

Samuli Immonen

# Tiedonsiirtojärjestelmien hyödyntäminen tuotekehityksen testauksissa

Konetekniikka

Insinööri

Kevät 2024



**KAMK • University  
of Applied Sciences**

## Tiivistelmä

**Tekijä:** Immonen Samuli

**Työn nimi:** Tiedonsiirtoväylien hyödyntäminen tuotekehityksen testauksessa

**Tutkintonimike:** Insinööri (AMK), konetekniikka

**Asiasanat:** Tiedonsiirto, CAN-väylä, metsäkone, tuotekehitys, Arcnet

Tässä opinnäytetyössä keskityttiin tutkimaan Ponsse Oyj:n valmistamien metsäkoneiden tiedonsiirtojärjestelmiä, kartoitettiin niiden hyödyntämismahdollisuuksia tuotekehityksen näkökulmasta ja tarvittavien mittauksien osalta sekä sitä, miten näitä järjestelmiä voitaisiin hyödyntää tuotekehityksen työkaluna.

Työssä perehdyttiin mahdollisiin laitteistoihin, joilla dataliikennettä voidaan lukea, tallentaa ja konfiguroida ymmärrettävään muotoon. Työ tehtiin Ponsse Oyj:lle tavoitteena tehostaa metsäkoneiden testaamista tiedonsiirtoväyliä hyödyntämällä.

Työ aloitettiin kartoittamalla Ponssen metsäkoneiden tiedonsiirtoratkaisut. Tiedonsiirtoratkaisuja oli kaksi: Arcnet ja CAN-tiedonsiirto. Molempia tekniikoita tutkittiin ja päädyttiin lopputulokseen, että CAN-tiedonsiirto on paras ja kustannustehokkain tapa mitata ja tutkia koneiden toimintaa.

Kartoituksen pohjalta aloitettiin CAN-väylätekniikan opiskelu ja sen toiminnan selvittäminen. Kun riittävä ymmärrys CAN-tiedonsiirrosta oli saavutettu, siirryttiin Ponsselle ennakkoon hankitun CAN-tallennuslaitteen käyttöönottoon ja ohjelmistoihin perehtymiseen. Laitteiden sekä ohjelmistojen riittävän tuntemuksen jälkeen aloitettiin laitteen ja ohjelmistojen testaaminen fyysisessä metsäkoneessa. Näitä testauksia tehtiin useamman viikon aikana ja samalla tuotettiin Ponssen tuotekehitykselle dataa koneiden toiminnoista. CAN-laitteen ja väylien testauksissa kartoitettiin ja varmistettiin Ponsse-koneiden kaikkien CAN-väylien toiminta.

CAN-tietokanta (DBC) nousi myös suureen roolin työn edetessä: DBC tiedostojen käyttäminen, ymmärtäminen ja luominen on CAN-tiedon siirron analysoinnissa keskeisessä roolissa. Työhön kuului oleellisena osana myös CANopen-öljynvirtausturbiinin käyttöönotto. Käyttöönotto tarkoitti käytännössä virtausturbiinin ominaisuuksien muokkaamista turbiinin objektiikirjastossa, jotta kyseinen laite saatiin tuotekehityksen aktiiviseen käyttöön.

Työni aikana konfiguroitiin tuotekehityksen käyttöön lukuisia muitakin antureita, joilla mitataan eri fysiikan suureita. Yhtenä työn merkittävimpänä tavoitteena oli luoda koulutusmateriaali Ponssen tiedonsiirtoon ja sen hyödyntämiseen tuotekehityksessä. Lisäksi aiheesta pidettiin kaksipäiväinen koulutus, joka oli suunnattu testausinsinööreille ja tuotekehityksen asentajille.

Työn tuloksena saatiin kattava selvitys Ponsse-metsäkoneiden tiedonsiirtorakenteista, keskeisistä viesteistä mittauksen näkökulmasta ja siitä, kuinka näitä viestejä voidaan parhaiten hyödyntää mittauksissa. Lisäksi löydettiin uusia mittalaiteratkaisuja ja -antureita, jotka konfiguroitiin sopiviksi tuotekehityksen tarpeisiin. Työn tuloksena toteutettiin myös kattava koulutus Ponssen tiedonsiirrosta ja siihen liittyvien laitteiden käytöstä. Tietoisuus ja ymmärrys lisääntyivät huomattavasti Ponssen tuotekehityksen testausosastolla. CAN-pohjaiset ratkaisut mittauksissa säästää merkittävästi kalusto- ja henkilöresursseja.

Nykyään tiedonsiirrolla on keskeinen rooli modernien työkonien toiminnassa. Se mahdollistaa laajojen tietomäärien tallentamisen ja lukemisen koneiden toiminnasta. Tämä tarjoaa merkittäviä etuja niin vika-diagnostiikassa kuin tuotekehityksessäkin.

## Abstract

**Author:** Immonen Samuli

**Title of the Publication:** Utilizing Communication Bus Systems in Product Development Testing

**Degree Title:** Bachelor of Engineering, Mechanical Engineering

**Keywords:** Data transmission, CAN Bus, forestry machine, product development, ARCNET

Today, data transfer plays a central role in the operation of modern work machinery. It enables the storage and reading of extensive data regarding the machines operations, offering significant advantages in both fault diagnostics and product development. This thesis aim was to investigate the data transfer systems of forest machines manufactured by Ponsse Oyj, and to explore utilization possibilities from a product development perspective and in terms of necessary measurements. Additionally, it explored systems utilized as tools for product development.

Firstly, Ponsse's forest machine data transfer solutions were surveyed, identifying two options: Arcnet and CAN data transfer. Both technologies were examined, leading to the conclusion that CAN data transfer is the most effective and cost-efficient method for measuring and studying machine operations. Based on the survey, were studied CAN bus technology and its operation. Thereafter, the study proceeded with the implementation of a pre-acquired CAN recording device from Ponsse and familiarization with related software. Accordingly, testing was initiated on a physical forest machine. As the tests were conducted, simultaneously data was generated the machines operations for Ponsse's product development. The testing mapped and ensured the functionality of all CAN buses in Ponsse machines.

The CAN database (DBC) is important, for the use, understanding, and creation of DBC files. The analysis of CAN data transfer is significant.

A CANopen oil flow turbine was implemented. Practical implementation involved modifying the turbine's characteristics in the turbine object library to enable its active use in product development. Numerous other sensors were configured for product development use, covering various types of sensors designed to measure different physical quantities.

The thesis created training material on Ponsse's data transfer and its utilization in product development. Additionally, a two-day training session was held on the topic, aimed at testing engineers and product development assemblers.

The result of the thesis was a comprehensive overview of Ponsse forest machine data transfer structures, focusing on key messages from a measurement perspective and utilizing messages in measurements. Additionally, new measurement equipment solutions and sensors were discovered and configured to suit product development needs. The thesis resulted in comprehensive training on Ponsse's data transfer and the use of related equipment. Increased awareness and understanding were notable outcomes within Ponsse's product development testing department, with CAN-based solutions in measurements significantly saving equipment and personnel resources.

## **Alkusanat**

Haluan aluksi kiittää kaikkia niitä, jotka ovat tukeneet minua tämän opinnäytetyön tekemisessä. Erityiskiitos Juho-Pekka Karppiselle, Toni Kohiolle ja Simo Tauriaiselle, jotka ovat olleet korvaamaton apu matkan varrella.

Tämä työ tehdään Ponsen toiminnan kehittämiseksi ja uuden tiedon tuottamiseksi. Toivon, että tämä työ hyödyttää muita opiskelijoita, asiantuntijoita, kollegoita ja sidosryhmiä, jotka ovat kiinnostuneita CAN-väylistä ja -laitteista.

Työtä aloitettaessa minulla oli jo pitkän raskaskalustomekaanikon urani vuoksi peruskäsitys ja osaaminen CAN-väylätekniikasta. Peruskäsitys kattaa CAN-väylän fyysisen kerroksen ja perustoimintaperiaatteen. Syvemmälle tietotekniikkaan ja binäärijärjestelmiin minulle ei ollut käytännössä ennen tämän työn tekemistä juurikaan tietoa.

Toivon, että tämä opinnäytetyö antaa lukijalle arvokasta tietoa ja ajattelemisen aihetta tiedon siirtoon ja datan käsittelyyn liittyen. Olen nauttinut tämän työn tekemisestä ja oppinut paljon matkan varrella. Toivon, että työni voi olla hyödyksi sekä toimia inspiraation lähteenä muille opiskelijoille ja kollegoille.

Kiitos, että luet opinnäytetyötäni, ja toivotan mielenkiintoisia hetkiä sen parissa.

Ystävällisin terveisin,  
Samuli Immonen

## Sisällys

1	Johdanto .....	1
2	Toimeksiantaja.....	2
3	Tiedonsiirtoväylä .....	3
4	Binäärijärjestelmä.....	5
	4.1.1 Bitti.....	5
	4.1.2 Tavu.....	5
	4.2 Datatyypit.....	6
	4.2.1 Unsigned .....	7
	4.2.2 Signed.....	7
5	CAN-väylä.....	8
	5.1 CANin fyysinen kerros .....	9
	5.2 CAN-topologia .....	10
6	CAN-protokollat.....	13
	6.1 SAE J1939 .....	13
	6.1.1 PGN .....	13
	6.1.2 SPN.....	14
	6.2 CANopen.....	15
	6.2.1 CANopen-objektikirjasto.....	17
	6.2.2 SDO (Service data object) .....	17
	6.2.3 PDO (Process data object) .....	18
	6.2.4 NMT .....	18
	6.2.5 Error control.....	19
7	CAN-viestintä.....	21
	7.1 DBC (DataBaceCan) .....	22
	7.1.1 Big Endian .....	26
	7.1.2 Little Endian.....	26
8	Arcnet .....	27

9	Työn tulokset .....	28
10	Päätäntä.....	29
	Lähteet .....	31
	Liitteet	

## Symboliluettelo

ACK	(Acknowledgment) tiedonsiirrossa käytettävä viestin kuittausbitti.
Arcnet	Arcnet oli lähiverkkotekniikka mikrotietokoneille, myöhemmin suosittu myös sulautetuissa järjestelmissä.
Binääri	Kaksikantainen lukujärjestelmä (tietokonekieli).
CAN	(Controller Area Network) CAN-väylä on autoissa ja työkoneissa laitteiden ja antureiden kommunikoimiseen käytetty väylä.
CAN-H	CAN-väylän johdin, jonka jännitetaso nousee dominanttilassa.
CAN-L	CAN-väylän johdin, jonka jännitetaso laskee dominanttilassa.
CANopen	CIA:n kehittämä ja ylläpitämä CAN-väylän korkeamman tason protokolla.
CiA	CAN in Automation on kansainvälinen käyttäjien ja valmistajien voittoa tavoittelematon yhdistys.
COB-ID	CANopen-viestin tunnistekenttä.
CRC	(Cyclic Redundancy Check) tiedonsiirrossa käytettävä virheen tarkastus.
DBC	CAN-datan dekodaus-tiedosto.
EOF	(End of frame) CAN-viestin lopetusbitti.
Hexa	Heksadesimaali 16-kantainen lukujärjestelmä.
ID	Toimilaitteen yksilöintinumero.
Loggaaminen	Tietojen tallennus sellaisenaan, nauhoitetaan dataliikennettä.
Opti	Ponssen ohjausjärjestelmän tietokone.
PDO	(Process data object) toimintatieto-objekti, käytetään ohjaus- ja tilannetietojen lähettämiseen

PGN	(Parameter Group Number) 18-bittinen parametriryhmän tunnistenumero laajennetussa ID-kentässä
SAE	(Society of Automotive Engineers) Autoalan standardisointijärjestö.
SAE J1939	SAE:n kehittämä ja ylläpitämä CAN-väylän korkeamman tason standardi.
SDO	(Service data object) palvelutieto-objekti, mahdollistaa pääsyn kaikkiin objektikirjaston merkintöihin.
SOF	(Start of frame) CAN-viestin aloitusbitti.
Solmu	Tietoliikenneverkkoon kytketty laite.
Stand alone	Automaattinen tiedontallennus, laite tekee tehtävänsä itsenäisesti.
Token	Kiertävä datapaketti.
Topologia	Tiedonsiirtojärjestelmän fyysinen rakenne.
Triggeri	Laukaisu tai käynnistys, jonkin toiminnon aloittava tai lopettava signaali.

