



Juho Ketola

SR3-veturin apuvoimalaitteen väyläjärjestelmät

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Ajoneuvotekniikka

Insinöörityö

23.10.2024

Tiivistelmä

Tekijä: Juho Ketola
Otsikko: SR3-veturin apuvoimalaitteen väyläjärjestelmät
Sivumäärä: 26 sivua
Aika: 23.10.2024

Tutkinto: Insinööri (AMK)
Tutkinto-ohjelma: Ajoneuvotekniikka
Ammatillinen pääaine: Autosähkötekniikka
Ohjaajat: Lehtori Pasi Kovanen

Insinööriyön tavoitteena oli VR Kunnossapito Oy:n toimeksiannosta tutustua Sr3-veturiin ja sen väyläjärjestelmiin tarkoituksena kehittää menetelmä, jolla apuvoimalaitteen saa käynnistettyä huoltotilanteessa ilman veturissa sijaitsevia hallintalaitteita.

Työssä kuvataan raideliikenteen väyläjärjestelmiä, CAN-väylän peruseriaatteita ja eri protokollia erityisesti J1939-protokollan osalta. Lisäksi työssä tarkastellaan datalukijan käyttöä, viestien analysointia ja lähettämistä. Työssä selvitettiin veturin apuvoimalaitteen väyläprotokolla ja tarvittavat käynnistämiseen liittyvät viestit dataluvun avulla.

Luetun datan analysointi onnistui, ja CAN-väylän viestintäprotokolla pystyttiin määrittämään. Mahdollinen käynnistyskäsky löytyi, ja useita apuvoimalaitteen tilatietoja sisältäviä viestejä tunnistettiin. Vaikka jotkin viestit jäivät tuntemattomiksi, niiden uskotaan liittyvän laitteen omaan vikadiagnostiikkaan. Datan lukemista varten valmistettu laitteisto todettiin toimivaksi ja valittu ohjelmisto toimi tarkoituksenmukaisesti.

Aikarajoitukset estivät apuvoimalaitteen käynnistämisen tässä työssä, mutta väyläjärjestelmän toiminta saatiin selvitettyä ja jatkotoimenpiteet määriteltyä. Työn tuloksia on esitelty yleisellä tasolla salassapitovelvollisuutta noudattaen ja tarkemmat tulokset kirjataan erilliseen raporttiin työn tilaajalle. Tämän työn myötä ymmärrys apuvoimalaitteen väyläjärjestelmistä täydentyi ja tilaajalle pystyttiin selventämään vaatimuksia laitteen käynnistämiseksi.

Avainsanat: CAN-väylä, J1939, Train Communication Network

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author: Juho Ketola
Title: Bus Systems of the Auxiliary Power Unit in the SR3 Locomotive
Number of Pages: 26 pages
Date: 23 October 2024

Degree: Bachelor of Engineering
Degree Programme: Automotive Engineering
Professional Major: Automotive Electrics
Supervisor: Pasi Kovanen, Senior Lecturer

The aim of this thesis, commissioned by VR Fleetcare, was to familiarize with the Sr3 locomotive and its bus systems with the purpose of developing a method to start the auxiliary power unit during maintenance without using the control devices located in the locomotive.

The thesis describes the bus systems of rail transport, the basic principles of the CAN bus, and various protocols, especially the J1939 protocol. Additionally, the thesis examines the use of a datalogger, and the analysis and transmission of messages. The bus protocol of the locomotive's auxiliary power unit and the necessary messages for starting it were determined using data reading.

The analysis of the read data was successful, and the communication protocol of the CAN bus could be defined. A possible start command was found, and several messages containing status information of the auxiliary power unit were identified. Although some messages remained unknown, they are believed to be related to the device's own fault diagnostics. The equipment prepared for datalogging was found to be functional, and the selected software worked appropriately.

Time constraints prevented the starting of the auxiliary power unit in this work, but the operation of the bus system was clarified, and further actions were defined. The results of the work have been presented at a general level in compliance with confidentiality obligations, and more detailed results will be recorded in a separate report for the client.

Keywords: CAN bus, J1939, Train Communication Network

Sisällys

Lyhenteet

1	Johdanto	1
2	Sähköveturin tekniikka	2
2.1	Sr3-veturi	3
2.2	Apuvoimalaiteyksikkö	4
3	Väyläjärjestelmät sähkövetureissa	5
3.1	Train Communication Network	6
3.2	Controller Area Network	7
3.2.1	Fyysinen kerros	9
3.2.2	Siirtoyhteyskerros	10
3.2.3	Sovelluskerros	14
3.3	Väyläprotokollat ja standardit	14
4	Datalukijan käyttö väyläviestinnässä	18
4.1	Datalukijan esittely	18
4.2	Väyläviestien lukeminen ja analysointi	19
4.3	Väyläviestien lähettäminen	22
5	Käytännön toteutus ja testaus	23
5.1	Toteutusympäristö ja laitteisto	23
5.2	Testausprosessin kuvaus	24
5.3	Testitulokset ja niiden analysointi	25
6	Yhteenveto	26
	Lähteet	27

Lyhenteet

- APU: Auxiliary power unit. Apuvoimalaiteyksikkö. Dieselkäyttöinen generaattori.
- Sr3: VR:n käyttämä sähköveturi.
- CAN: Controller Area Network. Automaatioväylä, jota käytetään eri laitteiden väliseen kommunikointiin.
- TCN: Train Communication Network. Junissa ja vetureissa käytetty raide liikenteen oma väyläjärjestelmä.
- J1939: CAN-väylän sovellustason standardi. Käytössä erityisesti raskaassa liikenteessä.

1 Johdanto

Insinööriyön tavoitteena oli tutustua Sr3-veturin apuvoimalaitteen väyläjärjestelmään ja kehittää menetelmä, jolla apuvoimalaitteen saa käynnistettyä huoltotilanteessa ilman veturissa sijaitsevia hallintalaitteita. Sen avulla laitteen toiminta pystytään testaamaan välittömästi huollon jälkeen jo huolto paikalla, jolloin suurta laitetta ei tarvitse asentaa takaisin veturiin ennen kuin sen toiminta on varmistettu. Tämä mahdollistaa työn tilaajalle apuvoimalaitteiden itsenäisen huollon, mikä säästää merkittävästi resursseja.

Työssä tarkastellaan sähköveturin tekniikkaa ja sen väyläjärjestelmiä erityisesti apuvoimalaitteen osalta. Lisäksi kuvataan, kuinka väyläviestejä luetaan ja uudelleen lähetetään datalukijan avulla.

Työn tilaajana toimi VR Fleetcare viralliselta nimeltään VR Kunnossapito Oy. Yrityksen emoyhtiönä toimii VR-Yhtymä Oyj, joka on Suomen valtion omistama logistiikkakonserni. VR Fleetcare on raideliikenteen asiantuntijayritys, jonka palveluihin kuuluvat muun muassa raidekaluston huollot, modernisointihankkeet, komponenttipalvelut ja elinkaaren hallinta sekä digitaaliset palvelut. Yrityksellä on konepajat Helsingissä ja Pieksämäellä. Näiden lisäksi yrityksellä on projekti-keskus Oulussa ja kuusi varikkoa Suomessa. VR Fleetcaren pääkonttori sijaitsee Helsingin Ilmalassa.

2 Sähköveturin tekniikka

Sähkövetureiden käyttö on alkanut jo 1800-luvun lopulla, mikä tekee siitä yhden perinteisimmistä sähköisen liikkumisen muodoista. Ensimmäiset yritykset veturin sähköistämiseksi teki Robert Davidson Skotlannin Aberdeenissa vuosina 1837–1838. [1, s. 31.] Sähkörautateiden syntymävuotena pidetään vuotta 1879, jolloin Werner von Siemens esitteli sähköveturinsa. Tämän jälkeen tekniikka kehittyi vauhdilla ja ensimmäinen nykyisenkaltaista teknologiaa käyttävä vaihtovirtaveturi otettiin käyttöön Saksassa vuonna 1905. Veturi toimi yksivaihteisella 5500 voltin ja 15 hertsin vaihtosähkövirralla. [2, s. 6.]

