ani „din viața sa” , durata ce depășește vârsta universului de 150 de ori [49]. Pentru a afla cheia master, atac ce trebuie rulat din nou, ceea ce face aflarea cheii master imposibila într-un timp favorabil.

Un alt avantaj al algoritmului AES este efectul de avalanșă asupra datelor criptate, adică o variație mică a datelor de intrare pot duce la o schimbare majora a datelor criptate, sau vice versa, declanșând astfel un efect de avalanșă [27] . Criptând textul „Hello Word” obținem “f7e63cf44dd4e4e8a511f688c8ab5786”, iar schimbând caracterul o in a , „Hella Word” , ajungem la textul criptat “0fbfa517a5e8a6dbd87a1265e41e567c”, ce este complet diferit față de criptarea textului original [50], astfel , chiar daca 2 biți sau mai mulți, sunt interschimbați între ei, acțiune ce nu declanseaza checksum-ului daca biții sunt schimbați de către emițător înainte de a cripta datele, outputul v-a diferi drastic în caractere și numărul de biți setați pe 1, declanșând astfel checksum-ul. Așadar, utilizând criptarea cu AES și un checksum , putem garanta integritatea și confidelițialitatea datelor.

Pentru a menține anonima identitatea participanților la comunicare, și a sigura autenticitatea partenerului pe timpul sesiunii putem folosi tokenuri generate aleatoriu interschimbate între cele două părți conectate, astfel partenerii știind „semnătura „ provizorie a partenerului , detectând neregularitățile când aceasta se schimbă. Pentru a garanta anonimitatea dispozitivelor pe post de cheie securizată putem folosi funcția MAC Randomization disponibile pe dispozitivele mobile Android sau Apple ce are funcția de a schimba la un interval regulat adresa MAC transmisă dispozitivelor din jur pentru a nu permite localizarea dispozitivului[51] , adresa reală fiind transmisă doar dispozitivului la care este conectat.

Având aceste principii adresate mai avem de adăugat o metoda pentru a preveni atacurile de tip replay. Pentru a preveni pachetele din trecut să fie redate din nou serverului, și executate de către acesta, putem adăuga countere pentru pachete, astfel serverul putând ține cont de ce pachet ar trebui sa primească , având posibilitatea de a detecta și respinge pachetele vechi.

Știind principiile ce trebuie respectate și datele ce trebuie adăugate pentru prevenirea atacurilor putem defini structura exactă a pachetelor de comunicații ce v-or trebui să respecte structura:

* {KSes}KMaster  Pentru transmiterea cheii de sesiune
* {TkClient, TkServer, Counter, Random, Paiload, Checksum}KSes Pachetul comunicației

**3.2.2 Protocolul de comunicații**

Un sistem de securitate tipic constă dintr-un număr de entități, precum oameni, companii, calculatoare și cititoare de carduri magnetice, care comunică utilizând diverse canale, inclusiv telefoane, e-mail, radio, infraroșu și prin transportul datelor pe dispozitive fizice, cum ar fi cardurile bancare și biletele de transport. Protocoalele de securitate sunt regulile care guvernează aceste comunicații. Acestea sunt, de obicei, concepute astfel încât sistemul să reziste actelor rău intenționate, precum oameni care mint la telefon, guverne ostile care bruiază semnalele radio sau falsificatori care modifică datele de pe biletele de tren. Protecția împotriva tuturor atacurilor posibile este adesea prea costisitoare, așa că protocoalele sunt, de regulă, proiectate pe baza unor presupuneri despre amenințări. [52]

În proiectarea protocolului v-om ține cont de logica BAN „Burrows-Abadi-Needham” care este un set de reguli pentru a defini și analiza protocoale de schimb de informații. Mai precis, logica BAN ajută utilizatorii să determine dacă informațiile transmise sunt de încredere și sigure .[53] Logica BAN are ca și idee de bază acceptarea autenticității pachetului primit dacă este criptat cu cheia potrivită și dacă este „proaspăt” [52], logică ce presupune ca toate datele sunt interschimbate într-un mediu vulnerabil și transparent, idee ce a evoluat în bine cunoscuta mantră „Don’t trust the network”.[53]

Având în vedere cerințele BAN și conceptele de siguranță anterior, protocolul nostru v-a avea ca și primă instrucțiune efectuarea schimbului de cheie de sesiune, cheie ce ne garantează autenticitatea pachetului, criptat de un partener ce este în posesia secretului comun ( Cheia Master) , cu ajutorul căreia s-a putut obține cheia de sesiune, iar împreună cu contorul ales în faza de proiectare, cheia de sesiune poate garanta prospețimea datelor, care decriptate cu cheia potrivita, asigura proveniența pachetului din această sesiune de comunicare .

Conform notațiilor ban acest schimb de cheie de sesiune între client (C) și server (S) este:

A black background with a black square

Description automatically generated with medium confidence

Fig. 3.2 Schimbul cheii de sesiune

Ambii participanți având cheia de sesiune putem trece la schimbul de tokenuri ce ne v-a ajuta la viitoarele transmiteri de date. Pentru aceasta v-om aborda tehnica „challenge-response” în care un participant începe comunicarea cu o provocare ( challenge ), iar partenerul urmând să ofere un răspuns ( response ) , schimbând tokenurile ( care sunt niște nounce-uri , generate aleatoriu) print-o tehnică asemănătoare cu „3 way handsake” întâlnită în TCP. 3way handshake în TCP are rolul de a sincroniza clientul și serverul pentru o conexiune stabilă prin intermediul pachetelor SYN , SYN-ACK si ACK [54], schimbând numerele de secvență cu tokenurile participanților la comunicare, trimițând tokenul partenerului pentru a-l avea pe parcursul sesiunii de comunicare, iar acesta trimițându-l înapoi pentru a ne asigura că a recepționat valoarea corectă .

A black screen with white text

Description automatically generated

Fig. 3.3 Schimbul de tokenuri

Conform notației BAN avem următoarele declarații și concluzii care descriu pașii schimbului de tokenuri:

1. C∣≡C↔KSes​S,S∣≡C↔KSes​S

Serverul și Clientul cred faptul că KSes este o cheie sigură pentru comunicații

1. C⊢{TkServer, RandServer}KSes​​⇒C∣≡S∣∼TkServer

Serverul inițializează schimbul de tokenuri

1. S⊢{TkServer, TkClient, RandClient}KSes​​⇒S∣≡C∣∼TkClient

Clientul Trimite tokenul său alături de confirmarea primirii tokenului oferit de server

1. C⊢{TkClient, RandServer}KSes​​⇒C∣≡S∣≡TkServer, TkClient

Serverul trimite confirmarea tokenului oferit de client

După acest schimb de tokenuri și cheie de sesiune, partenerii de comunicație sunt pregătiți să iși transmită date conform structurii stabilite la partea de proiectare.

{TkClient, TkServer, Counter, Random, Paiload, Checksum}KSes

Pachetul comunicației

Pentru tratarea cazurilor în care câteva din pachet sunt eronate, ca urmare a unor probleme în canalul de comunicații sau a unor încercări de atacuri, putem implementa un mecanism de tratare, ce sa poată ignora aceste pachete eronate și să reia comunicația de la ultimul pachet sigur, avertizând partenerul de comunicații de această eroare, însă în cazul aplicațiilor de tip IoT, în care resursele sunt limitate și consumul redus de energie este necesar, avem posibilitatea mai simplă de a implementa un „buton de panica” care în cazul detectării unei neconcordanțe în datele primite, pur și simplu încheie conexiunea cu partenerul de comunicații , urmând pașii unei deconectări normale adică, generarea unui nou token pentru următoarea comunicare, ștergerea tokenului partenerului și în cazul clientului generarea unei noi chei de sesiune, iar în cazul nostru, unde avem comunicație prin serial intre modulul ESP32 și Arduino Mega putem goli forțat bufferele pentru a nu rămâne date nedorite ce pot interfera cu pachetele corecte primite de la client .

A screenshot of a computer

Description automatically generated

Fig. 3.4 Diagrama de secvență protocol comunicare Server-Client

**3.3 Bluetooth Low Energy și Bluetooth 6.0**

Bluetooth Low Energy , BLE, cunoscut anterior și sub denumirea de Bluetooth Smart, este o tehnologie pentru rețele wireless personale [43], ce are ca scop optimizarea tehnologiei Bluetooth Classic prin reducerea consumului de energie, având ca țintă sistemele IoT (Internet of things). Sistemele IoT sunt de cele mai multe ori alimentate de baterii pentru a putea fii versatile în ceea ce privește utilizarea lor în diferite locuri, astfel BLE vine în ajutor cu consumul sau redus de energie. Acest consum a fost posibil datorită perioadelor neactive în care nu sunt trimise date, dispozitivele intră într-un mod de consum redus, datele se transferă la o viteza de până la 2Mbps , favorizând transferul rapid de date cu pauze între ce vin cu un plus din punct de vedere energetic.

A diagram of a service

Description automatically generated

Fig 3.6 Structura de organizare a datelor BLE

Sursa: https://www.electrosoftcloud.com/en/ble-in-esp32-bluetooth-low-energy-connection/

Structura de date este bazată pe Profile , Servicii si Caracetrisitci[Fig.3.6], astfel oferind o organizare a datelor implementată direct în tehnologia de comunicare, ușurând treaba dezvoltatorilor pentru gestionarea și organizarea datelor[44] . Această separare a datelor ne v-a fi de folos, în sistemul nostru urmând să implementăm două caracteristici ce v-or avea rolul de TX si RX, replicând legăturile serial UART dintre Arduino Mega și ESP.

Distanța pe care poate comunica această tehnologie este între 10 și 100 de metri, în unele moduri de long range putând fii posibile distanțe mai mari în aer liber, dar dacă sunt prezente și alte obiecte sau clădiri, distanța scade iar viteza are de suferit[45], dar pentru cazul nostru, unde trebuie sa aproximăm distanța la doar câțiva metri distanța față de server, aceste limitări nu ne afectează.

Totul sună bine și frumos , dar BLE are niște dezavantaje față de Bluetooth Classic, viteza acestuia este mai mică, la doar 2 Mbps , capacitatea redusă pentru transferuri mari, BLE fiind optimizat pentru transferuri mici de date , periodice , nu streaming . Cu toate acestea limitările BLE nu impun o problema pentru aplicarea noastră , având nevoie să transmitem packete de 128 biți de câteva ori pe secundă.

Bluetooth 6, cea mai nouă și inovatoare apariție în lumea tehnologiilor wireless, apăruta pe 3 Septembrie 2024, vine cu îmbunătățiri pe partea de securitate și eficiență, aducând o funcție nouă pentru măsurarea distanțelor dintre dispozitive[46].

Funcția revoluționară ce apare cu Bluetooth 6 este „Channel Sounding” , funcție ce permite măsurarea precisă a distanțelor între dispozitive folosind RTT ( round-trip time ) și PBR ( pahse-based ranging) [47] ce măsoară timpul luat de fiecare pachet de date să ajungă de la emițător la destinatar, eliminând posibile atacuri de man-in-the-middle , detectând discrepanțele în distantele măsurate. Totuși pentru ca această funcție să poată funcționa dispozitivele trebuie sa fie sincronizate din punct de vedere al timpului, etapa ce se poate realiza la conectare .