## 1 Johdanto

Tiedonsiirto työkoneissa ja ajoneuvoissa on kasvanut huomattavasti 2000-luvun alun jälkeen. Nykyään lähes kaikissa sarjavalmisteisissa koneissa ja ajoneuvoissa käytetään jonkinlaista tiedonsiirtojärjestelmää. Työkoneissa ei kuitenkaan ole vakiintunutta standardia ohjausjärjestelmän tiedonsiirtoa varten, joten kukin koneen valmistaja voi määrittää omat tiedonsiirtoväylänsä ja -menetelmänsä.

Tässä opinnäytetyössä perehdytään suomalaisen metsäkonevalmistaja Ponsse Oyj:n valmistamien metsäkoneiden tiedonsiirtojärjestelmään sekä sen hyödyntämiseen tuotekehityksen näkökulmasta. Ponssen tuotekehityksessä testataan prototyyppikoneita sekä olemassa olevien konemallien muutostekijöitä, joilla ratkaistaan koneissa ilmenneitä ongelmia sekä kehitetään koneista kustannustehokkaampia, tuottavampia ja luotettavampia.

Opinnäytetyössä sivutaan myös Ponssen tuotekehityksen käytössä olevia antureita sekä mittalaitteita, jotka hyödyntävät CAN-tiedonsiirtoa. Lisäksi opinnäytetyö keskittyy yhteen CAN-datan tallennuslaitteeseen, joka tulee tuotekehityksen käyttöön tämän opinnäytetyön myötä, sekä CANopen-anturin konfigurointiin sekä käyttöönottoon tuotekehityksen työkaluna. Lisämausteenä opinnäytetyön myötä valmistuu koulutusmateriaalia CAN-väylien hyödyntämiseen Ponssen prototyyppiosaston insinööreille sekä asentajille.

Ponssen tuotekehitykseen kuuluu protorakennus ja testaus -osasto, joka on erikoistunut prototyyppirakentamiseen, konetestaukseen ja teräsrakenteiden rasitusmittauksiin sekä väsytestesteihin. Ponssen konetestausosasto käyttää paljon erilaisia mittaus- ja analysointilaitteita sekä -ohjelmistoja.

Opinnäytetyön aihe on erittäin laaja ja kattaa useita eri osa-alueita, jotka jo itsessään olisivat riittäviä AMK-tasoisena opinnäytetyön aiheiksi. Opinnäytetyössä oli tarpeen syventyä tiedonsiirtoon kokonaisvaltaisesti, opiskella ohjelmointia, binäärijärjestelmää sekä tutustua useisiin eri standardeihin, protokolliin ja laiteprofiileihin.

## 2 Toimeksiantaja

Ponsse Oyj on suomalainen metsäkonevalmistaja ja yksi maailman johtavista metsäkoneiden valmistajista. Yhtiö perustettiin vuonna 1970 pääkonttorillaan Vieremällä, Pohjois-Savon maakunnassa Suomessa. [1.]

Ponsse on erikoistunut metsäkoneiden ja metsänhoitokoneiden suunnitteluun, valmistukseen ja markkinointiin. Alusta alkaen perheyriyksenä toiminut Ponsse on kasvanut merkittävästi vuosien varrella ja on nyt kansainvälisesti tunnettu toimija metsäkoneiden alalla. [1.]

Tuotevalikoimaan kuuluvat harvesterit puuston kaatamiseen ja karsimiseen sekä ajokoneet, jotka kuljettavat puutavaran korjuupaikalta metsätien varteen. Lisäksi Ponsse tarjoaa erilaisia lisävarusteita ja huoltopalveluita metsäalan ammattilaisille. [1.]

Yhtiö on laajentanut toimintansa globaaliksi ja sillä on tytäryhtiöitä ja jälleenmyyjiä ympäri maailmaa. Ponsse-tuotteita käytetään laajasti eri puolilla Eurooppaa ja Amerikkaa sekä muilla metsäteollisuuden merkittävillä markkina-alueilla. [1.]

Ponsse on tunnettu korkeasta teknologisesta osaamisestaan ja jatkuvasta innovoinnistaan. Yhtiö kehittää uusia teknologioita ja ratkaisuja, jotka tehostavat metsänhoitotöitä ja vähentävät ympäristövaikutuksia. Ponsse on myös sitoutunut kestävään metsätalouteen ja ympäristövastuuseen pyrkien valmistamaan ympäristöystävällisiä metsäkoneita ja edistämään vastuullista metsänhoitoa. [1.]

Lisäksi Ponsse on merkittävä työnantaja Suomessa ja muissa maissa, joissa se toimii. Yhtiöllä on laaja ja osaava henkilöstö, joka tukee sen toimintaa. Ponsse Oyj on esimerkki suomalaisesta menestyksestä metsäkoneiden valmistuksessa ja se on vahvasti mukana metsäalan kehittämisessä niin Suomessa kuin kansainvälisestikin. [1.]

### 3 Tiedonsiirtoväylä

Tiedonsiirtoväylät ovat tietotekninen ratkaisu tietojen siirtämiseen eri yksiköiden välillä. Väylätekniikkaa käytetään laajasti teollisuuden automaatiossa, työkoneissa, ajoneuvoissa ja tietokoneiden keskinäisessä kommunikoinnissa. [2.]

Alun perin osa tiedonsiirtomenetelmistä kehitettiin tietokoneiden ja toimistolaitteiden keskinäiseen kommunikointiin, mutta myöhemmin niitä hyödynnettiin myös sulautettuna verkkoteknologia teollisuuden tarpeissa ja muissa järjestelmissä. Tiedonsiirtoa voidaan käyttää erilaisten laitteiden, kuten antureiden, ohjainten, tietokoneiden ja näyttöjen välillä. [3.]

Väylätekniikka on todella laaja käsite, sillä erilaisia väyliä on olemassa jopa kymmeniä. Tiedonsiirtoväyliä on lukuisia ja niiden käyttökohteet vaihtelevat kohdejärjestelmän ominaisuuksien sekä toimilaitteiden mukaan. Tässä poikkileikkaus yleisimmistä väylistä:

CAN (Controller Area Network): yleisin käyttökohde on ajoneuvojen ja teollisuusautomaation järjestelmissä. Se tarjoaa luotettavan ja nopean tiedonsiirron monipistemoduuliverkossa. [4.]

Ethernet: Alun perin suunniteltu paikallisiin tietokoneverkkoihin (LAN). Ethernet on yleisesti käytössä sekä toimistoissa että teollisuusympäristöissä. Se tukee erilaisia tiedonsiirtonopeuksia. [5.]

Arcnet (Attached Resource Computer NETwork): Alkuperäisesti suunniteltu teollisuusautomaatioon ja kontrollijärjestelmiin. Käyttää rengas- tai tähtitopologiaa sekä token passing-protokollaa. [6.]

Profibus: Käytetään teollisuusautomaatiossa ja prosessiteollisuudessa. Se tarjoaa nopean ja luotettavan tiedonsiirron automaatiojärjestelmien välisten laitteiden välillä. [7.]

Modbus: Yksinkertainen ja avoin viestintäprotokolla, jota käytetään teollisuusautomaatioissa. Modbus tukee master-slave-arkkitehtuuria. [8.]

LIN (Local Interconnect Network): Alhaisen nopeuden, yhden johtimen sekä yksinkertaisen viestintäväylän protokolla, joka on suunniteltu autojen sisäisten elektronisten komponenttien väliin tiedonsiirtoon. [9.]

## Ponssen väylätekniikka

CAN-väylä on alun perin kehitetty ajoneuvokäyttöön, esimerkiksi moottorin ja vaihteiston välisen tietojen siirtämiseen [4]. Nykyään CAN-väylä on yleinen ratkaisu myös teollisuuden tiedonsiirrossa [10.] Ponssen valmistamissa metsäkoneissa käytetään tiedonsiirtoon kahta eri teknologiaa. Ponsse hyödyntää kolmannen osapuolen polttomoottoritekniikkaa, joista suuri osa sisältää moottorinohjauksen lisäksi päästölaitteistot, jotka ovat vaatimuksia suuressa osassa Ponssen markkina-alueita. Näiden järjestelmien tiedonsiirto perustuu CAN-väylään.

Osassa Ponssen valmistamissa kuormatraktoreissa hyödynnetään kolmannen osapuolen valmistamaa voimansiirtotekniikkaa, joka käyttää CAN-väylään perustuvaa tiedonsiirtoa. Lisäksi Ponssen metsäkoneissa on lukuisia antureita ja anturimoduuleja, joiden tiedonsiirto perustuu CAN-väylään.

CAN-tiedonsiirto on yleistynyt huomattavasti anturitekniikassa sekä eri toimilaitteiden ohjauksessa. CAN-väylä onkin erinomainen tapa tietoa järjestelmään tai järjestelmässä. [10.] Ponssen metsäkoneissa CAN-väylän pääasiallinen tehtävä on toimia kolmansien osapuolien laitteiden kommunikointiväylänä Ponssen tiedonsiirtojärjestelmään sekä toimia mittausteknisenä ratkaisuna koneesta haluttavan datan tallentamista ja analysointia varten.

Ponssen valmistamien metsäkoneiden tiedonsiirto perustuu Arcnet-pohjaiseen ratkaisuun. Kappaletavaramenetelmään perustuvien metsäkoneiden tiedonsiirrolta vaaditaan erittäin suurta tiedonsiirtokapasiteettia. Datan määrä koneissa kasvaa jatkuvasti ja tämä haastaa tiedonsiirtoa metsäkoneissa. Näiden erilaisten tekniikoiden takia tiedonsiirto Ponssen metsäkoneissa toteutetaan hybridinä, joka käyttää CAN-väylää sekä Arcnetia.

## 4 Binäärijärjestelmä

Binäärijärjestelmä on tärkeä ja oleellinen osa eri tiedonsiirtoratkaisuja. Yleisesti tiedonsiirto käyttää aina binäärijärjestelmää toimintaansa, olipa kyse sitten langattomasta, langallisesta tai optisesta tiedonsiirtojärjestelmästä. Binäärijärjestelmän ymmärtäminen on tässä työssä keskeisessä osassa, mikäli ei ymmärrä binäärijärjestelmän toimintaa ei voi myöskään kommunikoida binääristen järjestelmien kanssa. Tässä osassa työtä käsitellään myös binääri-, hexa-desimaali- ja desimaalijärjestelmien korrelaatiota sekä kahta eri tietotyyppiä, jotka liittyvät vahvasti binäärijärjestelmään sekä ohjelmointiin. Tietotyypit ovat välttämättömiä tiedon esittämisessä oikein, näin ollen myös luettaessa tietoa on tiedettävä, mikä tietotyyppi on kyseessä. [11.]

### 4.1.1 Bitti

Bitti on pienin mahdollinen osa tietotekniikassa käsiteltävää tietoa, ja bitit voivat olla kahdessa eri tilassa, jotka tyypillisesti ilmaistaan 1 tai 0. Nämä tilat kuitenkin poissulkevat toisensa ja näin bitin arvo voi olla joko 1 tai 0. Bitin tilan voi määrittää moni eri asia riippuen järjestelmästä, missä sitä käytetään. Esimerkiksi valokuitutekniikassa valopulssi määrää bitin tilan; sähköisessä viestinnässä yleensä bitin tilan määrää jännite, mutta on myös järjestelmiä, jotka hyödyntävät sähkövirtaa eri loogisten tilojen ilmaisemiseen. Taulukossa 1 havainnollistetaan eri pituisten binäärijoukkojen suurimpia mahdollisia desimaaliarvoja sekä niiden muodostamia käsitteitä. [11.]

Taulukko 1, bittien eri tilojen suurin mahdollinen määrä.

1 bitti = $2^1 = 2$
2 bittiä = $2^2 = 4$
4 bittiä = $2^4 = 16$ (puoli tavu)
8 bittiä = $2^8 = 256$ (tavu)
16 bittiä = $2^{16} = 65536$ (kaksi tavua)
32 bittiä = $2^{32} = 4\,294\,967\,296$ (neljä tavua)
64 bittiä = $2^{64} = 18\,446\,744\,073\,709\,551\,616$ (kahdeksan tavua)

### 4.1.2 Tavu

Tavu on binäärijärjestelmässä yksi tietoyksikkö, joka sisältää nykyisin kahdeksan bittiä. Tavuja on aikaisemmin ollut käytössä myös 6–16 bitin kokoisina, mutta tästä on pääasiassa luovuttu. Tavu

käsittää nykyään melkein poikkeuksetta kahdeksan bittiä. Tavu on siis tietotekniikassa pienin mahdollinen tallennettava yksikkö. CAN-viestien datakentät esimerkiksi ovat standardisoitu 64 bitin mittaisiksi. Tällöin viesti sisältää kahdeksan tavua ja kussakin tavussa on kahdeksan bittiä. [11.]

