



MISKOLCI EGYETEM
GÉPÉSZMÉRNÖKI ÉS INFORMATIKAI KAR



TUDOMÁNYOS DIÁKKÖRI DOLGOZAT

Autóbuszok CAN üzenetforgalmának szimulálása távdiagnosztikai szűrőalgoritmusok fejlesztése céljából

Biró Zoltán

II. éves mérnök informatikus MSc hallgató

Konzulens:

Trohák Attila

egyetemi adjunktus

Automatizálási és Kommunikáció-technológiai Tanszék

Miskolc, 2012

Tartalomjegyzék

1.	Bevezetés	3
2.	CAN buszrendszer, jármű csoportok összehasonlítása	5
3.	Közepes és nagyteljesítményű járművek belső hálózata.....	12
4.	Autóbuszokon végzett mérések	28
5.	CAN üzenetforgalom szimulációja	37
6.	Összefoglalás	42
7.	Irodalomjegyzék	44
8.	Mellékletek	45

1. Bevezetés

A járművek diagnosztizálása fontos és egyre elterjedtebb, mert ezzel gyorsan megállapítható az egyes részekységek hibás működése, ez felgyorsítja és hatékonyabbá teszi a karbantartási folyamatokat. Ide tartozik a hibakeresés és a hibaelhárítás mellett a járművek gondozására és javítására vonatkozó tanácsadás is. A távdiagnosztika a hagyományos diagnosztikai eljárások telekommunikációs eszközökkel való támogatása, azaz jelen esetben azt jelenti, hogy a jármű és a diagnosztikát végző személy távol van egymástól. Az On-Board (fedélzeti) diagnosztika szerepe, hogy a járművek alrendszerait menetközben folyamatosan felügyelhetjük, ezért a távdiagnosztizáláshoz mindenképpen vezeték nélküli kommunikációs hálózatot kell választanunk. A hálózattal szemben támasztott legfontosabb követelmény a teljes rendelkezésre állás és a lefedettség. Ma Magyarországon csak a GSM hálózat felel meg ezeknek az elvárásoknak, ami 99 százalékos lefedettséget biztosít. Az adatátvitel a GSM hálózaton GPRS technológia segítségével oldható meg.

A járművek belső hálózata különböző, két csoportba oszthatóak. A következő fejezetben a járművek CAN buszrendszerének működését mutatom be általánosan. Ezután a két csoportot hasonlítom össze a diagnosztikai szabványok, a fizikai interfész és egyéb szempontok alapján.

Az összehasonlítás után a harmadik fejezetben a közepes és nagy teljesítményű járművek belső hálózata által használt SAE J1939-es szabványt részletezem. Az egyes vállalatoknak a flottaüzemeltetés, logisztika, személyszállítás területén ilyen járműből lehet sok, és a távdiagnosztizálásukkal növekedne a járművek megbízhatósága, a meghibásodott járművek gyorsan és egyszerűen diagnosztizálhatóvá válnak a telephelyről, ahonnan a szerelők a problémára felkészülten, a megfelelő felszerelések birtokában indulhatnak a jármű javítására, mentésére. Én az autóbuszok távdiagnosztizálásával foglalkozom, amik ugyancsak ebbe a csoportba tartoznak.

A jármű belső CAN kommunikációs rendszerének a sebessége sokkal nagyobb, mint ami elérhető a GPRS technológia segítségével, ezért valamilyen adat csökkentési

módszert kell alkalmazni, hogy megoldható legyen a távolból való diagnosztizálás. A szűrési eljárások megvalósításához mindenképpen fel kell mérni egy jármű hálózatának a tényleges kommunikációját. A negyedik fejezetben bemutatom az autóbuszokon elvégzett mérési eredményeimet.

Egy autóbusz beszerzése és üzembe tartása magas költségvonzattal jár, ezért szimuláció segítségével állítom elő a belső CAN hálózatának üzenetforgalmát. Ezáltal megvalósítható a kidolgozott szűrőalgoritmusok működése és a használatukkal elért adatmennyiségek elemzése, összehasonlítása. A szimulációt a mérésekre alapozva készítettem el. Ennek részletezése az ötödik fejezetben található.