La capitolul eficienta Bluetooth 6 vine cu îmbunătățiri datorita „Channel Sounding”, acum dispozitivele având capacitatea de a determina distanța dintre ele și dacă sunt folosite, cum ar fi în cazul unei tastaturi bluetooth , aceasta poate detecta când laptopul nu mai este în aproprierea sa , aceasta intrând într-un mod de economisire a energiei, iar pe partea de Low Energy , pachetele sunt publicate mai eficient , consumând mai putina energie, astfel oferind senzorilor și dispozitivelor industriale să ruleze ani întregi pe baterii.

Ultimul aspect îmbunătățit de „Channel Sounding” este securitatea conexiunii, Bluetooth 6 oferind o autentificare îmbunătățită , chei criptografice mai puternice și un handshake mai robust între dispozitive , iar posibilitatea de a măsura distanța la momentul comunicării dintre dispozitive elimină posibilitatea atacurilor de tip man-in-the-middle și relay. Luând în considerare aceste avantaje și securitatea crescută, Bluetooth 6 v-a fi noul standard, mai ales în domeniul auto pe sistemele keyless entry ce au la bază ca și cheie doar telefonul mobil.

Din cauza apariției recente a Bleutooth 6 , în timpul redactării acestei lucrări de licență, nu sunt dispozitive sau microcontrollere ce să utilizeze aceasta tehnologie , însă în următorul an v-or începe să apară telefoane și accesorii cu Bluetooh 6 ce sa permită poziționarea acestora , urmând domeniul auto și sistemele keyless entry, urmând să apară module dedicate aplicațiilor IoT .

**3.4 Algoritm și diagrame de funcționare**

Pentru a înțelege modul de funcționare a sistemului Keyless Entry , Keyless Go, acest subcapitol prezintă diagrame de stare și secvență ce ilustrează interacțiunile dintre componentele majore ale sistemului, algoritmii principali și fluxul de date .

A screenshot of a computer

Description automatically generated

Fig. 3.7 Diagrama de secvență a componentelor principale

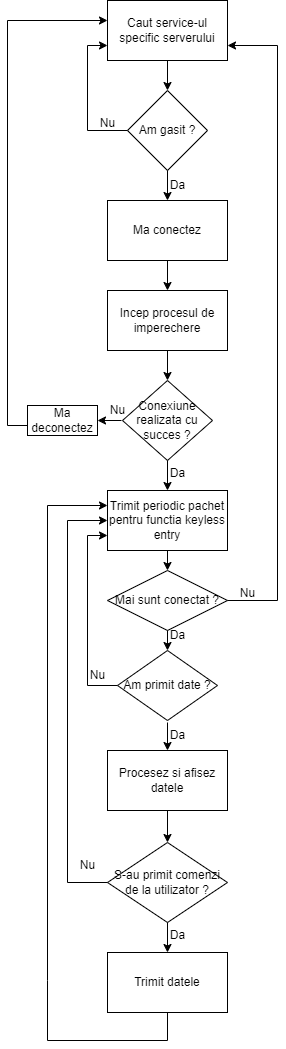


Fig. 3.8 Diagrama de stare a clientului

A group of white rectangles with black text

Description automatically generated

Fig. 3.9 Diagrama de stare a serverului ESP

A screenshot of a black screen

Description automatically generated

Fig. 3.10 Diagrama de stare a serverului Arduino

**A black screen with white text

Description automatically generated**

**Fig. 3.11 Diagrama de clasa a clientului Android**

**Capitolul 4**

**Implementare**

**4.1 Proiectare și implementare hardware, PCB**

Având în vedere cerințele analizate la capitolul 3, putem începe proiectarea generală a sistemului, începând cu modulul din interiorul mașinii ce v-a avea rolul de Server BLE și de a interacționa cu interfețele mașinii pentru funcțiile de pornire motor, aprindere becuri și blocare / deblocare . Acest modul trebuie împărțit în 5 părți, fiecare cu funcționalități bine stabilite :

* Alimentare
* Server BLE
* Control logic și decizional
* Comunicare între microcontrollere
* Interacțiune cu interfețele mașinii

La capitolul alimentare trebuie să ținem cont de tensiunea de alimentare primită de la mașină și tensiunile necesare pentru ca și componentele noastre să poată opera în condiții optime din punct de vedere al tensiunii de intrare, fiecare având o asemenea cerință. Pentru a scădea tensiunea primită de la mașină , ce poate varia de la 13,5V -14V în timp ce motorul este pornit la 9,5V – 8,0 V in timpul pornirii , avem de ales între două metode de a scădea tensiunea , convertor de tip buck sau regulator linear .

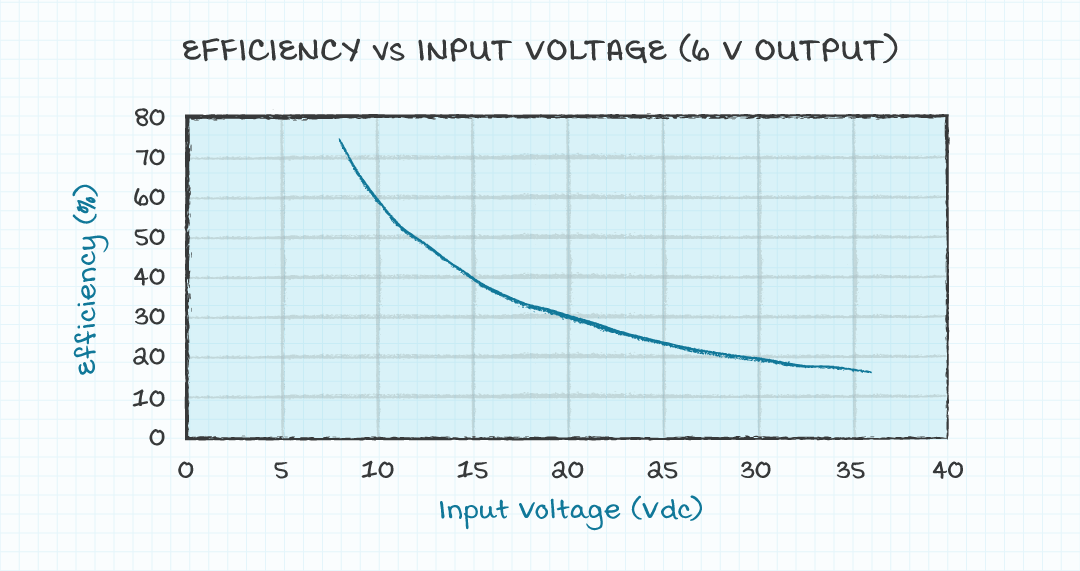
A diagram of a computer

Description automatically generated

Fig. 4.1 Exemplu diagramă implementare a unui

regulator linear

Regulatoarele lineare sunt cele mai simple de utilizat, având nevoie doar de niște rezistențe pentru a seta tensiunea dorită la ieșire, în unele cazuri nefiind nevoie de acestea, dacă se folosește un modul cu o tensiune de ieșire prestabilită din fabrică, și de doi condensatori pentru a atenua fluctuațiile de tensiune, plasați la intrarea în modul și la ieșirea din acesta [Fig. 4.1] . Aceasta simplitate în implementare vine cu niște dezavantaje care în cazul nostru nu pot fi ignorate. Modulele de regulare lineară, scad tensiunea la iesire prin convertirea diferenție dintre Vin( Voltaj intrare ) și Vout( voltaj iesire ) în căldură, introducând un risc de incendiu daca modulul consuma mai multa energie , care din formula puterii electrice, P=IxV, deducem că o crestere în puterea folosită de modul când tensiunea este constantă, duce la creșterea intensității curentului, curent ce are aceeași valoare la intrare în modul cât și la ieșire din acesta, deci puterea disipata de către regulator v-a creste odată cu consumul modulului , curent ce este convertit sub formă de căldură . Al doilea minus adus de acest tip de regulatoare este ineficiența sa extrem de redusă [30].



Grafic 4.1 Eficiența regulatoarelor lineare în

funcție de tensiunea de intrare

Sursa : <https://www.cn.cui.com/blog/a-comparison-between-dc-switching-regulators-and-linear-regulators>

Conform formulei eficienței (<https://www.renesas.com/en/document/apn/r34an0006-linear-vs-switching-regulators-taking-accurate-efficiency-measurements>), cu cât diferența dintre tensiunea de intrare și tensiunea de ieșire este mai mare, cu atât eficiența regulatorului scade, apropiindu-se de 0% eficiență conform Graficului 4.1, eficiența scăzută care în timp duce la un consum în exces și inutil de energie, ce scade durata de timp maxim posibila între pornirile motorului pentru reîncărcare.

A diagram of a circuit

Description automatically generated

Fig. 4.2 Exemplu diagrama implementare a unui

Convertor Buck

A graph showing different colored lines

Description automatically generated

Grafic 4.2 Temperatura de operare a unui modul buck

în funcție de tensiunea de intrare

Sursa: <https://learn.sparkfun.com/tutorials/buck-regulator-hookup-guide/all?print=1>

Deși utilizarea unui convertor de tip buck este mai complexă, având nevoie de componente adiționale [Fig. 4.2] precum bobină, diodă și condensatori electrolitici acesta rămâne la o temperatură confortabilă , aproape de cea a unui telefon mobil [Grafic 4.2] în timp ce regulează o putere de sub 12 W, putere pe care nu o v-om depășii în timpul funcționării modulului nostru. Așadar, datorita eficienței crescute și a temperaturii scăzute v-om utiliza un IC ( Integrated Circuit ) de tip Buck Convertor pentru a scădea tensiunea mașinii la 5V , un IC precum LM2596DP-5.0 deoarece are un preț redus, implementările sunt bine documentate de producători și tensiunea de ieșire este prestabilită, fără a trebui adăugate rezistențe în plus pentru a o seta .

Pentru ESP32 v-om avea nevoie de o tensiune de 3.3 V deoarece nu v-om folosi un modul de dezvoltare ce are integrat un regulator de tensiune de la 5V furnizati de USB la 3.3V necesari modulului, din cauza suprafeței crescute adusă de toate componentele opționale. ESP32 în timpul rulării unui server Bluetooth Low Energy are un consum de aproximativ 95~100 mA [31], iar ca tensiune de intrare pentru IC-ul de regulare a tensiunii v-om folosi cei 5V furnizați de convertorul Buck , iar ca tensiune de ieșire avem nevoie de 3.3V. Datorita diferenței mici de tensiune și a consumului mic de curent al ESP-ului în sarcină putem folosi un integrat de regulare lineară a tensiunii, dezavantajele fiind insesizabile, rămânând totuși cu plusul simplicității implementării. Din aceleași motive ca și cele pentru care am ales IC-ul LM2596DP-5.0, preș mic, documentație clară și tensiune prestabilita v-om folosi regulatorul linear AP2112K-3.3TRG1.

Pentru a furniza un server Bluetooth Low Energy v-om utiliza un modul ESP32 după cum am stabilit anterior, doar modulul ESP32, fără restul de componente pentru regularea tensiunii si programare care se regăsesc pe modulul de dezvoltare. Lipsa unui regulator de tensiune integrat în modulul ESP, am adresat-o anterior , folosind un regulator linear, AP2112K-3.3TRG1, regulator ce se și găsește pe multe plăci de dezvoltare. Pentru programarea modulului v-om avea nevoie de puțin ajutor din exterior, din partea unui modul de programare specializat sau a unui alt ESP32.