Kuvassa 1 käsitellään desimaalisten, binääristen sekä hexa-desimaalisten lukujen korrelaatiota. Esimerkiksi binäärisen tavun 1111 1111, desimaalinen arvo lasketaan seuraavasti.  $2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 = 255$ . Tämän lisäksi tavu voi saada arvon 0000 0000, joka on desimaalisena myös 0. Tällöin todellisia mahdollisia tiloja kahdeksalle bitille on 256.

Decimal	Hex	Binary
1	01	00000001
2	02	00000010
3	03	00000011
4	04	00000100
5	05	00000101
6	06	00000110
7	07	00000111
8	08	00001000
9	09	00001001
10	0A	00001010
11	0B	00001011
12	0C	00001100
13	0D	00001101
14	0E	00001110
15	0F	00001111
16	10	00010000
17	11	00010001

**Binary to Decimal**

$2^{10}$	$2^9$	$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
1024	512	256	128	64	32	16	8	4	2	1

					1	0	1	1	0	0	1
--	--	--	--	--	---	---	---	---	---	---	---

$1011001_2 = 1 \times 64 + 1 \times 16 + 1 \times 8 + 1 \times 1 = 89$

Jokainen heksadesimaali merkki sisältää 4 bittiä, eli yhdessä tavussa on kaksi hexadesimaalia.

1 0 1 1 0 1 1 0 binääriluku muutetaan desimaaliksi:

$$1 * 2^7 + 0 * 2^6 + 1 * 2^5 + 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 = 182 \text{ dec}$$

Heksadesimaali FE muutetaan desimaaliksi:

$$15 * 16^1 + 14 * 16^0 = 254 \text{ dec}$$

Kuva 1, Heksadesimaalit desimaaleiksi.

## 4.2 Datatyypit

Datatyypeillä tarkoitetaan erilaisia binääriesityksiä. Tarkoituksena näillä esitystavoilla on lähinnä negatiivisten lukujen sekä desimaalilukujen esittämisen mahdollistaminen binäärisesti. Nämä datatyypit ovat nimetty etumerkillisiksi (signed) ja etumerkittömiksi (unsigned). Tällä tarkoitetaan, että etumerkitön on aina positiivinen luku ja etumerkillisessä on määritelty, onko luku positiivinen vai negatiivinen. Binääriluvuilla ei sellaisenaan pystytä esittämään kuin vain positiivisia lukuarvoja. [12.] [13.]

#### 4.2.1 Unsigned

Unsigned (merkitön): tarkoittaa, että luku voi olla vain positiivinen, eli se voi olla nolla tai positiivinen, mutta ei negatiivinen. Esimerkiksi unsigned voi saada siis vain positiivisia kokonaislukuja. Unsigned-tyypillä voi olla odottamattomia tuloksia yhteenlaskuissa ja vähennyksissä, erityisesti, jos yritetään vähentää suurempaa arvoa pienemmästä. Unsigned-tietotyyppi luetaan sellaiseenaan. [12.] [13.]

#### 4.2.2 Signed

Signed (merkitty): tarkoittaa, että luku voi olla sekä positiivinen että negatiivinen, mukaan lukien nolla. Kokonaislukujen merkki (positiivinen tai negatiivinen) määritellään binäärimuodossa käyttämällä niin kutsuttua kahden komplementin esitystapaa. [13.] Kahden komplementin esitystavassa negatiiviset kokonaisluvut esitetään ottamalla luvun absoluuttinen arvo, muuntamalla se binäärimuotoon ja sitten kääntämällä kaikki bitit (0 -> 1, 1 -> 0) ja lisäämällä lopuksi 1 tulokseen. Tämä muunnos mahdollistaa sen, että negatiiviset luvut voidaan esittää samalla tavalla kuin positiiviset luvut, mutta merkki määräytyy ensimmäisen bitin perusteella. Esimerkki 8-bittisestä kahden komplementin esitystavasta:

Positiivinen luku 5: 00000101

Negatiivinen luku -5

Laske 5 binäärimuotoon: 00000101

Käännä kaikki bitit: 11111010

Lisää 1: 11111011 = -5

Tässä esimerkissä ensimmäinen bitti (vasemmalta) toimii merkinä. Jos se on 0, luku on positiivinen; jos se on 1, luku on negatiivinen.

Tämän ansiosta negatiivinen luku voi jakaa saman binääriesityksen kuin vastaava positiivinen luku, mutta sen ensimmäinen bitti erottaa sen merkityksen. Tämä on kätevää tietokoneissa, koska se mahdollistaa saman operaation käyttämisen sekä positiivisille että negatiivisille luvuille. [12.] [13.]

## 5 CAN-väylä

Controller Area Network (CAN) on laajalti käytetty verkkoprotokolla ja sarjaliikennejärjestelmä, joka on erityisesti suunniteltu ajoneuvojen sisäiseen verkkoliikenteeseen. Se on vakiintunut standardi ajoneuvoteollisuudessa, ja sen sovellukset ulottuvat autoista ja kuorma-autoista maatalouskoneisiin sekä raskaisiin teollisuuslaitteisiin. [10.]

BOSCH-yhtiön insinöörit kehittivät CAN-väylän 1980-luvun alussa autojen sisäiseen verkkoliikenteeseen ja hallintalaitteiden tarpeisiin. Tämä sarjaliikenneväyläprotokolla on suunniteltu kestämään häiriöitä ja tarjoamaan luotettavan tavan laitteiden väliseen kommunikointiin. [4.]

CAN-väylä mahdollistaa nopeat tiedonsiirtonopeudet, mikä tukee reaaliaikaisen datan liikkumista ajoneuvon eri komponenttien välillä. Sitä voidaan käyttää monien toimintojen hallintaan ajoneuvossa, kuten moottorin säätöön, ilmastoinnin ohjaukseen, turvajärjestelmien valvontaan ja viihdejärjestelmien toimintaan. [14.]

Yksi tekniikan vahvuuksista on sen kyky kestää häiriöitä. Se on suunniteltu ympäristöihin, joissa sähkömagneettinen häiriö ja muut ulkoiset tekijät voivat vaikuttaa tiedonsiirtoon. Tämä tekee siitä erityisen sopivan käytettäväksi liikkuvissa ajoneuvoissa, joissa haasteita voivat aiheuttaa esimerkiksi moottorin sähkömagneettiset kentät. [14.]

Tämä väylä tarjoaa helpon liittämistavan eri laitteiden välillä ja on joustavasti sovellettavissa monimutkaisiin järjestelmiin. CAN-väylä käyttää usein standardoituja protokollia, kuten CAN 2.0A (11 bit id frame) ja CAN 2.0B (29 bit id frame), mikä helpottaa eri laitteiden yhteensopivuutta. [15.]

CAN-väylä on perusajatukseltaan usean isännän (multi-master) sarjaliikenneväylä, joka on suunniteltu mahdollistamaan useiden itsenäisten solmujen samanaikainen tiedonsiirto ja hallinta. Tämä tarkoittaa, että useat laitteet voivat lähettää ja vastaanottaa tietoa väylällä ilman yhden hallitsevan solmun rajoituksia. CAN-väylä tarjoaa monimutkaisissa järjestelmissä tehokkaan keinoon laitteiden välisten tietojen siirtämiseen ja koordinointiin. [16.]

Koska CAN-väylä tarjoaa nopean ja tehokkaan tavan laitteiden väliseen kommunikaatioon, sen käyttö on laajentunut muihin sovelluksiin. CAN-väylää voidaan käyttää myös monimutkaisten

teollisuusautomaatiojärjestelmien hallintaan, kuten robotiikassa ja tehdasautomaatiossa. Sen avulla eri laitteet voivat tehokkaasti viestiä keskenään, mikä on olennaista tehokkaan ja joustavan tuotantoympäristön luomiseksi. [16.]

Metsäkoneissa käytetään CAN-väylää tiedonsiirtojärjestelmissä. Tässä työssä tutkitaan tarkemmin kahden yleisimmin käytetyn SAE-J1939- ja CANopen-protokollan käyttöä metsäkoneiden tiedonsiirrossa ja niiden viestirakenteissa. Viestirakenteet ovat oleellisessa osassa viestien sisällön määrittelemisessä.

### 5.1 CANin fyysinen kerros

CAN-väylän tiedonsiirtoyhteytenä käytetään yleensä paripunottua kuparijohdinta. Punottu pari-kaapeli tarkoittaa kahta erillistä johtoa, jotka on kierretty toistensa ympärille tietyllä taajuudella. [17.]

CAN-väylässä paripunotun johdon suurin etu on sen sähkömagneettisten häiriöiden sietokyky, sillä CAN-signaali koostuu kahdesta signaalitasosta, CAN-high ja CAN-low, jotka ovat sähköisesti vastakkaisia toisiinsa nähden. Tämä tarkoittaa, että kummallekin signaalille on oma johtonsa, mikä mahdollistaa fyysisen sähkömagneettisen häiriönsuojauksen. [14.]

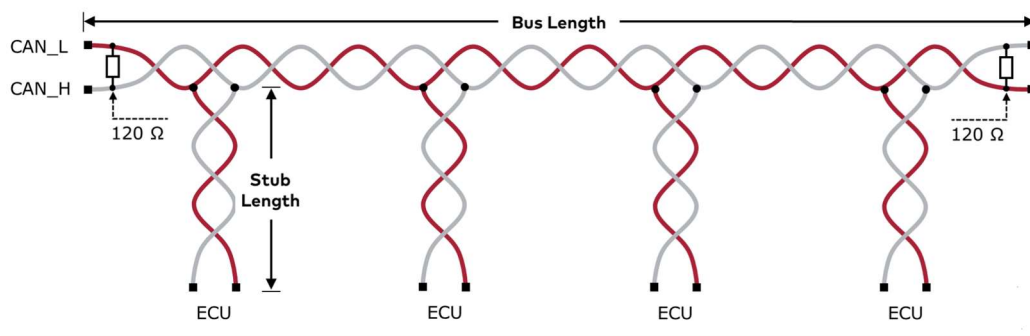
Kun paripunottu CAN-johdin on alttiina sähkömagneettisille häiriöille, ne vaikuttavat molempiin johtimiin samanaikaisesti. Tällöin signaalien sähköinen jännitetaso voi muuttua negatiiviseen potentiaaliin nähden. [14.]

Sähkömagneettisen häiriön sietokyvyn kannalta keskeistä on, että molempien signaalien jännitetaso muuttuu samanaikaisesti, jolloin CAN-high- ja CAN-low-signaalien välinen potentiaaliero pysyy vakiona. [14.]

Tämä varmistaa sen, että signaali voidaan edelleen tulkita oikein viestin vastaanottavassa laitteessa. Yleinen käytäntö CAN-väylässä on käyttää 60 ohmin kuormaa, joka estää sähköpulslien takaisinheijastumisen väylään. CAN-väylän fyysisen kerroksen toteutukseen vaikuttaa oleellisesti väylän tiedonsiirtonopeus. Mitä nopeampi tiedonsiirtonopeus on väylässä, sitä lyhyempi väylän fyysinen kerros voi olla. [14.]

CAN-väylän fyysinen rakenne käsittää runkoväylän sekä väylän eri toimilaitteelle menevät oksat, mikäli toimilaitte ei sijaitse väylän päässä. Väylässä käytettävän oksan pituus riippuu väylän tiedonsiirtonopeudesta. Mitä suurempi tiedonsiirtonopeus, sitä lyhyempi väylän oksan tulee olla. Kuvassa kaksi havainnollistetaan CAN-väylän fyysistä rakennetta päätevastuksineen sekä väylän oksan (Stub Length) merkitystä väylässä [18].

CAN-väylän runkoväylä sekä oksien pituus eli (stub length) esitetty kuvassa 2, maksimipituus rajoittuu aina väylässä käytettävän tiedonsiirtonopeuden mukaan. Mitä suurempi on tiedonsiirtonopeus, sitä lyhyempi väylän sekä oksien tulee olla, että väylä olisi toimiva ja stabiili. Kuvassa 2 havainnollistetaan CAN-väylän topologiaa sekä johtimien paripunontaa, häiriöiden minimoimiseksi. [18.]