2. CAN busrendszer, jármű csoportok összehasonlítása

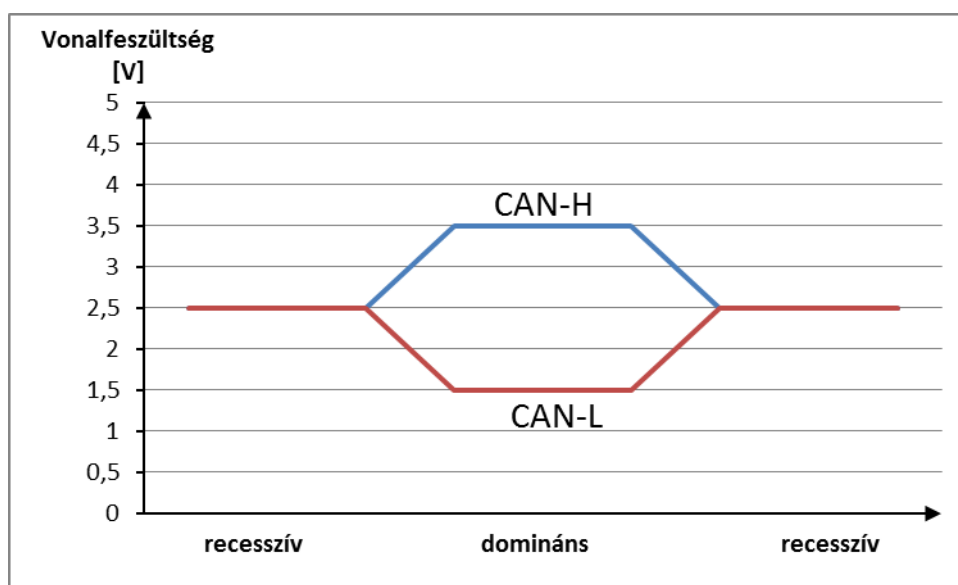
A Robert Bosch GmbH 1983-ban kezdett el egy belső projektet, amely célja egy járműfedélzeti hálózat kifejlesztése volt. A CAN (Control Area Network) protokoll hivatalos bemutatója 1986-ban történt meg, a Society of Automotive Engineers (SAE) kongresszuson, Detroit-ban. Az első CAN vezérlő chipet 1987-ben adta ki az Intel és a Philips cég együttműködve. A Bosch által továbbfejlesztett változat 1991-ben debütált CAN 2.0 néven. Már ebben az évben bevezetett egy CAN alapú magasabb rétegű protokollt a Kvaser cég. Egy évvel később, 1992-ben az első CAN hálózatot használó autó gördült le a Mercedes-Benz gyártósoráról és még ebben az esztendőben jött létre a CAN in Automation (CiA), amely felhasználók és gyártók nemzetközi csoportja. A CAN alap szabványát, az ISO 11898-at 1993-ban tették közzé. A CAN teljesíti az amerikai OBD-II jármű diagnosztikai standard előírásait, mely 1996-tól érvényes az USA-ban, és az EOBD szabványt, mely az európai benzinüzemű járművekre 2001-től, dízelekre pedig 2004-től alkalmazható.[1]

A CAN busz célja elsősorban az volt, hogy az autóipar decentralizációs törekvéseihez megbízható eszközként szolgáljon, ezért az alábbi célkitűzéseknek kellett megfelelnie:

- rendkívüli üzembiztosság,
- hibamentes átvitel,
- igen rövid ciklusidő, viszonylag magas átviteli sebesség mellett,
- adattartalom legyen optimális az autóipari érzékelők részére (0 - 8 bájt),
- busz kiépítése, állomások csatlakoztatása legyen egyszerű,
- broadcast támogatás.

