Upravljanje kontrolnom pločom automobila korištenjem CAN sabirnice

Sirovec, Borna

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:681829

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-16**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Sveučilišni studij

Upravljanje kontrolnom pločom automobila korištenjem CAN sabirnice

Diplomski rad

Borna Sirovec

Osijek, 2023.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za diplomski ispit

Osijek, 06.12.2023.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za diplomski ispit

Ime i prezime Pristupnika:	Borna Sirovec
Studij, smjer:	Automobilsko računarstvo i komunikacije
Mat. br. Pristupnika, godina upisa:	D-62ARK, 08.10.2021.
OIB studenta:	65872886133
Mentor:	doc. dr. sc. Ivan Vidović
Sumentor:	,
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	izv. prof. dr. sc. Tomislav Matić
Član Povjerenstva 1:	doc. dr. sc. Ivan Vidović
Član Povjerenstva 2:	izv. prof. dr. sc. Ivan Aleksi
Naslov diplomskog rada:	Upravljanje kontrolnom pločom automobila korištenjem CAN sabirnice
Znanstvena grana diplomskog rada:	**Arhitektura računalnih sustava (zn. polje računarstvo)**
Zadatak diplomskog rada:	U ovom radu potrebno je istražiti dostupna rješenja za upravljanje kontrolnom pločom automobila. Nakon istraživanja potrebno je razviti vlastito sklopovlje i programsku podršku za komunikaciju i upravljanje kontrolnom pločom korištenjem CAN sabirnice. Osim izrade vlastitog rješenja, potrebno je isto i testirati na proizvoljno odabranoj kontrolnoj ploči.
Prijedlog ocjene pismenog dijela ispita (diplomskog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene od strane mentora:	06.12.2023.

Potvrda mentora o predaji konačne verzije rada:	*Mentor elektronički potpisao predaju konačne verzije.*
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 24.01.2024.

Ime i prezime studenta:	Borna Sirovec
Studij:	Automobilsko računarstvo i komunikacije
Mat. br. studenta, godina upisa:	D-62ARK, 08.10.2021.
Turnitin podudaranje [%]:	7

Ovom izjavom izjavljujem da je rad pod nazivom**: Upravljanje kontrolnom pločom automobila korištenjem CAN sabirnice**

izrađen pod vodstvom mentora doc. dr. sc. Ivan Vidović

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD 1
2. Pregled postojećih rješenja 3
3. CAN protokol i CAN komunikacija 6
3.1. Povijesni razvoj CAN protokola 6
3.2. CAN kontroler i CAN primopredajnik 7
3.3. CAN mreža i CAN čvorovi 8
3.3.1. Prijenos podataka CAN mrežom 8
3.3.2. Primjer slanja CAN poruke 9
3.4. CAN okviri 10
3.4.1. Standardni i produženi data okviri 10
3.4.2. Remote okvir 11
3.4.3. Nadopuna bitova 12
4. Sklopovsko rješenje 13
4.1. Korištene komponente 13
4.1.1. dSPACE MicroLabBox 13
4.1.2. Eksperimentalna pločica 13
4.1.3. Kontrolna ploča 14
4.2. Realizacija sklopovskog rješenja 15
5. Programsko rješenje 17
5.1. Korišteni alati 17
5.1.1. Matlab Simulink 17
5.1.2. dSPACE ControlDesk 18
5.2. CAN adrese kontrolne ploče 19
5.2.1. Prikupljanje CAN adresa 19
5.2.2. Testiranje CAN adresa 21

5.3. Realizacija programskog rješenja 24
5.3.1. Simulink model CAN Controller aplikacije 24
5.3.2. Blok poruke 0x05A0 24
5.3.3. Blok poruke 0x280 25
5.3.4. Blok poruke 0x050 26
5.3.5. Blok poruke 0x470 27
5.3.6 Blokovi poruka 0x3D0 i 0x2A0 28
5.3.7. CAN Controller ControlDesk aplikacija 29
6. Zaključak 32
LITERATURA 33
SAŽETAK 35
ABSTRACT 36
ŽIVOTOPIS 37

1. UVOD

Pojam *simracing* podrazumijeva virtualnu simulaciju utrka automobila koja se igra na računalu ili drugim gaming platformama. U simracingu se koriste sofisticirane računalne igre i simulatori vožnje koji omogućuju igračima da simuliraju vožnju različitih vrsta automobila, na različitim stazama i u različitim uvjetima. Često se koristi za trening i pripremu stvarnih vozača, ali i kao zabavna aktivnost za ljubitelje automobilizma. Simracing je postao vrlo popularan tijekom pandemije COVID-19 kada su se mnogi natjecateljski događaji uživo morali otkazati ili odgoditi, pa su se ljubitelji automobilizma okrenuli virtualnom svijetu utrka. Ova, sve popularnija forma zabave kombinira strast prema vožnji i tehnologiju. U svijetu simulacija automobilskih utrka, entuzijasti mogu doživjeti uzbuđenje utrkivanja iz udobnosti svog doma. Simracing je postao toliko napredan da vozači koriste visokokvalitetne periferne uređaje, poput volana, papučica i upravljačkih ploča, kako bi imitirali stvarnu vožnju na stazi.

U ovom diplomskom radu istraženo je upravljanje kontrolnom pločom automobila korištenjem CAN sabirnice. CAN (eng. Controller Area Network) je standardizirani komunikacijski protokol koji se često koristi u automobilskoj industriji za prijenos podataka između različitih dijelova vozila. Inicijalno je planirano koristiti Arduino platformu u implementaciji rada, no otrkivene su poteškoće u uspostavljanju stabilne veze između Arduino platforme i kontrolne ploče. Dodatno, ukazala se prilika za izradu aplikacije za kolegij "Metode i tehnike ispitivanja automobilske programske podrške" korištenjem Matlab i dSPACE alata. Ovaj odabir alata omogućio je preciznije modeliranje i simulaciju sustava, prilagođeno potrebama predmeta, te je pružio napredne mogućnosti istraživanja i implementacije.

Cilj ovog diplomskog rada je istraživanje mogućnosti upravljanja kontrolnom pločom automobila putem CAN sabirnice, koristeći Simulink model, dSPACE ControlDesk aplikaciju i MicroLabBox uređaj. Istraživanje uključuje analizu CAN protokola, implementaciju komunikacije između upravljačke ploče i računala te integraciju s MicroLabBox uređajem kako bi se željene CAN poruke prenijele na upravljačku ploču. Kroz ovaj diplomski rad poboljšano je razumijevanje upravljanja kontrolnom pločom automobila korištenjem CAN sabirnice, istražene su mogućnosti prilagodbe i poboljšanja iskustva simuliranog utrkivanja.

Diplomski rad podijeljen je u šest poglavlja: Uvod, Pregled područja teme, CAN protokol i CAN komunikacija, Sklopovosko i Programsko rješenje te Zaključak. U poglavlju Pregleda područja teme predstavljena su postojeća rješenja problema, i navedene su inspiracije za ovaj diplomski rad. Slijedi poglavlje u kojem je objašnjena CAN komunikacija i CAN protokol, a nakon njega, u poglavljima sklopovskog odnosno programskog rješenja predstavljena je realizacija teme

diplomskog rada.

2. Pregled postojećih rješenja

Incijalna motivacija za ovaj diplomski rad je projekt Youtube korisnika „amstudio“ u kojem je sa simulacijskom igrom Asseto Corsa, putem SimHub softvera povezana instrument ploča automobila BMW E39.[1] Nedostatak ovog projekta je što je kontrolna ploča analogna, tj. ne sadrži neki od protokola za komunikaciju (kao npr. CAN protokol).

U projektu „vag-can-bus-gauge-cluster-controller“ github korisnika „mygithubadel“, videoigra BeamNG drive povezana je s kontrolnom pločom automobila Volkswagen Touran [2]. Telemetrija je dohvaćena OutGauge protokolom, te je Wifi mrežom poslana na Arduino. Arduino tada CAN modulom šalje zapakirane CAN poruke i upravlja kontrolnom pločom.

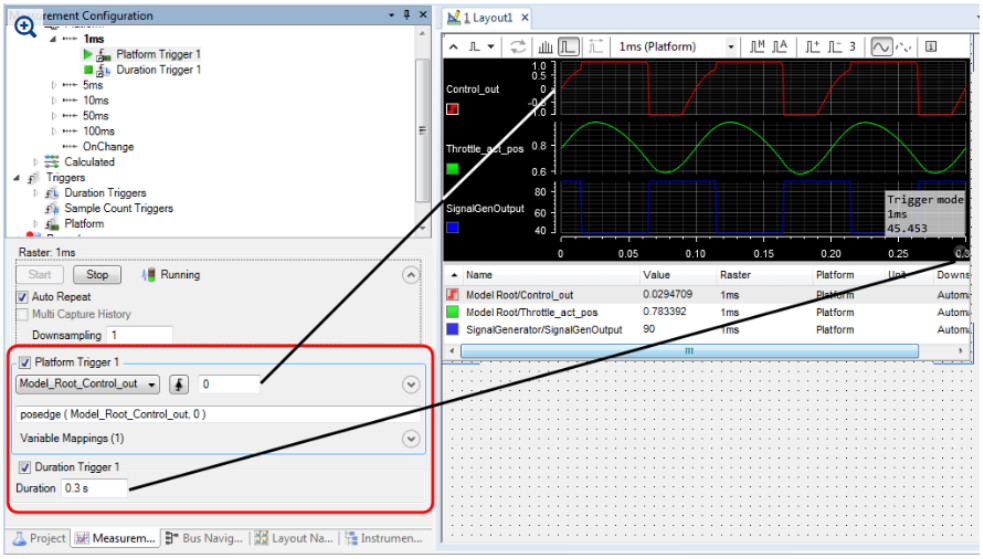
Autor diplomskog rada je za potrebe kolegija „Ugradbeni računalni sustavi“ napravio projekt imena „Custom-Dashboard-STM32-BluePill-SimHub“ [3] u kojemu je upravljačkom pločom kreiranom na bazi ploče automobila Fiat Bravo, manipulirao putem SimHub-ovog custom kontrolera i Ncalc formule. Slika 2.1. prikazuje rezultat navedenog projekta.

Slika 2.1. Upravljanje kontrolnom pločom realizirano u radu [3].

Projekt „VW-Instrument-Cluster-Controller“ čine kontrolna ploča automobila Volkswagen Up i Arduino u kombinaciji s CAN komunikacijom. [4] Autor, GitHub korisnik „an-ven“ uspješno upravlja pločom (Slika 2.2.) putem serijske veze, slanjem određenih naredbi preko serijskog monitora (eng. *Serial Monitor*).

Slika 2.2. Rezultat projekta „VW-Instrument-Cluster-Controller“

Autor diplomskog rada obavio je stručnu praksu u tvrtki dSpace Engineering d.o.o. [5] Tijekom obavljanja stručne prakse upoznat je sdSPACE razvojnim alatima poput Ethenet Configuration Tool, Configuration Desk, Control Desk te s dSPACE hardverom poput Scalexia te MicroLabBoxa. Stekao je znanje o simuliranju i testiranju u automotiv industriji, te o kreiranju runtime aplikacija u Control Desk alatu. Slika 2.3. predstavlja jednu od aplikacija za praćenje signala izrađenu na stručnoj praksi.



Slika 2.3. Control Desk aplikacija izrađena na studentskoj praksi.

Ovim diplomski radom predstavljena je adaptacija i proširenje projekta „VW-Instrument-Cluster-Controller“ [4]. Znanje stečeno iz projekta „VW-Instrument-Cluster-Controller“ te znanje stečeno na odrađenoj studentskoj praksi u tvrtki dSPACE Engineering d.o.o. iskorišteno je za upravljanje pločom automobila putem Simulink modela i ControlDesk aplikacije.

3. CAN protokol i CAN komunikacija

3.1. Povijesni razvoj CAN protokola

Početkom 1980-ih inženjeri u Boschu procjenjivali su postojeće sustave serijskih sabirnica u pogledu njihove moguće upotrebe u osobnim automobilima. Budući da niti jedan od dostupnih mrežnih protokola nije mogao ispuniti zahtjeve automobilskih inženjera, Uwe Kiencke je 1983. započeo razvoj novog sustava serijske sabirnice. Novi protokol uglavnom je trebao dodati nove funkcionalnosti, redukcija potrebnih žica zapravo je samo nuspojava. Računalski gigant Intel kao i automobilski gigant Mercedes uključili su se u razvoj novog protokola, a profesor dr. Wolfhard Lawrenz daje novom mrežnom protokolu ime CAN.

U veljači 1986., tvrka Bosch GmbH predstavila je CAN sustav serijske sabirnice na kongresu društva automobilskih inženjera. Prva verzija CAN protokola objavljena je 1986. godine, 1987. Intel dostavlja prvi čip za CAN kontroler, a 1993. standardiziran je kroz nekoliko ISO dokumenata. U terminologiji referentnog modela, ISO 11898-1 standard opisuje fizički sloj i sloj podatkovne veze. Standardi ISO 11898-2 i ISO 11898-3 pokrivaju dva podsloja fizičkoga sloja – CAN fizički sloj visoke i niske brzine. Iako je standardiziran tek 1993. godine, prvi automobil s CAN protokolom bio je Mercedes S klase, oznake W140 kojim je Mercedes 1991. godine uz CAN, uveo i mnoge druge inovacije. 1990-ih godina, Dr. Bernd Muller u suradnji s tvrtkom Bosch, stručnjacima iz industrije poluvodiča i stručnjacima iz akademskih istraživanja definira tzv. „Time-triggered communication on CAN“ (TTCAN). Ovo proširenje protokola omogućilo je vremenski ekvidistantni prijenos okvira i implementaciju upravljanja zatvorenom petljom pomoću CAN-a.

