Optimizarea procesului de încărcare a firmware-ului pe un microcontroler

Munteanu Ciprian-Marian

REZUMAT

Proiectul își propune încărcarea firmware-ului pe un microcontroler într-un mod rapid și securizat prin păstrarea integrității datelor și a zonei de memorie.Scrierea greșită sau cu date eronate poate duce la deterioarea unor zone de cod rezervate, pierderea unor funcționalități sau distrugerea în mod ireversibil a firmware-ului existent.

Proiectul realizează optimizarea acestui proces prin aducerea următoarelor îmbunătățiri:

- siguranța comunicației între microcontroler și dispozitivul care ne transmite datele
- folosirea în mod sigur și eficient a memoriei
- trimiterea datelor într-un timp cât mai mic
- posibilitatea ca o modificare apărută în firmware(îmbunătățire , funcționalitate nouă, rezolvare bug) să impacteze într-o măsură mică firmware-ul deja existent pe microcontroler.

Programul este construit pentru a putea fi folosit pentru microcontrolere din industria auto deci, sunt unele particularități de implementare ce țin de standardele folosite . Mai jos este un tabel cu modelul OSI și nivelele folosite.

Aplicație	UDS
Prezentare	-
Sesiune	-
Transport	ISOTP
Rețea	ISOTP
Legătura de date	CAN
Fizic	CAN

La nivelul fizic și legătura de date vom folosi protocolul CAN(Controller Area Network) pentru a implementa comunicația între microcontrolere și nodul central. Rețeaua CAN este o rețea de tip multi-master în care toate nodurile sunt conectate între ele prin magistrala de CAN. Orice nod poate pune mesaje pe CAN , mesaje care sunt recepționate de toți membri rețelei, dar care sunt interpretate doar de membri pentru care au fost destinate, ceilalți ignorand mesajele.

Pentru nivelul rețea și transport vom folosi standardul ISOTP(15765-2) , standard internațional de trimitere a pachetelor de date pe magistrala de CAN . ISOTP presupune folosirea a 4 tipuri de cadre:

- SF(Single Frame) - cadru pentru mesaj până la 7 bytes
- FF(First Frame) - mesaj mai mare de 7 bytes , conține lungimea pachetului de trimis și datele inițiale
- CF(Consecutive Frame) - conține restul datelor din pachet mai mari de 7 bytes
- FCF(Flow Control Frame) – folosit cand primim un răspuns mai mare de 7 bytes de la destinatar, asemănător cu CF

Nivelurile sesiune și prezentare nu sunt implementate.

La nivelul aplicației vom folosi protocolul UDS(Unified Diagnostic Services), protocol de diagnoză folosit în domeniul auto pentru a controla funcțiile unui microcontroler. Acest protocol are o serie de 6 funcționalități, iar fiecare funcționalitate are o serie de servicii.

Pentru a realiza siguranța căii de comunicație între microcontroler și dispozitiv vom folosi un serviciu de acces securizat la microcontroler printr-un concept de tip “Seed and Key”, în care atât dispoztivul prin care încărcăm firmware-ul cât și microcontroler au o cheie privată, cu ajutorul cheii și a unui seed trimis de microcontroler se construiește un mesaj securizat .

În funcție de memoria microcontroler-ului codul va fi scris în anumite zone de memorie astfel încât să nu interfereze cu zonele rezervate iar scrierea să necesite cât mai puțin resursele microcontrolerului.

Îmbunătățirea timpului de trimitere a datelor pe comunicație va fi realizată printr-un concept de tip “delta file”, în care vom cunoaște versiunea firmware-ului de pe microcontroler, versiunea noului firmware iar pe baza lor vom calcula doar diferența dintre ele(delta file), delta file-ul va fi trimis pe microcontroler , iar în interior microcontroler își reconstruiește zona care trebuie modificată salvând astfel modificarea celorlalte zone dacă nu sunt afectate de schimbare. Acest lucru este un alt bonus la securitate deoarece nu trimitem întreg fișier cu date.

Prin acest proiect dorim să aducem îmbunătățiri în procesul de sciere a firmware-ului pe un microcontroler, care să ducă la o scriere mai rapidă, eficientă și securizată.