A CAN egy multi-master üzenetszórásos, soros busz. Azaz busz topológiájú, a rácsatlakozó eszközök egyenrangúak, ezért mindenképpen biztosítani kell az arbitrációt, ezt a CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) buszhozzáférési eljárás alkalmazásával oldja meg. A buszrendszer adatátviteli sebessége 5 kbps-tól 1 Mbps-ig változhat. A CAN-busznak két protokollja is létezik, a gyors ISO 11898 és a lassú ISO 11519-2. A két protokoll csak a fizikai szinteken tér el egymástól, azaz vezetéken összekötve nem kompatibilisek. Mivel az ISO 11898-at

alkalmazzák szélesebb körben és a járművek is ezt használják, ezért a továbbiakban csak ennek a tulajdonságait részletezem. A buszhossza átlagosan 40 méter lehet, de ez függ a kábelezéstől és az adatátviteli sebességtől is. A buszra csatlakoztatott eszközök száma a kiviteltől függ, de általában legfeljebb 30 busz résztvevő lehet. Több fizikai kialakítása is létezik, az ISO 11898 szerint NRZ (Non Return to Zero) kódolást használ átviteli módként, tehát az egymás után következő 1-es bitértékek között a feszültségszint nem esik vissza nullára. Ennek előnye, hogy egy bit átviteléhez egy bitidő szükséges, viszont a szinkronizáció nehezebb, mint a minden bit átvitelénél átmenetet biztosító kódok esetén. Az átviteli közeg, azaz a kábelezés árnyékolt, vagy árnyékolatlan sodrott érpár lehet. A CAN busz a jelátvitelhez két vezetékot alkalmaz, a CAN-H (High) és CAN-L (Low) vezetékot (ellensodrott, esetleg árnyékolt). A zavarvédelem miatt az ezeken futó jelek ellenfázisúak, azonos jelekre ellenkező feszültségszint irányokba térnek ki.

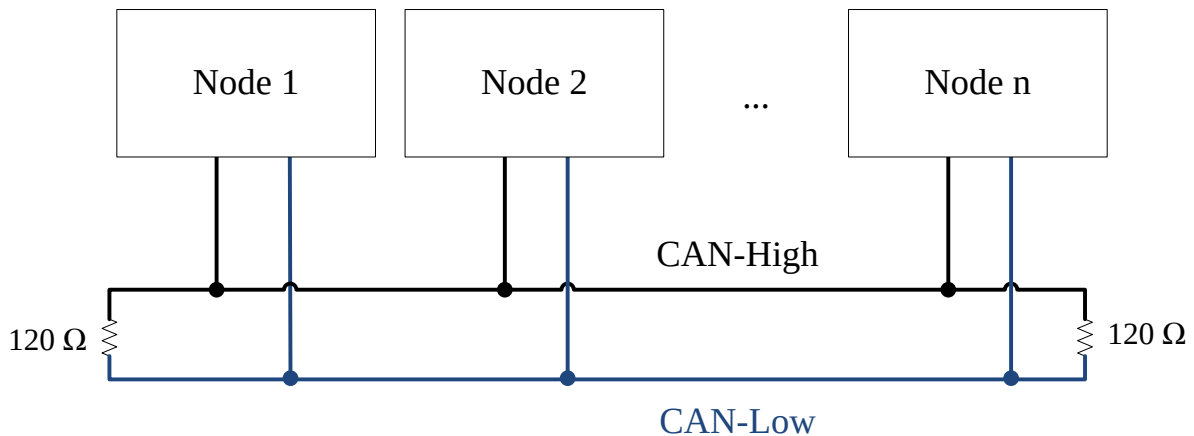


2.1. ábra: A CAN jelszintek recesszív és domináns bit esetén

A CAN-H és CAN-L buszvezetékek közötti feszültségkülönbségből állapítják meg, hogy domináns vagy recesszív bit érkezett. A recesszív bit a logikai 1-et szimbolizálja, ennek az átvitelkor 0 V a feszültségkülönbség, mivel mindkét vonalon 2,5 V feszültség van. Domináns bit, azaz logikai 0 átvitelkor 2 V a feszültségkülönbség, mivel a CAN-H vonal 3,5 V, a CAN-L vonal pedig 1,5 V-on van.

A megengedett feszültségkülönbségek recesszív bit esetén maximum 0,5 V, domináns bit esetén minimum 0,9 V.[2]

Az ISO 11898 által meghatározott CAN buszrendszer technikai jellemzői közé tartozik a busz fajlagos ellenállása, ami 70 M Ω /m, a busz késleltetési idő, ami 5ns/m és a véglezáró ellenállás értéke, ami 120 Ω -os általában, de minimum 85, maximum 130 Ω lehet.