U prosincu 2010.godine, organizacija „CAN in Automation “ (CiA), objavljuje 601-1 standard kojim predstavlja nadogradnju postojećeg protokola u vidu brzine prijenosa i fleksibilnosti nazvanu „CAN Flexible data Rate“ (CAN FD). Trenutno, u vrijeme pisanja ovog rada provodi se standardizacija „CAN eXtended Length“ (CAN XL) nadogradnje. Ona predstavlja odgovor na rastuće zahtjeve za propusnošću podataka u industriji autonomnih vozila i složenih industrijskih sustava. [6]

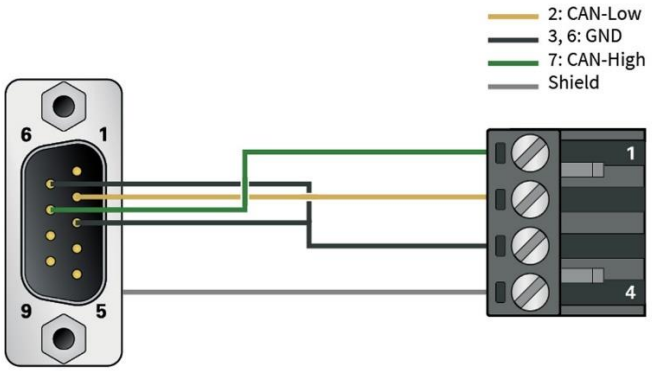
3.2. CAN kontroler i CAN primopredajnik

CAN protokol je s obzirom na OSI model, implementiran kroz dva sloja: fizički sloj i sloj podatkovne veze. Kroz fizički sloj implementiran je CAN primopredajnikom, CAN medijem sabirnice te CAN konektorom. Na sloju podatkovne veze, implementaciju predstavlja CAN kontroler.

ECU (eng. *Electronic Control Unit*) koji želi sudjelovati u CAN komunikaciji zahtjeva CAN sučelje. Sučelje sadrži kontroler i primopredajnik. Postoje dvije vrste kontrolera, Full CAN kontroler i Basic CAN kontoler, Full CAN kontroler pruža kompletnu podršku za sve funkcionalnosti CAN protokola, uključujući generiranje, prijem, filtriranje, detekciju grešaka i upravljanje. To uključuje napredne funkcionalnosti poput konfigurabilnih filtara, više kanala, bufferske memorije i druge napredne kontrole. S druge strane, Basic CAN kontroler obično podržava osnovne funkcionalnosti poput prijema i slanja podataka, ali može imati ograničene ili jednostavnije mogućnosti filtriranja i upravljanja.

CAN primopredajnik je elektronička komponenta koja služi za povezivanje CAN kontrolera s fizičkim slojem CAN magistrale. Njegova glavna funkcija je provođenje električnih signala između logičkog sučelja CAN kontrolera i fizičke sabirnice. Sastoji se od dva pina sabirnice, po jednog za CAN High liniju (CANH) i jednog za CAN Low liniju (CANL), a razlikujemo ih po podržanoj brzini prijenosa. Primopredajnici visoke brzine podržavaju prijenos do 1Mbit/s, dok oni niske brzine prenose do 125kbit/s.

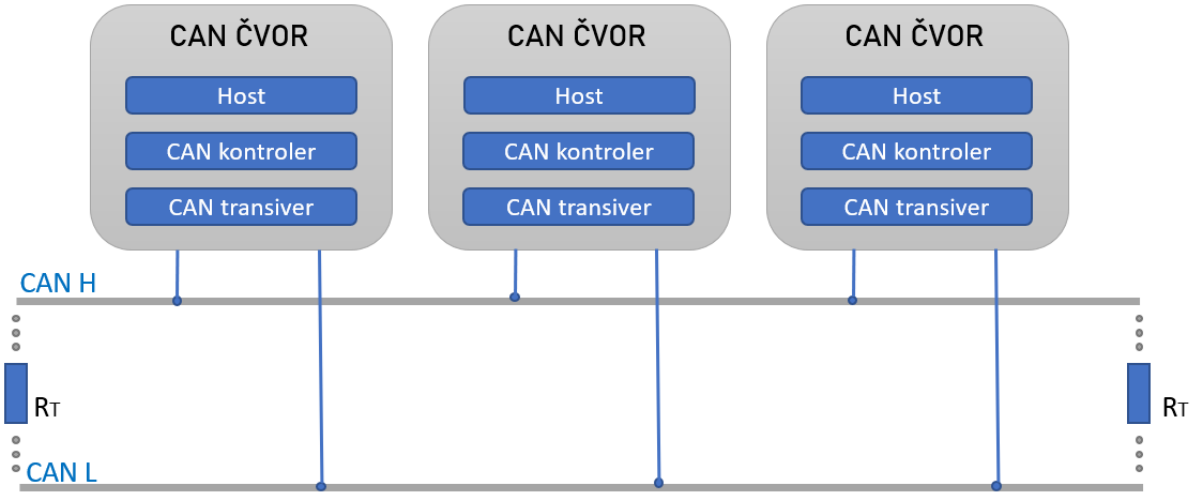
Standardni priključak za povezivanje CAN primopredajnika s CAN sabirnicom je „D-Sub“ konektor (Slika 3.1.), a sam prijenos obavlja se preko UTP-a (eng. *Unshielded twisted pair*) žica. [7]



Slika 3.1. Prikaz D-Sub konektora za CAN komunikaciju.

3.3. CAN mreža i CAN čvorovi

CAN mreža za komunikaciju sastoji se od CAN čvorova povezanih medijem za fizički prijenos (Slika 3.2.). Najčešće je predstavljena topologijom linije, linijskom sabirnicom na koju su spojeni ECU-ovi koji podržavaju CAN protokol. Svaki od tih ECU-ova predstavlja jedan CAN čvor. Trenutno, prosječan broj ECU-ova u modernom automobilu je 70. [8]



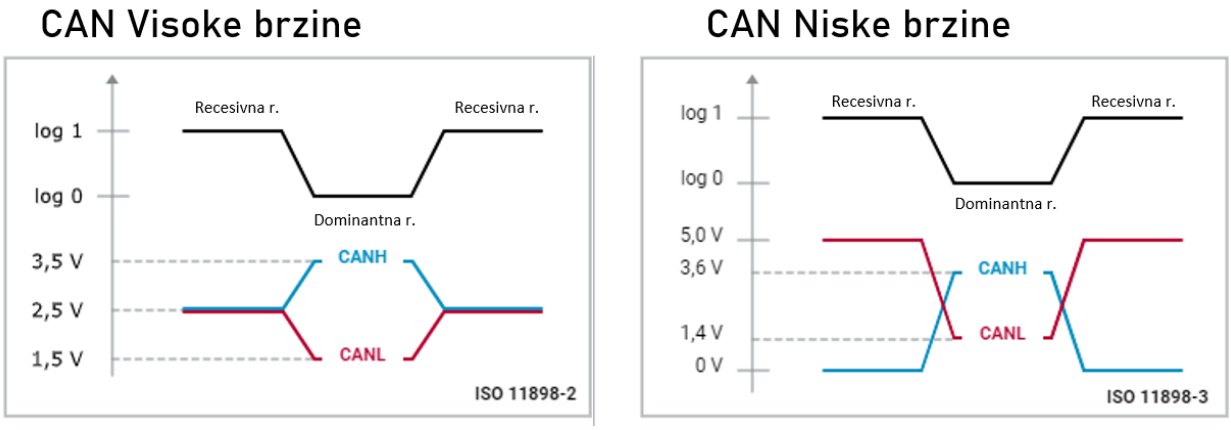
Slika 3.2. Prikaz CAN mreže i CAN čvora.

Prema ISO 11898 standardu, mreža smije biti dugačka do 40m, a na krajevima se nalaze otpornici za terminaciju sabirnice. Naime, zbog konačnog širenja signala velikom brzinom postoji mogućnost refleksije signala. Kako bi se spriječila refleksija, u mrežama visoke brzine na rubove su dodani otpornici od 120 Ohma. Maksimalan broj čvorova u mrežama niske brzine iznosi 32, a u mrežama visoke brzine iznosi 110. [7]

3.3.1. Prijenos podataka CAN mrežom

Fizički prijenos signala CAN mrežom baziran je na prijenosu diferencijalnih napona. Vrijednost diferencijalnih napona ovisi o sučelju sabirnice – koristimo li CAN niske ili visoke brzine (Slika 3.3.). ISO 11898-2 pridružuje logičku "1" tipičnom diferencijalnom naponu od 0 V. Logička "0" pridružena je tipičnom diferencijalnom naponu od 2 V. CAN prijemnici velike brzine kao dominantnu razinu unutar opsega zajedničkog način rada (obično između 12 V i -12 V) smatraju diferencijalni napon veći od 0,9V. Međutim, ispod 0,5 V, diferencijalni napon smatra se recesivnom razinom. Histerijski krug povećava otpornost na smetnje u naponima. ISO 11898-3

pridružuje tipični diferencijalni napon od 5 V logičkoj "1", dok tipični diferencijalni napon od 2 V odgovara logičkoj "0". [8]



Slika 3.3. Naponske razine CAN sabirnice.

CAN mreža koristi decentralizirani pristup sabirnici, što znači da svaki čvor ima pravo pristupa sabirnici u bilo koje vrijeme. Svaki čvor može poslati poruku na sabirnicu u mreži, a prijenos poruke nije determiniran nekom vremenskom sekvencom već je „event-driven“. Komunikacijski kanal zauzet je samo ako se neka nova informacija treba poslati, te upravo ovo dovodi do brzog pristupa sabirnici – posljedica je prijenos podataka u realnom vremenu s brzinom od 1Mbit/s. Dodatno, informacije koje se šalju CAN sabirnicom vide svi čvorovi, tj. koristi se „Broadcast“ način slanja.

3.3.2. Primjer slanja CAN poruke

Neka je zadana sljedeća situacija: neka mrežu čine četiri čvora spojenih na sabirnicu i neka čvor A šalje poruku s identifikatorom 0x12 čvoru B putem sabirnice. Podijelimo ovaj primjer u 6 koraka.

Korak 1: Kao odgovor na event, host čvora A zapisuje informacije u TX Buffer njegovog kontrolera. Kontroler tada šalje sadržaj buffera čim sabirnica bude slobodna.

Korak 2: Netom prije slanja na sadržaj buffera dodaje potrebne informacije i tvori CAN okvir. Nakon što je stvorio CAN okvir, broadcasta ga drugim čvorovima u sabirnici.

Korak 3: Čvorovi B,C i D primaju okvir.

Korak 4: Čvorovi B,C i D provjeravaju identifikator okvira kojeg su primili. U situaciji opisanoj na početku, čvor B treba primiti poruku – čvor B prihvaća okvir, čvorovi C i D ga odbacuju.

Korak 5: CAN kontroler čvora B javlja hostu da je primio poruku. Host čita poruku iz RX Buffera.

Korak 6: Host čvora B procesira poruku. Slanje poruke uspješno je obavljeno. Opisan primjer dostupan je na animaciji [9].

3.4.CAN okviri

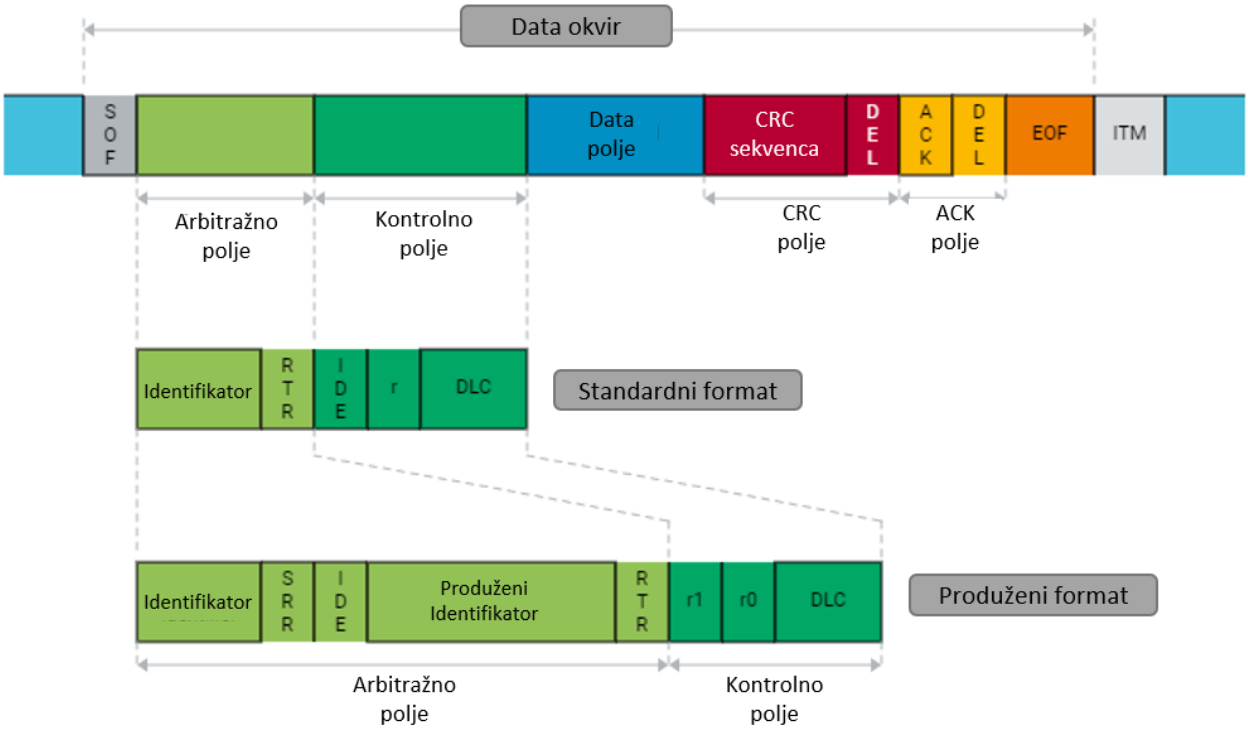
CAN protokol definira tri različite vrste okvira: standardni data okvir, prošireni data okvir te remote okvir. Data okviri služe za prijenos podataka i mogu prenijeti do 8 B podatka. Razlika između standardnog i proširenog okvira je u tome da prošireni okvir ima polje za prošireni identifikator. U standardnom okviru, identifikator je veličine 11 b, dok je u proširenom veličine 29 b pa zato i postoji potreba za dodatnim poljem. Remote okvir služi za zahtjevanje podataka od drugog čvora, samim time nema podatkovno polje jer ne šalje podatke. [10]