Kuva 2, havainnollistava kuva paripunotusta CAN high- ja CAN low -johtimesta. [18]

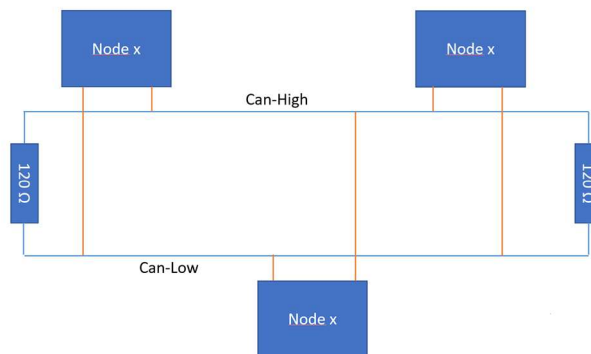
## 5.2 CAN-topologia

CAN-väylä on niin sanottu usean isännän väylä (multi master), joka tarkoittaa, ettei väylässä ole master/slave-ominaisuutta, vaan kaikki väylän kytketyt solmut ovat tasavertaisia keskenään. Tällaisessa ratkaisussa jokainen väylään kytketty solmu voi lähettää väylään viestejä yhtä aikaa muiden solmujen kanssa. [16.]

Myös kaikki solmut saavat kaikki viestit ja voivat näin ollen myös lukea ne. Viestien priorisointi sekä oikeiden viestien lukeminen oikeissa solmuissa toteutetaan itse viestirakenteessa tai lukevassa laitteessa, (riippuen protokollasta). Väylään kytketyt solmut itse valikoivat niille oleellisia

viestejä viestin sisältämän binäärikoodin perusteella. Yhteen CAN-väylään voidaan kytkeä parhaillaan 127 tai 255 solmua, riippuen tiedonsiirtoon käytettävästä protokollasta.

Solmujen määrää voidaan tarvittaessa laajentaa kytkemällä johonkin keskeiseen solmuun toinen väylä, jolloin CANopen-väylässä voi olla master-slave-toiminto. [16.] Yleensä masterina toimii jokin teollisuuslogiikka eli PLC. CANopen-väylässä voidaan laitteiden eri ominaisuuksia muokata laitteiden objekti kirjaston mukaan. Topologialla tarkoitetaan ja kuvataan CAN-väylän fyysisen kerroksen rakennetta. [19.] Kuvassa 3 havainnollistetaan CAN-välien perusrakennetta.



Kuva 3, Kuva perinteisestä väylätologiasta.

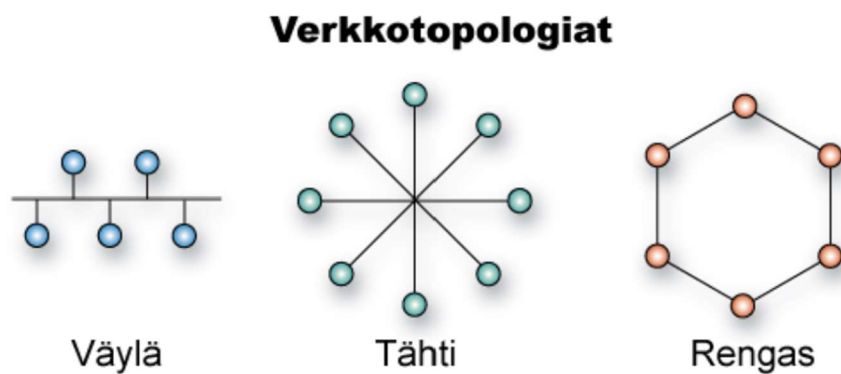
Kaksi rinnan kytkettyä  $120 \Omega$  vastusta muodostaa väylän kokonaisvastukseksi  $60 \Omega$ . Kun vastukset kytketään rinnan kytkentään, noudattavat ne ohmin lakia seuraavan kaavan mukaisesti. Virtapiiirin rinnan kytkettyjen vastusten kokonaisresistanssi lasketaan ottamalla kytkettyjen vastusten käänteislukujen summa ja laskemalla tämän summan käänteisluku. [20.]

$$\frac{1}{R_{total}} = \frac{1}{R_1} + \frac{1}{R_2} \rightarrow \frac{1}{120\Omega} + \frac{1}{120\Omega} = \frac{1}{60\Omega}$$

CAN-väylässä käytettävät vastukset ovat olennainen osa järjestelmää monista syistä. Ne hallitsevat signaalien taajuuskomponentteja ja varmistavat signaalien vakauden väylällä. Lisäksi ne auttavat hallitsemaan virrankulutusta ja toimivat terminointina estäen signaalihäiriöitä ja heijastumia. Näiden ominaisuuksien ansiosta vastukset varmistavat luotettavan ja häiriöttömän tiedonsiirron CAN-väylällä. [21.]

CAN-järjestelmä voi olla topologialtaan myös tähti, rengas, verkko tai näiden topologioiden yhdistelmä. Mikäli CAN-väylää käytetään muissa kuin väylä- ja rengas-topologioissa, vaatii väylä

toimiakseen jonkin toimilaitteen, joka jakaa tietoa eri väylien välillä. On siis huomioitava, että kaikki muut kuin väylätopologia koostuu kuitenkin toimilaitteeseen liitetystä yksittäisistä väylisistä. Ajoneuvotekniikassa toimilaitetta nimitetään yleisesti keskusyhdyskäytäväksi (Central Gateway). Muissa järjestelmissä käytetään termiä hub. Tähtimallisen topologian suurena etuna on, että voidaan yhdistää erilaisia väyliä keskenään, esimerkiksi eri tiedonsiirtonopeuksia tai jopa kokonaan eri tiedonsiirtoratkaisuja. [22.] Tämä tietenkin riippuu täysin toimilaitteen rakenteesta sekä ohjelmistosta. Kuvassa 4 havainnollistetaan erilaisten tiedonsiirtoratkaisujen topologioita.



Kuva 4, Esimerkkejä eri topologioista.

## 6 CAN-protokollat

CAN-protokolla (Controller Area Network) on ylemmän tason protokolla, joka määrittelee, miten tiedonsiirto CAN-verkon kautta tapahtuu. CAN-protokolla tarjoaa perusviestintäkehyksen ja mahdollistaa tiedonsiirron CAN-väylän kautta, mutta se ei määritä, miten tietoja siirretään tai tulkitaan. [16.] Sen sijaan alemmat protokollat, kuten J1939 ja CANopen, rakentuvat CAN-protokollan päälle ja määrittelevät tarkemmin, miten tietoja siirretään ja mitä ne tarkoittavat erityisissä sovelluksissa. [18.] [23.]

### 6.1 SAE J1939

J1939 on yksi CAN-protokollan käytännön sovelluksista ja standardi erityisesti raskaissa ajoneuvoissa, kuten kuorma-autoissa ja maanrakennuskoneissa. Se määrittelee tiettyjä viestityyppejä, viestien rakenteita ja parametreja ajoneuvojen keskinäiseen viestintään. [24.]

J1939-standardi määrittelee PGN (Parameter Group Number) -tunnisteet ja SPN (Suspect Parameter Number) -lukuarvot, jotka auttavat määrittämään, mitä tietoja viesteissä siirretään. J1939-protokolla mahdollistaa eri ajoneuvon komponenttien, kuten moottorinohjauksen, voimansiirronohjauksen ja jarrujärjestelmän välisen tiedonjakamisen. Tämä standardi on kehitetty yhteistyössä SAE Internationalin (Society of Automotive Engineers) kanssa ja tunnetaan virallisesti nimellä "SAE J1939". [24.]

#### 6.1.1 PGN

PGN (Parameter Group Number) on osa J1939-viestirakennetta ja määrittää, miten tieto on organisoitu ja käytetään J1939-verkossa. Vaikka J1939 ja CAN (Controller Area Network) ovat kaksi erillistä protokollaa, J1939 on perustettu CAN-tekniikkaan, ja ne voivat toimia yhdessä. Alla on yleiskatsaus siitä, miten J1939-protokollan PGN toimii. [24.] [25.]

PGN koostuu 18-bittisestä numerosta, joka tunnistaa tietyn tietoryhmän tai parametriryhmän J1939-verkossa. Se voidaan jakaa kahteen osaan, PDU Format (PF): Ensimmäiset 8 bittiä määrittelevät viestin tyyppikoodin ja lähettäjän laitteen tyyppinumeron. PDU Specific (PS): Seuraavat 8 bittiä määrittelevät viestin tarkemman tyypin tai sisällön. [24.]

Kun viesti luodaan J1939-verkossa, PGN sisältyy viestiin sen tunnistamiseksi ja oikean vastaanottajan tunnistamiseksi. [24.]

Lähettävä laite (esim. ajoneuvon ohjausjärjestelmä) pakkaa tarvittavat tiedot PGN:ksi ja lähettää sen J1939-verkkoon. PGN voi sisältää erilaisia tietoja, kuten ajoneuvon nopeuden, lämpötilan, moottorin tilan ja paljon muuta. Vastaanottavat laitteet kuuntelevat J1939-verkkoa ja tarkkailevat PGN:ää. Kun oikea PGN havaitaan, vastaanottava laite pystyy purkamaan sen ja käsittelemään sisältämiään tietoja. [24.] [25.]

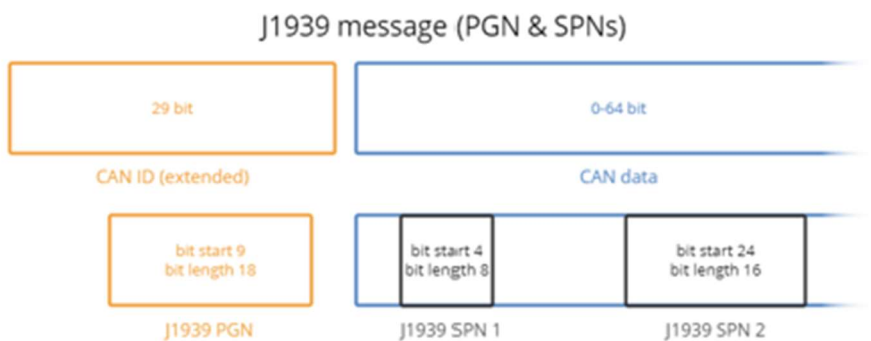
J1939-protokollan PGN-toiminta on erityisen hyödyllinen ajoneuvoissa ja raskaskoneissa, joissa eri järjestelmät ja komponentit tarvitsevat tietoa toisistaan. Se mahdollistaa tietojen tehokkaan jakamisen ja seurannan ajoneuvon eri osien ja laitteiden välillä. [24.] [25.]

Vaikka J1939-protokollan PGN on perustettu CAN-tekniikkaan, se on silti oma protokollansa, ja sen rakenne ja tarkoitus eroavat tavallisesta CAN-väylän viestirakenteesta. J1939 tarjoaa erityisiä suuntaviivoja ja määrittämiä ajoneuvojen ja raskaskoneiden tietoliikenteelle, mikä mahdollistaa tiettyjen standardien ja yhteensopivuuden säilyttämisen kyseisillä aloilla. [24.] [25.]

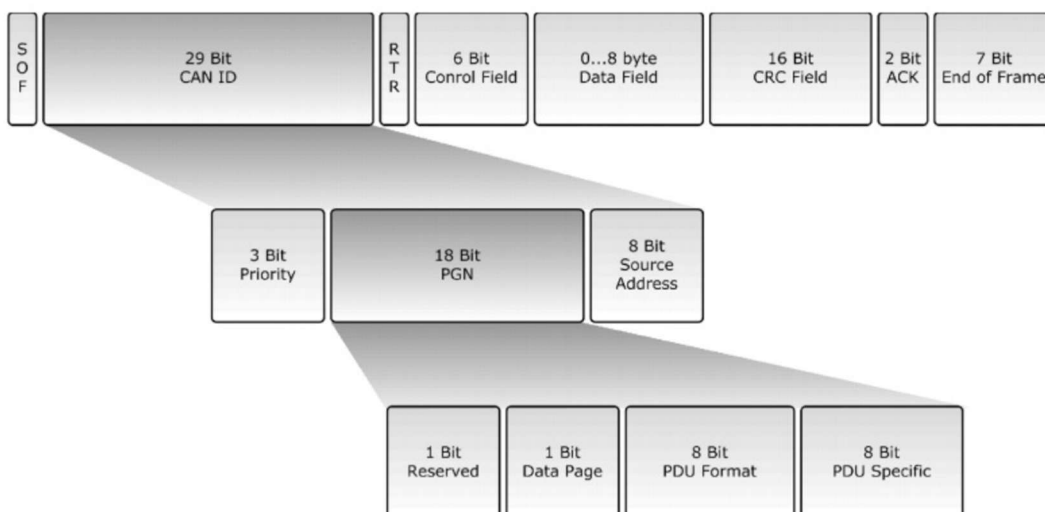
### 6.1.2 SPN

J1939-protokolla käyttää SPN:ää määrittämään tietyn parametrin tai sensorin arvon. SPN on numerokoodi, joka tunnistaa tietyn parametrin. Jokaisella SPN:llä on liittyvä PGN (Parameter Group Number) ja suurempi ryhmä nimeltä PG (Parameter Group). SPN ja PGN auttavat järjestelmää tulkitsemaan ja ymmärtämään CAN-väylän kautta siirtyvää tietoa raskaiden ajoneuvojen ja teollisuuslaitteiden järjestelmissä. [24.] [25.]