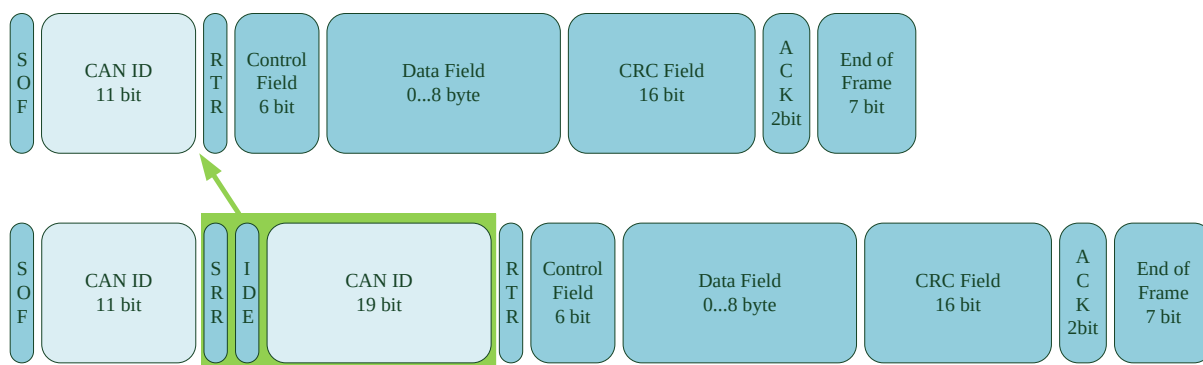
2.2. ábra: A CAN buszrendszer topológiája

A CAN üzenetorientáltan működik, vagyis ha egy résztvevő adatokat akar küldeni, összeállítja és azonosítóval látja el a telegramot. Az azonosító csak egyszer létezik, ezért ha egy résztvevő felismeri azt a buszon, egyértelműen tudja, hogy mely adatok következnek. Így a busz minden résztvevője az azonosító alapján el tudja dönteni, hogy akarja-e venni az üzenetet, függetlenül attól, hogy ki küldi. A CAN rendszerben tehát az adatok az azonosító révén, tartalom szerinti címezéssel jutnak el a megfelelő vevőhöz. Minden vevőnek kellő „intelligenciával” kell rendelkeznie az adatok azonosításához.

Öt fajta üzenetkeretet támogat:

- normál üzenetkeret,
- kibővített üzenetkeret,
- kéréses üzenetkeret,
- hiba üzenetkeret,
- túlterheltség üzenetkeret.

A CAN 2.0-nak kétfajta szabványa létezik, amelyek csak az üzenetkeretben térnek el. A CAN 2.0A szabvány nem támogatja a kibővített keretformátumot, így alap üzeneteket csak normál kerettel küldhetünk és fogadhatunk. A CAN 2.0B támogatja a kibővített üzenetkeretet is, azaz ezzel normál és kibővített típusú üzeneteket is küldhetünk, fogadhatunk. A két keret közötti különbség, hogy a normál üzenetkeret 11 bites azonosítót használ, míg a kibővített 29 biteset.



2.3. ábra: A normál és kibővített üzenetkeret

Összehasonlítás

A járművek belső hálózata két csoportba osztható. Első csoportba tartoznak a személygépjárművek és a kis teljesítményű járművek, amelyeken a diagnosztizálás az ISO 15031-es szabvány szerint definiált, a másik csoportba a közepes és nagy teljesítményű járművek, melyek szabványa a SAE J1939. Ezt a két szabványt hasonlítom össze a diagnosztikai szabványok, a fizikai interfész, a csatlakozók, a protokoll és a hibakódok alapján. A fő különbség, hogy míg a SAE J1939 egyes részeiben definiált az összes szabvány a közepes és nagyteljesítményű járművekhez, addig a másik csoportban az ISO 15031 nem tartalmaz mindent és összhangban van néhány másik SAE szabvánnyal is. A protokoll szinten az ISO 15031 a CAN 2.0A-t, a J1939 a 2.0B-t alkalmazza, azaz az első 11 bites, a második 29 bites azonosítót használ. A J1939-et a jármű hálózatán normál kommunikációra és diagnosztizálásra is használják, viszont a másik szabványt csak diagnosztizálásra. Elsődleges funkcionálisága a J1939-nek a diagnosztikai üzenetek, amelyeket ciklikusan küldenek, az ISO 15031-nek az egyedi kommunikáció Service ID-kkel. Hibakódokat mindkét szabvány használ, de a J1939 még esemény számlálását is tartalmaz e mellett. Az ISO 15031 egy figyelmeztető lámpát határoz meg, míg a J1939 négyet.[3]