3.4.1. Standardni i produženi data okviri

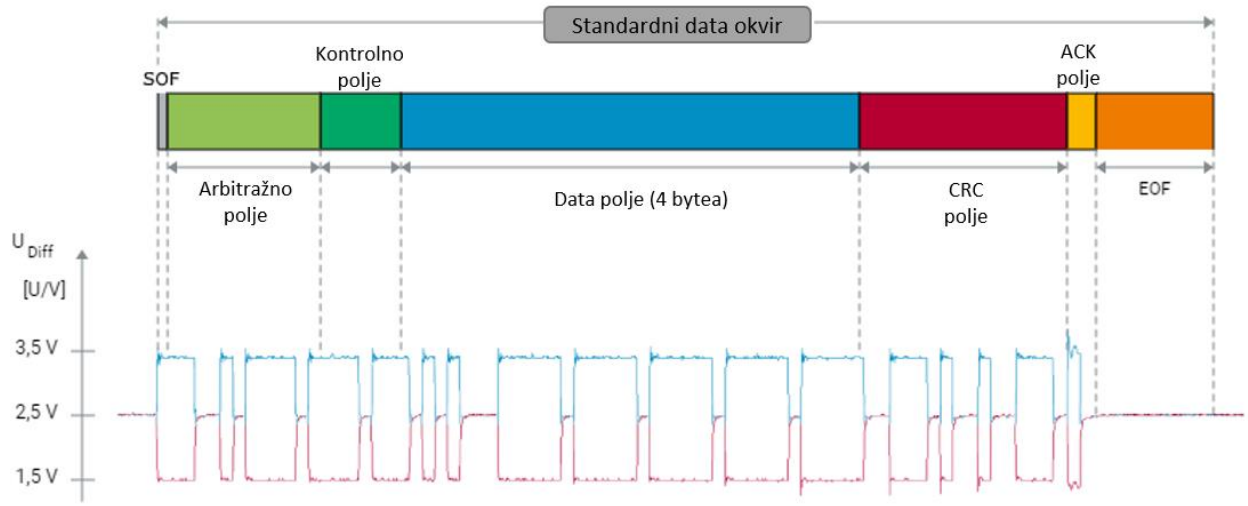
CAN okviri (prikazani na Slici 3.4.) za prijenos podataka sastoje se od sljedećih polja:

- SOF (eng. *Start of Frame*) – predstavlja početni bit kojim počinje prijenos okvira,
- Arbitražnog polja – ovo polje sadrži identifikator (u slučaju proširenog okvira, sadrži i produženi identifikator) i Remote Transmission Request (RTR) bit koji označava vrstu okvira (je li okvir Remote ili ne). Recesivan RTR bit (vrijednost „1“) označava Remote okvir, a dominantan (vrijednost „0“) Data okvir. Identifier Extension (IDE) bit govori o tome ima li okvir produženi identifikator,
- Kontrolnog polja – sadrži Data Length Code (DLC) koji primatelje obavještava o veličini payload bajtova,
- Data polja – u ovom polju se prenosi podatak, maksimalne veličine 8 B,
- CRC (eng. *Cyclic Redundancy Check*) polja – podatak koji se prenosi, zaštićen je kontrolnom sumom koju računamo Cikličkom provjerom redundancije,
- ACK (eng. *Acknowledgement*) polja – ovisno o rezultatu cikličke provjere redundancije, primatelji šalju pozitivnu ili negativnu potvrdu. Završava delimiter bitom,
- EOF (eng. *End of Frame*) – zadnje polje u okviru, nakon njega prijenos okvira je terminiran sa sedam recesivnih bitova. [10]

Slika 3.5 prikazuje prijenos fizički prijenos podatkovnog okvira.



Slika 3.4. CAN okviri za prijenos podataka.

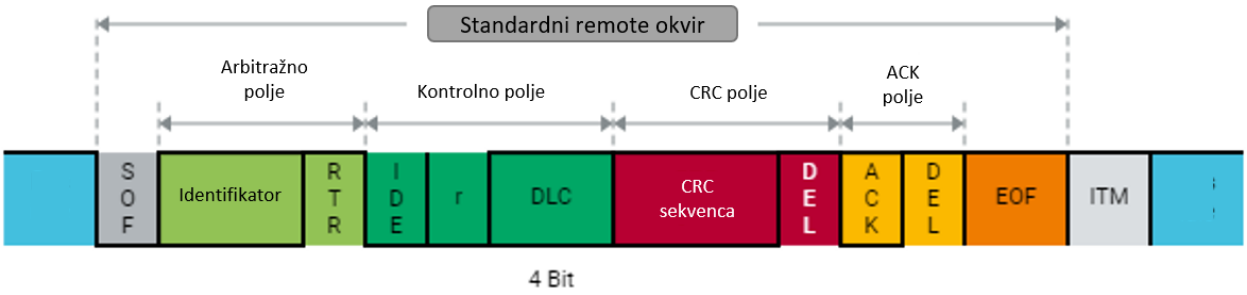


Slika 3.5. Fizički prijenos podatkovnog okvira.

3.4.2. Remote okvir

Remote okviri (Slika 3.6.) služe za zahtjevanje neke informacije ili okvira od drugih CAN čvorova. Upravo zbog toga rijetko se koriste u automotive aplikacijama, budući da prijenos nije zasnovan na zahtjevu (eng. *request*) već je vođen događajem (eng. *event-driven*). Ono što remote

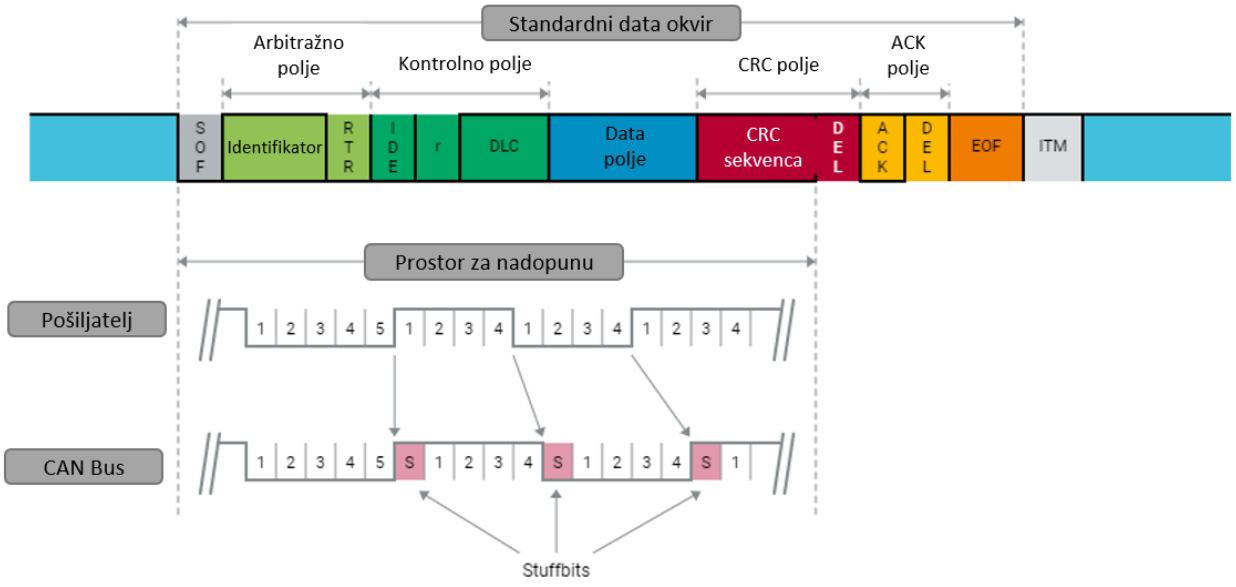
okvir razlikuje od standardnog je to što ne sadrži Data polje. U idealnom slučaju, odmah se odgovara na zahtjev, no postoji mogućnost da CAN poruke višeg prioriteta budu umetnute između zahtjeva i odgovora. [10]



Slika 3.6. Remote okvir.

3.4.3. Nadopuna bitova

Osnovni preduvjet za ispravan prijenos podataka su sinkronizirani sudionici komunikacije. Sinkronizacija započinje signalom SOF bita, a daljnju sinkronizaciju omogoćava mehanizam resinkronizacije koji traje do kraja poruke. Mehanizam resinkronizacije temelji se na evaluaciji recessive-to-dominant signalnih rubova. Ti rubovi osigurani su mehanizmom nadopune bitova. ISO 11898-1 standard nalaže da pošiljatelji moraju poslati komplementarni bit nakon najviše pet homogenih bitova – dodaje se jedan bit nadopune čak i ako komplementarni bit slijedi nakon pet homogenih bitova (Slika 3.7.).



Slika 3.7. Mehanizam nadopune bitova.

4. Sklopovsko rješenje

U ovom poglavlju opisan je sklopovski dio rješenja diplomskog rada. Navedene su informacije o MicroLabBox uređaju i kontrolnoj ploči te je opisana realizacija sklopovskog rješenja.

4.1. Korištene komponente

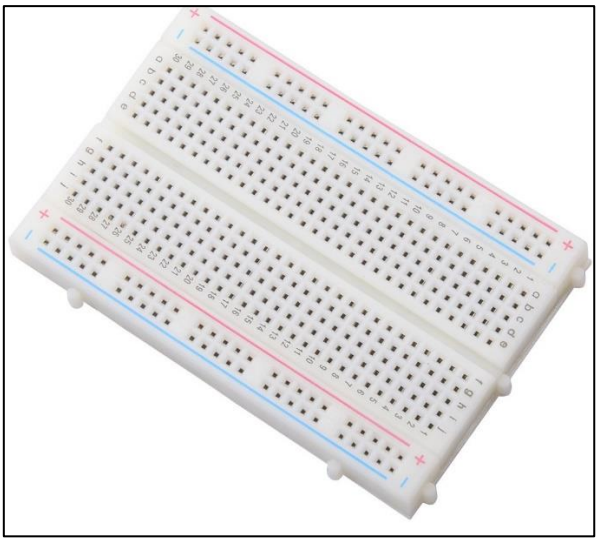
4.1.1. dSPACE MicroLabBox

MicroLabBox (Slika 4.1.) tvrtke dSPACE je kompaktni modularni uređaj koji kombinira osobitosti sustava za automatsko testiranje i mjerenje s prednostima laboratorijskih mjernih instrumenata. Namijenjen je za upotrebu u širokom rasponu inženjerskih aplikacija, uključujući razvoj i testiranje kontrolnih sustava, mjerenje signala, procesiranje podataka i slične svrhe. Omogućuje jednostavnu integraciju u različite laboratorijske postavke, pruža fleksibilnost glede konfiguracije i programiranja podržavanjem različitih jezika i okolina. Dodatno, u kombinaciji sa dSPACE softverskim alatima čini savršenu okolinu za RT (eng. *real-time*) i HIL (eng. *hardware in the loop*) testiranja. [11]

Slika 4.1. MicroLabBox tvrtke dSPACE.

4.1.2. Eksperimentalna pločica

Eksperimentalna ploča (eng. *protoboard*) je ploča koja ima matricu malih rupa na koje se komponente mogu pričvrstiti bez lemljenja (Slika 4.2.). Koristi se za privremenu konstrukciju i testiranje električnih i elektroničkih krugova [12]. U svrhu dodatnog napajanja, korišten je adapter za od 12 V za napajanje ploče automobila.



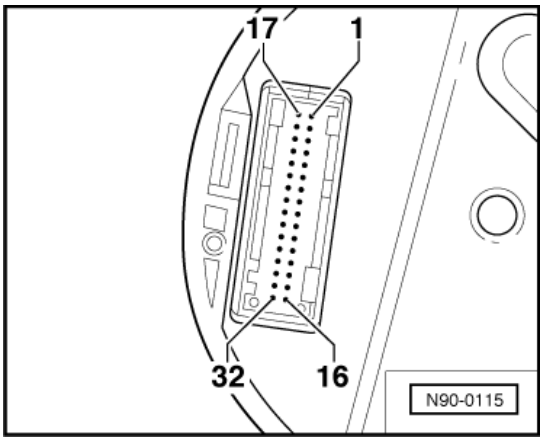
Slika 4.2. Eksperimentalna pločica.