Suomen ensimmäinen sähköistetty rataosuus rakennettiin jo vuonna 1903 Mustion ruukin ja rautatieaseman välille. Radan rakennuttajana toimi Hjalmar Linder, ja veturit olivat saksalaisen AEG:n valmistamia. Muita sähköistettyjä rataosuuksia tehtiin erityisesti teollisuuden tarpeisiin kuljettamaan tavaraa tehtailta rautatieasemille. Ensimmäiset varsinaiset valtionrahoitteiset sähkörautatiet rakennettiin vasta vuonna 1965. Myöhäisen aloituksen etuna oli, että radoilla voitiin käyttää uusinta tasasuuntaustekniikkaa. [3.] Ratojen sähköverkoksi valikoitui 25 kV:n ja 50 Hz:n yksivaihejärjestelmä, joka on vielä nykyäänkin paras sähkörautatietekniikka. Ensimmäisiksi juniksi valikoituivat Valmet Oy:n ja Oy Strömberg Ab:n valmistamat Sm1-sähkömoottorijunayksiköt, joita tilattiin vuonna 1968 yhteensä 30 kappaletta ja seuraavien vuosien aikana vielä 20 lisää. Viimeisimmät Sm1-yksiköt poistettiin käytöstä huhtikuussa 2016. [4.]

Nykyisenkaltaisen sähköveturi on sähköistetty raiteen yläpuolella sijaitsevasta ajojohtimesta saatavalla virralla. Tämä virta johdetaan ajojohtimesta veturiin sen katolla sijaitsevien virroittimien avulla muuntajille, jotka muuntavat jännitteen vastaamaan moottorin käyttämää jännitettä. Ajojohtimista saatavaa virtaa käytetään myös veturin muissa laitteissa.

2.1 Sr3-veturi

Sr3-veturi (kuva 1) on uusin VR:n käyttämästä kolmesta sähköveturisarjasta. Veturin on valmistanut Siemens Mobility. Veturi perustuu Siemens Mobilityn Vectron-veturimalliin, joka pohjautuu tekniikaltaan sitä edeltäneisiin Eurosprinter- ja Eurorunner-veturimalleihin. Moderni ja energiapihi veturi on muokattu Suomen haastaviin olosuhteisiin. Siinä on Vectronin laaja varustepaketti, joka sisältää muun muassa leveämmät telit, radiokauko-ohjausjärjestelmän, apuvaimalaitteiston ja talviolosuhteisiin tarkoitettuja varusteita. [5.]

Sr3-veturin sarjatunnuksen "S" tarkoittaa sähköä ja "r" raskasta akselipainoa (yli 15 tonnia). Veturin sähkömoottorit tuottavat 6400 kW jatkuvaa tehoa, ja sen huippunopeus on 200 km/h. Veturit on suunniteltu kestäväksi jopa 40 asteen pakkasia sekä pölyävää lunta. Vetureita voi nähdä niin matkustaja- kuin tavara- liikenteessä. [5.] Sen edeltäjät Sr1 ja Sr2 liikennöivät edelleen.



Kuva 1. Sr3-veturi [5].

VR Group tilasi helmikuussa 2014 yhteensä 80 Sr3-veturia, joiden toimitukset alkoivat vuonna 2016. Tällä hetkellä liikenteessä on 63 veturia, ja loput vetureista toimitetaan vuoteen 2026 mennessä. [6.]

2.2 Apuvoimalaiteyksikkö

Siemens Vectron vetureiden Last Mile -versiot on varustettu yhdellä tai kahdella Steyrmotorsin valmistamalla dieselvirtamoduulilla eli apuvoimalaiteyksiköllä. Apuvoimalaite toimii perinteisen dieselgeneraattorin tavoin. Polttomoottorin tuottama teho muunnetaan generaattorin avulla sähköksi, joka muunnetaan edelleen taajuusmuuttajan avulla vastaamaan veturin moottoreiden käyttämää jännitettä ja syötetään moottoreille. Sr3-vetureissa näitä apuvoimalaitteita on kaksi kappaletta, ja niiden tuottama yhteisteho on 360 kW. Tämä mahdollistaa veturin liikkumisen myös sähköistämättömillä rataosuuksilla. [7.] Apuvoimalaitteet on sijoitettu ohjaamon takana sijaitsevaan moottoritilaan ja ne on integroitu saumattomasti veturin sähköjärjestelmään nopean ja tehokkaan siirtymisen mahdollistamiseksi käyttövoimajärjestelmien välillä.

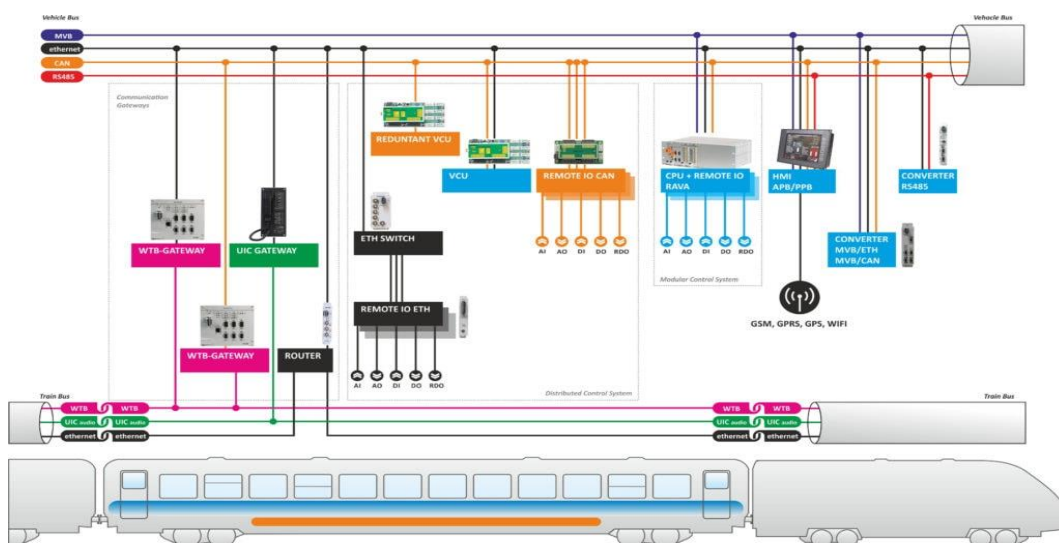
Apuvoimalaite on oma itsenäinen yksikkö. Yksikössä on turboahdetun dieselmoottorin lisäksi polttoainejärjestelmä, raitisilma- ja pakokaasujärjestelmä, generaattori liitinkoteloineen, sähkölaite- ja relekotelot, hydrostaattijärjestelmä, käynnistysakusto, jäähdytys- ja lämmitysjärjestelmät sekä palotorjuntalaitteisto. Moottorin polttoainetankit sijaitsevat laitteen ulkopuolella veturissa. Moottorin ja generaattorin ohjaus tapahtuu laitteen omalla moottorinohjausjärjestelmällä. Apuvoimalaitteessa on kaksi CAN-väylää: yksi sisäinen diagnostiikkaväylä ja toinen veturin kanssa kommunikointiin käytettävä väylä. Yksikön etuseinässä on näyttö laitteen diagnosointia varten. [8.]

3 Väyläjärjestelmät sähkövetureissa

Väyläjärjestelmiä käytetään tiedonsiirtoon ja kommunikointiin eri laitteiden välillä. Näitä hyödynnetään laajasti monilla aloilla, kuten tietokoneissa, teollisuusautomaatiossa ja ajoneuvoissa.

Aikaisemmin junien väyläjärjestelmät olivat usein junakohtaisia, mekaanisia ja sähkömekaanisia järjestelmiä. Monien vanhempien junien laitteita ohjattiin pneumaattisilla ja analogisilla sähkösignaaleilla. Usein laitteita ohjattiin laitekohtaisilla tiedonsiirtoväylillä, jotka tekivät väyläjärjestelmistä monimutkaisia ja raskaasti johdotettuja. Nykyisenkaltaiset vetureiden väyläjärjestelmät tulivat käyttöön 1990-luvulla, jolloin väyläjärjestelmiä standardisoitiin turvallisuuden ja selkeyden lisäämiseksi. [9.]

Nykyään yleisimmät väyläjärjestelmät sisältävät useita eri standardeja, jotka on suunniteltu erilaisiin sovelluksiin. Raideliikenteessä käytetään tyypillisesti Train Communication Network (TCN) -nimistä väyläjärjestelmää. [10.] Tämä järjestelmä toimii rinnakkain eri väylästandardien kuten, Controller Area Networkin (CAN) ja UIC-väylän kanssa. Näiden lisäksi junissa on erilaisia radio-ohjattavia järjestelmiä, I/O-järjestelmiä sekä Ethernet-verkkojärjestelmiä. TCN-yhdyskätävä mahdollistaa viestinnän eri standardien välillä. (Kuva 2.)



Kuva 2. Havainnekuva junan väyläjärjestelmästä [11].