A következő táblázat szemlélteti a járművek két csoportját és a hozzájuk tartozó SAE és ISO diagnosztikai szabványokat.

1. táblázat

	SAE	ISO
Személyautók és kisteljesítményű járművek	J1930 – feltételek és definíciók J1962 – csatlakozó J1978 – vizsgáló eszköz J1979 – diagnosztikai szolgáltatások J2012 – hiba kódok J2186 – kapcsolat biztonság J2534 – pass-thru J1699 – OBD megfelelők	ISO11898 (5 rész) – CAN ISO15765 (4 rész) – diagnosztika CAN-en ISO15031 (7 rész) – OBD CAN-en
Közepes és nagyteljesítményű járművek	J1939 (összetett részek) J2403 – feltételek és definíciók	Nincs

Egyes esetekben több szabvány keveredik ugyanazon a járművön. A közepes és nagyteljesítményű járműveknél ez sokkal egyszerűbb, mivel azok kommunikációját jól átfogja a SAE J1939.

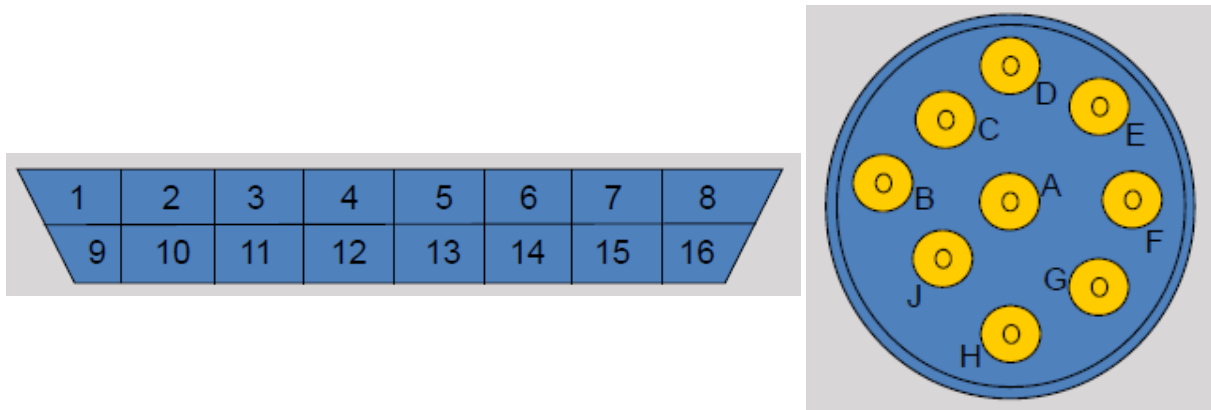
Az ISO 15031 egyes részei leképezhető SAE szabványokra a következőképpen:

- ISO 15031-2: J1930 feltételek és definíciók
- ISO 15031-3: J1962 diagnosztikai csatlakozó
- ISO 15031-4: J1978 diagnosztikai eszközzel szembeni elvárások
- ISO 15031-5: J1979 diagnosztikai szolgáltatások
- ISO 15031-6: J2012 hiba kód definíciók
- ISO 15031-7: J2186 adatkapcsolat biztonság

Fizikai szinten a SAE J1939 11 és 15-ös része alkotja az egyik csoportot, a másikat az ISO 15031-3, ISO 11898-2 és ISO 15765-4. A második csoport busz sebessége kétszer akkora, azaz a J1939 250 Kbps, a másinak 500 Kbps. A J1939/11-es szabványrész sodrott, árnyékolt érpárt határoz meg, míg a többi árnyékolatlant. A SAE szabvány

meghatározza a maximális ECU számot, a 11-esnél 30, a 15-ösnél pedig 10, ezzel szemben az ISO szabványoknál nincs meghatározva maximum.