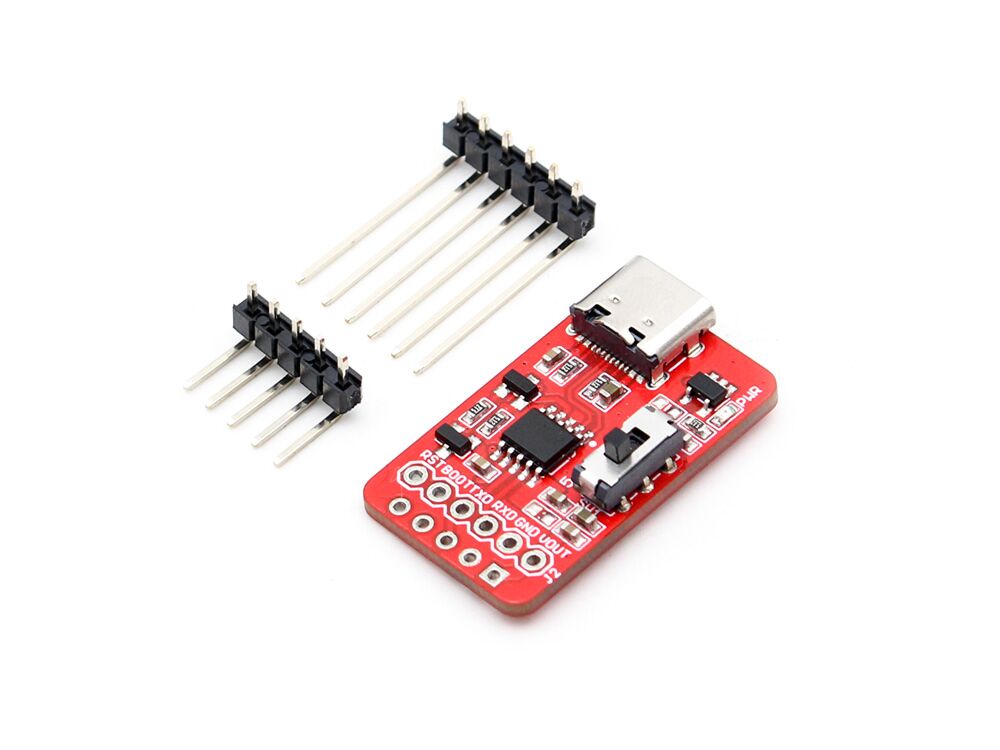


Fig. 4.3 Modul programare USB2UART CH340K

Sursa: <https://www.makerfabs.com/esp32-programmer-usb2uart-ch340k.html>

A diagram of a circuit board

Description automatically generated

Fig. 4.4 Configurarea de baza a modulului ESP32 pentru

funcționare și programare

Folosind metoda programatorului propriu-zis, poate reduce numărul de componente, ne mai fiind nevoie de prezenta butoanelor pentru a interacționa cu pinii modulului ESP ce controlează dacă chipul este activ sau dacă acesta intră sau nu în boot mode, conectând pinii modulul de programare cu pinii ESP-ului conform diagramei furnizate de producător. Pentru a utiliza metoda unui ESP adițional pentru programare v-a trebui să adăugăm pinilor EN ( chip enable , responsabil de activarea și dezactivarea modulului ) și IO0 ( responsabil pentru selectarea modului boot normal sau al programării modulului) conform FIG. 4.4 , butoanele fiind conectate la GND , împreuna cu o rezistențe pull-up de 10 kOhm .

A close-up of a circuit board

Description automatically generated

Fig. 4.5 Conexiuni între module ESP32 pentru programare

Sursa: <https://www.youtube.com/watch?v=ndEeFQ5mi_g&t=357s>

Pe partea modulului ce v-a avea rol de modul de programare v-a trebui să dezactivăm chipul ESP32 prin conectarea pinului EN la GND , astfel modulul de pe plăcuța de dezvoltare fiind dezactivat, oferindu-ne permisiune de a ne folosi de programatorul acestuia. Restul conexiunilor sunt simple conform Fig. 4.5, fiind nevoie de alimentare , 3.3V și GND , iar pinii pentru comunicare serial RX și TX fiind conectați direct .

Ca și microcontroller ce să se ocupe de gestionarea conexiunii cu clientul și de a executa comenzile primite am ales un Arduino Mega ce lucrează la o frecvență de 16 MHz, suficientă pentru operațiuni simple de comparare a numerelor și a activa pini , având în vedere faptul ca modulul ESP o să se ocupe de acestea, având o putere de calcul mult mai mare și o frecventa de 240MHz fiind ca o rachetă pe lângă Arduino , dar insuficiența pinilor ne obliga se avem doua microcontrollere.

Pentru a permite celor două microcontrollere să comunice între ele avem de ales între două metode principale de comunicare: I2C și Serial(UART).

I2C este o metoda de comunicare serială, sincronă , ce permite comunicarea mai multor dispozitive folosind două fire pentru comunicare, SDA( Serial Data Line) care asigură transferul de date și SCL( Serial Clock Line) ce poartă semnalul generat de dispozitivul master pentru sincronizarea transferului de date. Ca avantaje ale acestui protocol ar fi acomodarea mai multor dispozitive pe aceeași magistrală, fiind nevoie de doar două fire pentru orice număr de dispozitive conectate, însă ca dezavantaj ar fi necesitatea de rezistente pull-up pe firele SCL și SDA , viteza relativ scăzută, aceasta ajungând la maxim 3.4 Mbps în modul High-Speed[32], și nevoia ca fiecare dispozitiv ce dorește să comunice are nevoie de o adresă unică.

Serial (UART – Universal Asynchronous Receiver-Transmitter) este o metoda de comunicare point-to-point ce permite transferul de date între două dispozitive folosind un canal de recepție și unul de transmitere, TX(Transmit) și RX( Receive) . Singurele condiții ce trebuie îndeplinite de cele doua dispozitive pentru a putea comunica este sa își seteze viteza de transfer ( baud rate) la aceeași valoare, iar conexiunile să fie inversate , RX-TX și TX-RX, aceasta simplitate face ca UART-ul să fie extrem de ușor de folosit, având un proces de setup simplu și eficient. Cu toate că viteza sa este mult mai mica fata de cea a I2C-ului ( 115kbps , cea mai răspândită viteză pentru comunicații rapide și cu o fiabilitate mare, unele harware-uri putând atinge 4 Mbps dar la o fiabilitate scăzută), simplitatea acestuia, prezența doar a două microcontrollere ce trebuie să comunice între ele și cantitatea redusă de date ce trebuie transmisă face ca UART-ul sa fie o alegere simplă , eficientă și sigură.

A circuit board with numbers and symbols

Description automatically generated

Fig. 4.6 Modul conversie tensiuni pentru comunicații serial

Cunoscând protocolul de comunicare, acum trebuie adresată problema tensiunilor diferite ale celor doua microcontrollere. ESP32 folosește 3.3V iar Arduino folosind 5V, aceste tensiuni regăsindu-se și la pinii ce se ocupă cu comunicația serial, această diferență oprindu-ne din a conecta direct modulele unul la celălalt, fiind nevoie de un mediator între cele două. Integratul TXS0102DCUR este o alegere perfectă pentru cazul nostru, încercând să menținem un budget redus cu funcționalități mari. Ușurința folosirii acestui IC adusă de simplitatea conexiunilor ce trebuie realizate și a lipsei componentelor adiționale externe [Fig. 4.6], fac utilizarea acestuia foarte simplă, iar viteza sa maxima de 24 Mbps ( viteză incomparabila cu cei 115 kbps ce vor fi folosiți în aplicarea noastră ) fac acest ic alegerea ideală.

Ultima funcționalitate a modulului nostru ce trebuie adresată este interacțiunea acestuia cu interfețele mașinii. Pentru a putea începe trebuie să stabilim cu ce funcționalități ale mașinii dorim să interacționăm. În acest caz v-om controla pornirea mașinii, becurile acesteia și blocarea/deblocarea. Știind funcționalitățile trebuie să vedem cum interacționăm cu ele, începând cu blocarea și deblocarea, putem folosi un tranzistor sau un releu pentru a simula acționarea butonului de închidere/deschidere folosind un singur pin de comandă și o masă comună, dar luând în considerare dimensiunile componentelor și sunetul în plus produs de un releu în timpul acționării v-om alege un tranzistor NPN pentru această sarcina.

A diagram of a circuit

Description automatically generated

Fig. 4.7 Utilizarea tranzistorului NPN

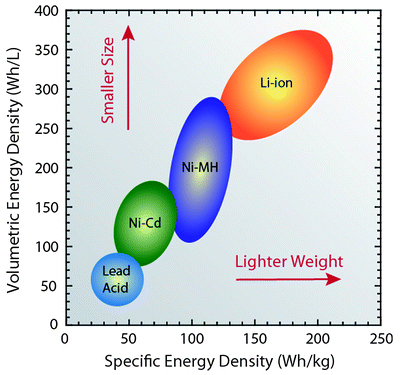
Pornirea motorului și aprinderea becului sunt asemănătoare, ambele acționând componente ce solicita un amperaj mare când sunt active, forțându-ne să trecem de tranzistori micuți sau relee de 5V și să ne îndreptăm atenția la mosfeturi și relee auto. Mosfeturile sunt asemănătoare cu tranzistorii, acestea având nevoie de un semnal pentru a permite fluxul de curent prin acesta, singura diferență fii faptul că un tranzistor controlează curentul prin acesta în funcție de intensitatea curentului aplicat la pinul de control al acestuia, iar mosfetul răspunde în funcție de voltaj, amperaj prin pinul de control gate fiind aproape de 0 . În cazul nostru doar becurile pot consuma in jur de 100W ceea ce ar rezulta la un curent de aproximativ 8 amperi , curent care dacă ar trece printr-un mosfet ar produce o cantitate semnificativă de căldură și ar avea nevoie de un radiator de răcire adecvat, astfel ajungem la releele auto de 12V ce pot comuta curenți și de până la 80 de amperi, dar v-a fi nevoie de un tranzistor suplimentar prin care să acționăm aceste relee datorita tensiunii de 12V ce trebuie aplicată releului și tensiune de 5V a pinilor microcontroller-ului. Tranzistorul v-a primi un semnal de control de 5V din partea microcontroller-ului , acesta urmând să permită curentului de 12V să treacă prin el, pornind releele [Fig. 4.7]

Trecând la partea de cheie digitala, avem de împărțit funcționalitatea în 5 părți:

* Alimentare
* Protecția și încărcarea bateriei
* Control logic și decizional
* Regularea tensiunii
* Interfața de intrare/ieșire

La capitolul alimentare trebuie să utilizam o baterie pentru ca acest dispozitiv să

poată fi unul de sine stătător, furnizând curent întregii cheii.



Grafic 4.3 Raport densitate energie în funcție de volum și masă

Sursa: https://www.epectec.com/batteries/cell-comparison.html

Pentru baterie avem de ales între 4 tipuri bine cunoscute de baterii :

* Baterii cu acid și plumb ce sunt cele mai utilizate în industria auto, acestea fiind rezistente la variații mari de temperatură, oferind un amperaj foarte mare când acesta este solicitat
* Ni-Cd si Ni-MH ( Nickel-cadmium și respectiv Nickel- Metal Hydroxide ) întâlnite des în bateriile reîncărcabile ce erau populare în perioada anilor 2000 , acestea având o tensiune de 1.5V , la fel ca cele AA sau AAA , acestea având în plus capabilitatea de a fi reîncărcabile, luând locul bateriilor tradiționale destul de repede
* Li-ion și LI-Po cele mai recente ți performante baterii, fiind folosite în 90% din dispozitivele pe care le întâlnim zi de zi, de la telefoane și laptopuri și până la mașini electrice. Acestea sunt cele mai avansate tehnologii de baterii, oferind cel mai bun raport între energia stocată și volum/masă.