3.1 Train Communication Network

Train Communication Network on nimensä mukaisesti rautatieajoneuvojen väyläjärjestelmä, joka on suunniteltu tiedonsiirtoon ja kommunikointiin vetureiden ja junan muiden vaunujen eri laitteiden ja järjestelmien välillä. TCN-järjestelmän kehitys alkoi 1990-luvun alkupuolella ja se standardisoitiin virallisesti 1999. [12.] Se on nykyään olennainen osa modernien junien toimintaa. TCN-järjestelmä varmistaa, että kaikki junan järjestelmät voivat kommunikoida luotettavasti ja tehokkaasti keskenään riippumatta käytetystä standardista.

Viestintäjärjestelmässä on yleensä kaksi pääväylää: junanlaajuinen väylä WTB (Wire Train Bus) ja vaunukohtainen sisäinen väylä MVB (Multifunction Vehicle Bus). Sisäinen väylä vastaa viestinnästä yksittäisen vaunun tai veturin sisällä, kun taas junanlaajuinen väylä huolehtii tiedonsiirrosta koko junan välillä. Yleisesti käytössä olevat TCN-yhdyskäytävät ovat modulaarisia, ja niitä voi muokata tarpeen mukaan. Perinteisesti yhdyskäytävä koostuu keskustietokoneesta, virtalähteestä ja yhdestä tai useammasta liityntäraajapinnasta. Liityntä tapahtuu yleensä valokuitukaapelilla tai kierretyllä parikaapelilla.

Wire Train Bus on pääväylä, joka yhdistää veturin ja kaikki vaunut yhdeksi kokonaisuudeksi. Sen avulla veturi voi hallita koko junan toimintoja keskitetysti. WTB on suunniteltu tukemaan pitkiä junakokoonpanoja, joissa jokainen vaunu toimii viestien välittäjänä. Tämä mahdollistaa keskeytymättömän tiedonsiirron koko junan läpi, vaikka vaunuja lisätään tai poistetaan junasta. WTB-väylän tiedonsiirtonopeus on 1 Mbit/s, se tukee 32:ta eri solmua, ja sen maksimipituus on 860 metriä. Sen rajapinnassa on yleensä kaksi liitintä, joissa kummassakin on integroidut redundantit linjat. Viime vuosina WTB-väylää on korvattu Ethernet Train Backbone (ETB) -väylällä. ETB on monipuolisempi ja nopeampi väylä, sillä se kykenee nopeuteen 100 Mbit/s ja tukee 63:a solmua. [12.]

Multifunction Vehicle Bus on vaunun sisäinen väylä, joka yhdistää vaunun tai veturin sisäiset laitteet ja järjestelmät. Tämä mahdollistaa tiedonsiirron eri alijärjestelmien, kuten moottorinohjauksen, jarrujen, ovien, valaistuksen ja ilmastoinnin välillä. MVB on suunniteltu erityisesti näiden järjestelmien tiedonsiirtoon

korkean vikasietoisuutensa ja reaaliaikaisen viestinnän takia. MVB tukee useita erilaisia viestityyppejä, kuten prosessi-, sanoma- ja diagnostiikkatietoja. Väylä priorisoi kriittisten viestin käsittelyn, mikä varmistaa, että turvallisuuteen ja toiminnan luotettavuuteen liittyvät viestit käsitellään aina ajallaan.

Junien viestintäjärjestelmät ja erityisesti Train Communication Network on määriteltä IEC 61375 -standardissa. Myös kansainvälisen rautatieliiton (Union Internationale des Chemins de fer) UIC-standardit määrittävät TCN:n rakennetta. IEC 61375 tarjoaa laajemman ja kansainvälisemmän viitekehyksen, kun taas UIC-standardeissa voi olla tarkempia ohjeita ja vaatimuksia, jotka on suunniteltu erityisesti eurooppalaiseen kontekstiin. Esimerkiksi monet vaunujen liityntärajapinnat on standardisoitu UIC:n toimesta, mikä varmistaa yhteensopivuuden eurooppalaisissa rautatiejärjestelmissä.

3.2 Controller Area Network

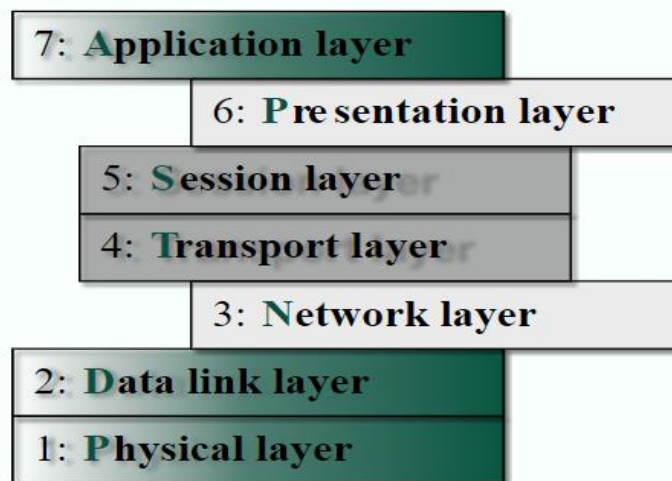
CAN-väylä on yksi yleisimmistä ajoneuvoissa, koneissa ja teollisuudessa käytetyistä väyläjärjestelmistä. Sen monipuolisuus, nopeus, yksinkertaisuus ja hyvä häiriönsietokyky tekee siitä suosittu valinnan erilaisiin sovelluksiin. Väylä mahdollistaa laitteiden eri alijärjestelmien välisen reaaliaikaisen kommunikoinnin tehokkaasti ja luotettavasti.

Väylän kehitys alkoi erityisesti autoteollisuuden tarpeista. Monimutkaiset laitteelta laitteelle tehdyt johtosarjat ja vanhentuneet verkkoprotokollat eivät täyttäneet autoteollisuuden vaatimuksia ja niistä haluttiin eroon. Väylän kehitys alkoi vuonna 1983, ja Robert Bosch GmbH esitteli CAN-väylän Society of Automotive Engineersin (SAE) kongressissa helmikuussa 1986. Alkuperäinen väyläprotokolla toimi lähinnä prototyyppinä ja kehitysversiona. Se virallistettiin vuonna 1991, ja samana vuonna Mercedes Benz esitteli ensimmäinen CAN-väylällä varustetun autonsa W140, joka käytti CAN 2.0A -protokollaa. [13.]

Vaikka CAN-väylä on suunniteltu ensisijaisesti ajoneuvojen käyttöön, sitä käytetään jonkin verran myös junissa. Monissa moderneissa junissa käytetään

teollisuuskäyttöön suunniteltuja yleiskäyttöisiä järjestelmiä, jotka hyödyntävät CAN-väylää tiedonsiirrossa. TCN:n ansiosta tämä ei ole ongelma, sillä se mahdollistaa CAN-väylän kommunikoinnin myös muiden väyläjärjestelmien kanssa.

Standardin mukaisesti CAN-väylän tietoliikennejärjestelmää kuvataan 7-kerroksisen OSI-mallin (Open Systems Interconnection) mukaan (kuva 3). Tässä järjestelmässä jokainen kerros käyttää yhtä alemman kerroksen palveluita ja tarjoaa palveluita yhtä kerrosta ylemmäs. Alkuperäisessä CAN-standardissa määritellään vain fyysinen kerros ja siirtoyhteyskerros. Sovelluskerrosta määritetään vain väylään pohjautuvissa sovelluslaajennuksissa. Muita kerroksia ei yleensä käytetä, sillä väylän sovelluksissa näille toiminnoille ei ole yleensä tarvetta.



Kuva 3. CAN-standardin OSI-malli [14].

CAN-väylän OSI-mallin kerrokset ovat seuraavat:

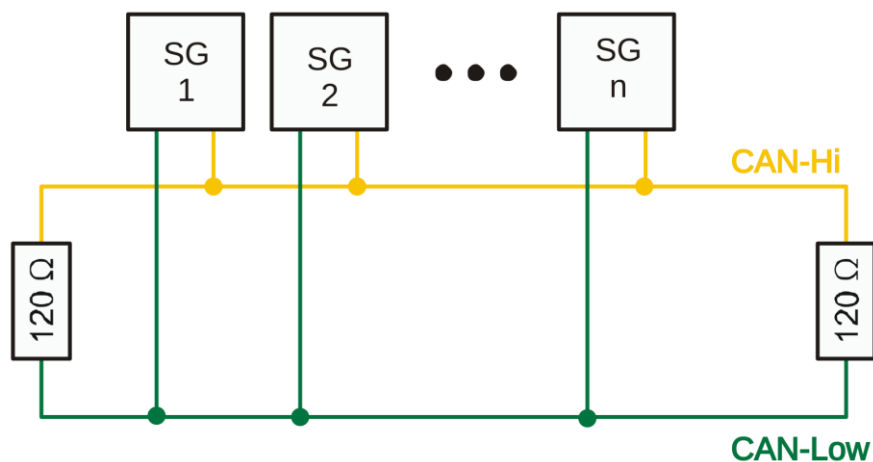
- Fyysinen kerros (Physical layer) määrittää väylän signaalien fyysistä siirtoa, kuten kaapeloinnin liittimet, jännitetasot ja bittiajoitukset.
- Siirtoyhteyskerros (Data link layer) vastaa CAN-viestien muodostamisesta, lähettämisestä, vastaanottamisesta ja virheenkorjauksesta. Se määrittelee, miten data pakataan viesteiksi, miten viestien lähettäminen

ja vastaanottaminen synkronoidaan ja miten väylän käyttöoikeus hallitaan esimerkiksi prioriteettien avulla.