A diagnosztikai csatlakozók is eltérőek a szabványoknál. A ISO 15031-3-ban 16 tűs a csatlakozó, amit a J1962 is definiál. A J1939/13 szabványrész 9 tűs csatlakozót használ diagnosztizálásra.



2.4. ábra: Diagnosztikai csatlakozók (bal oldalon az ISO 15031-3, jobb oldalon a SAE J1939-13 által meghatározott)

A CAN protokoll az alapja mindkét szabványnak, ezen belül az ISO 15031 a normál üzenetkeretet használja, azaz 11 bites azonosítót.

Az ISO 15031-ben az OBD kezeli az ECU-k azonosítását a következő funkciókra:

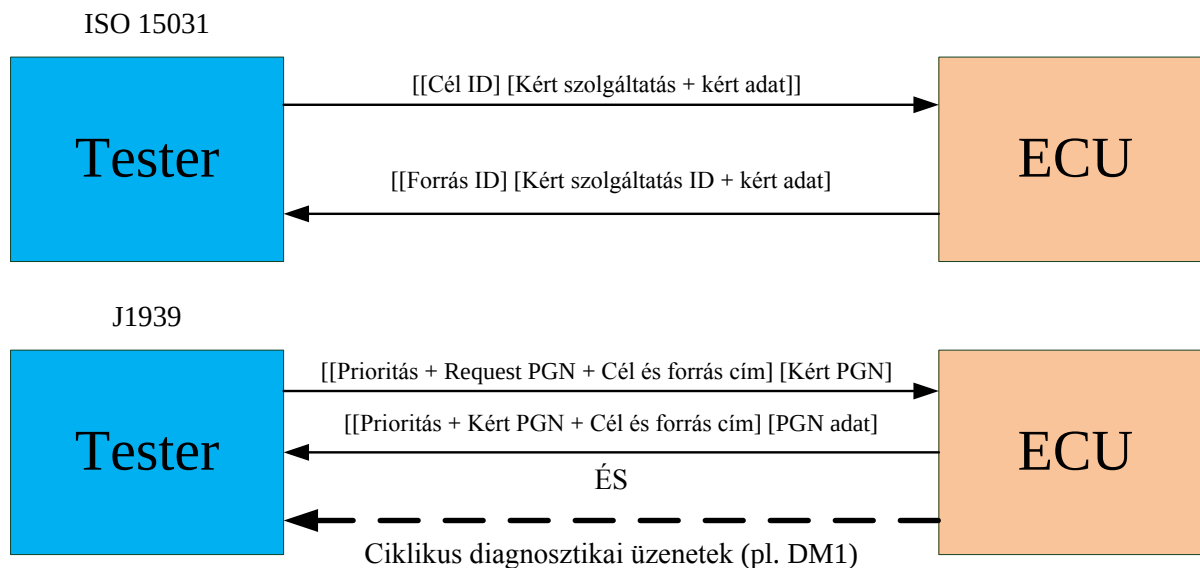
- OBD diagnosztikai kérésekhez funkcionális Request ID (forrás cím nem szükséges, mivel csak egy diagnosztikai teszter megengedett a buszon egy időben),
- forrás ECU azonosító a diagnosztikai válaszokhoz,
- a legtöbb OEM-nek (Original Equipment Manufacturer) van saját azonosító hozzárendelési szabványa.

A J1939 a kiterjesztett CAN üzenetformátumot alkalmazza (29 bites azonosító).

Három komponens, amit a szabvány definiál az azonosítóban:

- üzenet priorítás,
- Parameter Group Number (meghatározza az adatot a DATA területen – SAE szabványosított és tulajdonosi PGN-ek lehetségesek),
- forrás cím.

Diagnosztikai üzenetek struktúrájának összehasonlítása:



2.5. ábra: Az ISO 15031 és J1939 által definiált diagnosztikai folyamatok

A hibakódokat a J1939/73-as szabványrész definiálja. A DTC (Diagnostic Trouble Code) felépítése a következő ábrán látható.