A close-up of several objects

Description automatically generated

Fig. 4.8 Baterii Li-ion folosite în țigări electronice de unică folosință

Sursa: https://www.mdpi.com/2313-0105/10/7/236

Mulțumită densității energetice crescute și a popularității, v-om folosi baterii de tip Li-ion, putând obține aceste baterii din țigările de unică folosință[Fig. 4.8] ce conțin astfel de baterii, de o capacitate 500mAh și 800mAh în țigările mai mari, astfel împiedicând poluarea mediului prin eliminarea acestor baterii din țigări înainte să ajungă la gunoi și în centre de reciclare unde pot prezenta un risc de incendiu [33] , ajutând în același timp și bugetul acestui proiect, obținând aceste baterii gratis de la persoanele ce le folosesc.



Fig. 4.9 Modul încărcare și protecție baterii Li-Ion

Sursa: <https://www.emag.ro/circuit-de-incarcare-1-acumulator-li-ion-18650-cu-type-c-usb-si-protectie-tp4056-tp-4056-type-c/pd/DT9H9QMBM/?ref=hdr-favorite_products>

Ajungând la a doua funcționalitate, protecția și încărcarea acestor baterii, avem nevoie de un integrat care să controleze încărcarea sigură a bateriilor de la o sursa de 5V și unul pentru protecția acestea când vine vorba de supra-încărcare și sub-descărcare. Căutând pe internet putem găsi module complete cu aceste funcționalități, ce utilizează 3 IC-uri și un port USB . Căutând integratele pe internet putem observa că s-a folosit un TP4056 ce are rolul de a încărca bateria Li-Ion, acesta primind 5V volți furnizați de portul USB , un DW01A ce monitorizează tensiunea bateriei pentru a tine-o între 2.9 V și 4.2V , acesta decuplând bateria de restul circuitului prin intermediul a două mosfeturi integrate în IC-ul FS8205A. Datorita documentației bogate și a popularității acestei combinații de componente , v-om folosi pentru cheia noastră aceeași combinație de IC-uri împreuna cu un port USB-C pentru încărcarea ușoară cu conectorul standard impus de Uniunea Europeana [34] .

Logica și funcționalitatea acestei chei v-a rula pe același modul ales pentru modulul din interiorul mașinii, adică ESP32 modul puternic, cu capabilități wireless Wi-Fi și Bluetooth ce este perfect pentru aplicarea noastră, mulțumită dimensiunilor foarte mici[31]. ESP-ul trebuie să ruleze un client Bluetooth Low Energy , să primească comenzi de la utilizator și să monitorizeze voltajul bateriei pentru a informa utilizatorul de nivelul de energie disponibil în baterie .

Tensiunea de intrare acceptată de ESP32 este cuprinsa între 3.6V și 2.7 V , de preferat ca aceasta să se afle la 3.3V, însă voltajul unei baterii Li-Ion poate varia între 4.2V și 2.9V, interval care dacă depășit poate duce la un incendiu, interval prea mare pentru a putea conecta direct bateria la ESP, așadar v-om avea nevoie din nou de un regulator de tensiune. Conform analizei anterioare, avem de ales între convertor Buck sau regulator linear. Mulțumita lipsei componentelor adiționale și a spațiului restrâns a cheii v-om folosi același regulator linear, anume AP2112K-3.3TRG1.

Ultimul aspect ce trebuie definit este modalitatea prin care utilizatorul interacționează cu cheia și cum acesta poate primi date referitoare la mașina. Pentru a trimite comenzi de bază v-om utiliza 4 butoane pentru blocare , deblocare , lumini și pornire/oprire motor, iar pentru a vedea starea mașinii și datele despre aceasta v-om utiliza un ecran oled de 1.3” monocolor conectat la ESP prin I2C pentru a eficientiza spațiul, utilizând doar două fire pentru comunicarea cu ecranul .

Putem începe procesul de dezvoltare prin proiectarea hardware a celor două componente principale, modulul pentru mașină și cheia smart, știind funcționalitățile și componentele dorite, codul putând fii dezvoltat ulterior și uploadat pe module.

V-om începe cu modulul mașinii, mai exact cu sursele de curent, microcontroller-ele și conexiunile dintre ele, apoi cu componentele ce interacționează cu mașina . După cum am stabilit anterior pe partea de alimentare, pentru a aduce tensiunea furnizată de mașină v-om utiliza convertorul buck LM2596DP-5.0, care conform datelor furnizate de producător[35] tensiunea de ieșire este prestabilita din fabrică, ne mai fiind nevoie de a o seta prin intermediul unor rezistori externi.

A diagram of a circuit

Description automatically generated

Fig. 4.10 Implementarea regulatorului de tensiune LM2596DP-5.0A screen shot of a graph

Description automatically generated

Grafic 4.4 Tensiunile înainte / după inductor, și după condensator

Sursa: https://iopscience.iop.org/article/10.1088/1742-6596/2312/1/012059/pdf

Acest integrat dispune de 5 pini ce trebuie adresați pentru buna funcționare a acestuia VIN,VOUT ,GND, FEEDBACK, ON/OFF, din denumirile acestora ne putem și da seama de funcționalitatea unora, cum ar fi GND și VIN , acești pini furnizează integratul cu tensiunea mașinii, un condensator electrolitic de 680UuF fiind adăugat pentru a finisa și stabiliza tensiunea mașinii , daca aceasta are scăderi bruște, de scurtă durată, condensatorul putând furniza diferența pentru o tensiune constantă la intrarea în integrat. Pinul ON/OFF permite pornirea și oprirea modulului la comanda, dar fiindcă dorim ca acesta să fie pornit în permanență , acesta va fi conectat la GND. Următorii doi pini , VOUT și FEEDBACK sunt puțin mai complicat de înțeles , Modulul Buck crează un semnal PWM ( Pulse Width Modulation) ce încarcă parțial bobina, aceasta eliberând tensiunea când semnalul PWM este în faza de 0 V, tensiune ce este finisată cu ajutorul condensatorului de după bobină, iar pinii FEEDBACK este conectat pentru a informa integratul cum să varieze duty cycle-ul ( perioada în care tensiunea este la VCC sau 0V ) pentru ca la ieșire să obținem 5V. În graficul 4.4 se poate observa semnalul PWM generat de convertor , voltajul după bobină și voltajul finisat după condensator. Această tensiune constantă și regulată de 5V poate fi furnizată Arduino-ului și regulatorului de 3.3V.

A diagram of a circuit

Description automatically generated

Fig. 4.11 Implementarea regulatorului linear AP2112K-3.3TRG1

Folosirea integratului AP2112K-3.3TRG1 este una extrem de simplă , fără componente adiționale pentru regulare sau feedback, acesta având nevoie doar de o tensiune de 5V pinilor VIN și GND, pinul EN (chip enable) fiind conectat la 5V pentru a fi mereu în funcțiune, iar pinul VOUT furnizându-ne cei 3.3V pe care îi putem folosi pentru ESP , mai fiind nevoie doar de doi condensatori la intrare și ieșire pentru finisarea tensiunii.

A circuit board with many wires

Description automatically generated

Fig. 4.12 Implementarea modulului ESP32

Având tensiunile necesare putem trece la adăugarea microcontroller-elor, începand cu Arduino Mega. În cazul nostru v-om folosi un modul de dezvoltare aftermarket Arduino Pro Mega ce are o dimensiune redusă față de cel original de la Arduino și un preț mai micuț. Acesta fiind un modul complet de dezvoltare nu trebuie să conectăm niciun modul de programare la acesta , doar tensiunea de 5V, o masa (GND) și conexiunile pentru comunicarea serial cu ESP-ul și interfețele malinii. Trecând la ESP , acesta v-a fi conectat la masa și 3.3V pentru alimentare , dar pentru a putea programa modulul trebuie adăugate două butoane pinilor EN și IO0 ce sunt responsabili cu pornirea și oprirea modulului ( funcție de repornire) și selectare a modului de boot normal sau de programare, aceste butoane fiind conectate la masă , având o rezistență pull-up la 3.3V de 10 KOhm pentru funcționarea normală a modului, mai fiind nevoie sa adăugăm doua puncte de contact pentru pinii RX si TX ai modulului ce v-or fi conectați pentru programare.

Acum că avem microcontroller-ele pregătite de programare și funcționare trebuie să ne asigurăm că acestea pot comunica între ele prin intermediul interfeței serial . Un avantaj pe care îl avem prin utilizarea acestor module este prezența mai multor UART-uri pentru comunicări serial, permițându-ne să putem programa modulele fără a le deconecta unul de la celălalt din cauza naturii UART-ului de a permite comunicarea doar între doua dispozitive, făcând programarea imposibilă pe pinii principali UART care sunt folosiți pentru încărcarea instrucțiunilor. Așadar v-om lăsa liberi pinii dedicați programării pe Serial0 și ne v-om folosi de Serial1. Tensiunile prezente pe pinii comunicației serial sunt cele furnizate modulelor pentru funcționare , 3.3V și 5V făcând imposibilă conectarea lor directă fără a avea probleme de comunicare fiind nevoie să folosim un „Voltage Level Shifter „ . Acest shifter poate fi construit din câțiva tranzistori și rezistențe adecvate, dar căutând pe internet am ajuns la integratul TXS0102DCUR , un Voltage Level Shifter ce poate face legătura între cele două Seriale la tensiuni diferite , la un preț asemănător cu cel construit din tranzistori și rezistențe, fără bătăile de cap și de implementare, acest IC oferind suport pentru comunicații de până la 24Mbps[36] , iar folosirea acestuia în circuitul nostru fiind simplă, fără a avea nevoie de componente adiționale, doar conexiuni la cele două tensiuni, masa și cele două seriale .

În cele din urmă ajungem la interacțiune cu interfețele mașinii, unde am stabilit să folosim relee de 12V pentru pornire și oprire, iar tranzistori pentru a simula apăsarea de buton.