- Sovelluskerros (Application layer) tarjoaa rajapinnan sovelluksille, jotka käyttävät CAN-väylää tiedonsiirtoon. Se määrittelee, miten laitteet kommunikoivat toistensa kanssa sovellustasolla, esimerkiksi kuinka anturit ja toimilaitteet vaihtavat tietoja.

3.2.1 Fyysinen kerros

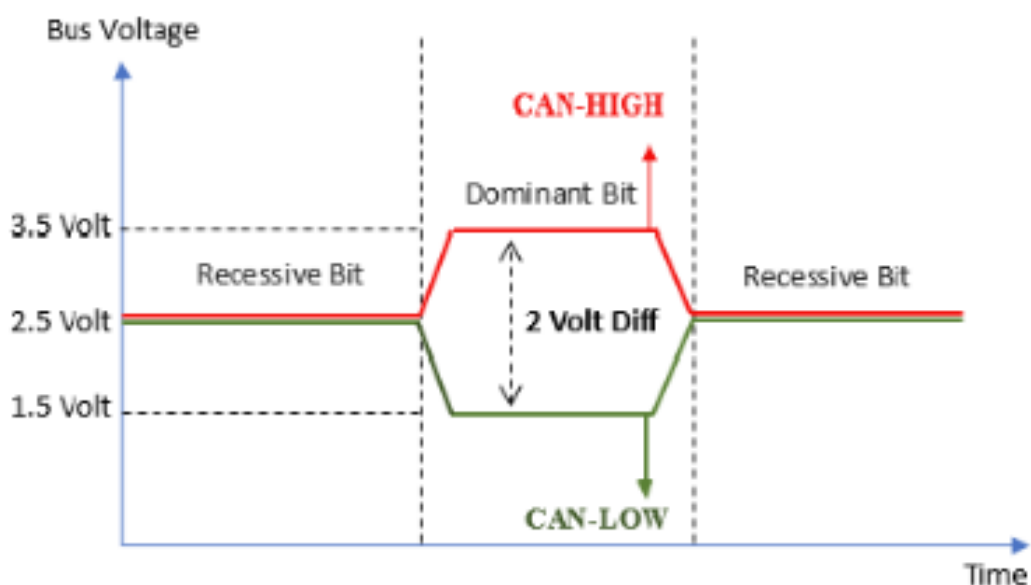
CAN-väylän fyysinen kerros määrittelee käytännössä väylän topologian, kaapeloinnin ja liittimet, signaalit ja jännitetasot sekä muita fyysisiä ominaisuuksia. CAN-väylän topologiassa (kuva 4) kaikki laitteet eli solmut ovat yhdistettynä toisiinsa ja muodostavat verkon, jossa viestit kulkevat kaikkien solmujen kautta. [15, s. 131.]



Kuva 4: CAN-väylän topologia.

Standardin mukaisesti kaapelina toimii kaksinapainen parikaapeli, joka on kierretty 40 kertaa metriä kohden. Sen impedanssiksi on standardissa määritetty 120 ohmia ja väylän päätevastuksina toimii 120 ohmin vastukset. Näillä toimilla vähennetään signaalihäiriöitä ja parannetaan tiedonsiirron laatua. Johtimista käytetään nimiä CAN-High ja CAN-Low. Väylän signaalit perustuvat näiden

johtimien väliseen jännite-eroon. Lepotilassa eli resessiivisessä tilassa molempien johtimien jännite on noin 2,5 voltia; tämä jännite-ero tulkitaan järjestelmässä binääriarvona yksi. Kun CAN-verkossa lähetetään arvo nolla eli dominantti signaali, nousee CAN_H-johdon jännite 3,5 volttiin ja CAN_L-johdon jännite laskee 1,5 volttiin (kuva 5). Tällä menetelmällä väylän viestit rakentuvat. [15, s. 132–133.]



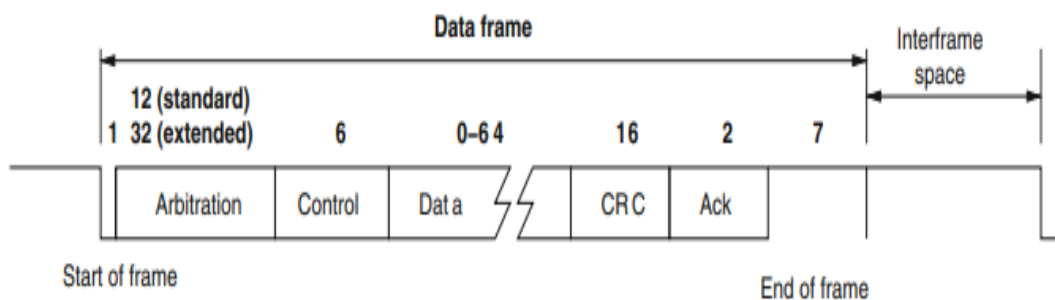
Kuva 5. CAN-viestin signaali [19, s. 2].

3.2.2 Siirtoyhteyserros

Siirtoyhteyserros määrittelee nimensä mukaisesti CAN-väylän tiedonsiirtomenetelmät. Niihin kuuluvat kehysrakenteet, virheenkorjausmenetelmät, viestien priorisointi, väylän ja viestien hallinta sekä verkkoyhteyksien hallinta. CAN-väylän ominaispiirre on viestien lähetystapa, jossa väylä koostuu verkosta, jossa eri ohjainlaitteet eli solmut ovat yhteydessä toisiinsa. Kun yksi solmu lähettää signaalin, se välittyy kaikille verkossa oleville solmuille, jolloin solmu päättää itse reagoiko se viestiin millään tavalla. [15, s. 23.]

CAN-väylä viestii neljällä erityyppisellä kehyksellä:

- datakehykset (Data Frame), jotka sisältävät solmun lähettämän datan
- kyselykehykset (Remote Frame), joita käytetään tietopyyntöön solmulta
- virhekehykset (Error Frame), jotka lähetetään aina, kun verkon solmu havaitsee virheen
- ylikuormituskehykset (Overload Frame), joita käytetään ylikuormituneessa väylässä pyytämään lisää aikaa data- tai kyselykehysten lähettämiseksi.



Kuva 6: Datakehys CAN-väylässä [16, s. 14].

Data- ja kyselykehysten rakenne on pitkälti samanlainen, mutta kyselykehys ei sisällä datakenttää. Datakehystä (kuva 6) käytetään tiedon lähettämiseen ja kyselykehystä tietopyyntöön. Ne koostuvat seitsemästä osasta, joita seuraa taukokenttä:

1. Start of Frame (SOF) eli viestin alku merkitsee viestin alkamisen. Se on aina yksi dominantti bitti (0).
2. Arbitration Field eli tasauskenttä sisältää viestin tunnisteen (ID) ja Remote Transmission Request (RTR) -bitin. RTR-bitti on dominantti (0) datakehyksissä ja resessiivinen (1) kyselykehyksissä. Tunnisteen pituus

riippuu väylän protokollasta. Standardiversiossa se on 11 bittiä ja laajennetussa 29 bittiä. Tunnistekenttä sisältää mm. viestin prioriteetin arvon. [16, s. 15.]