DTC																															
Byte1 Low Byte SPN MSB <----->								Byte2 Mid Byte SPN MSB <----->								Byte3 SPN 3 MSB + FMI								Byte4 Conversion Method +							
SPN																FMI						C M	OC								
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1

2.6. ábra: A J1939 által meghatározott hibakód felépítése

Az ISO 15031 által meghatározott Diagnostic Trouble Code pedig a következő ábrán.

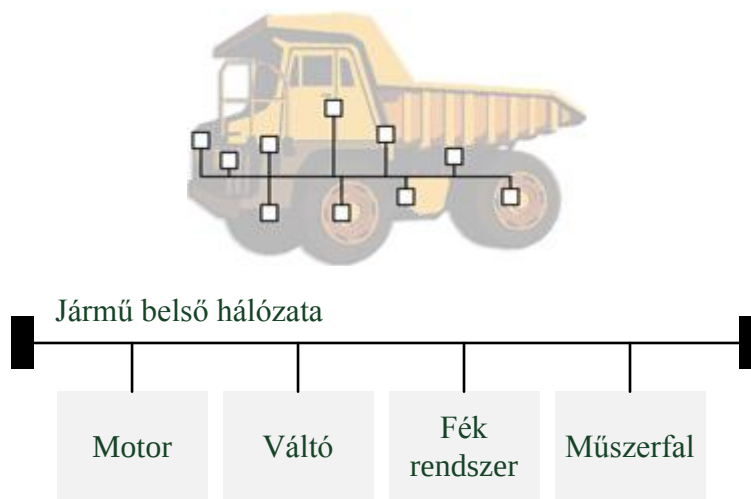
DTC																							
Byte 1								Byte 2								Byte 3							
SAE Code Number																FTB							
1.	2.	3.			4.			5.															
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1

2.7. ábra: Az ISO 15031-es szabványban definiált hibakód

Itt öt karakterből álló hibakódot határoz meg a szabvány. Az első és második karakter is két-két bit nagyságú, azaz 4 lehetőséget határoz meg, a harmadik, negyedik és ötödik karakterek 4 bit hosszúságúak, értékük 0-tól F-ig terjed. Az első karakter értéke P, C, B vagy U, a második karakter pedig 0, 1, 2, vagy 3.

3. Közepes és nagyteljesítményű járművek belső hálózata

A SAE J1939-et használják a haszongépjárművek területén, az azokban zajló kommunikációra. A SAE J1939 a CAN-t (ISO 11998) használja, mint fizikai réteg. Ez határozza meg, hogy milyen adatokat, valamint hogy azokat hogyan kell közölni az elektronikus vezérlőegységek (Electronic Control Unit – ECU) között a jármű hálózatán. Tipikus vezérlők a motor, fék, váltó, stb.



3.1. ábra: Tipikus J1939 jármű belső hálózat

Léteznek más szabványok is, amelyek a SAE J1939-ből származnak. Ezek a szabványok a SAE J1939 alapvető jellemzőit használják más és más paraméter csoportokkal és módosított fizikai rétegekkel. Ezek a szabványok a következők:[4]

ISO 11783 – Mezőgazdasági és erdészeti traktorok, gépek

Meghatározza a kommunikációt a traktor és a munkagépek között egy eszköz buszon. Részletez néhány szolgáltatást az alkalmazási rétegen, mint például virtuális terminál, traktor ECU, feladat vezérlő és fájl szerver. Hozzáfűz még egy kiterjesztett átviteli protokollt és munka készlet kezelést.

NMEA2000® - Tengerészeti elektronikus eszközök soros adat hálózata (National Marine Electronics Association)

Ez határozza meg a paraméter csoportokat a tengerészeti eszközök közötti kommunikációhoz. Specifikálja még a kiegészítő, gyors csomag alapú átviteli protokollt.

