

CAN platforma za logovanje i nadzor parametara u vozilima

Duško Gajinović, Đorđe Novaković, *Member, IEEE*

Apstrakt—U ovom radu prikazan je primer platforme za komunikaciju preko CAN mreže u vozilima. Osnovu projekta čini Arduino Uno ploča, na kojoj se nalazi ATmega328 mikrokontroler, koji preko SPI protokola komunicira sa CAN Bus dodatnom pločom. Na istoj ploči se nalazi MCP2515 CAN kontroler, koji preko MCP2551 uspostavlja komunikaciju sa samom CAN mrežom. Sastavni deo ove platforme je i GUI program, koji je realizovan u Python programskom jeziku, pomoću kojeg se mogu pratiti, filtrirati i slati poruke na CAN mrežu. Uređaj poseduje i funkcije koje su specifične za OBD2 protokol, koji se nalazi u većini modernih automobila, tako da se mogu pratiti razni parametri koji inače nisu dostupni vozaču.

Ključne reči—CAN Bus; Logger; OBD2; Arduino Uno; Python; GUI; PyQt5.

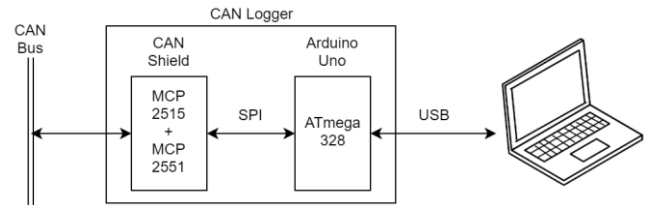
I. UVOD

Intenzivan razvoj automobilske industrije doveo je do uvođenja visokih standarda u segmentu kontrolnih sistema u okviru vozila. Takvi sistemi moraju posedovati veliki kapacitet i brzinu, sposobnost rada u realnom vremenu, kao i visoku pouzdanost. Koncept CAN[1] mreže prvi put se pojavio osamdesetih godina prošlog veka, kada je kompanija Bosch zajedno sa Intelom razvila novi uređaj za prenos podataka. Controller Area Network je serijski komunikacioni protokol projektovan upravo po ovim standardima i danas predstavlja dominantni protokol u automobilskim mrežnim sistemima.

Projekat je baziran na Arduino Uno[2] mikrokontrolerskoj ploči, koja pomoću CAN BUS Shield-a[3] vrši komunikaciju sa CAN mrežom. U okviru projekta se nalazi i program pomoću kojeg se mogu pratiti, filtrirati i slati poruke na CAN mreži. Takođe postoje određene funkcije koje su specifične za OBD2[4] protokol koji se nalazi u većini modernih automobila, tako da se ovim uređajem mogu pratiti razni parametri koji inače nisu dostupni vozaču. Sam mikrokontroler ne vrši nikakvu obradu podataka, već služi kao most između CAN mreže i računara. Sva obrada podataka se vrši na računaru u programu “CAN Logger”, koji je pisan u Python[5] programskom jeziku.

Duško Gajinović – Fakultet tehničkih nauka, Univerzitet u Novom Sadu, Trg Dositeja Obradovića 6, 21000 Novi Sad, Srbija (e-mail: gajinovic.dusko@gmail.com)

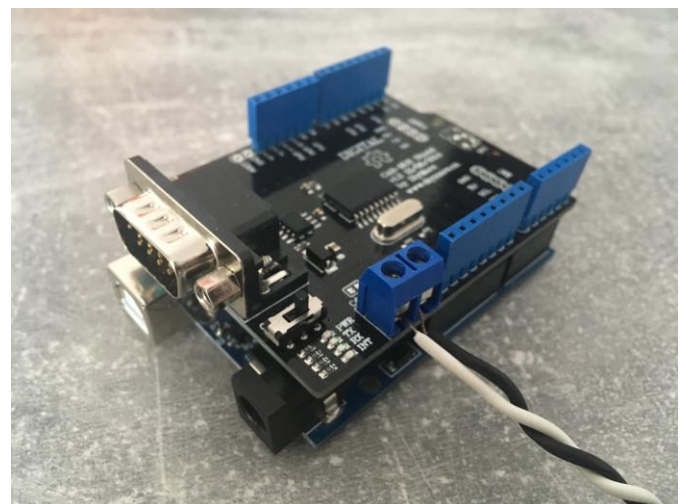
Đorđe Novaković - Fakultet tehničkih nauka, Univerzitet u Novom Sadu, Trg Dositeja Obradovića 6, 21000 Novi Sad, Srbija (e-mail: djordjenovakovic@uns.ac.rs)