- Standardiversiossa (CAN 2.0A) tasauskentän pituus on 12 (**11+1**) bittiä ja laajennetussa versiossa (CAN2.0B) 32 (**11+1+1+18+1**) bittiä.
 - Laajennetussa versiossa käytetään RTR-bitin lisäksi Substitute Remote Request (SRR) -bittiä, joka toimii aina resessiivisenä (1). Tämä bitti toimii eräänlaisena merkkipaikkana, joka auttaa varmistamaan, että väylän viestit käsitellään oikeassa järjestyksessä, kun sekä tavallisia että laajennettuja viestejä lähetetään samalla väylällä. SRR-bitti sijoittuu samaan kohtaan kuin standardi tunniste- kentän RTR-bitti, jonka jälkeen tulee IDE-bitti. Laajennetussa kehyksessä tunniste- kenttä jatkuu tämän jälkeen ja päättyy RTR- bittiin. Tämä varmistaa, että laajennetut viestit eivät yliaja standar- dimuotoisten viestien tärkeysjärjestystä. Laajennetun kehyksen Identifier Extension (IDE) -bitti on aina resessiivinen. IDE-bitti osoittaa, että kyseessä on laajennettu kehys. Standardikehyk- sessä IDE-bitti kuuluu ohjauskenttään ja toimii vastaavana merk- kinä siitä, onko kehys laajennettu vai ei.
3. Control Field eli ohjauskenttä määrittää datakentän pituuden ja muut oh- jausparametrit. Se sisältää 4 bitin pituisen Data Length Code (DLC) -ken- tän, joka kertoo datakentän pituuden. Ohjauskenttä sisältää myös vara- bittejä tulevaisuuden laajennuksia varten. [16, s. 15.]
 4. Data Field eli datakenttä sisältää itse viestin datan CAN 2.0 -protokol- lassa sen pituus on 0–8 bittiä ja CAN FD -protokollassa 0–64 bittiä.
 5. CRC Field (Cyclic Redundancy Check) eli CRC-kenttä tarkistaa viestin eheyden ja virheettömyyden. Kenttä sisältää CRC-koodin, joka lasketaan datakentän sisällöstä. [16, s. 15.]

6. ACK Field (Acknowledgement) eli vahvistuskenttä ilmoittaa, että viesti on vastaanotettu. Kenttä sisältää ACK-bitin, jonka vastaanottaja asettaa viestin vastaanottamisen merkiksi. [16, s. 15.]
 7. End of Frame (EOF) eli viestin loppu koostuu seitsemästä resessiivisestä bitistä (1111111), jotka varmistavat, että viestin vastaanottaminen on päättynyt.
- Intermission Field eli taukokenttä tarjoaa lyhyen tauon viestien välillä, jolloin verkko voi synkronoitua ja valmistautua seuraavaan viestiin. Kentän pituus on 3 bittiä. Se ei ole osa CAN-kehysten virallista rakennetta, mutta sitä käytetään yleisesti verkon synkronisoinnin helpottamiseksi. [16. s. 16.]

CAN-väylän virhekehukset koostuvat kahdesta pääelementistä: virheilmoituksesta (Error Flags) ja virhe-erottimista (Error Delimiters). Virheilmoitus on yksinkertainen signaali, joka ilmoittaa tapahtuneesta virheestä kuudella peräkkäisellä dominantilla tai resessiivisellä bitillä riippuen siitä, onko virheilmoitus aktiivinen (dominantti) vai passiivinen (resessiivinen). Virhe-erotin ilmaisee virhetilan päätymisestä yhdellä resessiivisellä bitillä. [16, s. 17.]

Ylikuormituskehysiä käytetään lisäämään viivettä ennen data- tai kyselykehysten lähetystä silloin, kun vastaanotin on ylikuormittunut tai ei kykene käsittelemään saapuvaa tietoa. Ylikuormituskehys koostuu kuudesta dominantista bitistä. Sen lähetys alkaa jo taukokentän kahden ensimmäisen bitin aikana, jotta signaali ylikuormitustilasta havaitaan kaikissa muissakin solmuissa. Muut solmut osallistuvat ylikuormituskehysten lähettämiseen enintään 12 dominantin bitin ajan. [16, s. 17.]

3.2.3 Sovelluskerros

CAN-väylän sovelluskerros on korkeimmalla tasolla CAN-väyläarkkitehtuurissa, ja se tarjoaa rajapinnan, jonka kautta ohjelmat ja laitteistot voivat kommunikoida toistensa kanssa CAN-väylän kautta. Vaikka CAN-standardi (ISO 11898) itsessään ei määrittele sovelluskerrosta, se on hyvin olennainen osa CAN-väylää.

Sovelluskerroksen keskeiset toiminnot liittyvät viestien tulkintaan ja koodaukseen, protokollien hallintaan, viestien priorisointiin sekä virheenkäsittelyyn ja diagnostiikkaan. Se mahdollistaa väylän mukauttamisen ja laajentamisen erilaisiin sovelluksiin ja tarpeisiin. [15, s. 23.]

3.3 Väyläprotokollat ja standardit

CAN-väylään on kehitetty sen syntymisestä asti erilaisia väyläprotokollia ja standardeja eri käyttötarkoituksia varten. Alla muutamia yleisimpiä väyläprotokollia:

- CANopen. Sovelluskerrosprotokolla. Erityisesti teollisuusautomaatiossa käytetty standardi. Yhteensopiva CAN 2.0A: ja 2.0B:n kanssa.
- J1939. Korkeamman tason sovellusprotokolla, jossa määritetään jonkin verran myös fyysistä-, verkko-, kuljetus- ja siirtoyhteyskerrosta. Erityisesti raskaan ajoneuvokaluston käytössä. Käyttää laajennettua CAN 2.0B standardia.
- ISO-TP. Kuljetuskerrosprotokolla. Käytetään yleisimmin ajoneuvon diagnostiikkaväylässä (On Board Diagnostics). Mahdollistaa suurten viestien jakamisen pienempiin osiin. Yhteensopiva CAN 2.0A ja 2.0B kanssa.
- NMEA 2000. Sovelluskerrosprotokolla, joka käyttää samankaltaista PGN-järjestelmää kuin J1939. Meriteollisuusstandardi. Käyttää laajennettua CAN 2.0B standardia.

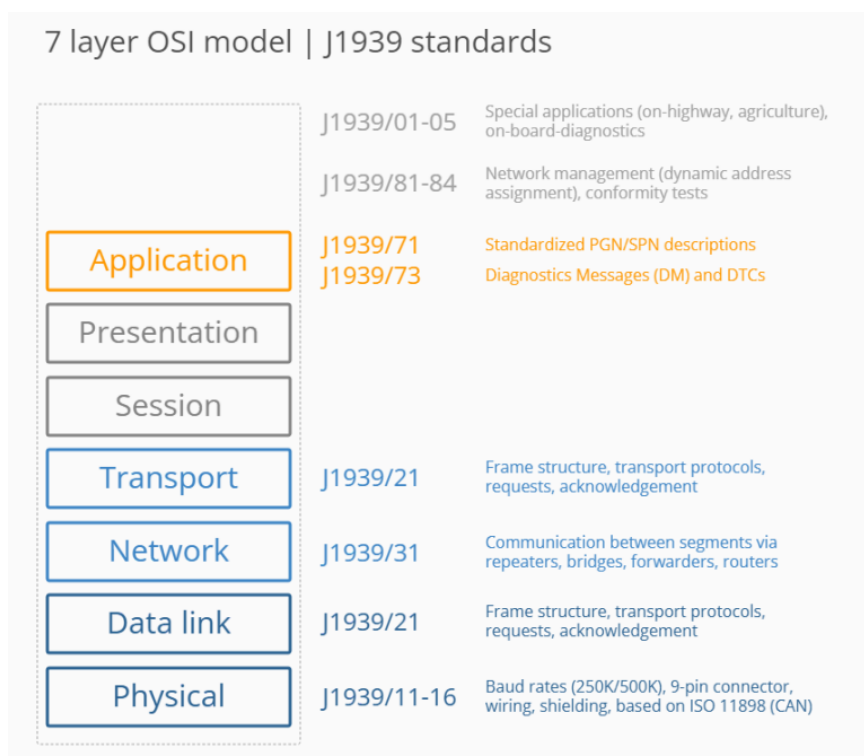
- ARINC 825. Ilmailuteollisuuden sovelluskerrosprotokolla. Käyttää laajennettua CAN 2.0B standardia.
- MilCAN. Sotilaskäyttöön kehitetty sovellusprotokolla. Käyttää laajennettua CAN 2.0B standardia.

Vaikka protokollat eroavat toisistaan, ne yleensä rakentuvat saman fyysisen ja siirtoyhteyskerroksen päälle. Joissain tapauksissa myös fyysisen kerroksen liittimiä määritellään eri protokollissa eri tavoin. Nykypäivänä CAN 2.0B on suosituin käytetty standardi laajennetun tunnistekenttensä ansiosta.