4.1.3. Kontrolna ploča

U svrhu izrade diplomskog rada korištena je kontrolna ploča automobila Volkswagen UP (Slika 4.3.). Ona podržava CAN komunikaciju te se CAN porukama može upravljati kazaljkama okretaja motora, brzine vozila i temperaturom rashladne tekućine. Dodatno, uz kazaljke, može se upravljati zaslonom, te paliti i gasiti neke od lampica upozorenja kao na primjer: pokazivači smjera, duga svjetla, svjetla za maglu, svjetla za zračni jastuk i pojas, itd.

Ploča sadrži 32-pinski konektor (Slika 4.4.) od kojih je za CAN komunikaciju potrebno samo pet. Za ostvarivanje CAN komunikacije, ploču treba spojiti na napanje od 12 V (pinovi 31 i 32), pin 16 je za ground, a pinovi 28 i 29 su CANH i CANL pinovi koji se spajaju na MicroLabBox.

Slika 4.3. Kontrolna ploča Volkswagen Up-a.

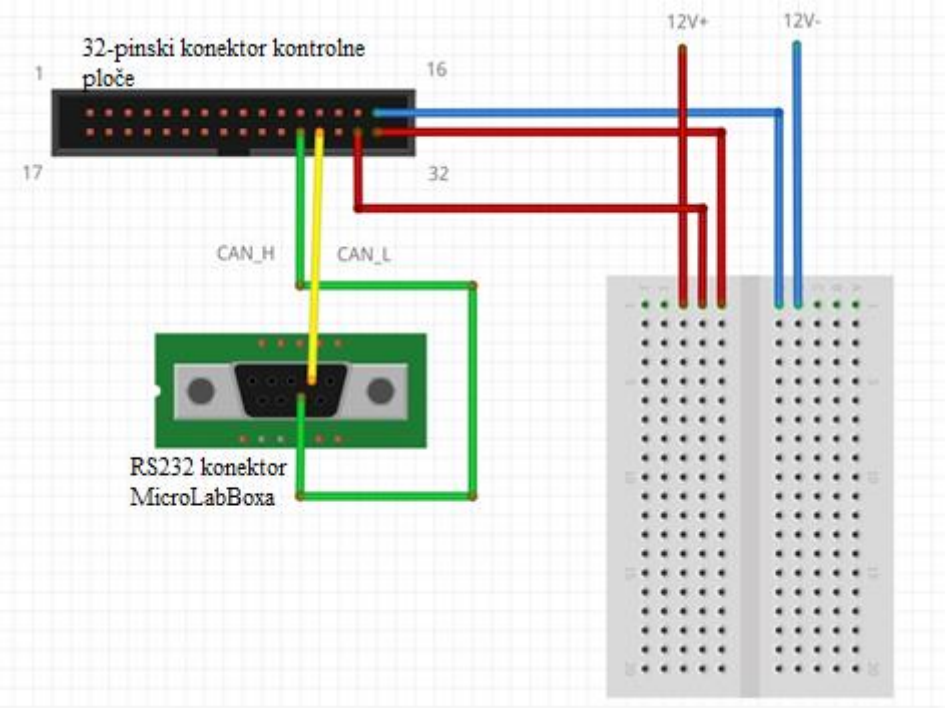


Slika 4.4. 32-pinski konektor.

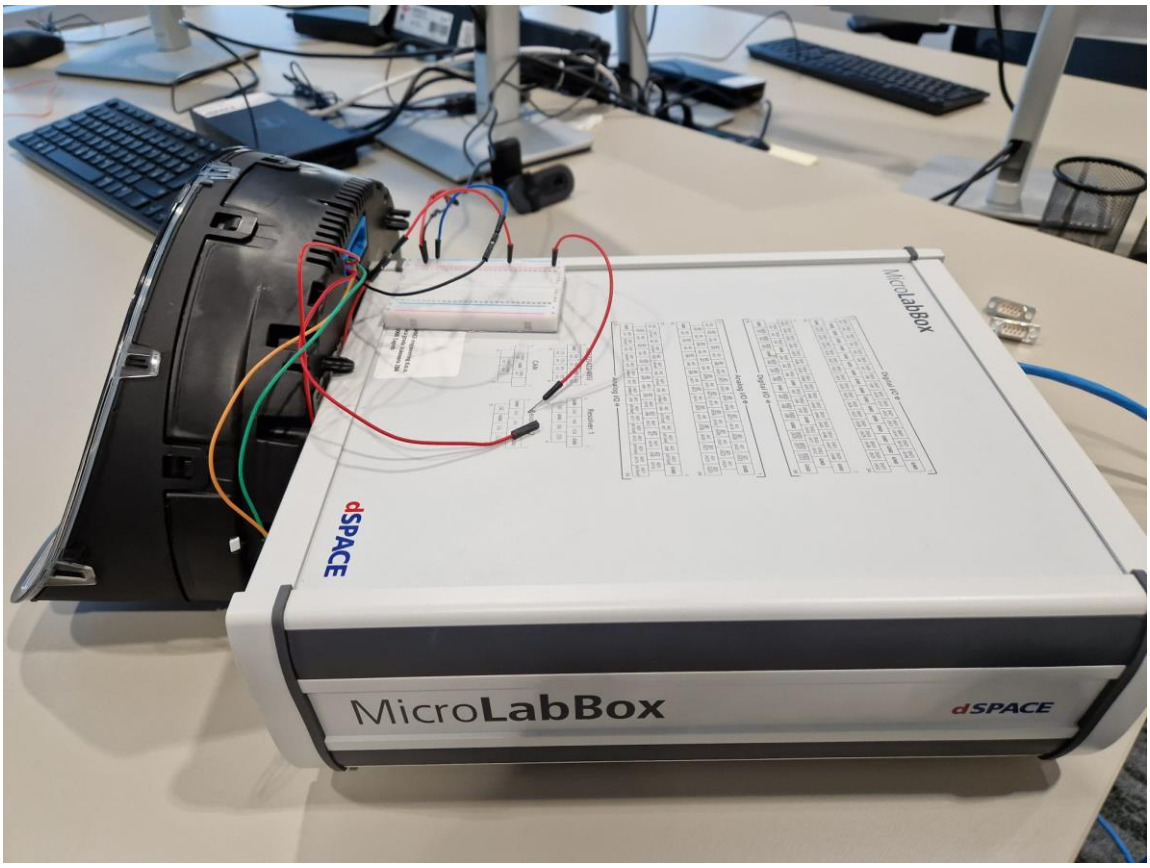
4.2. Realizacija sklopovskog rješenja

U ovom poglavlju objašnjeno je spajanje komponenata navedenih u poglavlju 4.1. MicroLabBox spojen je na računalo Ethernet kablom. CAN poruke šalje na upravljačku ploču putem pinova CAN H i CAN L koji se nalaze na RS232 konektoru (Slika.4.5.). Dodatno, potrebno je napajanje od 12V za upravljačku ploču. Schema spajanja dostupna je na slici 4.6., a sklopovsko rješenje na slici 4.7.

Slika 4.5. CAN_H i CAN_L spojevi na RS232 konektoru MicroLabBoxa



Slika 4.6. Schema spajanja sklopovskog rješenja.



Slika 4.7. Izgled sklopovskog rješenja.

5. Programsko rješenje

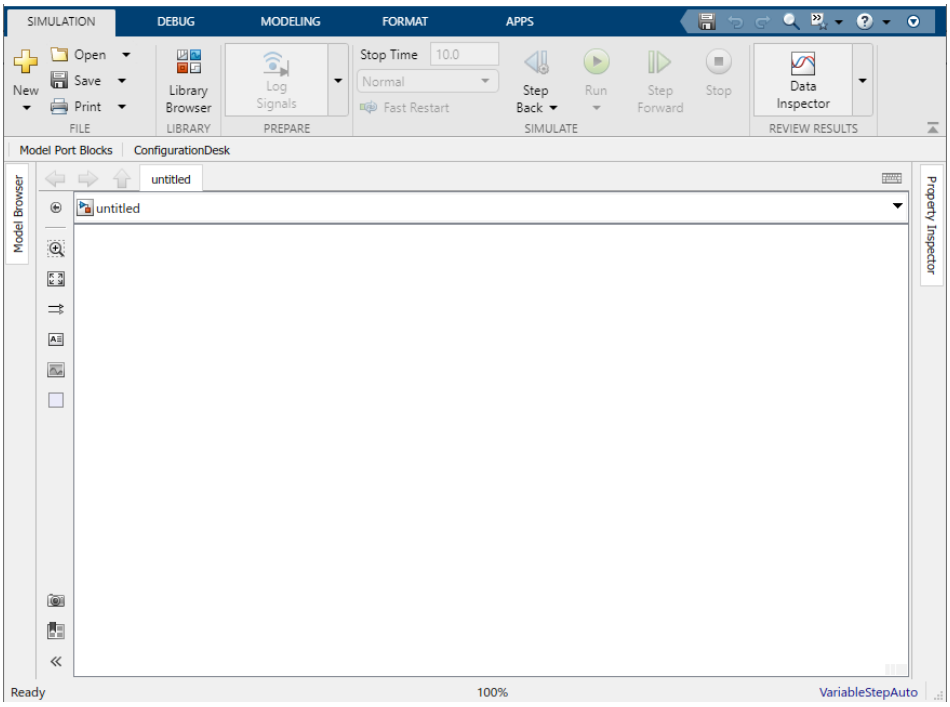
U ovom poglavlju opisan je Matlab Simulink model za generiranje i slanje CAN poruka, te ControlDesk aplikacija za upravljanje porukama i pločom.

5.1. Korišteni alati

U ovom poglavlju navedeni su i opisani softverski alati kojima je omogućeni razvoj i realizacija programskog rješenja.

5.1.1. Matlab Simulink

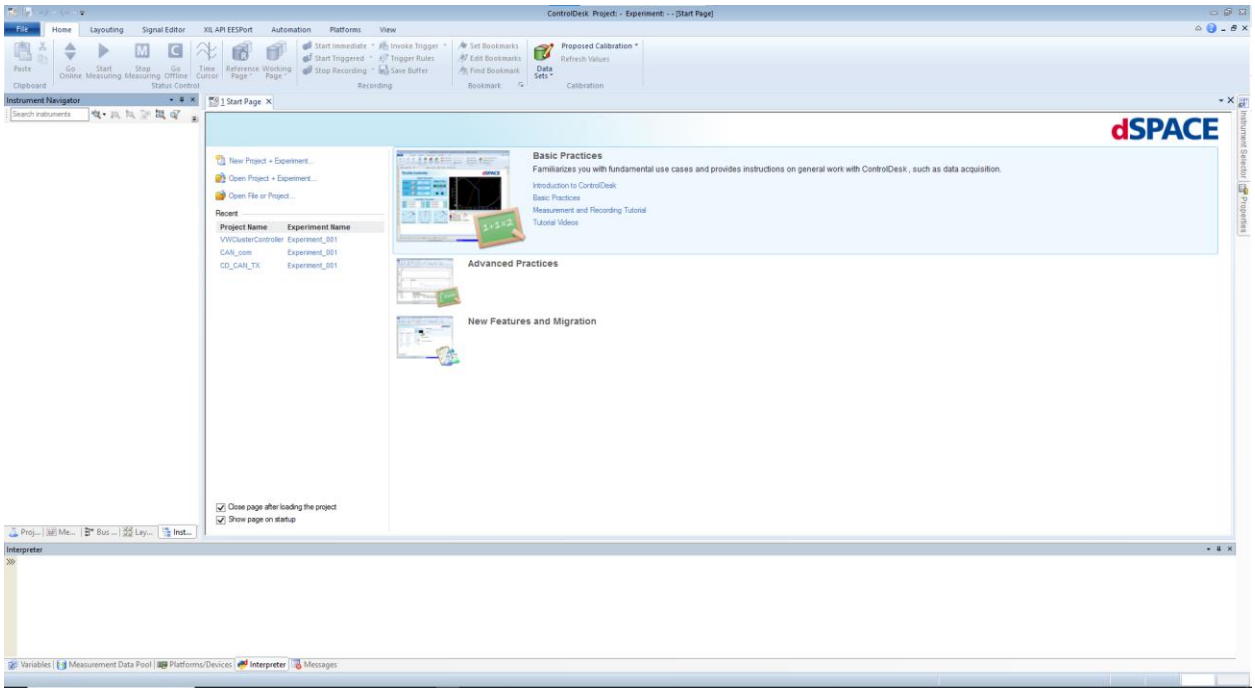
Simulink je grafički simulacijski okvir (Slika 5.1.) koji je razvila tvrtka MathWorks, a koristi se za modeliranje, simuliranje i analiziranje dinamičkih sustava. Simulink omogućuje inženjerima i znanstvenicima da stvaraju modele sustava pomoću blokova i linija koje predstavljaju komponente sustava i veze između njih. Dio je Matlab platforme, što znači da se integrira s Matlabom i koristi Matlab jezik za definiranje matematičkih izraza, analizu podataka i obradu rezultata simulacija. Kroz tu integraciju, korisnici mogu kombinirati moć analize podataka i programiranja u Matlabu s prednostima grafičkog modeliranja i simulacije u Simulinku. Simulink se često koristi u područjima kao što su kontrola sustava, komunikacije, signalna obrada, automobilska industrija, elektroenergetika i druge oblasti gdje je potrebno modeliranje i simulacija dinamičkih sustava. [13]



Slika 5.1. Početno sučelje Simulink alata.