Esimerkiksi, jos puhutaan moottorin öljynpaineesta, kyseiselle parametrille voidaan antaa SPN-numero. Tämä numero auttaa järjestelmää ymmärtämään, että kyseessä on moottorin öljynpaineeseen liittyvä tieto. SPN-numerot määritellään J1939-standardeissa, ja ne helpottavat tiedonsiirron yhtenäisyyttä raskaiden ajoneuvojen ja teollisuuslaitteiden väyläpohjaisissa järjestelmissä. [20.] Kuvissa 5 ja 6 havainnollistetaan J1939-Protokollan mukaisen ID-kentän sisältöä, PGN sijaitsee ID-kentässä ja SPN viestin datakentässä. [24.] [25.]



Kuva 5, SPN sijaitsee viestin varsinaisessa datakentässä. [25.]



Kuva 6, J1939-viestirakenne. [24]

## 6.2 CANopen

Controller Area Network Open (CANopen) -protokolla tarjoaa tehokkaan ja luotettavan viestintäympäristön monenlaisten laitteiden välillä. Tämä viestintäteknologia perustuu alun perin teollisuuden kontrolliverkkoihin tarjoten usean isännän viestintäympäristön. [19.]

CANopen-protokolla on monipuolinen ja sitä hyödynnetään laajasti eri teollisuuden aloilla, kuten automaatio, lääketieteelliset laitteet, rautatiejärjestelmät ja ajoneuvot. Protokolla mahdol-

listaa datanopeuksien säädön välillä 10 kbit/s – 1000 kbit/s, mikä antaa joustavuutta erilaisten sovellusten vaatimuksiin. [26.]

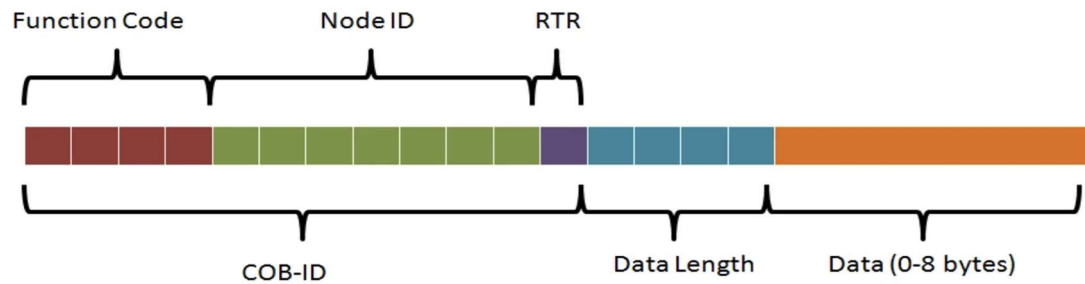
Yksi CANopen-protokollan vahvuuksista on sen tuki erilaisille laiteprofiileille. Nämä profiilit määrittelevät, miten laitteet kommunikoivat keskenään, mikä tekee CANopenista erittäin soveltuvan monenlaisiin käyttötarkoituksiin. Protokollan keskeiset käsitteet ovat objektit ja niihin liittyvät palvelut, jotka mahdollistavat tietojen tehokkaan jakamisen ja hallinnan. [26.]

CANopen-protokolla on suunniteltu modulaarisesti, mikä tekee siitä erittäin joustavan ja helpon sovittaa erilaisiin sovelluksiin. Tämä modulaarisuus helpottaa myös järjestelmän laajentamista ja päivittämistä. CANopen-protokolla nauttii vakiintuneesta asemasta ja laajasta tuesta teollisuudessa. Tämä tekee siitä houkuttelevan vaihtoehdon monenlaisille sovelluksille, missä luotettava ja monipuolinen viestintä on olennaista. [26.]

CANopen perustuu ISO 11898-1 -standardin määrittelemään tietoyhteyskerrokseen, ja sen bititajointus on tarkasti määritelty CiA 301 CANopen -sovelluskerroksessa. CANopen-protokolla muodostaa useita protokollia ja on siis käytännössä protokollaperhe. Sen modulaarinen suunnittelu mahdollistaa sen sovittamisen erilaisiin sovelluksiin ja tarjoaa mahdollisuuden laajentaa järjestelmää eri vaatimusten mukaisesti. [26.]

#### CANopen-viestirakenne

CANopen-protokolla käyttää erilaista viestirakennetta kuin raskaankaluston J1939-protokolla. CANopen viestirakenteessa ID-kentän pituus on 11 bittiä, siinä ei määritellä viestin prioriteettia, vastaanottavan solmun ositetta eikä 18-bittistä PGN-kenttää, toisin kuin J1939-protokollassa. CANopen-viestissä prioriteetin määrää solmun ID. 11-bittisestä COB-ID-kentästä varsinaisen solmun ID:lle on varattu 7 bittiä. Tämä seitsemän bitin ID-numero rajoittaa CANopen-laitteiden määrän samassa väylässä 127 kappaleeseen, seitsemän bitin maksimiarvo on 127. 7 bittiä voi saada siis yhteensä 128 arvoa nolla mukaan luettaessa,  $2^7 = 128$ . Arvoa 0 ei kuitenkaan yleisesti käytetä laitteiden yksilöintiin. Kuvassa 8 havainnollistetaan CANopen-protokollan viestirakennetta. [26.]



Kuva 7, CANopen-viestirakenne. [26.]

### 6.2.1 CANopen-objektikirjasto

CANopen-objektisanakirja on tärkein CANopen-laitteen konfiguroinnissa. Laitteen sisäisenä referenssinä käytetään 16-bittistä indeksiä, joka annetaan 4-numeroisena heksadesimaaliarvona. Kaikki CANopen-laitteen toiminnan määrittävät viestintä- ja sovellusparametrit hallitaan standardoidussa luettelossa, CANopen-objektisanakirjassa. Tässä objektisanakirjassa on kaikki tiedot, jotka on vaihdettava laitteen viestintäosan (CANopen-protokollapino) ja sovellusosan (sovellusohjelmisto) välillä, omistavat yksilölliset 24-bittiset osoitteet. Tämä osoite on jaettu 16-bittiseen indeksiin ja 8-bittiseen alaindeksiin. CANopen-objektisanakirjan kaikki merkinnät ovat luettavissa CANopen-viestintäpalvelujen kautta. [27.]

CANopen-protokolla koostuu useista alemman tason protokollista, nämä protokollat yhdessä muodostavat edellytykset toimivalle CANopen-järjestelmälle. Kaikissa CANopen-toimilaitteissa ei välttämättä ole käytössä kaikkia alemman tason protokollia. Tämä riippuu hyvin paljon laitteen käyttötarkoituksesta. [28.]

### 6.2.2 SDO (Service data object)

SDO-protokolla mahdollistaa CANopen-laitteen objektikirjaston muokkaamisen väylän kautta. Tämä tarkoittaa, että CANopen-laitteiden ominaisuuksia voidaan muokata SDO-ositteiden kautta, muokattavia ominaisuuksia voivat olla esimerkiksi väylänopeus, tapahtuma-ajastin, laitteen yksilöintitunnus tai datatyyppi, jolla laite kommunikoi. [29.] Esimerkiksi antureissa voidaan usein

muokata mitattavan suureen lähetettävää yksikköä esimerkiksi PA->bar, tai kulma-antureissa radiaaneista asteiksi.

Yksi SDO koostuu kahdesta eri datakehyksestä, joilla on omat ID-tunnisteet, toinen on vastaanottava kehys ja toinen lähettävä. Esimerkiksi jos halutaan muuttaa laitteen tiedonsiirtonopeus, lähetetään viesti johonkin SDO-osoitteeseen, ja anturi vastaa onnistuneesta muutoksesta toisesta SDO-osoitteesta. SDO yksilöidään aina lisäämällä SDO-osoite laitteen ID-numeroon, minkä mukaan identifioidaan muokattava laite. Esimerkiksi 600 h +node ID. [29.] (Liite 3)

### 6.2.3 PDO (Process data object)

PDO:t ovat tarkoitettu prosessidatan (Process Data) siirtämiseen laitteiden välillä. Tämä voi olla esimerkiksi anturien tai toimilaitteiden tuottamaa reaaliaikaista tietoa. PDO:t päivitetään tietyn ajan välein, mikä määritellään CANopen-verkossa. Tämä aika voi vaihdella sovelluksen mukaan, mutta se on tyypillisesti hyvin nopea, mikä mahdollistaa reaaliaikaisen tiedonsiirron. [30.]

CANopen-laitteiden käyttö vaatii määrittämiä järjestelmän laitteiden välillä. Nämä määrittäykset sisältävät tiedot siitä, mitä tietoa siirretään, missä muodossa ja kuinka usein. Asetukset ja määrittäykset tallennetaan CANopen-objekteina, mikä helpottaa niiden konfigurointia ja hallintaa. [24.]

PDO:t ovat tärkeitä CANopen-verkkojen dynaamisessa toiminnassa, ja ne mahdollistavat reaaliaikaisen datan tehokkaan tiedonsiirron laitteiden välillä. Ne tarjoavat joustavan tavan määrittellä, mitä tietoa ja milloin siirretään, mikä on kriittistä monimutkaisissa automaatio- ja ohjaussovelluksissa. [30.]

### 6.2.4 NMT

NMT-protokolla mahdollistaa verkon eri tilojen hallinnan, kuten käynnistämisen, pysäyttämisen ja uudelleen käynnistymisen. Tämä on tärkeää esimerkiksi teollisuusautomaatiossa, jossa laitteiden on voitava käynnistyä tai pysähtyä hallitusti. [31.]

NMT-protokolla tarjoaa mekanismin valvoa verkkokommunikaation tilaa ja suorituskykyä. Se voi esimerkiksi tarkistaa, onko verkon solmu aktiivinen, ja tarvittaessa käynnistää uudelleen tai ilmoittaa mahdollisista ongelmista. [31.]

Protokolla määrittelee erilaisia tiloja, joita solmut voivat ottaa verkossa. Tämä voi sisältää esimerkiksi esikäynnistyksen, käytön ja pysäytyksen tilat. NMT käsittelee myös erikoistilanteita, kuten hätätilanteet ja erikoiskomennot, jotka voivat olla tärkeitä järjestelmän turvallisuuden ja tehokkuuden kannalta. [31.] Kuvassa 9 on esitettyä yhden CANopen-laitteen kommunikaatio-osoitteet.

Communication object	COB-ID(s) hex	Slave nodes
NMT node control	000	Receive only
Sync	080	Receive only
Emergency	080 + NodeID	Transmit
TimeStamp	100	Receive only
PDO	180 + NodeID 200 + NodeID 280 + NodeID 300 + NodeID 380 + NodeID 400 + NodeID 480 + NodeID 500 + NodeID	1. Transmit PDO 1. Receive PDO 2. Transmit PDO 2. Receive PDO 3. Transmit PDO 3. Receive PDO 4. Transmit PDO 4. Receive PDO
SDO	580 + NodeID 600 + NodeID	Transmit Receive
NMT node monitoring (node guarding/heartbeat)	700 + NodeID	Transmit
LSS	7E4 7E5	Transmit Receive

Kuva 9, kuva erään CANopen-anturin konfigurointiohjeesta, kuvassa on esitetty kommunikointikanavat.

### 6.2.5 Error control

Error Control (virheen hallinta) on tärkeä osa CANopen-protokollaa ja monia muita tiedonsiirto-protokollia. Se on suunniteltu havaitsemaan ja hallitsemaan virheitä, jotka voivat esiintyä tie-

donsiirrossa CAN-väylällä. CANopen käyttää CRC-tarkistusta tietojen eheyden varmistamiseen. Tietojen mukana lähetetään CRC-arvo, joka vastaa lähettäjälaitteessa lasketun CRC:n arvoa. Vastaanottajalaitteessa CRC tarkistetaan ja virheelliset kehykset hylätään. [32.]