Sl. 1 Blok šema projekta

II. HARDVER

Sa hardverske strane, projekat nije previše kompleksan. Bazira se na Arduino Uno ploči na kojoj se nalazi ATmega328[6] mikrokontroler kao i USB to TTL čip preko kojeg mikrokontroler komunicira sa računarem. Sa druge strane imamo MCP2515 [7] koji je samostalni CAN kontroler, a komunikacija između njega i ATmega328 se vrši preko SPI[8] protokola. MCP2515 je zadužen za slanje i primanje poruka sa CAN mreže, i on radi nezavisno od mikrokontrolera. Bitno je napomenuti da njega nije moguće direktno vezati na CAN mrežu, već mora da se koristi primopredajnik, koji je u ovom slučaju MCP2551 [9]. Kako bi se izbegle smetnje i održala stabilnost mreže, na samoj ploči se nalazi i terminacioni otpornik od 120 oma. Ploča se može povezati sa CAN mrežom preko CAN H(high) i CAN L(low) terminala ili preko DB9 konektora. Povezivanje sa CAN mrežom preko terminala prikazano je na slici 2.



Sl. 2 Arduino Uno i CAN Bus ploča

Za komunikaciju sa vozilom se koristi kabel koji sa jedne strane ima OBD2 konektor, a sa druge DB9, što se i vidi na slici 3.



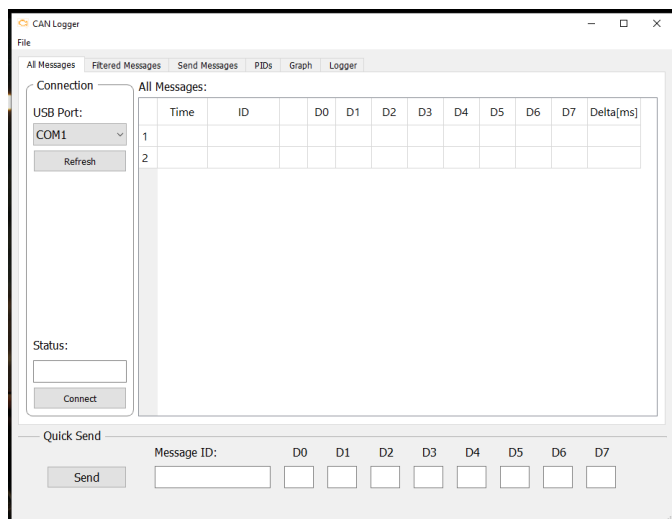
Sl. 3 CAN Logger sa OBD2 konektorom

III. PROGRAM I GUI

Program je realizovan u Python programskom jeziku, a sam GUI je dizajniran i implementiran pomoću PyQt5 [10] biblioteke i njenog Designer-a. Kada se pokrene program, prikazuje se glavna stranica “All Messages” preko koje korisnik može da uspostavi konekciju sa uređajem.