SAE J1939

Society of Automotive Engineers (SAE) julkaisi ensimmäisen version raskaiden ajoneuvon ja teollisuuden viestintään tarkoitettua J1939-protokollasta vuonna 1994. Protokollan ensimmäinen versio ei ollut CAN-standardin mukainen ja se liitettiin siihen vasta 2000-luvun alussa. Tällöin protokolla mukautui käyttämään CAN 2.0B -standardia, joka tarjosi laajennettuja 29-bittisiä tunnisteita parantaen väylän viestintämahdollisuuksia. [17.] Se on suosittu erityisesti dieselkäyttöisissä laitteistoissa niin ajoneuvo- kuin teollisuuskäytössä, sillä sen avulla esimerkiksi päästöjen kontrollointi nykypäivän vaatimusten mukaisesti on helppoa.

J1939 määrittelee OSI-mallin mukaisesti viisi kerrosta (kuva 7). Fyysinen kerros ja siirtoyhteyskerros eivät eroa merkittävästi CAN 2.0B -standardista. Suurimmat eroavaisuudet perinteiseen CAN 2.0B -standardiin verrattuna löytyvätkin verkko-, kuljetus- ja sovelluskerroksista, jotka puuttuvat CAN-standardista. J1939 mahdollistaa esimerkiksi viestien osoittamisen tietyille laitteelle.



Kuva 7: J1939-standardin OSI-malli [17].

J1939-protokollan kerrokset tarjoavat laajennettuja ominaisuuksia viestien hallintaan ja siirtoon:

- Verkkokerros (Network layer) käyttää Parameter Group Number (PGN) -järjestelmää viestien hallintaan. Tämä järjestelmä laajentaa CAN 2.0B -standardin tarjoamia ominaisuuksia kehittyneemmillä mekanismeilla viestien hallintaan, reitittämiseen ja priorisointiin.
- Kuljetuskerros (Transport layer) tarjoaa yhteydenhallintaa, virheenkorjausta ja uudelleenlähetysmekanismeja varmistuen viestien luotettavan ja virheettömän siirron. Kerroksen erityispiirteenä on suurten viestien lähettäminen jakamalla ne pienempiin paketteihin, jotka kootaan uudelleen vastaanottopäässä.
- Sovelluskerros määrittää viestien käsittelyn, esityksen ja hallinnan sovellustasolla. Kerros käyttää PGN-järjestelmää viestien tunnistamiseen ja

luokitteluun. Tämän lisäksi kerros käyttää Suspected Parameter Numbers (SPN) järjestelmää yksittäisten tietoparametrien tunnistamiseen. [18.]

J1939-protokollan viestit pohjautuvat CAN 2.0B:n 29-bittiseen tunnistekenttään, joka sisältää viestin prioriteetin sekä Parameter Group -numeron. Tunnisteen rakenteessa on neljä pääkomponenttia:

- Priority (3 bittiä). Määrittää viestin prioriteetin. Mitä pienempi arvo, sen tärkeämpi viesti. Pituus 3 bittiä.
- Reserved (1 bitti). Varabitti tulevaisuuden laajennuksia varten. Arvoltaan 0.
- PGN (18 bittiä). Määrittää viestin sisällön ja tarkoituksen. Koostuu kolmesta osasta:
 - Data Page (1 bitti): laajentaa mahdollisia PGN-ryhmiä.
 - PDU Format (8 bittiä): Määrittää, lähetetäänkö viesti osoitteellisenä (arvo < 240) vai toistetaan se kaikille laitteille (arvo 240–255).
 - PDU Specific (8 bittiä): Jos viesti toistetaan kaikille laitteille (PF arvo 240–255), tämä kenttä sisältää ryhmälaajennuksen.
- Source address (8 bittiä). Sisältää lähettävän laitteen yksilöllisen osoitteen CAN-verkossa.

4 Datalukijan käyttö väyläviestinnässä

Datalukija on elektroninen laite, jolla pystytään seuraamaan, tallentamaan ja analysoimaan dataa. Väyläviestinnässä ne ovat suhteellisen yleisiä erityisesti diagnostiikkaan liittyvässä vikakoodien lukemisessa. Datalukijoita käytetään myös laitteiden seurantaan, jotta mahdolliset viat löytyvät nopeasti ja huolto-tarve selviää ajoissa. Niiden käyttö on yleistynyt viime vuosina myös kuluttajata-solla erityisesti ajoneuvodiagnostiikassa.

CAN-väylän datalukija koostuu yksinkertaistettuna mikroprosessorista, väyläoh-jaimesta, lähetinvastaanottimesta ja liitännästä. Yleensä datalukijoita ei ole ter-minoitu, sillä oletuksena luettava väylä on jo oikeaoppisesti terminoitu. CAN-väylän toimintaperiaatteen ja rakenteen ansiosta väylän lukeminen on suhteelli-sen helppoa. Lukemistilanteessa laite liitetään CAN-väylään, jolloin se pystyy lukemaan ja tallentamaan kaikkea väylässä liikkuvaa tietoa.

4.1 Datalukijan esittely

Projektia varten datalukijoiden valintaan ei käytetty merkittäviä resursseja. Saa-tavilla oli muutamia erilaisia vaihtoehtoja, mutta valinta kohdistui käyttäjäystä-vällisimpiin ja helposti kannettaviin malleihin. Mitään erityisvaatimuksia väylän lukemiseen ei ollut, sillä J1939-protokollan lukemisen erityisominaisuudet löytyi-vät kaikista saatavilla olevista malleista.

Valinta kohdistui PEAK PCAN-USB-datalukijaan (kuva 8). Lukijassa on riittävät perustoiminnot, ja ne voidaan liittää tietokoneeseen USB-liitännällä sekä väy-lään D-sub-liittimellä. Lisäksi lukijoita oli käytettävissä kaksi kappaletta, joka mahdollistaisi kahden väylän lukemisen samanaikaisesti.



Kuva 8. PEAK PCAN-USB datalukija.

Lukemisohjelmistoksi valittiin Busmaster, koska se on suhteellisen monipuolinen ja perustuu avoimeen lähdekoodiin. Monet datalukuohjelmistot ovat ilmaisia, mutta niissä on usein rajoitteita analysoinnin ja datan käsittelyn osalta. Datat analysointiin käytettiin SavvyCAN-nimistä ohjelmistoa, koska se toimii saumattomasti Busmasterin lokitiedostojen kanssa ja on suunniteltu erityisesti käänteiseen suunnitteluun (reverse engineering). Näiden lisäksi myös Microsoft Excelin datan käsittelyominaisuudet osoittautuivat tarpeellisiksi.

4.2 Väyläviestien lukeminen ja analysointi

CAN-väyläviestejä lukiessa on tärkeää ymmärtää, mitä protokollaa väylä käyttää, toisin sanoen mitä "kieltä" se puhuu. Datalukijalla viestejä lukiessa väylässä liikkuvat viestit on jo käännetty binäärimuodosta desimaali- tai heksadesimaalimuotoon. Raakaviestit on yleensä jaoteltu aikaleimaan, vastaanottotietoon (RTR), tunnisteeseen (ID), prioriteettiin, datan pituuteen (DLC) ja itse viestidataan. Lukuohjelmistosta ja väylän protokollasta riippuen näissä voi olla erilaisia jaotteluominaisuuksia.

Busmaster-ohjelmistolla tallennettu viesti väylästä sisältää aikaleiman, viestin suunnan (receive/transmit), kanavan, tunnisteon, RTR-bitin, datan pituuden, ja itse datan (esimerkkikoodi 1).

```
10:10:01:0031 Rx 1 0xCF00401 x 8 FF FF 82 DF 1A FF FF FF
```

Esimerkkikoodi 1. CAN-väylästä tallennettu viesti.

J1939 viestin analysointi onnistuu CAN ID:n ja datan avulla. Tyypillisesti lukuohjelmisto näyttää viestit heksadesimaalimuodossa suurien viestimäärien takia. Alla olevaan taulukkoon 1 on sijoitettu viestin tunniste ja data myös binäärimuodossa.

Kenttä	Heksadesimaali (0x)	Binääri	Kuvaus
ID	CF00401	00001100 11110000 00000100 00000001	Viestin tunniste
Data	FF FF FF B0 15 FF FF FF	11111111 11111111 11111111 10110000 00010101 11111111 11111111 11111111	Viestin data

Taulukko 1. Viestin ID ja datakentät.

Heksadesimaaliarvon eteen lisätään usein 0x osoittamaan, että kyseessä on heksadesimaaliluku. Lisäksi alle 8-merkkisten CAN-tunnisteon eteen lisätään johtava nolla (0), jotta tunniste olisi aina samanpituisen. Tämä toimenpide ei muuta tunnisteon arvoa vaan yhdenmukaistaa viestin ulkoasun ja helpottaa sen lukemista ja analysointia. Tällöin tunniste tulee muotoon 0x0CF00401.