5.1.2. dSPACE ControlDesk

ControlDesk je softverski alat tvrtke dSPACE, kojeg se koristi u razvoju i testiranju automobila. dSPACE je priznata tvrtka koja pruža rješenja za razvoj ugrađenih sustava u automobilima, telemetriju i srodna područja. Ovaj alat sadrži sustav za vizualizaciju podataka, omogućujući korisnicima praćenje i analizu podataka generiranih tijekom testiranja u različitim formatima poput grafikona, dijagrama i brojčanih prikaza. Korisnici mogu postavljati i konfigurirati eksperimente za testiranje parametara te pratiti ponašanje sustava u kontroliranom okruženju. ControlDesk se često koristi kao sučelje za integraciju s drugim dSPACE alatima poput kontrolnih sustava, simulatora, modela simulacije i telemetrijskih sustava. Također podržava automatizaciju testiranja, omogućujući postavljanje i izvođenje testnih scenarija bez stalnog ručnog nadzora, čime se poboljšava učinkovitost testiranja. Alat je fleksibilan i prilagodljiv, pružajući mogućnost prilagodbe različitim aplikacijama i projektima. ControlDesk također podržava razne industrijske standarde, čineći ga kompatibilnim s drugim alatima i sustavima u automobilskoj industriji. [14] Na slici 5.2. prikazano je početno sučelje ControlDesk alata.



Slika 5.2. Početno sučelje alata ControlDesk.

5.2. CAN adrese kontrolne ploče

U ovom poglavlju opisano je provedeno istraživanje o CAN adresama kontrolne ploče vozila Volkswagen Up. (Poglavlje 4.1.3.)

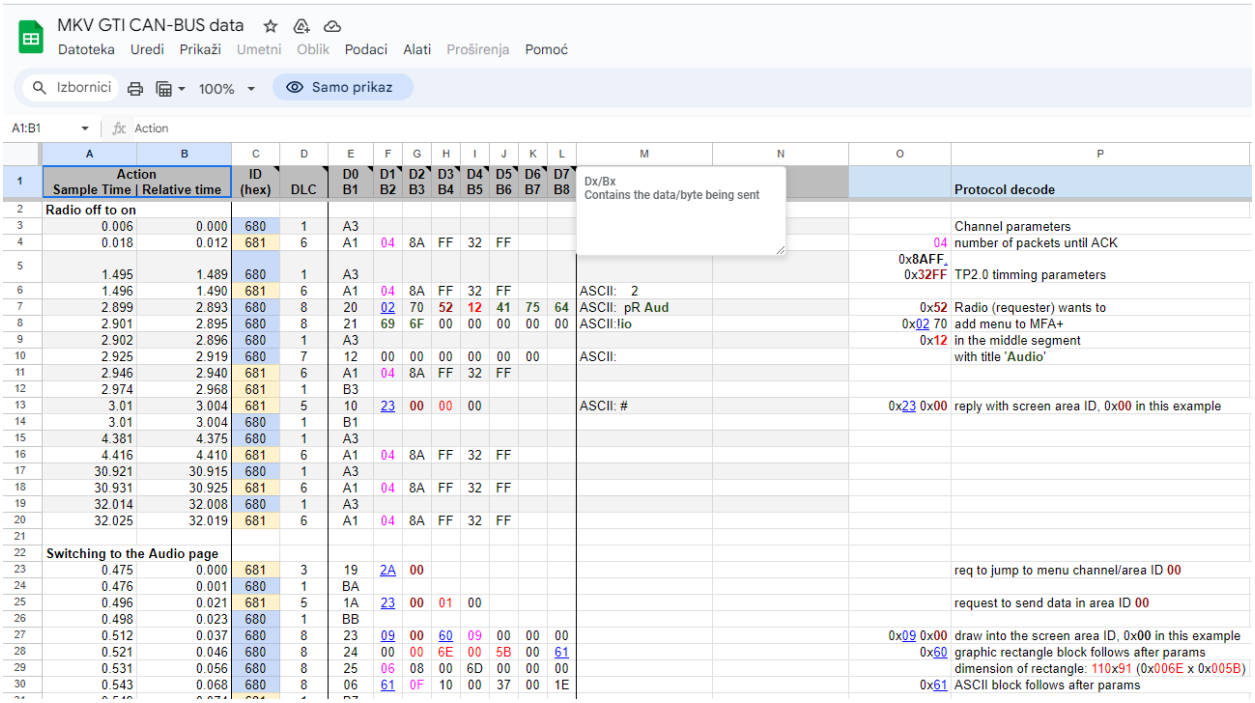
5.2.1. Prikupljanje CAN adresa

Kontrolna ploča automobila koristi CAN adresiranje kako bi omogućila komunikaciju između različitih dijelova sustava vozila. U ovom poglavlju analiziran je proces prikupljanja informacija o CAN adresama kontrolne ploče, s fokusom na strukturu adresa, njihove payloade te način komunikacije putem specifičnih adresa kako bi se postigli određeni efekti. Adrese korištene na kontrolnoj ploči sastoje se od tri heksadecimalne znamenke. Postupkom opisanim u poglavlju 5.2.2. dobiveno je kako na primjer, adresa 0x5A0 predstavlja jednu od adresa kojom je ostvarena komunikacija između kontrolne ploče i MicroLabBox-a.

Payload svake adrese sastoji se od osam polja, pri čemu je svako polje veličine 1 B. Polja su nazvana AA, BB, CC, DD, EE, FF, GG, i HH kako bi se olakšala njihova identifikacija i upotreba. Ovo osigurava standardiziran pristup pri analizi i manipulaciji podacima putem CAN komunikacije.

Za postizanje određene akcije na kontrolnoj ploči, poput paljenja lampice za ABS ili okretanja kazaljki, potrebno je poslati određeni payload na određenu adresu. Na primjer, za paljenje lampice za ABS, potrebno je poslati vrijednost 1 u polje DD na adresu 0x5A0.

Dio korištenih adresa dobiven je putem analize GitHub repozitorija [2][4], dio analizom Excel tablice „MKV GTI CAN-BUS data“ [15], a preostali dio generiran je pomoću "Generate Random Hex" web aplikacije [16]. Adrese generirane pomoću web aplikacije služile su za tzv. Napad uzastopnim pokušavanjem (eng. *Bruteforce attack*). Važno je napomenuti da su sve identificirane adrese sustavno zabilježene u Excel tablicu koja je popunjavana tijekom testiranja. Na slici 5.3. prikazana je tablica „MKV GTI CAN-BUS data“ [14], na slici 5.4. prikazano je generiranje nasumičnih adresa web aplikacijom „Generate Random Hex“ [16], a na slici 5.5. prikazan je dio tablice spremne za popunjavanje tijekom testiranja.



MKV GTI CAN-BUS data

Datoteka Uredi Prikaži Umetni Oblik Podaci Alati Proširenja Pomoć

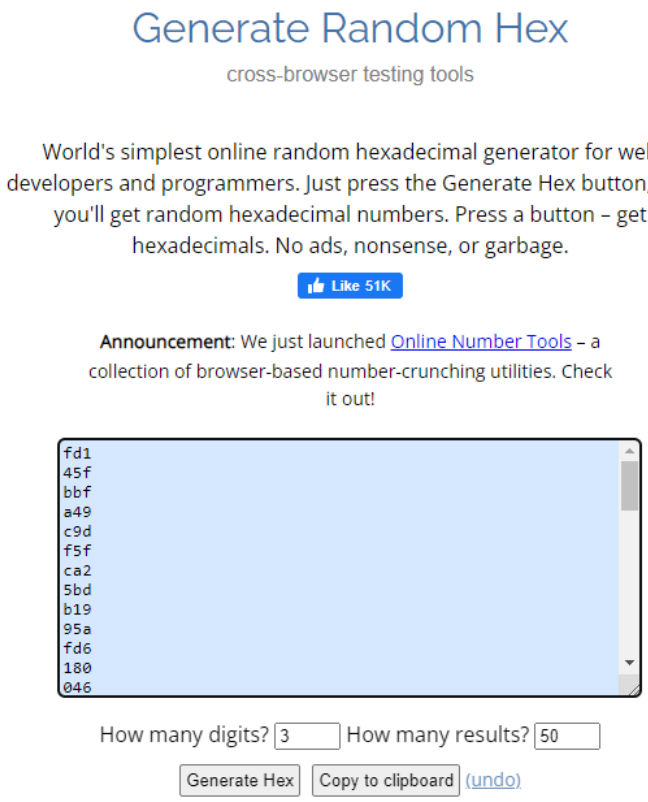
Izbornici 100% Samo prikaz

A1:B1 Action

Dx/Bx
Contains the data/byte being sent

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Action Sample Time	Relative time	ID (hex)	DLC	D0 B1	D1 B2	D2 B3	D3 B4	D4 B5	D5 B6	D6 B7	D7 B8				Protocol decode
2	Radio off to on															
3	0.006	0.000	680	1	A3											Channel parameters
4	0.018	0.012	681	6	A1	04	8A	FF	32	FF					04	number of packets until ACK
5	1.495	1.489	680	1	A3										0x8AFF, 0x32FF	TP2.0 timming parameters
6	1.496	1.490	681	6	A1	04	8A	FF	32	FF			ASCII: 2			
7	2.899	2.893	680	8	20	02	70	52	12	41	75	64	ASCII: pR Aud		0x52	Radio (requester) wants to
8	2.901	2.895	680	8	21	69	6F	00	00	00	00	00	ASCII:!io		0x02 70	add menu to MFA+
9	2.902	2.896	680	1	A3										0x12	in the middle segment
10	2.925	2.919	680	7	12	00	00	00	00	00	00		ASCII:			with title 'Audio'
11	2.946	2.940	681	6	A1	04	8A	FF	32	FF						
12	2.974	2.968	681	1	B3											
13	3.01	3.004	681	5	10	23	00	00	00				ASCII: #		0x23 0x00	reply with screen area ID, 0x00 in this example
14	3.01	3.004	680	1	B1											
15	4.381	4.375	680	1	A3											
16	4.416	4.410	681	6	A1	04	8A	FF	32	FF						
17	30.921	30.915	680	1	A3											
18	30.931	30.925	681	6	A1	04	8A	FF	32	FF						
19	32.014	32.008	680	1	A3											
20	32.025	32.019	681	6	A1	04	8A	FF	32	FF						
21																
22	Switching to the Audio page															
23	0.475	0.000	681	3	19	2A	00									req to jump to menu channel/area ID 00
24	0.476	0.001	680	1	BA											
25	0.496	0.021	681	5	1A	23	00	01	00							request to send data in area ID 00
26	0.498	0.023	680	1	BB											
27	0.512	0.037	680	8	23	09	00	60	09	00	00	00			0x09 0x00	draw into the screen area ID, 0x00 in this example
28	0.521	0.046	680	8	24	00	00	6E	00	5B	00	61			0x60	graphic rectangle block follows after params
29	0.531	0.056	680	8	25	06	08	00	6D	00	00	00				dimension of rectangle: 110x91 (0x006E x 0x005B)
30	0.543	0.068	680	8	06	61	0F	10	00	37	00	1E			0x61	ASCII block follows after params

Slika 5.3. Tablica „MKV GTI CAN-BUS data“. [14]



Generate Random Hex

cross-browser testing tools

World's simplest online random hexadecimal generator for web developers and programmers. Just press the Generate Hex button, and you'll get random hexadecimal numbers. Press a button – get hexadecimals. No ads, nonsense, or garbage.