A. All Messages Tab

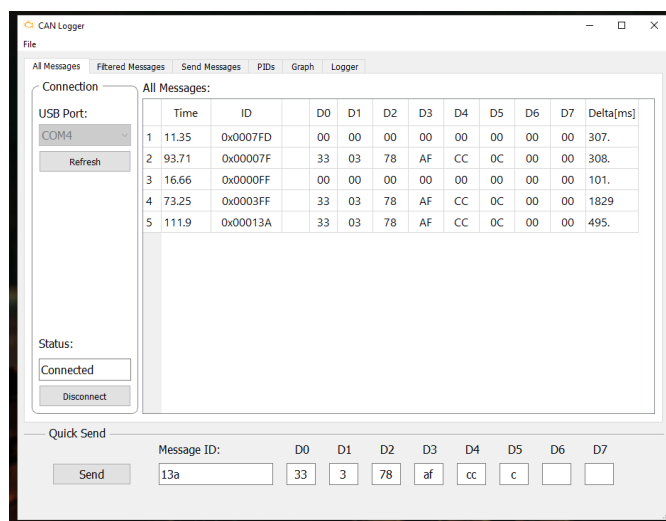
Sa leve strane se nalazi “Connection” grupa, gde se može izabrati na kom COM portu se nalazi uređaj i pritiskom na dugme “Connect” se uspostavlja veza. Nakon uspostavljanja veze, prikazuje se trenutno stanje uređaja, kao i opcija zatvaranja komunikacije dugmetom “Disconnect”. Nakon što se uspostavila konekcija sa uređajem, sve poruke koje se nađu na CAN mreži se prikazuju u sledećoj tabeli.



Sl. 4 All Messages Tab

Od trenutka kada se CAN Logger poveže na mrežu i kada uspostavi kontakt sa programom, preko USB porta, on počinje da osluškuje i očitava sve poruke koje se nađu na mreži.

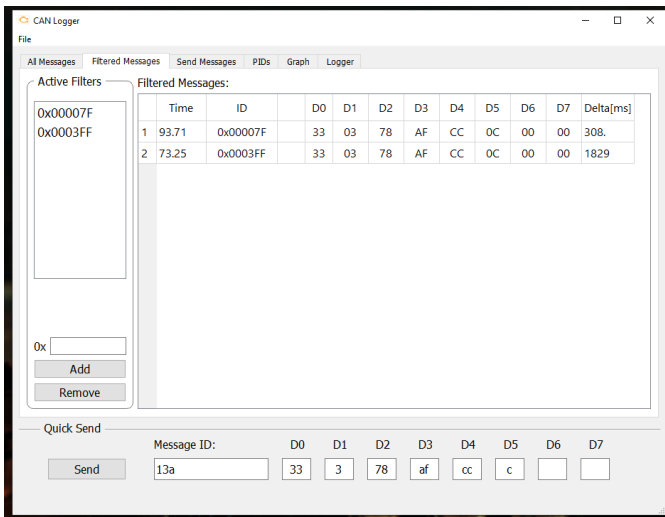
Kada MCP2515 detektuje početak poruke, on je beleži njen ID i sve data bitove u svoj interni registar. Nakon završetka poruke, proverava da li je poruka u celini preuzeta bez grešaka, i onda je šalje mikrokontroleru putem SPI protokola. Nakon preuzimanja poruke, mikrokontroler je formatira na određeni način, da ne bi došlo do greške pri očitavanju od strane programa na računaru. Sam firmver, koji se nalazi na mikrokontroleru je pisan tako da očitavanje poruke, formatiranje i slanje računaru traje što kraće, kako bi se izbeglo propuštanje novih poruka na CAN mreži. Kada program na računaru preuzme paket koji sadrži sve informacije o poruci, on ga raspakuje i redefiniše kako bi se lakše i brže sama poruka smestila u računarsku memoriju. Poruke se skladište na odgovarajući način kako bi se kasnije pretraživanje i sortiranje lakše obavljalo. Osim osnovnih parametara poruke, kao što su njen ID i osam data bajtova, samoj poruci se dodeljuje još par parametara nakon što se preuzme od mikrokontrolera. Kako bi lakše pratili poruke koje se nalaze na mreži, svakoj poruci dodeljujemo vremensku vrednost, koja nam prikazuje kada smo primili poruku u odnosu na trenutak kada je veza sa CAN Logger pločom uspostavljena. Takođe je bitno napomenuti da svaka poruka koja se nađe na mreži, a ima isti ID neće biti prikazana kao nova poruka u tabeli, već će parametri kao što su vreme poruke i njene data vrednosti biti ažurirane. Svaka poruka ima i polje pod imenom “Delta [ms]” pomoću kojeg možemo videti da li se neka poruka šalje na mrežu periodično, i sa kojim vremenskim periodom.