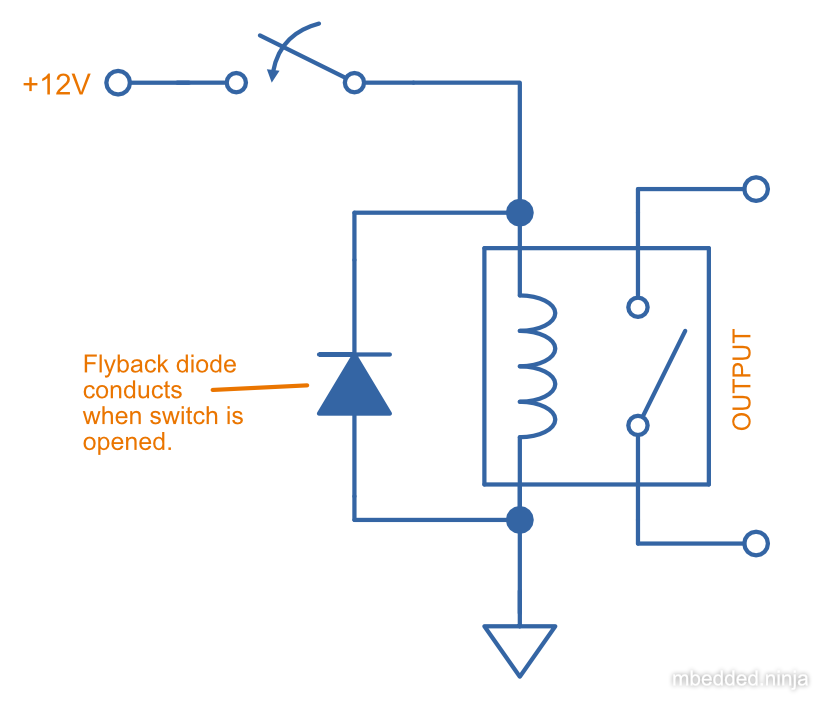


Fig. 4.13 Releu și diode flyback

Sursa : <https://www.cloudynights.com/topic/932174-12v-power-supply-for-observatory/>

Butucul rotativ folosit pentru pornire are 4 conductori , un fir ce are 12V furnizați de baterie, ci câte un fir pentru fiecare treaptă a butucului, accesorii , contact și pornire, fiecare fiind conectați la +12V când sunt activați , astfel releele noastre v-or fi amplasate în exteriorul modulului, cât mai aproape de conductorii responsabili pentru pornire, firul +12V fiind comun între relee, iar restul de 3 fire câte un releu separat pentru acționarea lui, împreuna cu o diodă flyback [Fig. 4.14] pentru a opri curentul furnizat de căderea câmpului electromagnetic după îndepărtarea alimentării bobinei, pentru a preveni defectarea tranzistorilor .

A diagram of a circuit

Description automatically generated

Fig. 4.14 Conectarea tranzistorilor

O parte a bobinei releelor va fi conectata la +12V ai mașinii, iar celălalt capăt v-a fi conectat la GND prin intermediul tranzistorului. Tranzistorul de tip NPN 2N222, este unul foarte răspândit ce poate permite trecerea curentului prin pinii Collector și Emitter ai acestuia dacă pe pinul Base se aplică un voltaj pozitiv. Tranzistorii sunt niște amplificatoare , ce amplifică curentul primit la Base în functie de un factor ce indică amperajul maxim ce poate trece prin acesta. Pentru a calcula rezistentele necesare pentru a tranzistor avem nevoie de rezistenta bobinelor releurilor pentru a calcula amperajul consumat de acestea. Rezistența bobinelor alese este de 80 Ohm , rezistență cu care putem calcula intensitatea curentului electric conform formulei I=U/R din care aflat I=150mA. Pentru a permite trecerea a 150mA trebuie sa calculăm câți mA avem nevoie la pinul Base , alegând cel mai rău factor de amplificare , care în cazul nostru este 35 conform documentației [37], care înlocuită în formula IBase = ICollector / Gain [38] aflăm că avem nevoie de un curent de aproximativ 5mA pe pinul Base. Pentru calcularea rezistenței necesare pentru a ajunge la un curent de 5mA avem nevoie de Base-Emitter saturation voltage [37] ce reprezintă căderea de tensiune dintre pinul Base și Emmiter, care în cazul nostru este de 0.7V, deci pentru a alfa rezistența din formula I=U/R înlocuim tensiunea cu 5V- 0.7 și obținem o rezistență de 800 Ohm , cea mai apropiată valoare fiind 1kOhm, iar pentru a preveni activările greșite cu cauza sarcinilor electrice din mediul înconjurător putem adăuga o rezistență pull down de 10kOhm pentru a ne asigura că tranzistorul nu se activează din greșeală.

A screenshot of a computer

Description automatically generated

Fig. 4.15 Integratul responsabil cu încărcarea bateriei

Terminând cu proiectare pentru modul, putem trece la cheia digitală unde lucrurile sunt mai simple. Începem cu circuitul de încărcare, care după cum am stabilit v-a avea ca piesă principală integratul TP4056. Conform documentației [39] are nevoie de câteva legături simple cum ar fi alimentarea pinilor temp și Prog fiind trași la GND, pinul Prog având o rezistență de 2kOhm ce setează curentul maxim de încărcare la 0.4A , valoare mai scăzută, dar care nu v-a pune stres asupra bateriilor, generând căldură suplimentară. În rest , plusul bateriei conectat la acest modul, iar minusul fiind conectat prin modulul de protecție la masă.

A diagram of a circuit

Description automatically generated

Fig. 4.16 Configurarea portului USB pentru alimentarea cu 5V

Pentru a furniza integratul de încărcare cu 5V trebuie să configurăm portul USB-C pentru a solicita sursei 5V. Conform documentației oficiale [40] pentru a seta portul în așa fel încât să poată primi 5V , trebuie să conectăm pinii CC1 și CC2 la GND prin intermediul unor rezistențe de 5.1 kOhm [Fig. 4.16] adăugând opțional un condensator de 10 uF pentru afinisa unele posibile neregularități sau căderi de tensiune .

A diagram of a circuit

Description automatically generated

Fig. 4.17 Circuit protecție baterie

A diagram of a circuit

Description automatically generated

Fig. 4.18 Circuit Mosfet pentru conectarea și deconectarea bateriei

Acum că avem partea de alimentare și încărcare putem trece la partea de protecție a bateriei cu ajutorul integratului DW01A [Fig. 4.17] ce ne oferă protecție pentru supra descărcare, ce oprește circuitul de descărcare dacă tensiunea scade sub 2.4 V, supra încărcare, dacă tensiunea creste de 4.3V oprește încărcarea prin deconectarea bateriei de la circuitul de încărcare și scurtcircuit detectând imediat curenți foarte mari, datorită unei conexiuni defectuoase, prevenind deteriorarea bateriei[41]. Conectarea la circuit a integratului este simplă ă, acesta având nevoie de o conexiune la baterie , împreuna cu o rezistență de 100 Ohm adăugată bateriei pentru a limita curentul acesteia prin integrat , un condensator pentru finisarea tensiunii, iar restul pinilor , OC, VM , OD [ Fig. 4.18] fiind conectați la un mosfet [Fig. 4.18] ce se va ocupa de conectarea și deconectarea bateriei conform indicațiilor oferite de DW01A. Pentru mosfeturile ce se ocupă cu conectarea și deconectarea, am decis să folosim după modelul circuitelor deja prezente pe piață, integratul FS8205A ce oferă două mosfeturi conectate între ele , având un Drain comun[42] , Gate-urile fiind conectate la pinii OC și OD ai integratului DW01A , bateria la Source 2 și circuitul la Sourec 1 având conectat și pinul VM cu o rezistență de 1kOhm pentru a putea monitoriza tensiunea bateriei.

A diagram of a circuit

Description automatically generated

Fig. 4.19 Regulator Linear de tensiune TPS76933DBVR

Acum că avem încărcarea și protecția pregătite mai avem un pas până a ajunge la ESP și componentele de intrare și ieșire, acest pas fiind cel ar regulatorului linear pentru a limita tensiunea bateriei la maxim 3.3V . Regulatorul linear TPS76933DBVR, ca și oricare alt regulator linear nu are nevoie de alte componente externe pentru a putea funcționa, în acest caz tensiunea fiind prestabilită din fabrica la 3.3V , tot ce ne mai rămâne este sa conectăm bateria și pinul EN la minusul bateriei , adăugând niște condensatori pentru atenuarea fluctuațiilor tensiunii, pe pinul OUT obținând cei 3.3V doriți.

Având toate componentele de bază pregătite acum putem să ne îndreptăm atenția către ESP și componentele de intrare ieșire. Ca și în cazul modulului din interiorul mașinii v-om folosi un modul ESP32 fără componentele adiționale pentru programarea acestuia , deci la fel ca în acel caz , v-om adăuga butoane pentru a permite repornirea și programarea modulului , conform Fig. 4.12, urmând să conectăm componentele ce ne permit interacționarea cu utilizatorul și voltage deviderul ce ne permite monitorizarea tensiunii bateriei. Pentru a prelua comenzile dorite de utilizator v-om folosi 5 butoane pentru blocare , deblocare , lumini , pornirea motorului și aprinderea ecranului , butoane ce v-or fi conectate la minusul bateriei( după ce tensiunea trece prin regulatorul linear ) și cu niște rezistențe pull-up interne ale ESP-ului.

Pentru a afișa datele legate de tensiunea bateriei cheii și a mașinii , a stării închiderii centralizate , a becurilor și a motorului v-om folosi un ecran OLED de 1.3” [Fig. 4.20], SH1106 ce poate comunica cu ESP-ul prin intermermediul interfeței I2C, și afișa eficient datele dorite , urmând să implementăm și o funcție de stingere a ecranului după câteva secunde de inactivitate.

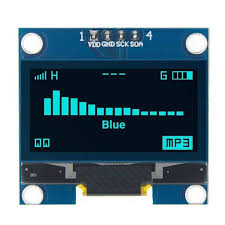


Fig. 4.20 Ecran OLED SH1106 I2C

Sursa: <https://uelectronics.com/producto/display-oled-azul-128x64-1-3-pulgadas-sh1106-i2c/>

Din Cauza componentelor alese, microcontrollere și IC-uri ajungem în punctul în care trebuie să proiectam PCB-ul ( Printed Circuit Board ) pentru a conecta cu ușurință componentele între ele și a reduce mărimea modulelor , mai ales forma cheii , organizând componentele în funcție de funcționalități așezându-le cât mai aproape posibil una de alta .

A blue circuit board with white and black circles and black dots

Description automatically generated

Fig. 4.21 PCB Cheie digitala

A blue circuit board with many small chips

Description automatically generated

Fig. 4.22 PCB Modul mașină

În figurile 4.21 și 4.22 putem observa un exemplu în care să organizăm componentele, firește acestea putând fii optimizate mai bine , dar funcționalitatea este cea mai importantă în această situație.

**4.2 Programarea funcționalității de Server pe modulului ESP32**

Pentru a funcționa ca server pentru sistemul nostru , modulul ESP32 trebuie să ofere funcționalitățile unui server Bluetooth Low Energy pentru a permite comunicării între cheia digitală securizată și modulul UCL , acesta prelucrând mesajele primite și transmiterea lor mai departe în forma corectă.

Pentru buna funcționare a serverului, modulul ESP32 trebuie să îndeplinească următoarele funcții principale și de bază :

* Inițializarea și configurarea serverului BLE
* Crearea caracteristicilor BLE pentru trimiterea și recepția datelor
* Funcționalități criptografice pentru prelucrarea datelor
* Adăugarea de date suplimentare pentru determinarea distanței dintre modul și client
* Citirea și transmiterea datelor către UCL prin intermediul comunicării serial

Pentru a începe programarea v-om utiliza Arduino IDE pentru scrierea de cod, compilare, upload și debugging, IDE creat special pentru programarea plăcuțelor Arduino , dar care din meniul „Boards manager” putem căuta și instala toate librăriile necesare scrise de Espressif. Începând cu inițializarea serverului BLE , avem nevoie de librăriile de bază Bluetooth Low Energy BLEDevice , BLEUtils și BLEServer pentru a crea serverul și a începe configurarea acestuia.