Like 51K

Announcement: We just launched Online Number Tools – a collection of browser-based number-crunching utilities. Check it out!

```
fd1
45f
bbf
a49
c9d
f5f
ca2
5bd
b19
95a
fd6
180
046
```

How many digits? 3 How many results? 50

Generate Hex | Copy to clipboard (undo)

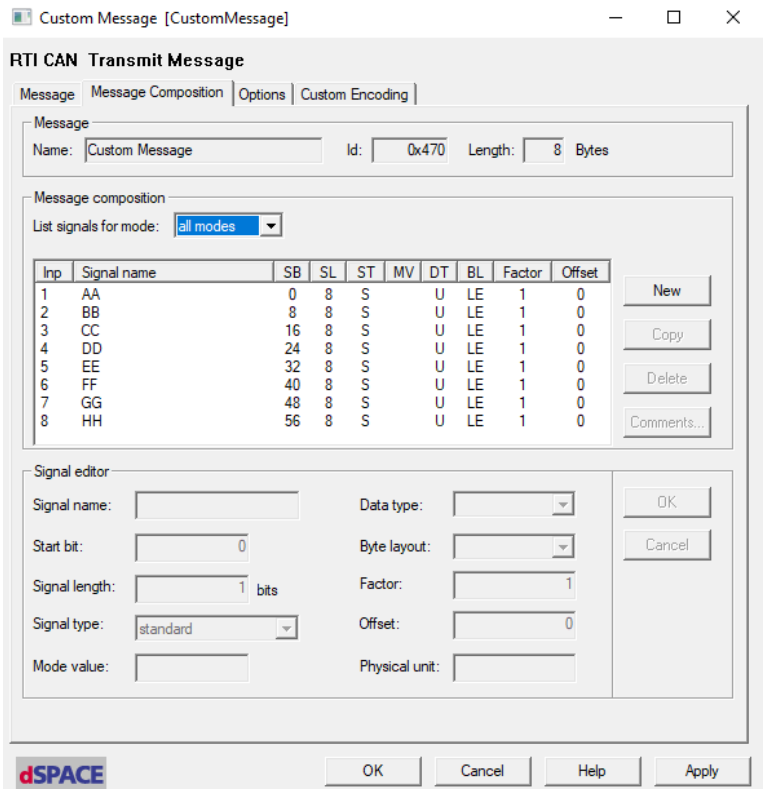
Slika 5.4. Generiranje nasumičnih adresa za testiranje web aplikacijom „Generate Random Hex“ [16]

0x5A0							
AA	BB	CC	DD	EE	FF	GG	HH
0x588							
AA	BB	CC	DD	EE	FF	GG	HH
0x100							
AA	BB	CC	DD	EE	FF	GG	HH
0x470							
AA	BB	CC	DD	EE	FF	GG	HH

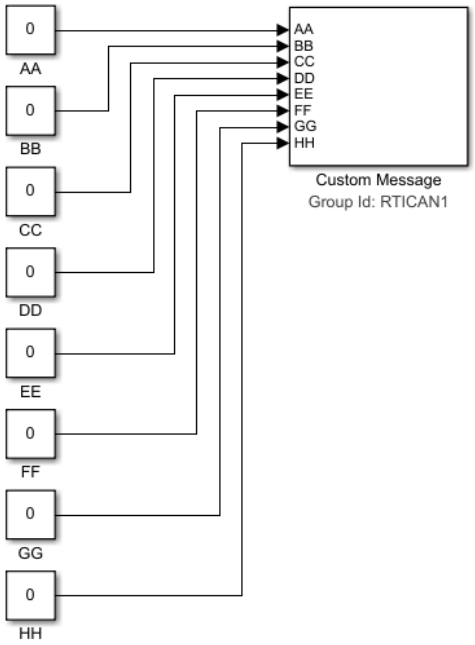
Slika 5.5. Tablica spremna za popunjavanje tijekom testiranja

5.2.2. Testiranje CAN adresa

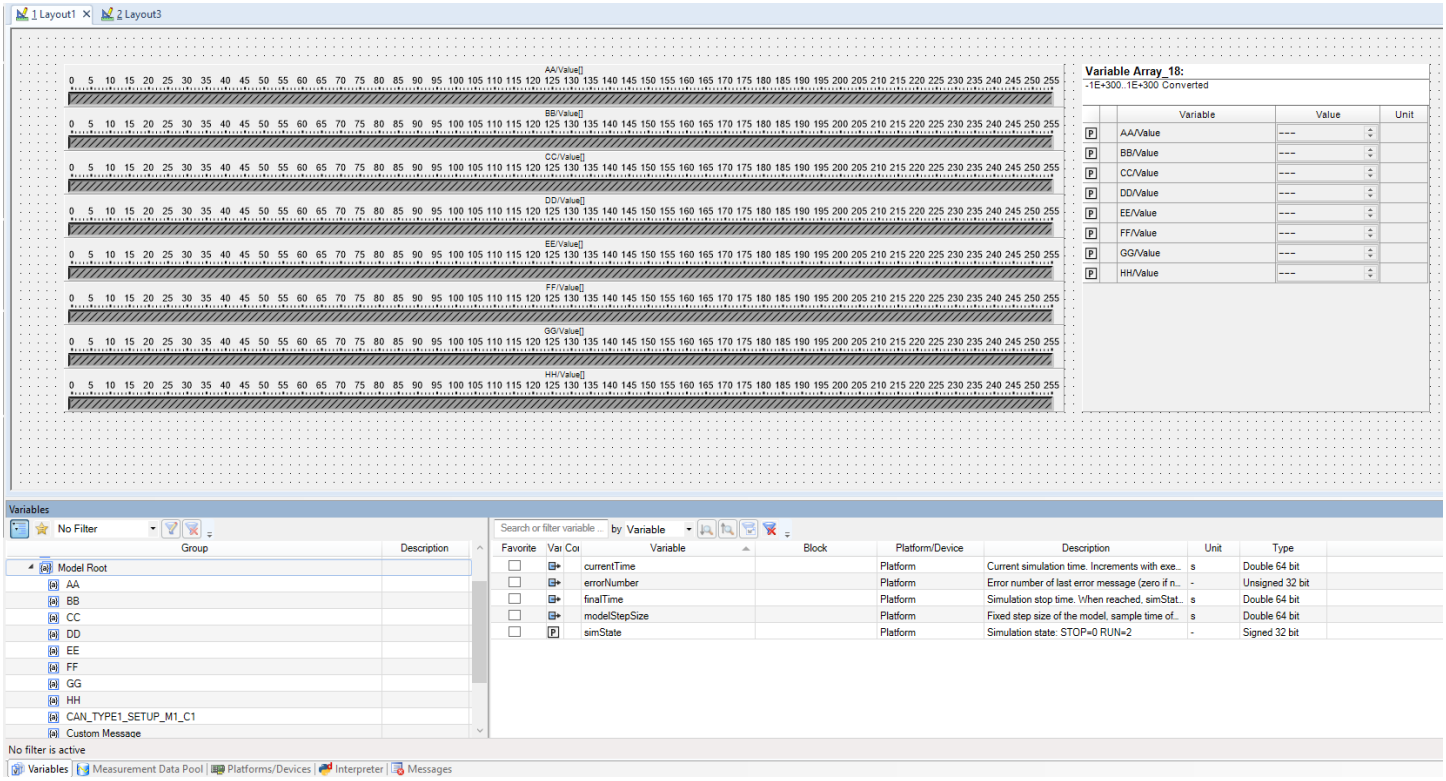
U svrhu provjere i potvrde funkcionalnosti identificiranih CAN adresa, provedeno je istraživanje koje uključuje niz koraka: kreiranje i izgradnja Simulink modela te izrada i izvršavanje ControlDesk aplikacije (podrazumijeva proces učitavanja i izvođenja aplikacije na MicroLabBoxu). Ovaj pristup omogućava precizno manipuliranje poljima CAN poruke i praćenje reakcija kontrolne ploče tijekom simulacije. Simulink model sastoji se od bloka CAN poruke s osam polja (Slika 5.6.). Polja CAN poruke predstavljena su tipa „Constant“ (Slika 5.7.), što omogućuje jednostavnu manipulaciju vrijednostima polja u ControlDesk aplikaciji. Izgradnjom modela, generirana je .sdf (*eng. System Description File*) datoteka u kojoj su sadržane informacije o modelu. Pri kreiranju ControlDesk aplikacije, .sdf datoteka je učitana i omogućeno je dodavanje varijabli iz Simulink modela u aplikaciju. Aplikacija se sastoji od osam klizača, svaki odgovoran za podešavanja vrijednosti od 0 do 255 za određeno polje CAN poruke (Slika 5.9.). Prilikom mijenjanja vrijednosti klizača, potrebno je istovremeno promatranje kontrolne ploče i dokumentriranje svake akcije na određenoj adresi. Testiranjem je otkriveno ukupno sedam aktivnih adresa od stotinjak testiranih. Aktivne adrese su 0x280, 0x5A0, 0X050, 0X470, 0X3D0, 0X2A0 i 0X58C. Na slici 5.9. dostupne su informacije koje su otkrivene testiranjem, a iskorištene u poglavlju 5.3.



Slika 5.6. Parametri bloka CAN poruke



Slika 5.7. Polja CAN poruke u Simulink modelu za testiranje.



Slika 5.8. Izgled aplikacija za testiranje CAN adresa

0x280 - Samo RPM							
AA	BB	CC	DD	EE	FF	GG	HH
	Okretaji	Okretaji					

0x5A0							
AA	BB	CC	DD	EE	FF	GG	HH
	brzina	brzina	ABS 1	Zvuk 64 na 63			
			ESP 2				
			ABS ESP 3				
			PritisakUGumama 8				

0x050 AirBag&PojasWL							
AA	BB	CC	DD	EE	FF	GG	HH
	Bit 0 - Airbag	Bit 2 - Pojas					
	Bit 1: Prikazuje "Airbag, ugašen" poruku						
	Bit 3: Prikazuje "Airbag Error" poruku						

0x470 VrataSvjetla							
AA	BB	CC	DD	EE	FF	GG	HH
Bit 0 - Lijevi žmigavac	Bit 0: Prednja lijeva	Osvjetljenje		Bit 0 - "Pritisnite kvačilo"	Bit 7: "Zamjenite bateriju ključa"		Bit 5: Duga svjetla
Bit 1 - Desni žmigavac	Bit 1: Prednja desna	0,100 - Full					Bit 6: Maglenka
Bit 7 - Akumulator (128+)	Bit 2: Zadnja lijeva	1-99 fade					
	Bit 3: Zadnja desna						
	Bit 4:Hauba						
	Bit 5: Gepek						

0x3D0							
AA	BB	CC	DD	EE	FF	GG	HH
	Servo						
	0-Nista						
	1 -Crven						
	2- Žut						
	3- Crven						

0x2A0							
AA	BB	CC	DD	EE	FF	GG	HH
						ESP - 0-7	
						TC - 8-16	
						bit2 @ DD 0x5A0 treba bit 1	

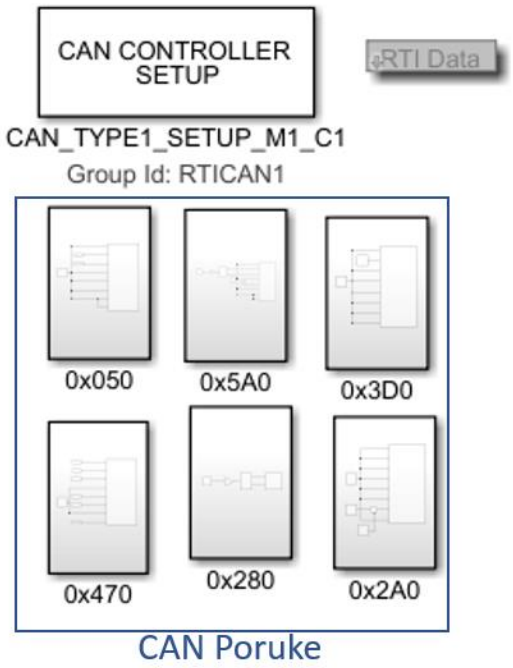
Slika 5.9. Rezultati testiranja CAN adresa.

5.3.Realizacija programskog rješenja

Za realizaciju programskog rješenja kao i za testiranje, bilo je potrebno kreiranje i izgradnja Simulink modela, te kreiranje i izvršavanje „Can Controller“ ControlDesk aplikacije. U Simulink model Can Controller aplikacije, dodane su CAN poruke za koje je testiranjem otkriveno da sadrže informacije za upravljanje kontrolne ploče.

5.3.1. Simulink model CAN Controller aplikacije

Simulink model CAN Controller aplikacije (Slika 5.10.) sastoji se od tri vrste blokova: poruke, Can Controller blok, RTI data blok. U Can Controller bloku podešava se brzina prijenosa, Can sučelje za MicroLabBox i iznos terminacijskog otpora. RTI data blok služi za siguran prijenos podataka, a blok poruka služi za konfiguriranje poruka.



Slika 5.10. Izgled Simulink modela CAN Controller aplikacija.

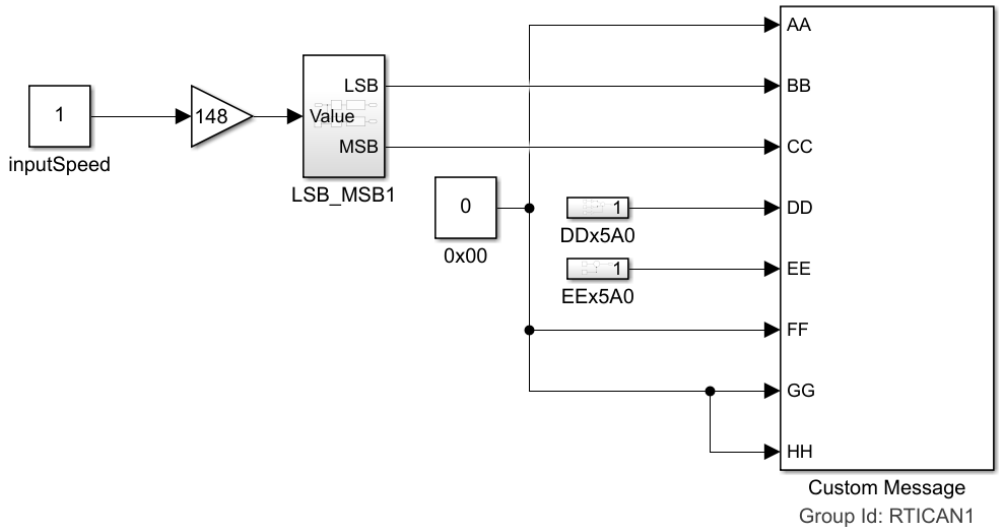
U poglavljima 5.3.2. – 5.3.7. opisani su blokovi CAN poruka za pojedine adrese.