Error control sisältää mekanismin, joka mahdollistaa virheen ilmaisemisen ja reagoinnin. Jos virhe havaitaan, voidaan lähettää erityinen virhekehys, joka ilmoittaa verkossa oleville laitteille, että virhe on tapahtunut. [32.]

Error control käyttää ACK-bittipalautetta, joka auttaa vahvistamaan, että viesti on vastaanotettu oikein. Jos vastaanottava solmu ei pysty vahvistamaan viestiä, lähettävä solmu voi havaita virheen. [32.]

#### Overload Frame (Ylikuormituskenttä)

Jos verkon kuormitus on liian suuri, CANopen voi käyttää ylikuormituskenttää ilmoittamaan laitteille, että ne saattavat olla ylikuormitettuja, ja että ne saattavat hylätä joitakin viestejä. [32.]

#### Error Signaling (Virheen ilmaisu)

Virheilmaisu CANopenissa voi tapahtua useilla tavoilla, esimerkiksi lähettämällä NMT-protokollalla erityisiä virheilmoituksia tai käyttämällä PDO-protokollaa ilmoittamaan laitteille virheellisestä tilasta. CANopen sisältää myös mekanismeja diagnostiikan tekemiseen virheiden havaitsemiseksi ja analysoimiseksi. Tämä auttaa tekemään järjestelmästä luotettavamman ja ylläpitämään parempaa suorituskykyä. [32.]

Yleensä ottaen virheen hallinta on keskeinen osa CANopen-protokollaa, mikä tekee siitä sopivan käytettäväksi sovelluksissa, joissa luotettavuus ja virheettömyys ovat ensisijaisen tärkeitä, kuten teollisuusautomaatiossa ja ajoneuvojen ohjausjärjestelmissä. [14.]

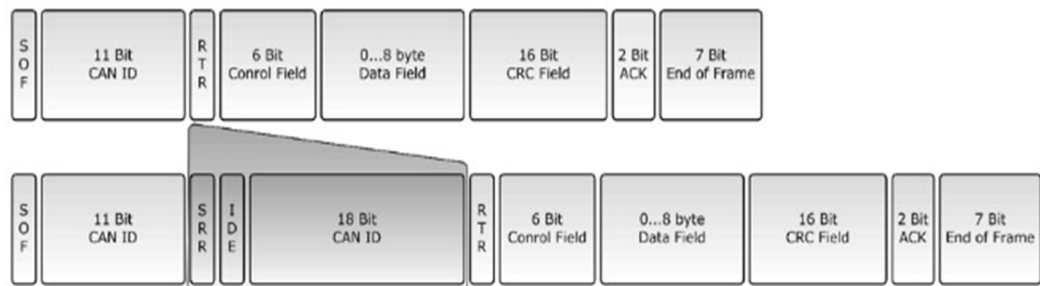
## 7 CAN-viestintä

CAN-väylän tiedonsiirto perustuu fyysisessä kerroksessa tapahtuvaan jännitemuutokseen, jotka tulkitaan binäärijärjestelmän arvoiksi 1 tai 0 [33.] CAN-viestit voidaan lukea väylältä binäärisenä ja muuntaa tarvittaessa heksadesimaalimuotoon. Kehittyneemmät CAN-viestien analysointiohjelmit näyttävät viestit binäärisenä sekä heksadesimaalisena (Liite 2).

Mikäli CAN-viestistä halutaan fyysinen arvo, esimerkiksi moottorin jäähdytysnesteiden lämpötila tai jonkin anturin lähettämä arvo esimerkiksi enkooderin kulma, muunnetaan binäärikoodi tai hexa-desimaali normaaliksi kymmenkantaiseksi lukuarvoksi. [34.]

Kymmenkantaiseksi muuttamisen jälkeen yleensä saatu arvo täytyy kertoa jollakin tietyllä kertoimella, jolloin arvo saadaan oikeassa yksikössä. On kuitenkin mahdollista, että jotkut signaalit ovat suoraan lähetetty jossakin tietyssä yksikössä, tällöin ei kerrointa tarvita, eli kerroin on 1. [34.]

CANopen-antureissa voidaan anturiin määrittää tietyissä rajoissa, miten ja missä yksikössä viesti lähetetään. Tällöin anturin signaalille kannattaa tehdä DBC-tiedosto. DBC-tiedosto sisältää kaikki oleelliset tiedot siitä, miten viesti luetaan ja skaalataan ymmärrettäväksi ihmiselle. [34.] Jos ostaa laitteen, joka ei ole CANopen, vaan esimerkiksi J1939-protokollan mukainen, toimitetaan sen mukana yleensä valmistajan tekemä DBC-tiedosto. Tämä tiedosto on melkein välttämätön, että tuotteeseen saadaan luotua toimiva kommunikaatio. J1939-protokolla itsessään määrittelee väylän viestirakenteen, joten J1939:ä hyödyntävän laitteeseen saadaan kommunikaatio yleisillä J1939 DBC-tiedostoillakin, valmistajan laitekohtainen DBC tarjoaa kuitenkin usein lisäominaisuuksia, joita standardi ei määrittele. [25.] Kuten kuvasta 10 huomataan, näiden kahden CAN-protokollan eroavaisuus on pelkästään CAN-ID kentän pituudessa. Kaikki muut viestikentän osat ovat samat.



Kuva 10, CANopen- ja J1939-protokollan mukaiset viestirakenteet. [21.]

CAN-viestin eri kentät selityksineen

**Start Of Frame (SOF):** Viestin alussa oleva bitti synkronoi kaikki verkon solmut yhden CAN-viestin ajanjaksoon. [33.]

**Arbitration field:** Tässä kentässä määritetään viestin prioriteetti CAN-tunnisteen ja Remote Transmission Request (RTR) -bitin avulla. RTR-bitti erottaa tunnistekentän ja muut datakentät toisistaan. [33.]

**Control field:** Tämä alue sisältää tunnistekentän, laajennusbitin (IDE) ja Data Length Code (DLC) -tiedon. IDE-bitti määrittää, käytetäänkö viestissä 11- vai 29-bittistä tunnistekenttää. [33.]

**Data field:** Tämä on sovellusdatan alue, joka voi vaihdella nolasta jopa kahdeksaan tavuun. Se sisältää itse viestin tiedot. [33.]

**CRC field:** Tässä lasketaan syklisen redundanssitarkistuksen summa, joka auttaa varmistamaan viestin eheyden. [33.]

**Acknowledge (ACK) field:** Vastaanottavat solmut vahvistavat viestin vastaanottamisen tässä kentässä. [33.]

**End Of Frame (EOF):** Tämä ilmaisee viestin lopun. [33.]

## 7.1 DBC (DataBaceCan)

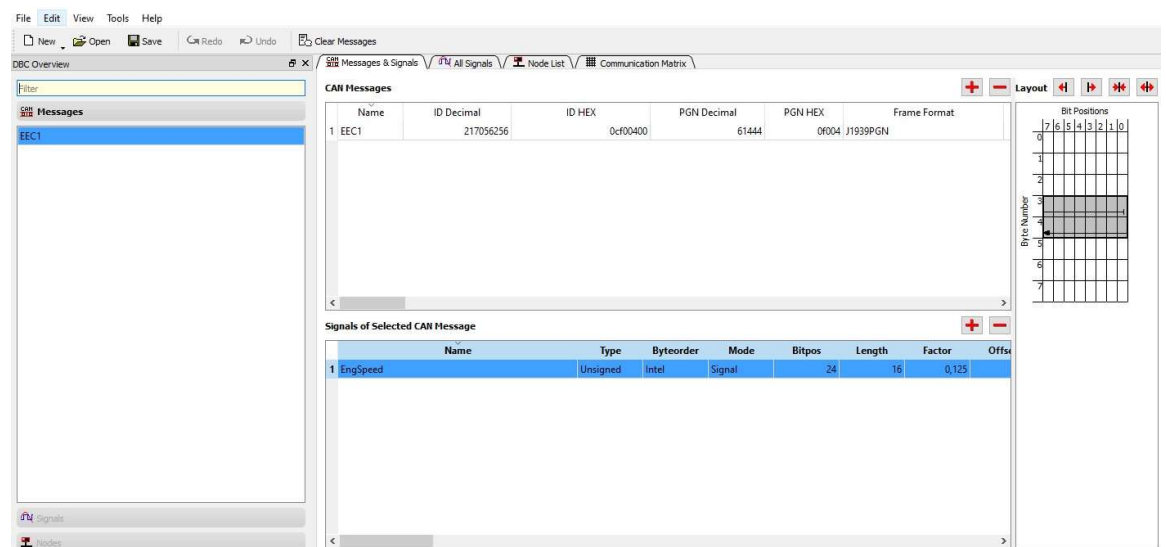
CAN-tiedonsiirto käyttää binäärijärjestelmää tiedon siirtämiseen. Binäärinen data ei itsesään ole ihmiselle ymmärrettävää dataa, että sitä voitaisiin tulkita ja analysoida. CanDataBase (DBC) on tekstitiedosto, jota CAN-analysointiohjelmat sekä jotkin mittalaitteet hyödyntävät datan muuntamiseen fyysisiksi arvioiksi. [34.]

DBC-tiedosto sisältää kaikki tarvittavat tiedot siitä, miten binäärinen CAN-viesti saadaan muutettua ymmärrettävään muotoon. Lisäksi se määrittää, miten ja missä järjestyksessä binääristä dataa luetaan, että luetaan haluttua tietoa. Yksi CAN-viesti voi sisältää useampia signaaleja. Signaalin tiedon määrä sekä haluttu resoluutio määrittävät, kuinka monta bittiä signaali vaatii. [34.]

Esimerkiksi J1939-protokollan mukainen viesti EEC1 (Electronic Engine Controller 1) sisältää dieselmoottorin pyörintänopeustiedon, joka lähetetään väylälle 16 bitin eli kahden tavun mittaisena. 16 bitin suurin mahdollinen kymmenkantaisen kantalukujärjestelmän arvo on 65535, joka kerrotaan DBC-tiedostossa kerrotulla kertoimella (factor), ja tällöin maksimikierroslukuksi saadaan 8191 rpm. Tiedon esittämiseen riittäisi siis kymmenkantaisen järjestelmän neljä merkkiä, saman tiedon esittämiseen binäärisenä tarvitaan kuitenkin 16 bittiä. Tämän signaalin desimaalinen arvo on tuhansia (Liite 2). Tällöin ei voida käyttää yhtä tavua eli kahdeksaa bittiä tiedon esittämiseen. Koska suurin mahdollinen desimaaliluku, joka saadaan esitettyä kahdeksalla bitillä, on 255, on käytettävä joka tapauksessa kahta tavua eli 16 bittiä.

On järkevää käyttää kaikki 16 bittiä tiedon esittämiseen, koska tällöin väylälle lähetettävän tiedon resoluutio parantuu moninkertaiseksi ja signaali voidaan skaalata DBC-tiedoston kertoimen avulla moottorin todelliseksi pyörintänopeudeksi.

Kuvassa 11 avattuna Kvaser DBC- editorilla esimerkksignaaleja J1939 protokollan mukaisista signaaleista, korostettuna tarkemmin kuvattu signaali moottorin kierrosluvusta.

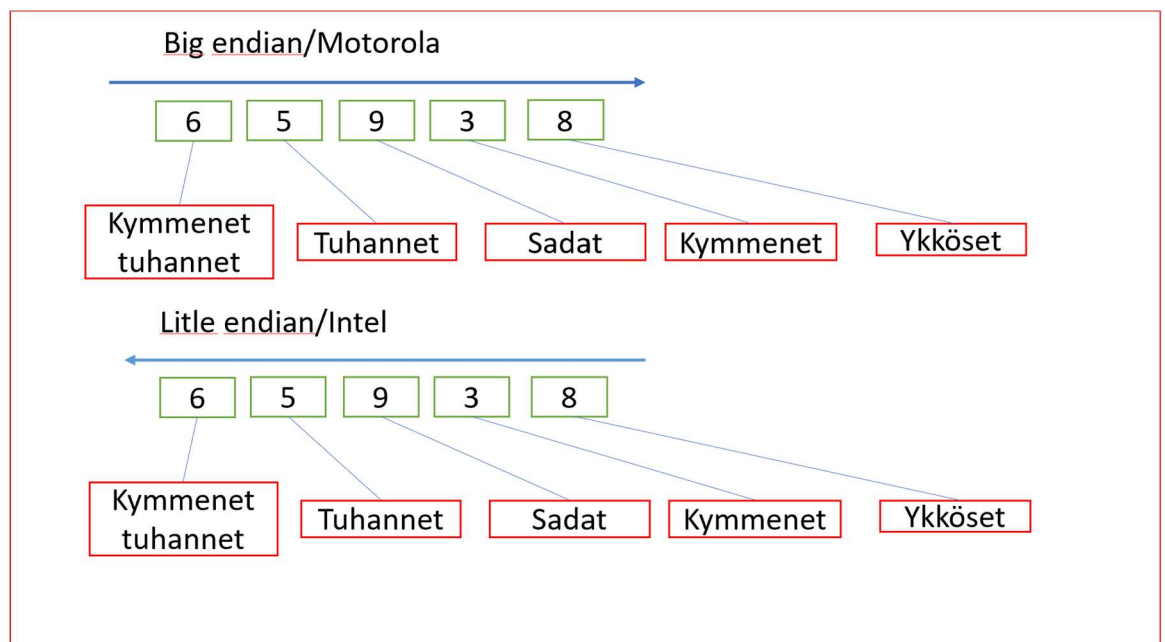


Kuva 11, Kuva J1939 protokollan mukainen moottorin kierroslukusignaali.