 BLEDevice::init("ESP32");

//Inițializează dispozitivul BLE și setează numele său ca “ESP32”

  server=BLEDevice::createServer();

//Crează serverul BLE ce gestionează conexiunile cu clienții săi și stochează pointerul în variabila globala server

  server->setCallbacks(new MyServerCallbacks());

//Asociază un set de callback-uri personalizate în care se v-or trata individual evenimentele de conectare și deconectare a serverului

  service=server->createService(BLEUUID(SERVICE\_UUID) , 32);

//Crează service-ul principal căruia îi atribuie UUID-ul SERVICE\_UUID și aloca 32 handles pentru caracteristici și descriptori

Având serverul și service-ul putem trece la crearea și customizarea caracteristicilor.

caracteristic = service->createCharacteristic(CHARACTERISTIC\_UUID, BLECharacteristic::PROPERTY\_READ |

                  BLECharacteristic::PROPERTY\_WRITE|

                  BLECharacteristic::PROPERTY\_NOTIFY |

                  BLECharacteristic::PROPERTY\_INDICATE);

//Crează caracteristica BLE asociată serviciului , pointer stocat în variabila characteristic ce v-a fi folosită pentru trimiterea de date

//Proprietatile sunt UUID-ul caracteristicii pentru identificare și proprietățile ce permit clientului să citească și scrie date și trimiterea de notificări și indicații

  caracteristic->addDescriptor(new BLE2902());

//Adaugă un descriptor caracteristicii , descriptor ce permite clientului să activeze sau dezactiveze notificările pentru această caracteristică

  caracteristicRecive=service->createCharacteristic(CHARACTERISTIC\_UUID\_2 , BLECharacteristic::PROPERTY\_READ |

                  BLECharacteristic::PROPERTY\_WRITE|

                  BLECharacteristic::PROPERTY\_NOTIFY |

                  BLECharacteristic::PROPERTY\_INDICATE);

  BLE2902 \* desc= new BLE2902();

  desc->setNotifications(true);

//Crează un nou descriptor și configurează opțiunea de notificări pentru a fi activate implicit pentru această caracteristică

caracteristicRecive->addDescriptor(desc);

//Asociază descriptorul caracteristicii caracteristicReceive

  caracteristicRecive->setCallbacks(new MyCharacteristicCallback());

//Setează un set de callback-uri personalizate pentru caracteristica characteristicReceive pentru a prelucra datele primite de la client

Având caracteristicile create și configurate, cu o clasă de callback-uri pentru prelucrarea datelor putem trece la decriptarea datelor primite și setarea cheii de sesiune.

void onWrite(BLECharacteristic\* data,esp\_ble\_gatts\_cb\_param\_t\* param){

//Metoda suprascrisă din clasa de baza

    if(availableSessionKey)// Daca este prezentă o cheie de sesiune , putem trece la procesarea datelor

    {

      decrypt(data->getData(),sessionKey,iv,output);

//Metodă pentru decriptarea datelor folosind AES128

      ::esp\_ble\_gap\_read\_rssi(param->read.bda);// metoda pentru obținerea RSSI

    }

    else

    {

      if(xSemaphoreTake(serialMutex,portMAX\_DELAY))

//Daca nu există o cheie de sesiune și primi un pachet de la client, înseamnă că este cheia de sesiune , astfel ne v-om folosi de un semafor Mutex pentru a putea salva această chei în siguranță fără a fi opriti de operațiile asincrone ale BLE

      {

        decrypt(data->getData(),masterKey,iv,sessionKey);

        availableSessionKey=true;

        output[0]=0x96;

        output[1]=0xAC;

        for(short i=0;i<16;i++) {

          Serial1.write(output[i]);

          output[i]=0x00;

        }

        xSemaphoreGive(serialMutex);

      //Decriptăm cheia de sesiune cu cheia master , salvând-o în memorie și anunțând UCL de prezența acesteia , urmând să eliberăm semaforul

}

    }

  }

Prin utilizarea metodei ::esp\_ble\_gap\_read\_rssi() ajungem la funcționalitatea de adăugare a datelor suplimentare pachetului original pentru a putea determina distanta aproximativă. Pentru folosirea acestei metode este nevoie de furnizarea unei metode ce poate fi apelată pentru acest eveniment.

BLEDevice::setCustomGapHandler(my\_gap\_event\_handler);

După apelarea metodei ::esp\_ble\_gap\_read\_rssi() pentru obținerea RSSI (Received signal strength indicator) , metoda predefinită de noi pentru tratarea acestor evenimente v-a fi apelată , metoda ce v-a primi ca parametrii date referitoare la datele transferate.

static void my\_gap\_event\_handler(esp\_gap\_ble\_cb\_event\_t event, esp\_ble\_gap\_cb\_param\_t\* param) {

  if(xSemaphoreTake(serialMutex, portMAX\_DELAY))

//Preluăm semaforul pentru a putea prelucra datele în siguranță

    {

      output[13]=(byte)(-param->read\_rssi\_cmpl.rssi);

      //Adaugăm valoarea RSSI obținuta din parametrul param, modificând-o în valoare pozitivă

for(short i=0;i<16;i++)

      {

        Serial1.write(output[i]);

        output[i]=0x00;

      }

//Transmitem datele către UCL folosind Serial1

      xSemaphoreGive(serialMutex);

//Eliberăm semaforul

    }

}

Având implementată transmiterea datelor către UCL , mai rămâne citirea datelor trimise de acesta, pentru a putea fi criptate și transmise clientului. Pentru obținerea acestei funcționalități ne putem folosi de funcția loop , în care v-om verifica Serial1 pentru datele primite de la UCL, în cazul în care se află 16 bytes în buffer (un pachet ) se poate citi , cripta și trimite clientului.

**4.3 Programarea funcționalității UCL Arduino Mega**

Pentru implementarea modului UCL (Unitatea de control si logica) , ce are la bază Arduino Mega, avem de luat în vedere funcționalitățile principale ale acestuia pentru buna operare și a menținerii securității sistemului Keyless Entry .Prin implementarea acestor funcții în cod avem de luat în considerare următoarele:

* Inițializarea comunicării Serial cu modulul ESP32
* Controlul hardware-ului mașinii prin intermediul interfețelor proiectate
* Securitatea comunicării și gestionarea sesiunilor de comunicare cu clientul
* Funcționalitate failsafe pentru accesul în caz de urgență

Pentru dezvoltarea codului v-om folosi Arduino IDE , începând cu determinarea modului intrare/ieșire pentru pinii responsabili cu senzorii pentru temperatură , HALL și interacțiunea cu interfața mașinii și configurarea Serial-ului pentru comunicarea cu serverul BLE, folosind o viteză mare, de 115200 bps pentru ca datele sa fie trimise între componente cât de repede posibil , fără a se putea aduna date primite de la client și pachetele vechi să se afle încă în faza de transmitere prin serial.

După toate configurările din funcția setup(), ajungem în loop ,unde UCL v-a verifica constant daca a primit date prin serial, urmând să efectueze operațiile corespunzătoare pentru fiecare comandă sau pachet pentru handshake și să transmită înapoi , după încheierea handshake-ului, datele referitoare la starea mașinii. După Transmiterea datelor , sau dacă nu s-a primit în primul rând, UCL verifică starea butonului de pornire și parametrii ce permit acțiune de a porni/opri motorul , urmând la final să verifice modulul RFID pentru a vedea dacă se află cartela necesară pentru operațiunile de failsafe.

Pentru funcția de keyless entry, deblocare/blocare automată în funcție de distanță v-om folosi RSSI-ul (Received Signal Strength Indicator) datelor primite, iar în funcție de valoarea acestuia urmând să efectuăm acțiunile corespunzătoare. Deși în interior, se poate determina poziția unui nod într-o rețea wireless cu un grad acceptabil de precizie, în funcție de aplicație, luând în considerare reflexiile semnalului și obstacolele ce le poate întâlni [55], în cazul unui sistem keyless entry ce are nodurile în aer liber ,valoarea RSSI nu ne poate oferi în mod direct o valoare ce sa corespundă unei distanțe datorită factoriilor externi ce influențează semnalul, precum temperatura, umiditatea, exponentul pierderii pe cale (PLE), fiind nevoie de un algoritm ce să ia în calcul toți acești parametri[56]. Cu toate acestea , în cazul sistemului nostru nu avem nevoie de distanța nodului cu cea mai mare precizie, ci de o zonă suficient de mare în jurul mașinii în care să se înregistreze valori pentru deschiderea portierelor , și de o zonă mai micuță pentru permiterea pornirii motorului. Ținând cont de faptul că valorile RSSI pot varia cu plus/minus câțiva dB v-om folosi o valoare care depășită v-a declanșa blocarea portierelor, iar una diferită, mai micuță cu 5 dB pentru deblocare, astfel , dacă o valoare a declanșat un eveniment ,următoarea să nu declanșeze al doilea eveniment dacă are o variație micuță față de valoarea anterioară. Pentru un plus în filtrarea valorilor RSSI mai putem folosi un contor pentru a tine minte daca valoarea anterioară a declanșat același eveniment cu cea actuală, permițând acțiunea de blocarea/deblocarea, în caz contrar, daca valoarea anterioară a declanșat un eveniment diferit v-om reseta acest contor, astfel fiind nevoie de două valori consecutive din aceeași interval pentru declanșarea evenimentului, algoritm ce în cod v-a avea următoarea formă:

if(RSSI>VALUE\_LOCK)

  {

    if(firstLockSent)

      if(!locked && !ovverride && keylessEnabled) lock();

    firstLockSent=true;

    firstUnlockSent=false;

  }else if(RSSI<VALUE\_UNLOCK){

    if(firstUnlockSent)

      if(locked && !ovverride && keylessEnabled) unlock();

    firstUnlockSent=true;

    firstLockSent=false;

  }

Urmând ca valorile VALUE\_LOCK și VALUE\_UNLOCK să fie înlocuite cu valori precise după testare, cu o diferență de 5/10 dB, astfel ușurând sarcinile microprocesorului folosind câteva state-uri if și două valori predefinite , fiind mai ușor ca un calcul algebric complicat ce să ia în vedere toți factorii externi.

**4.4 Implementarea clientului BLE**

Implementarea clientului este o etapă importantă în finalizarea sistemului keyless entry, clientul având roluri importante , precum inițierea conexiuni, trimiterea datelor și comunicarea cu serverul printr-un mediu sigur din punct de vedere al securității, criptat, format de cei doi participanți. Acest proces v-a fi realizat pe două platforme distincte:

* O aplicație Android ce oferă o interfață grafică intuitivă și ușor de utilizat pentru utilizator , ce oferă acces rapid la acțiunile și datele referitoare la starea mașinii.
* Un modul ESP32 ce v-a avea rolul de cheie securizată digitală , ce v-a permite lipsa telefonului pentru interacțiunea cu mașina, permițând oferirea acesteia altor persoane pentru a utiliza vehiculul
  + 1. **ESP32**

Inițializarea clientului BLE este asemănătoare cu cea a serverului, însă sunt câteva aspecte ce fac diferența între un client și server, alături de algoritmul de funcționare. Clientul BLE ce v-a rula pe modulul ESP32 vine cu niște sarcini în plus față de serverul BLE pentru a permite conectarea și gestionarea conexiunii cu serverul, funcții ce includ:

* Căutarea și conectarea la server
* Generarea cheilor de sesiune
* Procesarea datelor
* Gestionarea componentelor de intrarea/ieșire pentru a putea interacționa cu utilizatorul

După cum am menționat anterior, inițializarea clientului este asemănătoare cu cea a serverului fiind nevoie de a inițializa dispozitivul cu ajutorul metodei BLEDevice::int(„”) ce v-a permite modulului să utilizeze funcțiile BLE , oferindu-i dispozitivului numele primit ca și parametru. Diferențele apar acum , următorul pas fiind crearea clientului BLE cu ajutorul metodei BLEDevice::createClient() și stocarea pointerului acestui obiect într-o variabilă globală.