- Tunnisteon ensimmäiset kaksi tavua 0x0C (binäärimuodossa 00001100), sisältää viestin prioriteetin, varabitin ja data page (DP) -bitin. Ensimmäistä kolmea bittiä ei käytetä, koska tunniste koostuu vain 29 bitistä. Seuraavat kolme bittiä kertovat viestin prioriteetin (011), joka tässä tapauksessa on 3. Tämän jälkeen tulee varabitti (0) ja DP-bitti (0), jota käytetään tarvittaessa laajentamaan viestin PGN-ryhmiä.
- Seuraava tavu F0 (11110000) kertoo viestin PDU-formaatin (PF). Tässä tapauksessa arvo on 240, mikä tarkoittaa, että viesti toistetaan väylän kaikille laitteille.

- Seuraava tavu 04 (00000100) kertoo viestin PDU-specific (PS) arvon. Se sisältää ryhmälajennuksen, kun viesti toistetaan väylän kaikille laitteille.
- Viimeinen tavu kertoo lähettävän laitteen osoitteen 01 (00000001), joka on 1. [18.]

PGN muodostetaan tunnisteiden, PF, PS ja data page (DP) -arvon avulla. PGN muodostuu siis varabitistä (0), DP (0), PF (11110000) ja PS (00000100) kentistä. Heksadesimaalimuotoon käännettynä PGN arvoksi tulee F004, joka tarkoittaa J1939-standardin mukaan elektronista moottorinohjaus eli EEC-yksikköä. [18.]

Tässä viestissä datakentän bitit 1, 2, 3, 6, 7 ja 8 eivät ole käytössä eli niiden arvoksi on asetettu FF. Standardin mukaan bitit 4 ja 5 muodostavat moottorin kierrosnopeuden arvon. Tavujärjestys on nouseva (Little endian), joka tarkoittaa vähiten merkittävien tavujen tallentuvan ensin [18.]. Arvoksi tulee tällöin 0x15B0, joka on desimaalilukuna 5552. Arvon skaalaus on standardin mukaan 0,125 kierrosta minuutissa per bitti. Moottorin kierrosnopeudeksi tulee tällöin

$$5552 \times 0,125 \text{ rpm} = 694 \text{ rpm}$$

Väyläviestien lukemiseen ja analysointiin käytetään lähtökohtaisesti CAN data-base (DBC) -tiedostoja. DBC-tiedosto on yksinkertainen tekstitiedosto, joka mahdollistaa viestien ja signaalien purkamisen ymmärrettävään muotoon. Tiedosto sisältää tunnisteiden nimet ja purkamissäännöt signaaleille. DBC-tiedostot eivät kuitenkaan ole yleisesti saatavilla. Tällä menettelyllä varjellaan liiketalousalustoja, hallitaan tuotteen käyttöä ja mahdollistetaan omien tukipalveluiden kilpailukyky. DBC-tiedostot ovatkin hyvin mallikohtaisia esimerkiksi ajoneuvoteollisuudessa. Vahvasti standardoitujen yleisprotokollien DBC-tiedostoja on saatavilla, mutta niissäkin on usein laitekohtaisia viestejä, jotka ovat vain valmistajan tiedossa.

4.3 Väyläviestien lähettäminen

Yleisellä tasolla CAN-verkossa viestien lähettäminen on suhteellisen yksinkertaista, kunhan viestirakenne on standardin mukainen. Väylään lähetettävät viestit toistuvat kaikille laitteille, jolloin tärkeämpää onkin määrittää, miten viesti saadaan hyväksyttyä halutulle laitteelle. Tämä tarkoittaa huomion keskittämistä sovelluserroksen erityispiirteisiin, eli ajoitukseen, CAN-tunnisteen sisältöön ja itse dataan.

Fyysisesti liitännä olemassa oleviin järjestelmiin on järkevintä tehdä jonkin liittimen kautta esimerkiksi adapterin avulla. Väylän liityntäpinnaksi käy yleensä datalukija, sillä useimmat datalukijat pystyvät myös lähettämään viestejä väylään.

J1939-protokollassa jokainen laite tarvitsee osoitteen. Osoitteen osoittaminen laitteelle (Address claim) J1939-protokollassa toimii kahdella eri tavalla: Joko vaatimalla osoitetta, tai pyytämällä tätä väylältä (dynaaminen osoitteenmäärittäminen). Ensimmäisessä tapauksessa laite ilmoittaa osoitteensa väylään, jolloin väylän laitteet vertaavat sitä omaan verkossa olevien laitteiden taulukkoonsa. Jos osoite on jo käytössä korkeamman prioriteetin laitteella, laite lähettää viestin, joka kertoo osoitteen olevan jo käytössä. Toisessa tapauksessa laite pyytää väylältä osoitetta, jolloin väylän laitteet toimittavat omat osoitetaulukonsa. Tämä mahdollistaa myöhemmin käynnistyvien laitteiden saavan ajantasaisen osoitetaulukon, jonka avulla se voi vaatia vapaana olevaa osoitetta itselleen. Dynaamisen osoitteen määrittäminen on valinnainen ominaisuus J1939-protokollassa ja se löytyy lähtökohtaisesti vain laitteista, joiden odotetaan kohtaavan osoitekonflikteja. [18.]

5 Käytännön toteutus ja testaus

Käytännön toteutus koostui kahdesta veturin väylän ja yhdestä irrallaan olevan apuvoimalaitteen lukemisesta. Dataluvun tavoitteena oli selvittää apuvoimalaitteen käynnistyskäsky ja saada mahdollisimman monipuolisesti selville mitä viestejä väylässä liikkuu.

5.1 Toteutusympäristö ja laitteisto

Veturin väylän lukeminen suoritettiin varikon ratapihalla seisovassa veturissa. CAN-väylään liitettiin apuvoimalaitteen sisällä olevasta liittimestä huoltoluukun kautta (kuva 9). Tätä varten johtosarjaan oli rakennettu oma jatkojohto. Jatkojohtoon liittimet oli kopioitu vastaamaan alkuperäistä liittintä, mutta sen toisen pään Harting-liitintä muokattiin haaroittamalla CAN-väylän signaalit omille banaaniliittimilleen. Näihin banaaniliittimiin kiinnitettiin yksinkertainen d-sub-liitin, joka vastasi datalukijan pinnijakoa. Tällöin tietokoneen ja datalukijan kytkeminen järjestelmään onnistui siististi järjestelmää vahingoittamatta.



Kuva 9. Laitteeseen kytketty jatkojohto.

Pelkän apuvoimalaitteen luku suoritettiin huoltohallissa laitteen huollon yhteydessä. Väylään kytkeydyttiin samalla tavalla kuin edellä, mutta väylä käynnistettiin laitteen diagnostiikkatyökalujen avulla. Laittevalmistajan diagnostiikkatyökalulla kytkeydyttiin laitteeseen sen etupaneelissa olevan USB-portin kautta.

Väyläviestien tallentamiseen käytettiin Busmaster-ohjelmistoa. Tällä ohjelmistolla tallennettujen viestien jatkokäsittely oli sujuvaa, sillä lokitiedostot oli helppo tuoda taulukkotiedoiksi Exceliin tai avata ne suoraan Savvycanissa analysointia varten.

5.2 Testausprosessin kuvaus

Lukeminen aloitettiin tekemällä veturi virrattomaksi irti kytkemällä virroittimet. Tämän jälkeen laitteisto kytkettiin paikalleen ja datan lukeminen aloitettiin. Ensimmäiseksi veturiin kytkettiin virrat uudelleen, joka käynnisti apuvoimalaitteen EEC-yksikön ja väylään alkoi tulla viestejä. Seuraavaksi itse apuvoimalaite käynnistettiin noin minuutiksi, jonka jälkeen se sammutettiin, veturi tehtiin virrattomaksi ja datan lukeminen lopetettiin. Prosessi toistettiin muutaman kerran. Toisella lukukerralla prosessi pysyi samankaltaisena, mutta laitteisto kytkettiin veturin toiseen apuvoimalaitteeseen. Toisella lukukerralla testattiin myös hätäseis-toimintojen vaikutusta väyläliikenteeseen.

Pelkän apuvoimalaitteen kohdalla dataa tallennettiin noin minuutin ajan. Tämän datan avulla laitteen sisäiset viestit pystyttiin erottamaan helpommin laitteen ulkopuolisista viesteistä CAN-tunnisteita vertailemalla.