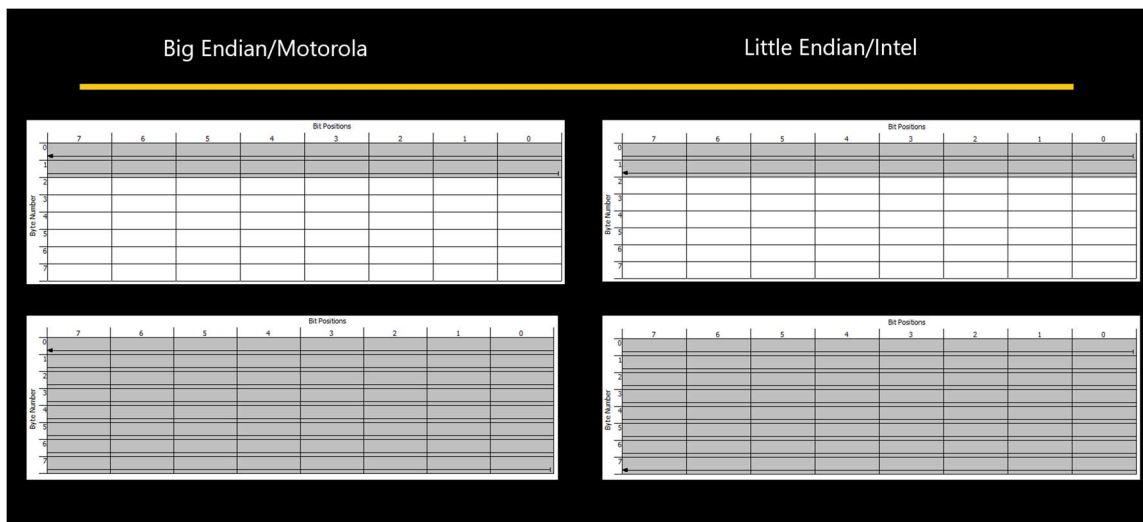
DBC-tiedostossa määritellään myös tavujen lukujärjestys (Byteorder). Tavujen lukujärjestyksiä on kaksi, eniten merkitsevä ensin (Big endian) ja vähiten merkitsevä ensin (Little endian). Big endian - ja Little endian -määritelmistä käytetään myös yleisesti nimityksiä Motorola ja Intel. Nimitykset johtuvat historiasta, kahden eri mikropiirivalmistajan käyttämien erilaisten tavujen lukujärjestyksistä. [35.]

Esimerkiksi jos tarkastellaan kymmenkantaisen luvun 65938 esitystä, molemmilla lukujärjestyksillä kuva 11. Byte order on siis nimensä mukaisesti tavujen lukujärjestys, bitit tavun sisällä luetaan täysin normaalisti oikealta vasemmalle. [35.]

Tavujen lukujärjestyksellä on erittäin suuri merkitys. Mikäli tavut luetaan väärin, saadaan viesti, joka on aivan väärä tai todennäköisemmin ei saada binääriviestistä mitään selvää. [35.] Kuvassa yksitoista sekä kaksitoista havainnollistetaan molempien lukujärjestyksien toimintaperiaatetta. Kuvissa 12 ja 13 havainnollistetaan tavujen lukusuuntaa kahdella eri tavalla.



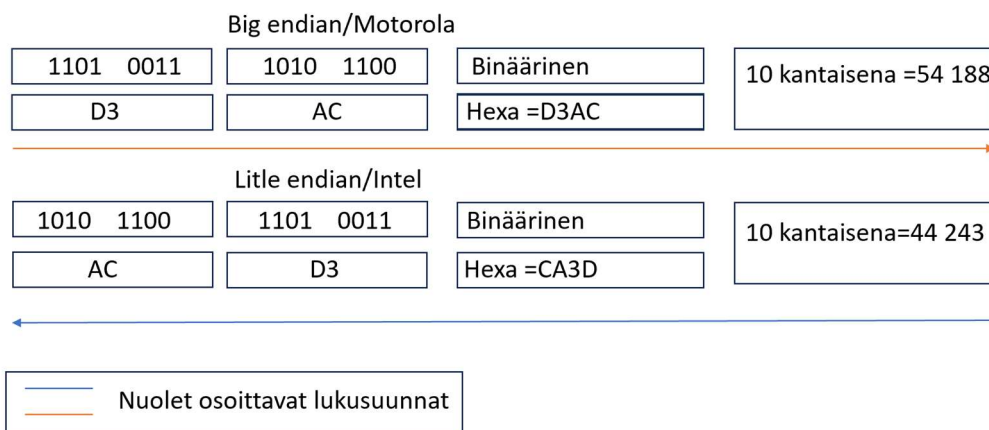
Kuva 12, Big endian ja Little endian havainnollistettu, sininen nuoli kertoo lukusuunnan.



Kuva13, Big endian - ja Little endian -viestit ja lukujärjestykset.

Bitit luetaan tavun sisällä aina vasemmalta oikealle, vain tavujen lukusuunta muuttuu. Kuten huomataan, yksittäiset tavut saavat saman arvon kymmenkantaisena esitettynä, lukusuunnasta riippumatta, mutta kun luetaan useampia tavuja eli sana niin kymmenkantainen arvo muuttuu riippuen tavujen lukujärjestyksestä. [35.]

Kuvassa 14 esimerkki tavujen lukujärjestyksestä, binaariluvusta Big endian 1101 0011 1010 1101. Yksi hexadesimaali esittää aina puolitavun eli neljän bitin arvon. Yhden tavun esittämiseen tarvitaan siis kaksi hexadesimaalia.



Kuva 14, Tavujen lukujärjestyksien merkitys.

### 7.1.1 Big Endian

Big endian on tavun lukujärjestys, jossa tavut luetaan eniten merkitsevistä vähiten merkitsevään. Esimerkiksi päivämäärä luetaan Yhdysvalloissa eniten merkitsevä ensin, eli vuosi, kuukausi ja päivä. Tämä tarkoittaa, että esimerkiksi CAN-viestin tavut luetaan samalla tavalla. [35.] Toinen hyvä esimerkki on perinteinen suomalainen numeroiden lukeminen, otetaan vaikka luku 5627, ensin luetaan tuhannet sitten sadat, kymmenet ja viimeisenä ykköset.

### 7.1.2 Little Endian

Little endian on päinvastainen lukujärjestys, jos käytetään Little endiana, niin luetaan vähiten merkitsevä ensin. Tästä paras esimerkki on mielestäni päivämäärän esitys Suomessa eli ensin päivä sitten kuukausi ja viimeisenä vuosi. Jos tätä verrataan normaaliin lukujen lukujärjestykseen, niin se on päinvastainen: ensin ykköset, sitten kymmit ja sataset, viimeisenä tuhannet jne. [35.] Esimerkiksi luku 5627, luetaan seitsemän, kaksi, kuusi ja viisi.

## 8 Arcnet

Arcnet (Attached Resource Computer Network) on lähiverkkotekniikka, jonka kehitti yritys nimeltä Datapoint Corporation. Arcnet on kehitetty 1970-luvun loppupuolella. Arcnet oli aikoinaan yksi suurimmista kilpailijoista nykyisin yleisesti käytössä olevan Ethernetin kanssa. [36.]

Arcnet oli aikoinaan melko suosittu toimistoautomaatiossa, nykyään Arcnet on suosiossa monissa sulautetuissa verkkoratkaisuissa, jota käytetään useissa sovelluskohteissa, kuten teollisuusohjauksissa, rakennusautomaatiossa ja robotiikassa. [36.]

Arcnet on suosiossa erityisesti Japanissa, Amerikassa, Euroopassa ja suosio on nousussa myös Kiinan markkinoilla. Arcnet on suhteellisen tuntematon verkkoratkaisu, koska Arcnetia käyttävät järjestelmät ja niiden valmistaja eivät yleensä mainosta käyttävänsä kyseistä tietoverkkoa sovelluksissaan. [36.]

Näiden järjestelmien käyttäjät eivät monesti edes tiedä järjestelmän käyttävän Arcnet-pohjaista tiedonsiirtoratkaisua. Arcnet on kuitenkin todella suosittu sulautetuissa järjestelmissä, mistä kertoo yli 22 miljoonaa myytyä Arcnet-solmua. [36.]

Viestintä Arcnetissä perustuu ”tokeneihin”, joka tarkoittaa kiertävää vuoroa hallita verkon tietoliikennettä, tokenin ollessa solmulla, kyseinen solmu on isäntänä verkossa (master), tällöin se hallitsee verkkoa ja voi lähettää verkkoon. Token siirtyy aina järjestyksessä seuraavalle solmun loogiselle naapurille, mikä tarkoittaa, ettei solmun loogisen naapurin tarvitse sijaita fyysisesti edellisen solmun vieressä. Looginen naapuri tarkoittaa solmun osoitenumeron määrittämää naapuria. [36.]

## 9 Työn tulokset

Tämän työn tuloksena syntyi selvitys Ponsse-metsäkoneiden tiedonsiirtotopologiasta sekä eri tiedonsiirtomenetelmien keskinäisistä rajapinnoista. Työn tavoitteena oli kouluttaa tuotekehityksen testausinsinöörejä ja asentajia hyödyntämään CAN-tiedonsiirtoväyliä koneen toiminnan mittauksissa ja kehittämisessä. Tämä tavoite täytettiin onnistuneesti, sillä tuloksena oli kurssi-materiaali aiheesta sekä kaksipäiväinen koulutus datan keräyslaitteiston.

Selvitystyön tulosten perusteella tiedonsiirron hyödyntämisessä päädyttiin keskittymään CAN-pohjaiseen ratkaisuun. CAN-pohjaiseen tiedonsiirtoon on tarjolla huomattavasti laitteistoja sekä ohjelmistoja, joten CAN-pohjainen tiedonsiirto on oikea ratkaisu tiedonsiirron hyödyntämiseen testauksissa. Tähän lopputulokseen päästiin tutkimalla saatavilla olevia laitteistoja sekä ohjelmistoja. Arcnet- tiedonsiirtoon on olemassa laitteita, muttei ohjelmistoja.

Työn tuloksena laadittiin myös dokumentaatio ja ohjeistus datankeräyslaitteiston käyttöönotosta sekä ohjeet laitteeseen liittyvien ohjelmistojen käytöstä.

Toimeksiantajalle saavutettuihin tuloksiin voidaan lukea tuotekehityksen tarpeisiin konfiguroitua ja CANopen-antureita sekä ohjeistusdokumentaatioita niiden konfiguroinnista. Työn tuloksena saavutettiin paljon tietotaitoa ja ymmärrystä CAN-tiedonsiirtoon ja Ponssen järjestelmiin liittyen, ja sitä voidaan jakaa kollegoille sekä sidosryhmille.

Toimeksiantajan tavoitteena oli kouluttaa tuotekehityksen testausosaston insinöörit sekä asentajat tekemäni työn pohjalta ja siten lisätä tietämystä sekä osaamista koko osastolla. Tämä koulutus suoritettiin kaksipäiväisenä kurssina, joka sisälsi yhden päivän teoriaa ja toisen päivän käytännön rastikoulutusta.

Myös kyky tehdä, muokata ja soveltaa DBC-tiedostoja eri tarkoituksiin tuotekehityksen eri mittauksissa ja muissa CAN-väylään liittyvissä toimissa voidaan lukea työni tuloksiin.