Având clientul putem începe căutarea și conectarea cu serverul, având nevoie de un obiect BLEScan ce îl putem obține cu ajutorul metodei BLEDevice::getScan(), oferind scannerului un obiect cu metoda necesară pentru procesarea callbackurilor. După configurarea scannerului putem trece la utilizarea acestuia pentru căutarea și conectarea cu serverul , salvând datele referitoare la scanare activă , pointer către adresa serverului și timpul ultimei scanări în variabile globale.

if (!connected && pServerAddress == nullptr && !scanning) {//Verificăm dacă putem începe o nouă sesiune de scanare

        pBLEScan->start(2, false);//Pornim scanarea activă ce v-a dura 2 secunde și nu v-a fi începută din noua după terminare

scanning = true;

        tm = millis();

    }

    if (scanning && (millis() - tm > 2000)) {scanning = false; pBLEScan->clearResults();} // Dacă au trecut 2 secunde de la ultima scanare și nu s-a găsit serverul , resetăm parametru scanning și golim rezultatele scanării din memorie .

În clasa oferită obiectului scanning , ce moștenește clasa de bază BLEAdvertised DeviceCallbacks urmează implementarea metodei ce v-a primi ca parametru un dispozitiv obținut în urma scanării și verificarea acestuia.

class MyAdvertisedDeviceCallbacks : public BLEAdvertisedDeviceCallbacks {

    void onResult(BLEAdvertisedDevice advertisedDevice) override {

        if (advertisedDevice.isAdvertisingService(BLEUUID(SERVICE\_UUID))) {

// Verificăm dacă UUID-ul service-ului dispozitivului găsit este cel al serverului nostru, urmând să salvăm adresa serverului și să oprim scanarea dacă s-a găsit service-ul corespunzător

            pServerAddress = new BLEAddress(advertisedDevice.getAddress());

            pBLEScan->stop();

            scanning = false;

        }

    }

};

Revenind în loop, având adresa serverului putem începe conectarea cu acesta, urmând si configurarea caracteristicilor pentru a putea comunica prin intermediul acestora.

if (!connected && pServerAddress != nullptr)

        connectToServer();

void connectToServer() {

    if (pClient->connect(\*pServerAddress)) {

        BLERemoteService\* pRemoteService = pClient->getService(BLEUUID(SERVICE\_UUID));

        if (pRemoteService == nullptr) {

            return;

            pClient->disconnect();

        }

// Dacă conectarea la server a fost efectuată cu succes , încercăm să obținem service-ul serverului pentru a configura caracteristicile, în caz contrar ne deconectam de la server.

        characteristicSend = pRemoteService->getCharacteristic(BLEUUID(TARGET\_CHARACTERISTIC\_UUID\_SEND));

        characteristicReceive = pRemoteService->getCharacteristic(BLEUUID(TARGET\_CHARACTERISTIC\_UUID\_RECEIVE));

//Obținem caracteristicile responsabile pentru primirea și transmiterea datelor

        if (characteristicReceive) {

            characteristicReceive->registerForNotify([](BLERemoteCharacteristic\* pBLERemoteCharacteristic, uint8\_t\* pData, size\_t length, bool isNotify) {})

//setăm metoda responsabilă de procesarea datelor, ce v-a fi apelată în momentul în care serverul trimite date către client

Reușind conectarea cu succes la server și obținerea caracteristicilor, putem trece la generarea cheii de sesiune , alături de procesarea datelor . Pentru criptarea și decriptarea datelor v-om folosi metodele prezente și pe serverul BLE , alături de cheia master și cheia de sesiune, creată cu ajutorul metodei random() ce primește ca parametru un interval pentru numărul generat , utilizând un pin liber (floating pin) ce recepționează radiațiile electromagnetice și generează un număr aleatoriu folosit la crearea umărului nostru. Având aceste date clientul poate începe handshake-ul conform protocolului stabilit, urmând după finalizarea acestuia să proceseze datele și să obțină parametrii referitori la starea mașinii. Ajungând în loop modulul verifică starea fiecărui buton pentru a verifica eventuale apăsări , urmând sa detecteze butonul apăsat și funcția asociată acestuia, urmând să pornească display-ul pentru o perioadă de timp , să transmită acțiunea aferentă butonului și să afișeze datele pe ecran.

**4.4.2 Android**

Aplicația Android joacă un rol important în interacțiunea utilizatorului cu sistemul keyless entry , oferind acestuia o interfață grafica intuitivă pentru transmiterea de comenzi și vizualizarea stării vehiculului. Pentru dezvoltarea acestei aplicații s-a folosit Android Studio împreună cu limbajul de programare Java, implementând pentru comunicarea BLE API-ul nativ Android pentru Bluetooth Low Energy.

Pentru dezvoltarea acestei aplicații putem începe cu interfața grafică, aceasta având cel mai mic grad de dificultate , având ulterior posibilitatea să conectăm funcționalitatea cu UI-ul. Folosind elemente native de tip TextView pentru afișarea de text și butoane fără text, cu fundalul transparent , alături de un foreground cu o pictogramă PNG putem realiza butoane ce descriu funcționalitatea acestora prin intermediul pictogramelor acestora.

Trecând la partea de funcționalitate, trebuie să ținem cont de faptul ca lucrăm cu un sistem de operare care limitează acțiunile unei aplicații fără drepturile și permisiunile necesare, așadar dezvoltarea funcționalității v-a începe în fișierul manifest care descrie succint aplicația noastră, permisiunile și feature-ile pe care le v-a folosi. Pentru utilizarea funcțiilor Bluetooth avem nevoie de următoarele permisiuni:

* Android.permision.BLUETOOTH
* Android.permision.BLUETOOTH\_ADMIN
* Android.permision.BLUETOOTH\_SCAN
* Android.permision.BLUETOOTH\_CONNECT
* Android.permision.ACCESS\_FINE\_LOCATION
* Android.permission.ACCESS\_COARSE\_LOCATION

Deși la prima vedere permisiunile legate de locație nu au ce căuta în contextul unei aplicații ce folosește BLE , Android consideră că scanarea poate permite localizarea dispozitivului , astfel fiind nevoie de acestea pentru a utiliza funcțiile de scanare pentru dispozitive. Alături de aceste permisiuni mai trebuie precizat și un service ce v-a fi folosit pentru gestionarea funcționalității Bluetooth și permisiuni de power management precum android.permission.WAKE\_LOCK și android.permission.REQUEST\_IGNORE\_BATTERY\_OPTIMIZATIONS pentru a permite rularea aplicației și în background , fără ca sistemul de operare să oprească aplicația după o perioadă de timp .

Începem dezvoltarea funcționalității cu service-ul Bluetooth ce v-a avea rol de mediator între interfața grafică și controllerul Bluetooth ce se v-a ocupa de comunicarea efectivă cu serverul, acesta obținând un obiect wakelock ce permite rularea aplicației în background, eliberând manual weakelock-ul când service-ul este distrus sau cu un timeout oferit ca un parametru funcției acquire(). Având wakelock-ul putem obține o referință către obiectul BluetoothController ce v-a fi o clasă de tip singletone ce permite existența unei singure instanțe a acestei clase, urmând apoi crearea unui obiect Handler ce v-a avea rolul de a trimite periodic date către server pentru funcția de keyless entry, iar dacă un buton a fost apăsat, trimiterea comenzii respective, apelând metoda postDelayed() în care adăugăm un obiect runnable cu funcția în care oferim controller-ului acțiunea dorită a fi trimisă , și un delay, ce reprezintă durata de timp la care dorim să se apeleze funcția , metoda ce v-a fi apelată și în metoda din interiorul runnable pentru ca aceasta să se execute la nesfârșit sau până la oprirea aplicației. Pentru preluarea comenzilor de la UI v-om folosi un câmp public static , care dacă nu este modificat v-a rezulta în trimiterea pachetului pentru funcția keyless entry , iar în cazul în care UI-ul dorește să trimită o acțiune , modifică valoarea din acest câmp care ulterior v-a fi trimisă, resetându-se la valoarea default după .

Trecând la clasa BluetoothController intervine o diferență față de clientul implementat pe ESP32, în acest caz fiind necesar sa adăugăm serverul BLE în lista dispozitivelor împerecheate din setările dispozitivului, ulterior verificând în cod pentru serverul nostru din acea listă de dispozitive, conectându-ne la acesta și ulterior obținând caracteristicile pentru trimiterea și recepția datelor, cu callback-urile acestora, funcțiile de criptare fiind preluate de clasa AES ce utilizează metode de criptare prezente în librariile Java .

**Capitolul 5**

**Rezultate și concluzii**

După implementarea fiecărui modul cu funcționalitățile sale și testarea acestora, și stabilirea valorilor RSSI cu o diferență de 7 dB între valorile de blocare și deblocare, 86 respectiv 79, s-a ajuns la o zonă stabilă de acces în jurul autovehiculului de aproximativ 2m fără blocări sau deblocări aleatorii din pricina factorilor externi ce afectează valoarea RSSI, totodată oferind un grad ridicat de securitate, oferind acces doar dispozitivelor autorizate ce au în posesie secretul comun și etapele utilizate de protocolul implementat pentru acest sistem, totodată fiind eficient din punct de vedere energetic folosind tehnologii de comunicații wireless moderne și eficiente.

Această lucrare a reușit cu succes analizarea securității sistemelor keyless entry prezente pe piață, determinând punctele forte ale fiecărui sistem și totodată vulnerabilitățile acestora, alegând astfel tehnologii și componente ce să țină cont de un buget redus, disponibilitatea componentelor, compatibilitatea și nu în ultimul rând securitatea avansată a sistemului prin implementarea unui protocol propriu de comunicare ce să poată combate cele mai des întâlnite atacuri.

Această lucrare a fost structurată într-o serie de etape, prima fiind cea a analizării sistemelor disponibile la momentul actual pe piață și analiza acestora, observând vulnerabilitatea sistemelor de bază ce utilizează RF, la atacuri simple de tip replay sau jamming, iar a celor prezente pe modele premium ce presupun un grad ridicat de securitate, (în unele cazuri precum Kia absent), având vulnerabilități software, probleme în detectarea atacurilor de tip replay, man-in-the-middle, jamming și relay.

A doua etapă a fost cea a determinării componentelor și tehnologiilor pentru crearea unui sistem scalabil, accesibil și compatibil cu o serie de dispozitive, astfel ajungându-se la microcontrollere precum Arduino Mega și ESP32 ce respectă pe deplin aceste cerințe și la Bluetooth Low Energy ce oferă compatibilitatea cu marea majoritate a dispozitivelor prezente pe piață, totodată având un grad de eficiență ridicat, urmând după aceea dezvoltarea protocolului de comunicații. Protocolul a avut de respectat principii de bază în securitatea sistemelor informatice precum integritatea, disponibilitatea și confidențialitatea, totodată prevenind și blocând cele mai răspândite atacuri la care sistemele din comerț au picat pradă, ajungându-se la un protocol ce utilizează o cheie master de criptare ce are rol de shared secret, criptare realizată cu AES128, cel mai puternic algoritm de criptare simetrică, utilizându-se doar pentru criptarea cheii de sesiune, astfel asigurându-ne că partenerul de comunicație este unul autorizat și blocând atacurile de man-in-the-middle, efectuându-se între partenerii la comunicație un handshake în 3 pași astfel stabilind nonce-uri valabile pe perioada sesiunii de comunicare ce asigură autenticitatea pachetelor, pachete după handshake ce folosesc numere generate aleator pentru a oferi pachetelor asemănătoare un aspect total diferit după criptare, a counterelor ce previn atacurile de tip replay, a nonce-urilor specifice clientului și serverului, alături de un checksum, toate acestea oferind un grad ridicat de securitate, oferind vulnerabilitate atacurilor de tip relay cu ajutorul aplicației android, care poate fi închisă pe parcursul nopții sau în zonele aglomerate.