Sl. 5 All Messages Tab - Poruke

B. Filtered Messages Tab

U ovom prozoru su poruke čiji se ID nalazi u listi sa leve strane, čime se smanjuje broj poruka na ekranu i lakše se prate samo one koje nas zanimaju.

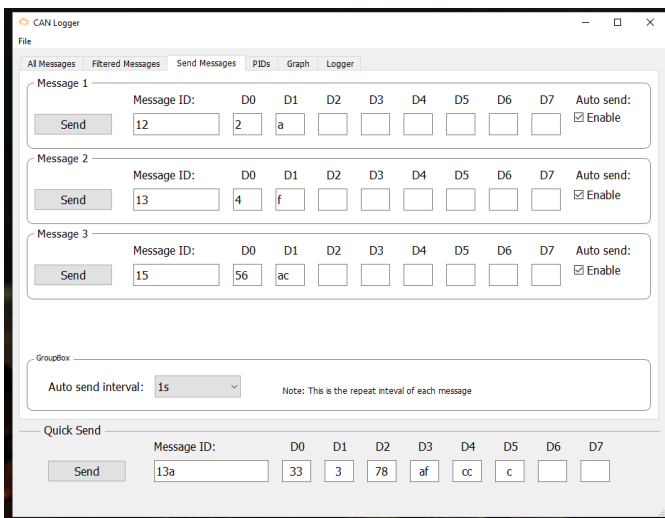


Sl. 6 Filitered Messages Tab

Filter se dodaje tako što se upiše ID poruke koju želimo da pratimo i pritiskom na dugme “Add” se taj ID dodaje u listu filtera. Takođe se u svakom trenutku može i ukloniti željeni ID sa liste.

C. Send Messages Tab

Ako želimo da šaljemo više poruka periodično, to se može postići preko Send Messages prozora. Postoje tri grupe poruka i one se mogu slati nezavisno jedna od druge pritiskom na taster “Send”.



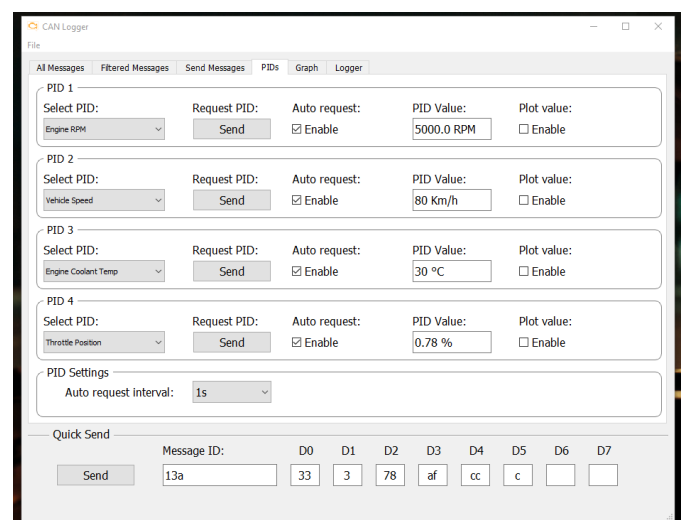
Sl. 7 Send Messages Tab

Moguće je i periodično slanje poruka označavanjem polja “Auto send Enable” sa desne strane prozora, i ta opcija je dostupna za svaku poruku posebno. Period slanja poruka se bira na dnu stranice padajućim menijem “Auto send interval”, koji se kreće od 1 s do 1 minuta. Interval između poruka se automatski skalira u zavisnosti od broja selektovanih poruka, tako da opterećenje same CAN mreže bude minimalno. Ako je, na primer, selektovano periodično slanje za dve poruke, na interval od jedne sekunde, period između te dve poruke će biti

500 ms, kako bi se izbeglo preterano opterećivanje mreže. U slučaju da se aktivira i treća poruka, period će se spustiti na 333 ms, i naravno vreme se skalira u zavisnosti perioda slanja.