## 10 Päätäntä

Opinnäytetyön aihe on todella laaja ja erittäin haastava. Sen vuoksi oli tarpeen rajata pois joitakin oleellisia osa-alueita, vaikka toiminnallinen osuus kuitenkin suoritettiin. Työni sisälsi useita eri testauksia ja mittauksia, joiden avulla osoitettiin työn tulosten toimivuus ja hankittiin paljon kokemusta tiedonsiirron hyödyntämisestä erilaisissa mittaus- ja testausvariaatioissa. Lisäksi työhöni kuului tuotekehityksen testausosaston kouluttaminen ja tekninen tuki, kun kyse oli tiedonsiirron hyödyntämisestä testauksissa.

Työn valmistumisen myötä voitiin todeta, että toimeksiantajan tavoitteet täyttyivät odotusten mukaisesti. Toimeksiantaja sai runsaasti lisätietoa ja osaamista CAN-tiedonsiirtoon liittyen. Lisäksi toimeksiantajan tuntemus tiedonsiirron osalta lisääntyi, ja tiedonsiirtojärjestelmän kuvaus yksinkertaistui huomattavasti luomani tiedonsiirtotopologiakaavion myötä. Työni myötä hankittiin hyvä ymmärrys datantallennuslaitteen käytöstä ja toiminnasta, mukaan lukien datantallennuslaitteeseen liittyvät ohjelmistot.

Työssäni tutkittiin myös Arcnetin mahdollisuuksia tuotekehityksen testausvälineenä, mikä lisäsi myös Arcnetin tuntemusta, vaikka se jäikin marginaaliseksi verrattuna CAN-tiedonsiirtoon. CAN-tiedonsiirron osalta ymmärrys ja tietämys lisääntyivät merkittävästi, mutta on kuitenkin huomioitava, ettei tämänkään työn myötä CAN-tiedonsiirtoon ole saavutettu täydellistä osaamista tai tietämystä. CAN on käsitteenä monimutkainen ja laaja, ja työssä käsiteltiin ainoastaan raskaan kaluston protokollaa sekä CANopen-protokollaa.

Ymmärrys lisääntyi myös yleisellä tasolla merkittävästi, kun puhutaan binäärijärjestelmistä ja järjestelmän käytöstä erilaisissa sovelluksissa. Työni antoi kattavan tietopohjan myös tulevaisuuden laitekommunikaatioon ja selvensi analogisen ja digitaalisen kommunikaation hyödyt ja haitat. Tämän tiedon valossa on helpompaa valita oikea tiedonsiirto oikeisiin kohteisiin.

Tiedonsiirtojärjestelmien hyödyntäminen tuotekehityksen testauksissa vaatii myös tulevaisuudessa panostusta, koska tiedon määrä kasvaa ja tiedonsiirtomenetelmät kehittyvät. Tässä työssä kartoitettujen CAN-laitteiden käyttöä sekä tuomia lisäominaisuuksia ja hyötyjä olisi myös syytä tutkia tarkemmin ja miettiä kustannus/hyöty-suhteita. Edellä mainituilla toimilla tiedonsiirron hyödyntäminen voitaisiin viedä uudelle tasolle.

Työ oli kaikin puolin onnistunut, mutta jos vastaava työ täytyisi tehdä uudelleen, aloittaisin työn hieman eri järjestyksessä sekä käyttäisin enemmän aikaa työhön liittyvien laitteiden ja ohjelmis-

tojen tutkimiseen. Myös ajan käytön kannalta tekisin toisin kuin tätä työtä tehdessäni; raportointiosuuden tekisin kartoitus- ja toteutusvaiheen kanssa yhtenäisesti. Tällä toimintamallilla välttyttäisiin raportoinnin viivästymiseltä verraten työn muiden vaiheiden valmistumiseen.

Työtä tehdessäni opin erittäin paljon tietotekniikasta ja digitaalisten järjestelmien tuomasta lisäarvosta käyttäjilleen. Työ oli kokonaisuutena todella tekninen ja se opetti minulle lähes kaiken, mitä tässä työssä käsitellään, lukuun ottamatta CAN-väylän fyysistä kerrosta sekä perustoimintaperiaatetta.

Työssä oli määritellyt tavoitteet, joihin tähdättiin. Tavoitteet saavutettiin kovalla opiskelulla sekä runsaalla järjestelmien testaamisella, minkä seurauksena muodostui erittäin kattava ja todennukainen käsitys tiedonsiirtojärjestelmien toiminnasta, niin teoriassa kuin käytännössäkin.

## Lähteet

1. Ponsse metsäkone valmistajan kotisivut. viitattu 21.04.2024 [internet]. Saatavilla: <https://www.ponsse.com/fi/etusivu#/>
2. Aalto-yliopisto Tiedonsiirron perusteet. Viitattu 20.04.2024 [internet]. Saatavilla: <http://www.cse.hut.fi/fi/opinnot/T-110.2100/2012/luennot-files/Tiedonsiirron.pdf>
3. Vector Informatik GmbH on saksalainen yritys, joka on erikoistunut sulautettujen järjestelmien ja ajoneuvoverkkojen kehitystyökaluihin ja ratkaisuihin. Viitattu 15.10.2023 [internet]. Saatavilla: <https://www.vector.com/int/en/products/products-a-z/software/preevision/automotive-ethernet-design/>
4. Bosch Can-väylän kehittäjän kotisivut. Viitattu 21.04.2024 [internet]. Saatavilla: <https://www.bosch.com/stories/the-controller-area-network/>
5. Network World tarjoaa syvällistä verkkoalueen asiantuntemusta. Viitattu 21.04.2024 [internet]. Saatavilla: <https://www.networkworld.com/article/970970/what-is-ethernet.html>
6. Arcnet Trade Association (ATA). Viitattu 21.04.2024 [internet]. Saatavilla: <https://arcnet.cc/history.htm>
7. PROFIBUS & BROFINET International (PI). Viitattu 21.04.2024 [internet]. Saatavilla: <https://www.profibus.com/download/profibus-technology-and-application-system-description/>
8. Modbus Organization. Viitattu 21.04.2024 [internet]. Saatavilla: <https://www.modbus.org/>
9. LOCAL INTERCONNRCT NRTWORK (Lin). Viitattu 21.04.2024 [internet] Saatavilla: <https://web.archive.org/web/20111005152436/http://www.lin-sub-bus.de/index.php?pid=5&lang=en&sid=252efac88ea251b38ba4af5e973d19fb>
10. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/can-knowledge/can/classical-can/>

11. Aalto-yliopisto johdatus kryptografiaan Viitattu 01.04.2024 [internet]. Saatavilla: <https://plus.cs.aalto.fi/cs-ej4404/2022/01-Johdanto/04-BititTavutBinaarioperaatiot/?hl=fi>
12. IBM (International Business Machines Corporation) on teknologiayritys, joka on tunnettu suurtietokoneiden ja raskaiden palvelimien valmistajana ja alkuperäisen IBM PC -arkkitehtuurin kehittäjänä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.ibm.com/docs/en/aix/7.3?topic=types-integer-data>
13. Tutorials Point on johtava yritys, joka pyrkii tarjoamaan parasta oppimateriaalia teknisistä ja ei-teknisistä aiheista. Viitattu 15.10.2023 [internet]. Saatavilla: <https://www.tutorialspoint.com/unsigned-and-signed-binary-numbers>
14. Heikki Saha Sandvik Tamrock Oy CAN-väylä. Viitattu 21.04.2024 [internet]. Saatavilla: <https://www.yumpu.com/fi/document/read/15394566/can-vayla-canopen>
15. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/news/cia-in-action/view/do-not-use-laboratory-slang/2018/5/14/>
16. Typhoon HIL on yritys, joka tarjoaa reaaliaikaisen Hardware-in-the-Loop (HIL) -alustan tehoelektronikan suunnitteluun, testaukseen ja varmennukseen. Viitattu 21.04.2024 [internet]. Saatavilla: [https://www.typhoon-hil.com/documentation/typhoon-hil-software-manual/References/can\\_bus\\_protocol.html](https://www.typhoon-hil.com/documentation/typhoon-hil-software-manual/References/can_bus_protocol.html)
17. Ditet on yksi johtavista ja tärkeimmistä toimittajista autojen diagnostisten työkalujen alalla Bulgarian markkinoilla. 25.04.2024 [internet]. Saatavilla: <https://autoditex.com/page/can-bus-physical-layer-74-1.html>
18. Vector Informatik GmbH on saksalainen yritys, joka on erikoistunut sulautettujen järjestelmien ja ajoneuvoverkkojen kehitystyökaluihin ja ratkaisuihin. Viitattu 15.10.2023 [internet]. Saatavilla: <https://www.vector.com/int/en/know-how/protocols/sae-j1939/#>
19. Tietotekniikkayritys Guang Zhou Han Ting Co. Ltd. on erikoistunut korkean teknologian valmistaja, joka tuottaa FM- ja TV-lähettimeä sekä niihin liittyviä tarvikkeita. Viitattu 25.04.2024 [internet]. Saatavilla: <https://fi.fmuser.net/content/?21055.html>
20. Electronics-tutorials on verkkosivusto, joka tarjoaa laajan valikoiman oppaita ja opetusmateriaaleja sähkötekniikan alalla. Viitattu 25.04.2024 [internet]. Saatavilla: [https://www.electronics-tutorials.ws/resistor/res\\_4.html](https://www.electronics-tutorials.ws/resistor/res_4.html)
21. CSS Electronics on ammattitason, helppokäyttöisten ja edullisten CAN-väylän tiedonkeruulaitteiden kehittäjä. Viitattu 25.04.2024 [internet]. Saatavilla: <https://www.csselectronics.com/products/terminal-resistor-can-bus>

22. Mindsensors.com tarjoaa korkealaatuisia osia Raspberry Pi-, LEGO Mindstorms NXT- ja EV3 Robotic -järjestelmille. Viitattu 25.04.2024 [internet]. Saatavilla: <http://www.mindsensors.com/content/86-can-and-its-topology>
23. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/canopen/>
24. Copperhill Technologies on yritys, joka keskittyy sulautettujen järjestelmien kehittämiseen ja myyntiin. Viitattu 13.10.2023 [internet]. Saatavilla: <https://copperhilltech.com/blog/sae-j1939-message-format-and-interpretation-of-pgns/>
25. CSS Electronics on ammattitason, helppokäyttöisten ja edullisten CAN-väylän tiedonkeruulaitteiden kehittäjä. Viitattu 25.04.2024 [internet]. Saatavilla: [https://www.csselectronics.com/pages/j1939-explained-simple-intro-tutorial?\\_pos=7&\\_sid=c63012af6&\\_ss=r](https://www.csselectronics.com/pages/j1939-explained-simple-intro-tutorial?_pos=7&_sid=c63012af6&_ss=r)
26. National Instruments, johtava ohjelmistoon kytkettyjen automaattisten testi- ja mittausjärjestelmien toimittaja. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.ni.com/en/shop/seamlessly-connect-to-third-party-devices-and-supervisory-system/the-basics-of-canopen.html#section-1484315868>
27. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/can-knowledge/canopen/device-architecture/>
28. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/canopen/>
29. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/can-knowledge/canopen/sdo-protocol/>
30. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/can-knowledge/canopen/pdo-protocol/>
31. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/can-knowledge/canopen/network-management/>
32. Can in automation (CiA) kansainvälinen Control Area Network käyttäjien ja valmistajien ryhmä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.can-cia.org/can-knowledge/canopen/error-control-protocols/>

33. National Instruments, johtava ohjelmistoon kytkettyjen automaattisten testi- ja mittausjärjestelmien toimittaja. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.ni.com/en/shop/seamlessly-connect-to-third-party-devices-and-supervisory-system/controller-area-network--can--overview.html#section-1870976089>
34. Influx-technology Yritys, joka tuottaa ratkaisua ja työkaluja CAN viestinnän tulkitsemiseen. Viitattu 13.10.2023 [internet]. Saatavilla: <https://www.influxtechnology.com/post/understanding-can-dbc>
35. IBM (International Business Machines Corporation) on teknologiayritys, joka on tunnettu suurtietokoneiden ja raskaiden palvelimien valmistajana ja alkuperäisen IBM PC -arkkitehtuurin kehittäjänä. Viitattu 13.10.2023 [internet]. Saatavilla: <https://developer.ibm.com/articles/au-endianc/#end>
36. Arcnet Trade Association (ATA). Viitattu 21.04.2024 [internet]. Saatavilla: <https://arcnet.cc/abtarc.htm>