Ultima etapă a presupus implementarea celor precizate mai sus folosind limbaje aproape de nivelul hardware pentru dezvoltarea codului pentru microcontrollere, precum C++ și limbaj portabil și performant Java alături de sistemul de operare Android pentru a oferi interfața grafică ușor de utilizat și o experiență fără probleme la rulare, toate acestea asigurând securitatea ridicată a întregului sistem.

În concluzie, lucrarea demonstrează posibilitatea dezvoltării unui sistem keyless entry accesibil cu un grad de securitate mult mai ridicat față de majoritatea sistemelor prezente pe piață, oferind totodată direcții pentru cercetări viitoare și îmbunătățiri, precum folosirea UWB, sau utilizarea Bluetooth 6 ce vine cu funcții în plus pentru determinarea corectă a distanței dintre noduri.

**Capitolul 6**

**Bibliografie**

[1]Company History

URL: <https://group.mercedes-benz.com/company/tradition/company-history/1885-1886.html>

[2] From car keys to perfectly keyless

URL:<https://www.bosch.com/stories/history-car-locking-systems/>

[3] Smart Key

URL :<https://en.wikipedia.org/wiki/Smart_key>

[4] Oswald, David F. "Wireless attacks on automotive remote keyless entry systems." Proceedings of the 6th international workshop on trustworthy embedded devices. 2016 [5] Mathilde Carlier ,Annual passenger car sales in the United States from 1951 to 2023

2024 URL:<https://www.statista.com/statistics/199974/us-car-sales-since-1951/>

[6] Veera Korhonen, Reported motor vehicle theft rate in the United States from 1990 to 2023 2024 URL:<https://www.statista.com/statistics/191216/reported-motor-vehicle-theft-rate-in-the-us-since-1990/>

[7] Marcia Wendorf , Car Keyless Entry Systems May Invite Relay Attacks, 2019 URL:<https://interestingengineering.com/transportation/car-keyless-entry-systems-may-invite-relay-attacks>

[8] David Bonnici, Keyless entry hacking leaves popular cars vulnerable, 2019 URL:<https://www.whichcar.com.au/car-news/popular-cars-vulnerable-to-thieving-hackers>

[9] Keyless cars and motorcycles that ADAC was able to unlock and make off with illegally URL:<https://media.product.which.co.uk/prod/files/file/gm-295a2572-487e-4a04-8ee1-5e79b2c17ba2-car-adac.pdf>

[10] Joachim Bjørge Ulven, Gaute Wangen, A Systematic Review of Cybersecurity Risks in Higher Education , MDPI, 2021

[11] Penny Hoelscher, What is a relay attack (with examples) and how can you prevent them?, 2023 URL:<https://www.comparitech.com/blog/information-security/what-is-relay-attack/>

[12] From Key Fob to UWB: How Hackers Hijack Vehicle Entry Systems URL:<https://vicone.com/blog/from-key-fob-to-uwb-how-hackers-hijack-vehicle-entry-systems>

[13] Avijit Mallik, MAN-IN-THE-MIDDLE-ATTACK: UNDERSTANDING IN SIMPLE WORDS, Jurnal Pendidikan Teknologi Informasi ,2018 URL:<https://jurnal.ar-raniry.ac.id/index.php/cyberspace/article/view/3453/2707>

[14] Christopher Malau, Jam and Replay Attack on Vehicular Keyless Entry Systems, 2018 URL:<https://github.com/trishmapow/rf-jam-replay>

[15] Subodh Gangan , A Review of Man-in-the-Middle Attacks URL:<https://arxiv.org/pdf/1504.02115>

[16]Replay Attack

URL: <https://en.wikipedia.org/wiki/Replay_attack>

[17] Anti-theft software tamps down viral theft trend targeting Hyundai, Kia vehicles, 2024

URL:<https://www.iihs.org/news/detail/anti-theft-software-tamps-down-viral-theft-trend-targeting-hyundai-kia-vehicles>

[18] Ibrahim, Omar Adel, et al. "Key is in the air: Hacking remote keyless entry systems.

" Security and Safety Interplay of Intelligent Software Systems: ESORICS 2018 International Workshops*,* Springer International Publishing, 2019 URL:<https://cri-lab.net/wp-content/uploads/2018/08/CarHacking.pdf>

[19] Andrew J. Hawkins , Kia and Hyundai’s antitheft software update seems to be working 2024 URL:<https://www.theverge.com/2024/8/8/24216042/kia-boys-hyundai-thefts-drop-software-update>

[20] Andy Greenberg , Millions of Vehicles Could Be Hacked and Tracked Thanks to a Simple Website Bug, 2024 URL:<https://www.wired.com/story/kia-web-vulnerability-vehicle-hack-track/>

[21] Vehicle Theft Data , 2022

URL:<https://www.nicb.org/media/3329/download>

[22] Csikor, Levente, et al. "Rollback: A new time-agnostic replay attack against the automotive remote keyless entry systems." ACM Transactions on Cyber-Physical Systems 8.1, 2024

[23] Zheng, Yong, et al. "Automotive Security in the Digital Era: A Comprehensive Survey of Attacks and Defenses for Keyless Entry System." International Conference on Computational & Experimental Engineering and Sciences. Cham: Springer Nature Switzerland, 2024

[24] Wavelength Formula & Calculation

URL:<https://study.com/learn/lesson/wavelength-formula-calculate.html>

[25] Bluetooth

URL:<https://en.wikipedia.org/wiki/Bluetooth>

[26] Ababii Nicolai , COMPATIBILITATE ELECTROMAGNETICĂ URL:<https://else.fcim.utm.md/pluginfile.php/132895/mod_resource/content/1/CEM_3.pdf>

[27] Advanced Encryption Standard (AES), Information Technology Laboratory National Institute of Standards and Technology Gaithersburg 2023 URL:<https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197-upd1.pdf>

[28] Trusted Platform Module

URL:<https://en.wikipedia.org/wiki/Trusted_Platform_Module>

[29] Memory dump

URL:<https://nordvpn.com/cybersecurity/glossary/memory-dump/>

[30] Linear vs Switching Regulators: Taking Accurate Efficiency Measurements, 2023, URL:<https://www.renesas.com/en/document/apn/r34an0006-linear-vs-switching-regulators-taking-accurate-efficiency-measurements>

[31] ESP32 URL:<https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>

[32] I2C URL:<https://en.wikipedia.org/wiki/I%C2%B2C>

[33] Andrew Gausden, Burak Can Cerik, Single-Use Vape Batteries: Investigating Their Potential as Ignition Sources in Waste and Recycling Streams, 2024 URL:<https://www.mdpi.com/2313-0105/10/7/236>

[34] Long-awaited common charger for mobile devices will be a reality in 2024 , 2022 URL:<https://www.europarl.europa.eu/news/en/press-room/20220930IPR41928/long-awaited-common-charger-for-mobile-devices-will-be-a-reality-in-2024>

[35] LM2596DP-5-0 URL:<https://www.lcsc.com/datasheet/lcsc_datasheet_2006081112_HTC-Korea-TAEJIN-Tech-LM2596DP-5-0_C596341.pdf>

[36] TXS0102

URL:<https://www.lcsc.com/datasheet/lcsc_datasheet_1810292010_Texas-Instruments-TXS0102DCUR_C53434.pdf>

[37] 2N222

URL:<https://www.lcsc.com/datasheet/lcsc_datasheet_2403071527_CBI-2N2222AU_C21714170.pdf>

[38] Dispozitive Electronice Indrumator de laborator, Universitatea Politehnica din Bucureşti Facultatea de Electronică Telecomunicaţii şi Tehnologia Informaţiei, 2009

URL: <https://wiki.dcae.pub.ro/images/7/70/Tbipolar.pdf>

[39] TP4056 URL:<https://www.lcsc.com/datasheet/lcsc_datasheet_2409241408_UMW-Youtai-Semiconductor-Co---Ltd--TP4056_C725790.pdf>

[40] Universal Serial Bus Type-C Cable and Connector Specification ,2019, URL:<https://www.usb.org/sites/default/files/USB%20Type-C%20Spec%20R2.0%20-%20August%202019.pdf>

[41] DW01A

URL:<https://www.lcsc.com/datasheet/lcsc_datasheet_1912111437_PUOLOP-DW01A_C351410.pdf>

[42] FS8205A

URL:<https://www.lcsc.com/datasheet/lcsc_datasheet_2411121100_TECH-PUBLIC-FS8205A_C2830320.pdf>

[43]BLE

URL: <https://en.wikipedia.org/wiki/Bluetooth_Low_Energy>

[44] Mohammad Afaneh, Bluetooth Low Energy (BLE): A Complete Guide, 2022

URL:<https://novelbits.io/bluetooth-low-energy-ble-complete-guide/>

[45] Jon Gunnar Sponas , Things You Should Know About Bluetooth Range, 2023

URL: <https://blog.nordicsemi.com/getconnected/things-you-should-know-about-bluetooth-range>

[46] Cosmin Vasile , Bluetooth 6.0 goes officially, here are all the new features and improvements , 2024

URL:<https://www.phonearena.com/news/bluetooth-6-new-features_id162270>

[47] Chandana Daryani ,The New and Improved Bluetooth 6.0, 2024

URL:<https://www.silabs.com/blog/the-new-and-improved-bluetooth-6-0>

[48] William Stallings CRYPTOGRAPHY AND NETWORK SECURITY Principles and practice, 7th Edition, Pearson Education Limited 2017

[49]How long would it take to brute force an AES-128 key

URL:<https://crypto.stackexchange.com/questions/48667/how-long-would-it-take-to-brute-force-an-aes-128-key>

[50] Avalanche Effect in Cryptography

URL:<https://www.geeksforgeeks.org/avalanche-effect-in-cryptography/>

[51] MAC randomization behavior

URL:<https://source.android.com/docs/core/connect/wifi-mac-randomization-behavior>

[52]Ros J. Anderson , Security Engeneering , Wiley Publishing, Inc, 2008

URL:<https://pub.deadnet.se/Books_on_Tech_Survival_woodworking_foraging_etc/security_engineering_a_guide_to_building_dependable_distributed_systems.pdf>

[53] Burrows–Abadi–Needham logic

URL:<https://en.wikipedia.org/wiki/Burrows%E2%80%93Abadi%E2%80%93Needham_logic>

[54] TCP 3-Way Handshake Process

URL:<https://www.geeksforgeeks.org/tcp-3-way-handshake-process/>

[55] Türkoral, T., Tamer, Ö., Yetiş, S., İnanç, E., & Çetin, L. (2017). Short range indoor distance estimation by using RSSI metric. *Electrica Journal, 17*(2), 3295–3302

URL: <https://electricajournal.org/Content/files/sayilar/58/3295-3302.pdf>

[56] Luomala, J., & Hakala, I. (2019). Analysis and evaluation of adaptive RSSI-based ranging in outdoor wireless sensor networks. Ad Hoc Networks, 87, 100-112

URL: <https://www.sciencedirect.com/science/article/pii/S1570870518307297>