5.3.2. Blok poruke 0x05A0

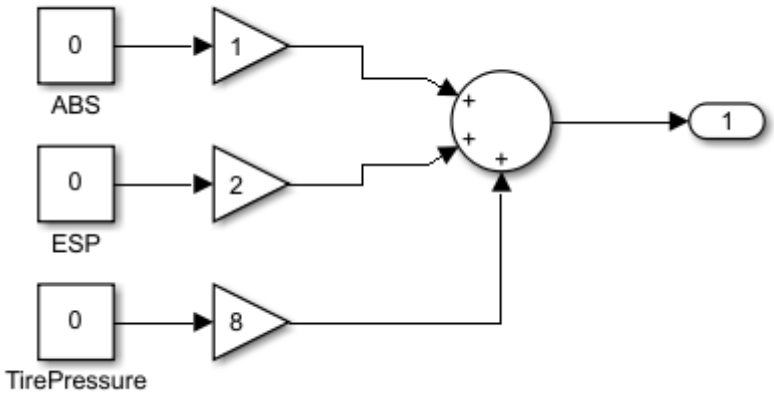
Provedenim istraživanjem u poglavlju 5.2. utvrđeno je kako se na adresi 0x5A0 nalaze informacije za upravljanjem kazaljkom brzine, informacije za paljenje lampica ABS-a (eng. *Anti-Lock Breaking System*), ESP-a (eng. *Electronic Stability Programme*) i pritiska u gumama, te informacija o proizvodnji zvuka upozorenja (Slika 5.9.). Polja BB i CC služe za upisivanje

vrijednosti brzine, polje DD za lampice, a polje EE za generiranje zvuka upozorenja. Po tim informacijama, konstruiran je blok za poruku adresi 0x5A0 dostupan na slici 5.11.

Testiranjem je utvrđeno kako je na polje BB i CC potrebno poslati prvih, odnosno drugih osam bitova, željene brzine pomnožene sa 148. Dodatno utvrđeno je da u polju DD vrijednosti 1, 2 i 8 pale različite lampice. U simulink modelu to je podržano na način da se za željenu lampicu vrijednost konstante postavi u 1 kako bi se vrijednost za polje DD promjenila. (Slika 5.12.)



Slika 5.11. Blok poruke za adresu 0x5A0.

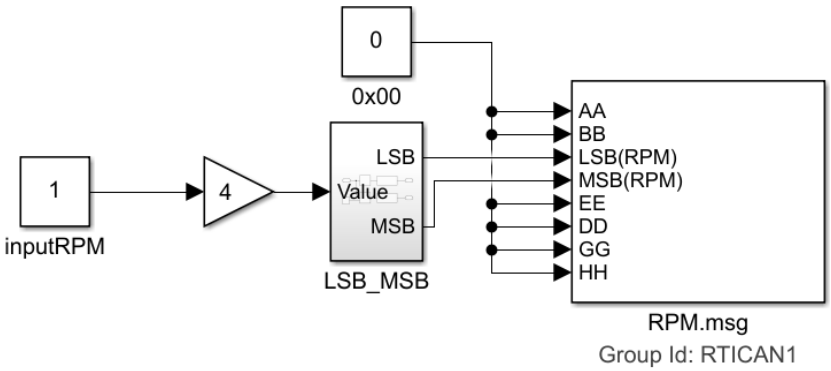


Slika 5.12. Blok polja DD poruke za adresu 0x5A0.

5.3.3. Blok poruke 0x280

Na adresi 0x280 nalaze se jedino informacije za upravljanje kazaljkom za okretaje. Željena vrijednost broja okretaja množi se s 4 te se predaje na analogan način kao vrijednost brzine

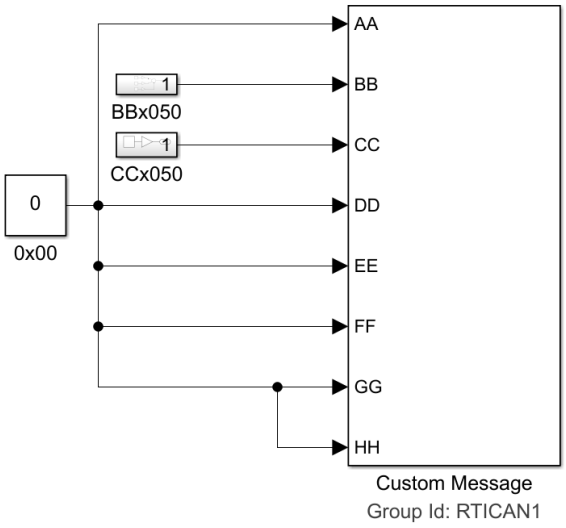
(opisano u poglavlju 5.3.2.). Na ostala polja adrese 0x280 šalje se vrijednost 0 budući da polja nisu od značaja za upravljanje kontrolnom pločom (Slika 5.13.).



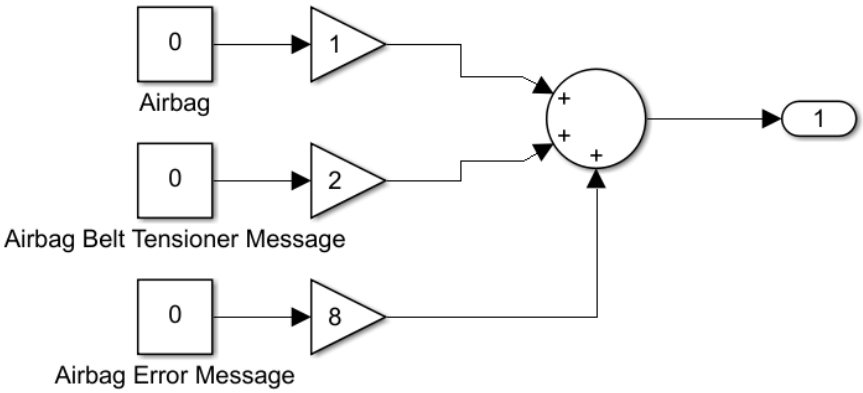
Slika 5.13. Blok poruke za adresu 0x280

5.3.4. Blok poruke 0x050

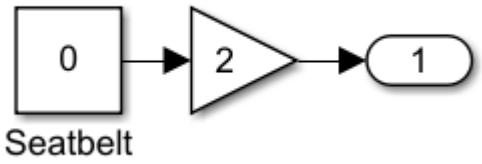
Kod adrese 0x050 relevantna su polja BB i CC. Na nulti bit polja BB vezana je lampica za zračni jastuk, a na prvi i treći bit vezane su poruka upozorenja za zračni jastuk. Stanje drugog bita polja CC određuje je li lampica za pojas upaljena odnosno ugašena. Slikom 5.14. prikazan je blok za poruku adrese 0x050,a slikama 5.15. i 5.16. prikazana je implementacija opisanih slučajeva za polja BB i CC.



Slika 5.14. Blok poruke za adresu 0x050.



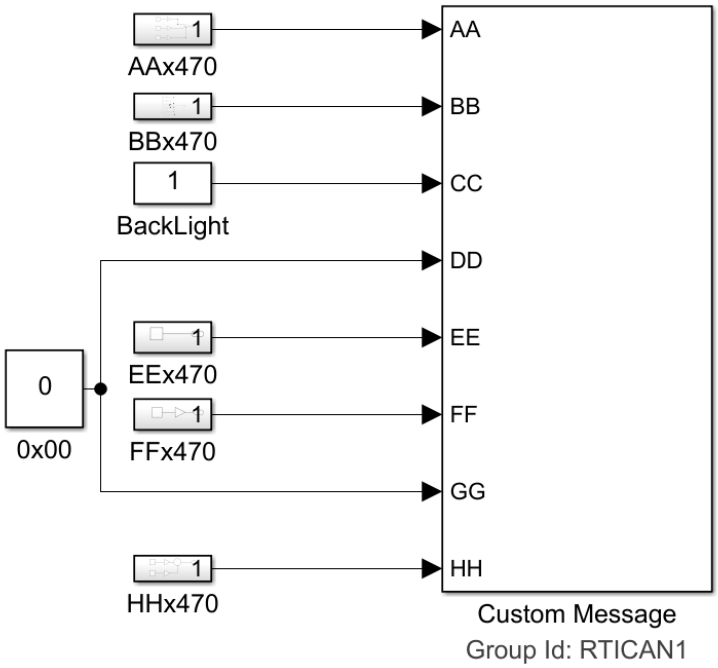
Slika 5.15. Polje BB poruke za adresu 0x050.



Slika 5.16. Polje CC poruke za adresu 0x050.

5.3.5. Blok poruke 0x470

Adresa 0x470, je najsadržajnija otkrivena adresa. Polje AA sadrži informacije o pokazivačima smjera, bitovi polja BB iscrtavaju na zaslonu prikaz otvorenih vrata, dok polje CC služi za podešavanje osvjetljenja kontrolne ploče. Na poljima EE i FF nalaze se poruke za pritisak kvačila i zamjenu baterije ključa, a na polju HH nalaze se bitovi za paljenje lampice maglenki i dugog svjetla. Samim time, blok poruke 0x470 je najopsežniji (Slika 5.17.).

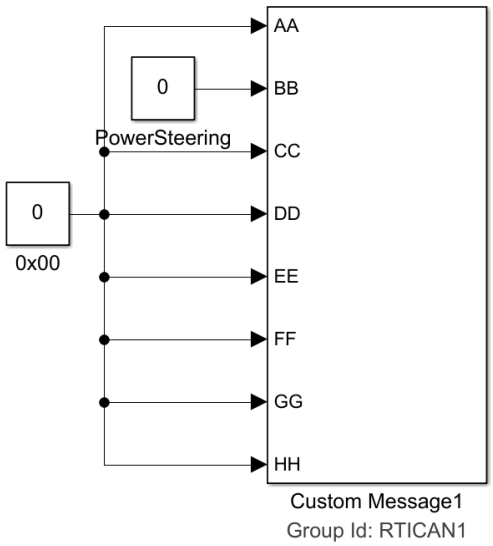


Slika 5.17. Blok poruke za adresu 0x470

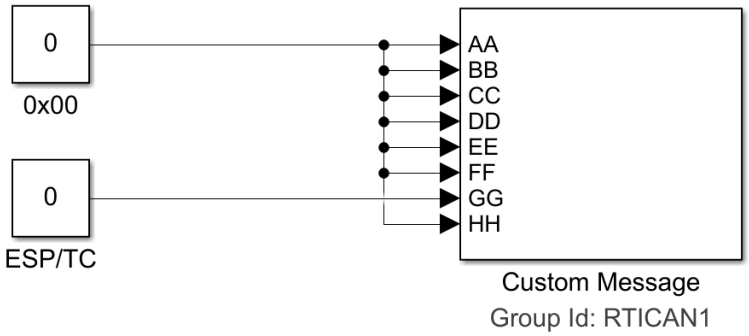
Paljenje lampica i iscrtavanje prikaza otvorenih vrata implementiran je kao i za polje BB adrese 0x050. (Slika 5.15.)

5.3.6 Blokovi poruka 0x3D0 i 0x2A0

Za adrese 0x3D0 i 0x2A0 relevatno je samo jedno polje. Za adresu 0x3D0 to je polje BB, koje u ovisnosti o stanjima bitova pali lampice za servo, dok polje GG adrese 0x2A0 pali lampice za ESP i proklizavanje. Blokovi poruka za adrese 0x3D0 i 0x2A0 dostupni su na slikama 5.18. i 5.19.