D. PIDs (Parameter ID) Tab

Ovaj prozor je specifičan za OBD2 port koji se nalazi na modernijim vozilima i pridržava se SAE J1979 [11] standarda. Ako je automobil opremljen ovim portom, moguće je pristupiti određenim parametrima unutar vozila. Kako bi došli do vrednosti samih parametara potrebno je prvo poslati zahtev automobilu u vidu CAN poruke. Većina parametara koji su dostupni putem OBD2[12] protokola su standardizovani i zahtevi za njih su dostupni svima. Naravno, postoje određene funkcije koje su specifične za svakog proizvođača koji nisu u obavezi da se pridržavaju bilo kakvih standarda. Pomoću ovog programa moguće je pristupiti oko dvadesetak različitih parametara, od kojih su obrtaji motora, brzina vozila, temperature rashladne tečnosti, pozicija papučice gasa i protok vazduha jedne od najbitnijih. Kada se pritisne taster “Send”, program na osnovu padajućeg menija sa leve strane pravi zahtev za željeni parametar i šalje ga samom mikrokontroleru. Preuzeti paket se parsira, izdvaja se ID poruke kao i njeni data bajtovi i šalju se MCP2515 kontroleru koji ih prosleđuje na mrežu. Ubrzo nakon poslanog zahteva, automobil odgovara sa novom porukom, poznatog ID-a, koja sadrži vrednost parametara u bajtovima D2-D5 iz kojih se uz pomoć predefinisanih formula može izračunati željena vrednost.

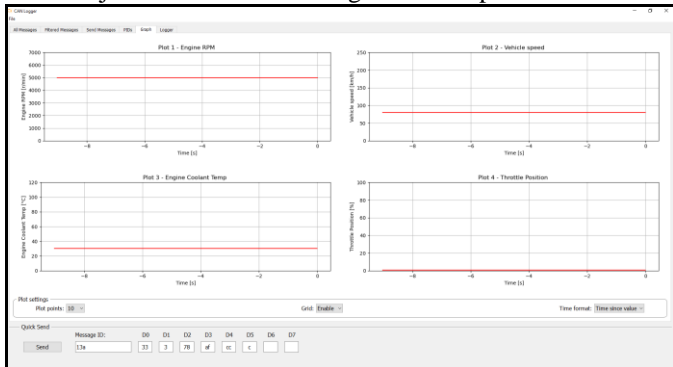


Sl. 8 PIDs Tab

CAN Logger dozvoljava praćenje do 4 parametara u isto vreme. Sa leve strane prozora se nalaze padajuće liste preko kojih biramo koje parametre želimo da pratimo. Možemo tražiti vrednost parametra pritiskom na dugme “Send” i njegova vrednost će se pojaviti u polju sa desne strane. Takođe se može uključiti automatsko osvežavanje za svaki od odabranih parametara, kao i interval osvežavanja preko padajuće liste na dnu prozora.

E. Graph Tab

U prethodnom prozoru "PIDs Tab" pored svake očitane vrednosti imamo opciju "Plot Enable", što uključuje iscrtavanje očitane vrednosti na grafik u Graph Tab-u.