Väylän nopeudeksi arveltiin 250 tai 500 kilobittiä sekunnissa (kbit/s), sillä nämä ovat yleisimpiä väylänopeuksia. Jos väylälukijan nopeuden asettaa vääräksi, viestit eivät näy oikein niitä luettaessa. Ensimmäinen testaus onnistuikin nopeudella 250 kbit/s, sillä väyläviestit näkyivät oikein.

Datalokeja tallennettiin yhteensä 10 kappaletta, joka tarkoitti noin 2 miljoonaa tallennettua väyläviestiä.

5.3 Testitulokset ja niiden analysointi

Tallennettua dataa oli runsaasti, joten lokit tuotiin Exceliin datan käsittelyn helpottamiseksi. Lokitiedostoja vertailtiin toisiinsa ja yritettiin etsiä eroavaisuuksia näiden välillä. Nopeasti kuitenkin huomattiin lokien olevan hyvin samanlaisia, joten analysoinnissa pystyttiin keskittymään pääasiassa vain yhteen veturissa tallennettuun lokitiedostoon ja yhteen irrallaan olevan apuvoimalaitteen lokitiedostoon.

CAN-tunnisteisiin tutustumalla niistä huomasi niiden noudattavan J1939-protokollaa. Käytössä oli myös J1939-protokollan DBC-tiedosto, jonka avulla muutamia tunnisteita DBC-tiedostoon vertaamalla tähän saatiin varmuus. Lokitiedoston ja DBC-tiedoston SavvyCAN-ohjelmistossa avaamalla, viesteistä pystyi näkemään yksittäisiä signaalitietoja ja niiden arvoja. Viesteissä oli kuitenkin useita laitekohtaisia tunnisteita, joita DBC-tiedostomme ei tunnistanut.

Apuvoimalaitteen käynnistyskäskyn löytämiseksi tuntemattomista viesteistä yritettiin löytää muuttuvia signaaleja käynnistykseen ajankohtana. Tällä tavalla datasta löytyikin viesti, joka viittasi apuvoimalaitteen käynnistykseen. Hypoteesin vahvistamiseksi veturissa ja irrallaan olleen apuvoimalaitteen lokitiedostoja käsiteltiin poistamalla veturin lokitiedostosta mittausten väliset identtiset osoitteet. Tällä menetelmällä viestien määrä väylässä laski merkittävästi ja käynnistysajankohtaan viittaavan viestin jäädessä tiedostoon voitiin sen olettaa liittyvän käynnistykseen.

6 Yhteenveto

Väylätutkimuksen tavoitteena oli selvittää apuvoimalaitteen käynnistyskäsky ja analysoida CAN-väylässä liikkuvia viestejä. Käytännön toteutus sisälsi veturin väylässä ja irrallaan olevan apuvoimalaitteen lukemisen datalukijan avulla. Väyläviestit tallennettiin Busmaster-ohjelmistolla ja analysoitiin Excelissä ja Savvy-can-ohjelmistolla.

Viestien analysointi onnistui, vaikka muutamien laitekohtaisten viestien sisältö jäi vielä hämärän peittoon. CAN-väylän viestintäprotokolla pystyttiin määrittämään, ja mahdollinen käynnistyskäsky löytyi. Lisäksi viesteistä löytyi useita apuvoimalaitteen tilatietoja sisältäviä viestejä sekä laiteyhteyden varmistava tunnisteviesti. Datan lukemista varten valmistettu laitteisto todettiin toimivaksi ja valittu ohjelmisto toimi tarkoituksenmukaisesti.

Vaikka jotkin viestit jäivät vielä tuntemattomiksi, niiden uskotaan liittyvän laitteen omaan vikadiagnostiikkaan. Apuvoimalaitteen käynnistämiseksi tulisi vielä tutkia laitteelle tulevien analogisten signaalien ajoituksia ja jännitteitä. Tällä tavoin väyläviestejä voitaisiin verrata analogisten signaalien ajoituksiin, mikä auttaisi käynnistysprosessin tarkemmassa selvittämisessä. Lisäksi polttoainepumpun ja turvajärjestelmien toimintaan on syytä perehtyä tarkemmin ennen laitteen käynnistämistä.

Valitettavasti aikarajoitukset estivät käynnistämisen tässä työssä, mutta väyläjärjestelmän toiminta saatiin kuitenkin selvitettyä. Tavoite väyläsignaalien selvittämiseksi toteutui työn tilaajan määrittelemien vaatimusten mukaisesti, ja mahdollinen käynnistyskomento löytyi. Vaikka apuvoimalaitteen käynnistäminen ei vielä toteutunut, jatkotoimenpiteet on nyt selvitetty. Tämän työn myötä ymmärrys apuvoimalaitteen väyläjärjestelmistä täydentyi ja tilaajalle pystyttiin selventämään vaatimuksia laitteen käynnistämiseksi.

Lähteet

- 1 Sheilah, Frey. 2012. Railway Electrification Systems & Engineering. Delhi: White Word Publications.
- 2 Steimel, Andreas. 2008. Electric Traction – Motive Power and Energy Supply. München: Oldenbourg Industieverlag GmbH.
- 3 Ratojen sähköistyksellä pitkät perinteet. 2011. Rautatietekniikka 2011–1 s.122–23.
- 4 Eonsuu, Tapio; Honkanen, Pekka & Pöyhö, Eljas. 1995. Suomen veturit. osa 2: Moottorikalusto. Helsinki: Elokuvan Maailma.
- 5 Vectron-veturi. 2024. Verkkoaineisto. VR Group. <<https://www.vrgroup.fi/fi/vrgroup/yrityksemme/liiketoiminta/vr-transpoint/vectron-veturi/>>. Luettu 13.8.2024.
- 6 Pohjosiin olosuhteisiin sovitettut Vectron-veturit saivat tyyppihyväksynnän Suomessa. 2017. Lehdistötiedote. Siemens AG. <<https://press.siemens.com/fi/fi/lehdistotiedote/pohjosiin-olosuhteisiin-sovitettut-vectron-veturit-saivat-tyyppihyvaksynnan>>. Luettu 14.11.2024.
- 7 Siemens Vectron. 2024. Verkkoaineisto. Siemens. <<https://www.mobility.siemens.com/global/en/portfolio/rolling-stock/locomotives/vectron.html>>. Luettu 16.8.2024.
- 8 Dieselvirtamoduulin käyttöohje. 2022. VR Group. Yrityksen sisäinen aineisto.
- 9 Ibanez-Guzman, Javier; Leitner, Andrea & Watzenig, Daniel. 2019. Validation and Verification of Automated Systems. Berliini: Springer Cham.
- 10 Kirrmann, Hubert; Zuber, Pierre A. 2001. The IEC/IEEE Train Communication Network. Sveitsi: ABB Corporate Research.
- 11 TCN Communication. 2016. Verkkoaineisto. Amit transportation. <<https://amit-transportation.com/en/reseni/tcn-communication/>>. Luettu 21.8.2024.
- 12 Train Communication Network (TCN). 2022. Verkkoaineisto. EKE-Electronics. <https://www.eke-electronics.com/wp-content/uploads/2022/08/EKE_Solutions-for-smarter-trains_Product-catalogue-8-2022_TCN_pages-14-15.pdf>. Luettu 20.8.2024.

- 13 Mercedes W140: First car with CAN. 2016. Verkkoaineisto. CAN-Newsletter. <https://can-newsletter.org/engineering/applications/160322_25th-anniversary-mercedes-w140-first-car-with-can/>. Luettu 20.8.2024
- 14 CAN: From physical layer to application layer and beyond. 2024. Verkkoaineisto. CAN in Automation. <<https://www.can-cia.org/can-knowledge>>. Haettu 23.8.2024.
- 15 Voss, Wilfried. 2008. A Comprehensible Guide to Controller Area Network. Greenfield, MA, USA: Copperhill Media Corporation.
- 16 Di Natale, Marco; Zeng, Haibo; Giusto, Paolo & Ghosal, Arkadeb. 2012. Understanding and Using the Controller Area Network Communication Protocol. New York: Springer Science+Business Media.
- 17 CAN bus the ultimate guide. 2024. Verkkoaineisto. CSS Electronics. <<https://www.csselectronics.com/pages/can-bus-ultimate-guide>>. Luettu 20.8.2024.
- 18 SAE J1939 Introduction. Verkkoaineisto. Kvaser. <<https://kvaser.com/about-can/higher-layer-protocols/j1939-introduction/>>. Luettu 20.8.2024.
- 19 Avatefipour, Omid; Hafeez, Azeem & Malik, Hafiz. 2018. Linking Received Packet to the Transmitter Through Physical-Fingerprinting of Controller Area Network. Rennes: IEEE International Workshop on Information Forensics and Security (WIFS).