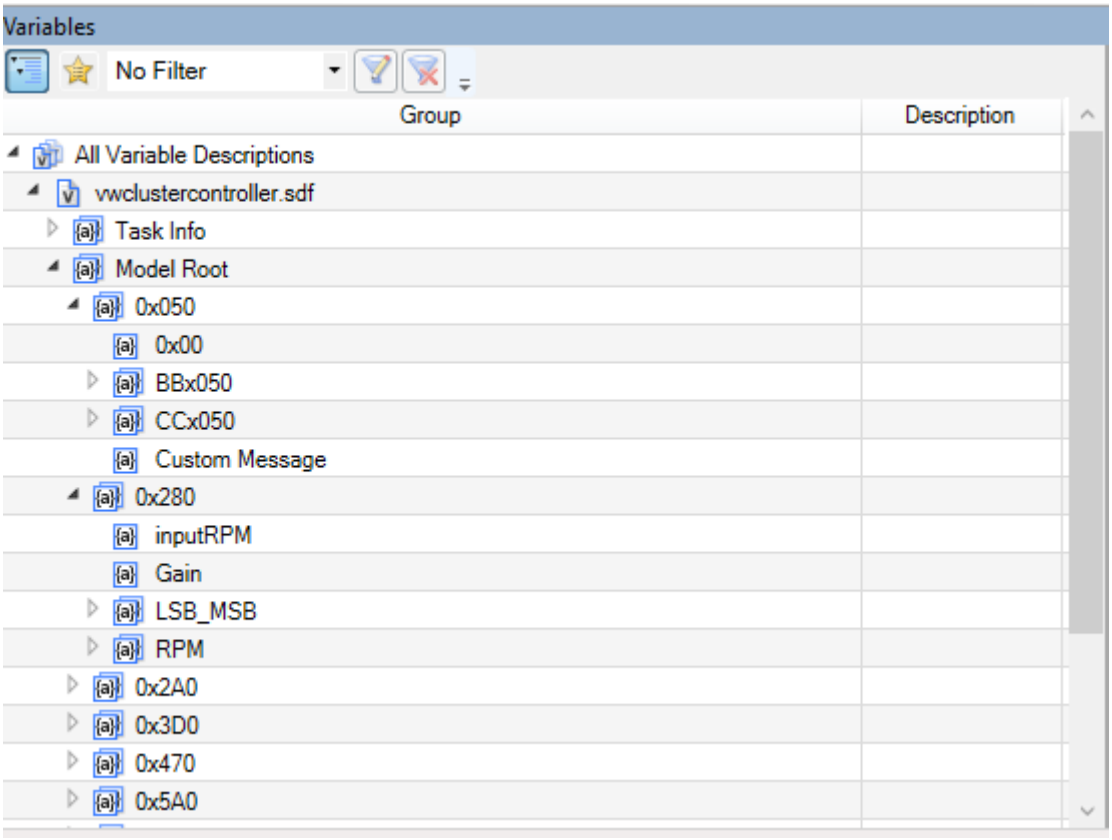
Slika 5.18. Blok za poruku adrese 0x3D0



Slika 5.19. Blok za poruku adrese 0x2A0

5.3.7. CAN Controller ControlDesk aplikacija

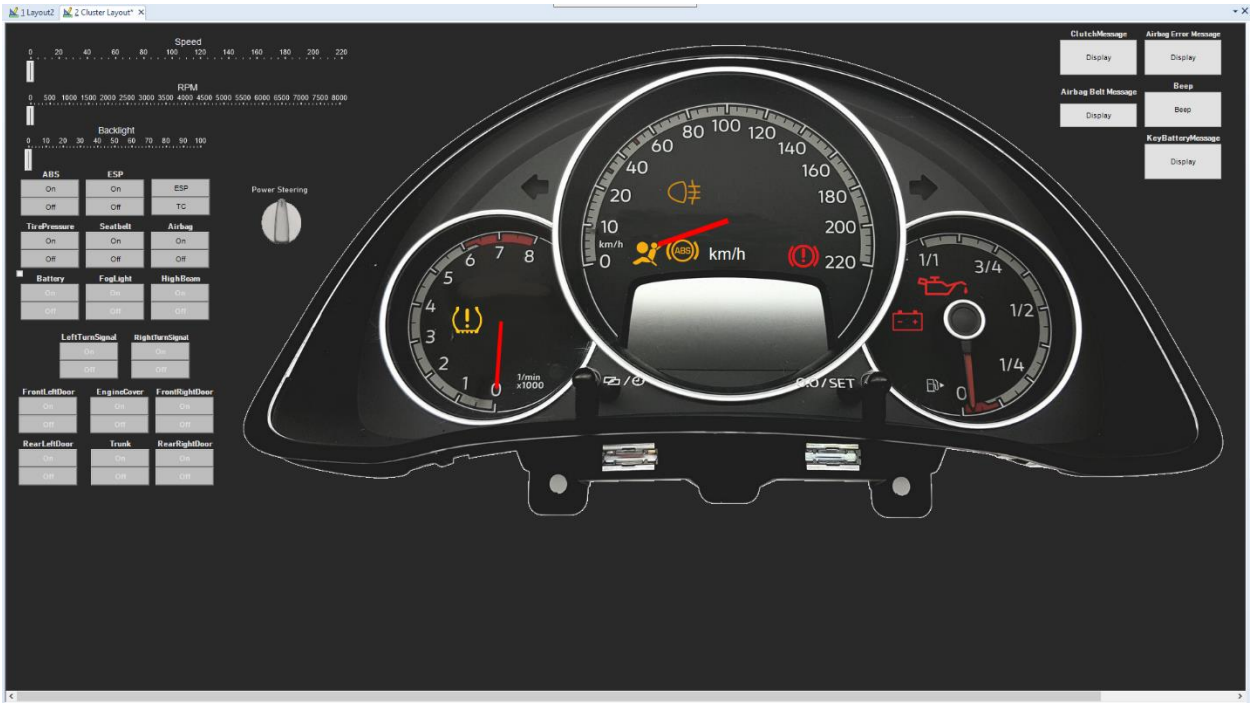
Izgradnjom Simulink modela, generiran je tzv. System Description File (skraćeno .sdf) kojim ControlDesk alat dobiva informaciju o generiranom modelu. Jednom kada je ControlDesk projekt kreiran i platforma za izvođenje povezana, učitava se .sdf datoteka i omogućena je manipulacija varijablama iz modela (Slika 5.20.).



Slika 5.20. Prikaz dostupnih varijabli iz modela u ControlDesku.

Iz dostupnih varijabli kreirano je grafičko sučelje za upravljanje kontrolnom pločom (Slika 5.21.). Varijablama za poruke i lampice se manipulira putem gumbova, a kazaljkama brzine i

okretaja upravlja se pomoću klizača. Dodatno, moguće je klizačem kontrolirati i osvjetljenje zaslona i kazaljki na ploči. Grafičko sučelje aplikacije napravljeno je tako da ploča u aplikaciji prati stanje prave, fizičke ploče. U programskom jeziku python, napisan je kod koji na pritisak određenih gumbova pali lampice i na pravoj i na virtualnoj ploči (Slika 5.22.). Isječak koda dostupan je na slici 5.23.



Slika 5.21. Grafičko sučelje ControlDesk aplikacije za upravljanje kontrolnom pločom.

Slika 5.22. Prikaz svih mogućih akcija na kontrolnoj ploči

```
def On_Application_LayoutManagement_Cluster0020Layout_ABS_btn_ButtonClicked(Sender, Button):
    """
    Syntax     : On_Application_LayoutManagement_Cluster0020Layout_ABS_btn_ButtonClicked(Sender, Button)
    Parameters: Sender, Button
    """
    if(Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["ABS_btn"].Value==1):
        Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["ABS_frame"].Background.Picture.Visible = True
    else:
        Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["ABS_frame"].Background.Picture.Visible = False
    print("On_Application_LayoutManagement_Cluster0020Layout_ABS_btn_ButtonClicked")

def On_Application_LayoutManagement_Cluster0020Layout_Airbag_ButtonClicked(Sender, Button):
    """
    Syntax     : On_Application_LayoutManagement_Cluster0020Layout_Airbag_ButtonClicked(Sender, Button)
    Parameters: Sender, Button
    """
    if(Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["Airbag"].Value==1):
        Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["AirBag_frame"].Background.Picture.Visible = True
    else:
        Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["AirBag_frame"].Background.Picture.Visible = False
    print("On_Application_LayoutManagement_Cluster0020Layout_Airbag_ButtonClicked")

def On_Application_LayoutManagement_Cluster0020Layout_FogLight_btn_ButtonClicked(Sender, Button):
    """
    Syntax     : On_Application_LayoutManagement_Cluster0020Layout_FogLight_btn_ButtonClicked(Sender, Button)
    Parameters: Sender, Button
    """
    if(Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["FogLight_btn"].Value==1):
        Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["Fog_frame"].Background.Picture.Visible = True
    else:
        Application.LayoutManagement.Layouts["Cluster Layout"].Instruments["Fog_frame"].Background.Picture.Visible = False
    print("On_Application_LayoutManagement_Cluster0020Layout_FogLight_btn_ButtonClicked")
```

Slika 5.23. Isječak python koda za upravljanje virtualnom pločom.

6. Zaključak

Upravljanje kontrolnom pločom automobila korištenjem CAN sabirnice predstavlja inovativan korak prema poboljšanju iskustva simuliranog utrkivanja u simracing zajednici. Kroz ovaj diplomski rad, istražene su mogućnosti integracije naprednih tehnologija kako bi se simulacija vožnje približila stvarnom iskustvu na stazi. Implementacijom komunikacije između kontrolne ploče i računala omogućeno je potencijalno prenošenje telemetrije iz videoigre na stvarnu kontrolnu ploču, poboljšavajući time autentičnost iskustva simuliranog utrkivanja.

Kroz diplomski rad temeljno je istraženo područje CAN protokola i CAN komunikacije. Dan je povijesni razvoj protokola i predstavljeni su pojmovi poput: CAN kontrolera, CAN prijemnika, mreže i čvorova. Pojašnjene su vrste CAN okvira i mehanizam nadopune bitova, te je time stečeno znanje za izradu sklopovskog i programskog rješenja.

U sklopu diplomskog rada kreirani su Matlab Simulink modeli za podešavanje i slanje CAN poruka te je ControlDesk alatom razvijena aplikacija za upravljanje kontrolnom pločom automobila. Aplikacija se izvršava na MicroLabBox uređaju, a postignuto je uspješno paljenje deset lampica upozorenja, interakcija s kazaljkama broja okretaja i brzine, podešavanje osvjetljenja kazaljki te prikaz poruka i grafike za otvorena vrata na automobilu. Uz akcije na fizičkoj kontrolnoj ploči, u aplikaciju je integrirana i virtualna kontrolna ploča koja pri izvršavanju aplikacije prikazuje akcije sukladno s fizičkom kontrolnom pločom.

Jedno od važnih područja proširenja odnosi se na mogućnost testiranja drugih vrsta kontrolnih ploča. Također, predviđeno je da će se ova aplikacija koristiti u okviru laboratorijskih vježbi kolegija "Metode i tehnike ispitivanja automobilske programske podrške". Ova integracija omogućit će studentima pristup inovativnom alatu za učenje i praktičnu primjenu u području ispitivanja automobilske programske podrške, stvarajući most između teorije i stvarnog rada u industriji.

LITERATURA

[1] Amstudio, video „HOW TO WIRE BMW SPEEDO CLUSTER FOR SIMULATOR | SIMHUB“, [online], dostupno na: https://www.youtube.com/watch?v=L59NL9EuEdM, [25.8.2023.]

[2] Mygithubadel, Github repozitorij „vw-can-bus-gauge-cluster-controller“, [online], dostupno na: https://github.com/mygithubadel/vag-can-bus-gauge-cluster-controller, [25.8.2023]

[3] Borna Sirovec, Github repozitorij „Custom-Dashboard-STM32-BluePill-SimHub“, [online], dostupno na: https://github.com/bosirovec/Custom-Dashboard-STM32-BluePill-SimHub, [25.8.2023.]

[4] Andraž Vene, Github repozitorij „VW-Instrument-Cluster-Controller“, [online], dostupno na: https://github.com/an-ven/VW-Instrument-Cluster-Controller [25.8.2023.]

[5] Borna Sirovec, dnevnik stručne prakse: „Development\test inženjer\ka za dSPACE Ethernet Configuration Tool“, Valpovo, 2023.

[6] CAN in Automation, članak History of CAN technology, [online], dostupno na: https://www.can-cia.org/can-knowledge/can/can-history [30.8.2023.]

[7] Steve Corrigan, Controller Area Network Physical Layer Requirements, [online], dostupno na: https://www.ti.com/lit/an/slla270/slla270.pdf?ts=1619435404297&ref_url=https%253A%252 [30.8.2023.]

[8] Vector, CAN_E, online trening tečaj, [online], dostupno na: https://elearning.vector.com/mod/page/view.php?id=333#section-3 [2.9.2023.]

[9] Vector, CAN Communication Principle, animacija, [online], dostupno na: https://elearning.vector.com/mod/page/view.php?id=343 [2.9.2023.]

[10] J.A.Cook, J.S.Freudenberg, Controller Area Network(CAN), 2008.

[11] dSPACE, brošura MicroLabBox proizvoda

[12] Wiktionary, Protoboard, [online], dostupno na: https://en.wiktionary.org/wiki/protoboard [5.9.2023.]

[13] Mathworks, stranica alata Simulink, [online], dostupno na: https://uk.mathworks.com/products/simulink.html [25.11.2023.]

[14] dSPACE, ControlDesk, Experiment and Instrumentation Software for ECU Development, [online], dostupno na: https://www.dspace.com/en/inc/home/products/sw/experimentandvisualization/controldesk.cfm, [25.11.2023.]

[15] Steven Wollman, tablica MKV GTI CAN-BUS data, [online], dosutpno na: https://docs.google.com/spreadsheets/d/1eirT8LbSRl4j06BpwgsiE4PM_2BGH9UStdWLXwKvHJw/edit?usp=drive_link [25.11.2023.]

[16] Browserling, web aplikacija Generate Random Hex, [online], dostupno na: https://www.browserling.com/tools/random-hex [25.11.2023.]

SAŽETAK

U ovom diplomskom radu istraženo je upravljanje kontrolnom pločom automobila putem CAN sabirnice u svrhu unaprjeđenja iskustva simuliranog utrkivanja u simracing zajednici. Korištenjem Simulink modela, ControlDesk aplikacije i MicroLabBoxa ostvarena je komunikacija između računala i stvarne kontrolne ploče, što je omogućilo slanje poruka iz ControlDesk aplikacije na stvarnu kontrolnu ploču. Ovaj rad predstavlja inovativni korak prema autentičnijem iskustvu vožnje u virtualnom svijetu i doprinosi rastućoj zajednici simracinga.

Ključne riječi: CAN, ControlDesk, kontrolna ploča automobila, MATLAB Simulink, MicroLabBox, simracing

ABSTRACT

Title: Car dashboard control using the CAN bus

This master's thesis explores the management of a car's dashboard through the CAN bus to enhance the experience of simulated racing in the simracing community. Utilizing Matlab Simulink model, ControlDesk application and MicroLabBox communication between the computer and the actual dashboard was established, enabling the transmission of CAN messages from the ControlDesk application to the real dashboard. This work represents an innovative step towards a more authentic driving experience in the virtual world and contributes to the growing simracing community.

Keywords: CAN, ControlDesk, car dashboard, MATLAB Simulink, MicroLabBox, simracing

ŽIVOTOPIS

Borna Sirovec rođen je 2. kolovoza 1999. godine u Osijeku. Osnovnu i srednju školu završava u Valpovu. Po završetku Opće gimnazije, 2018. godine, upisuje preddiplomski studij Matematike i Računarstva na Odjelu za Matematiku Sveučilišta Josipa Jurja Strossmayera u Osijeku. Preddiplomski studij završava 2021. godine, a tokom iste godine upisuje diplomski studij Automobilskog računarstva i komunikacija na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku.

_____

Potpis autora