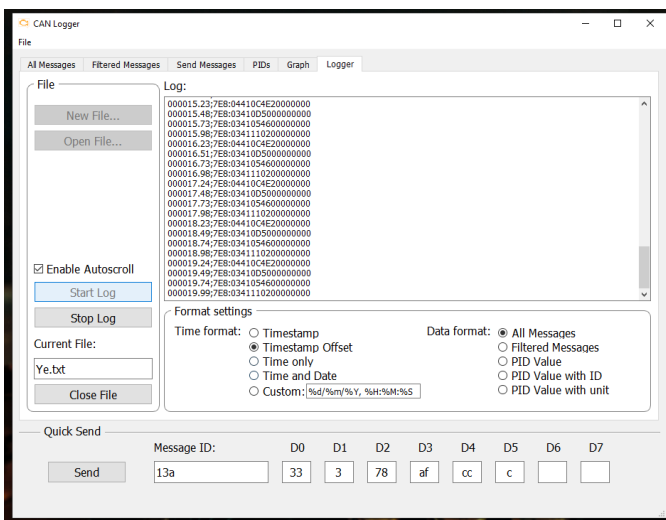


Sl. 9 Graph Tab

Iznad svakog grafika piše koji parametar se prikazuje, kao i njegova merna jedinica. Na X osi se prikazuje pre koliko sekundi je bila očitana vrednost. Ovo se može promeniti putem padajućeg menija u donjem desnom uglu "Time format". Takođe je moguće isključiti koordinatnu mrežu. Sa leve strane se može izabrati koliko će se tačaka unazad iscrtavati na graficima. U slučaju da se u prethodnom prozoru promeni parametar koji pratimo, grafik će automatski skalirati svoju minimalnu i maksimalnu vrednost, tako da prikazuje pun opseg trenutnog parametra.

F. Logger Tab

Ovaj prozor nam daje mogućnost da poruke čuvamo u tekstualnoj datoteci na računaru. Potrebno je napraviti ili otvoriti već postojeću datoteku, i nakon toga pritiskom na taster "Start Log" se ispisuje zaglavlje tekstualnog fajla i započinje zapisivanje svake poruke na mreži.



Sl. 10 Logger Tab

Ispod terminala koji prikazuje poruke postoje opcije za menjanje formata zapisa poruka.

IV. ZAKLJUČAK

Nakon realizacije mogu da zaključim da ova platforma uspešno komunicira sa CAN mrežom i kao takva otvara mogućnosti za dalje unapređivanje. Plan je da se implementira Bluetooth veza između ploče i računara, kao i skladištenje i praćenje podataka preko interneta.

ZAHVALNICA

Zahvaljujem se profesoru Platonu Sovilju na podršci i pomoći prilikom izrade ovog rada, kao i svim kolegama sa Katedre za električna merenja, na Fakultetu tehničkih nauka u Novom Sadu.

LITERATURA

- [1] Wikipedia, "Controller Area Network" [Online], Dostupno: https://en.wikipedia.org/wiki/CAN_bus
- [2] Arduino, "Arduino-Uno" [Online], Dostupno: <https://store.arduino.cc/arduino-uno-rev3>
- [3] Seed Studio, "CAN Bus Shield" [Online], Dostupno: <https://www.seedstudio.com/CAN-BUS-Shield-V1-2-p-2256.html>
- [4] Wikipedia, "OBD2" [Online], Dostupno: https://en.wikipedia.org/wiki/On-board_diagnostics
- [5] Python, "Python" [Online], Dostupno: <https://www.python.org/>
- [6] Microchip, "ATmega328" [Online], Dostupno: <https://www.microchip.com/wwwproducts/en/ATmega328>
- [7] Microchip, "MCP2515" CAN Controller [Online], Dostupno: <https://www.microchip.com/wwwproducts/en/en010406>
- [8] Wikipedia, "SPI" [Online], Dostupno: https://en.wikipedia.org/wiki/Serial_Peripheral_Interface
- [9] Microchip, "MCP2551" CAN Transceiver [Online], Dostupno: <https://www.microchip.com/wwwproducts/en/en010405>
- [10] Riverbank Computing, "PyQt5" [Online], Dostupno: <https://www.riverbankcomputing.com/software/pyqt/>
- [11] SAE International, "SAE J1979" [Online], Dostupno: https://www.sae.org/standards/content/j1979_201202/
- [12] Wikipedia, "OBD2" Paramater IDs [Online], Dostupno: https://en.wikipedia.org/wiki/OBD-II_PIDs

ABSTRACT

This paper presents an example of a platform for communication over the CAN network in vehicles. The basis of the project is the Arduino Uno board, which houses the ATmega328 microcontroller, that communicates with the additional CAN Bus board via the SPI protocol. On the same board there is a MCP2515 CAN controller, which establishes communication with the CAN network via the MCP2551. An integral part of this platform is the GUI program, which is implemented using the Python programming language, and can be used to monitor, filter and send messages to the CAN network. The device also has functions that are specific to the OBD2 protocol, which is found in most modern cars, so you can monitor various parameters that are not otherwise available to the driver.

CAN platform for logging and monitoring of vehicle parameters

Duško Gajinović, Đorđe Novaković