ISO 11992 – Digitális információ váltás a vontató és a vontatott jármű között (Truck-Trailer)

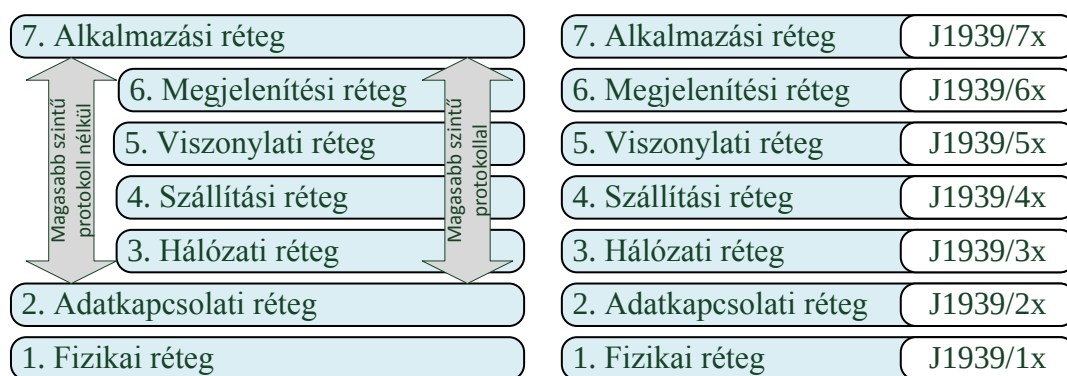
Meghatározza az információ cserét a közúti jármű és a vontatmány között. Ugyanazon paraméter csoportot használja mint a J1939, csak eltérő fizikai rétegen 125 kbit/s átviteli sebességgel.

FMS – Flotta menedzsment rendszer

Az FMS szabvány meghatároz egy átjárót a J1939 jármű hálózat és egy flotta menedzsment rendszer között.

A SAE J1939-et teherautók, kamionok és autóbuszok belső hálózata mellett dízel erőátviteli gépekben is alkalmazzák.

A következő ábrán látható az ISO/OSI 7-rétegű modellen alapulva a SAE J1939 szabvány csoport.



3.2. ábra: Az ISO-OSI modell és a SAE J1939 kapcsolata

A J1939 kihasználja a CAN legfontosabb jellemzőit, mint például a maximális megbízhatóság, kiváló hibaészlelés és feltárás, ütközésmentes busz arbitráció. [4,5]

A J1939 sajátos jellemzői:

- kiterjesztett CAN azonosító (29 bit),
- 250 kbit/s átviteli sebesség,
- peer-to-peer és broadcast kommunikáció,
- átviteli protokollok akár 1785 bájt adathoz,
- hálózat menedzsment,
- paraméter csoportok meghatározása haszongépjárművekhez és másokhoz,
- a gyártó specifikus paraméter csoportokat is támogatja,

- diagnosztikai funkciók,
- maximum 30 node (ECU) a hálózaton.

Paramétercsoportok

Egy paramétercsoport olyan paraméterek készlete, amelyek egyazon témához tartoznak és azonos az átviteli sebességük. A paramétercsoportok lényegesek az alkalmazások meghatározásához. A paramétereket megtalálhatjuk az alkalmazási réteg leírásában (SAE J1939-71).

A paramétercsoport hossza nem korlátozódik egy CAN keret hosszára. Általában egy paramétercsoport hossza 8 bájt, de akár maximum 1785 bájt is lehet. A több mint 8 bájtós paramétercsoportoknak szükségük van egy átviteli protokollra a továbbításhoz.

A CAN azonosító értelmezése

A J1939 üzenetnek a CAN azonosítója tartalmazza a paramétercsoport számát (továbbiakban PGN – Parameter Group Number), a forrás címet, a prioritást, két adatlap bitet (alap és kiterjesztett) és a cél címét (csak a peer-to-peer paraméter csoport esetén).

Az azonosító összetétele a 3.3. és 3.4. ábrán látható és az alábbiakból áll:

Az első 3 bit határozza meg a prioritást az arbitrációs folyamat közben. Ez 8 prioritási szintet határoz meg, a 0-s érték (000) a legmagasabb prioritású, míg a 8-as (111) a legalacsonyabb. Látható, hogy a 0-s érték a domináns ugyanúgy, mint ahogy fentebb bemutattam. A magas prioritású üzenetek az időkritikus adatoknak vannak fenntartva, mint például a forgatónyomaték vezérlő adat a váltótól a motorhoz. Az alacsony prioritásúak alkalmasak a nem időkritikus adatoknak, mint például a motor konfigurációs adatai.

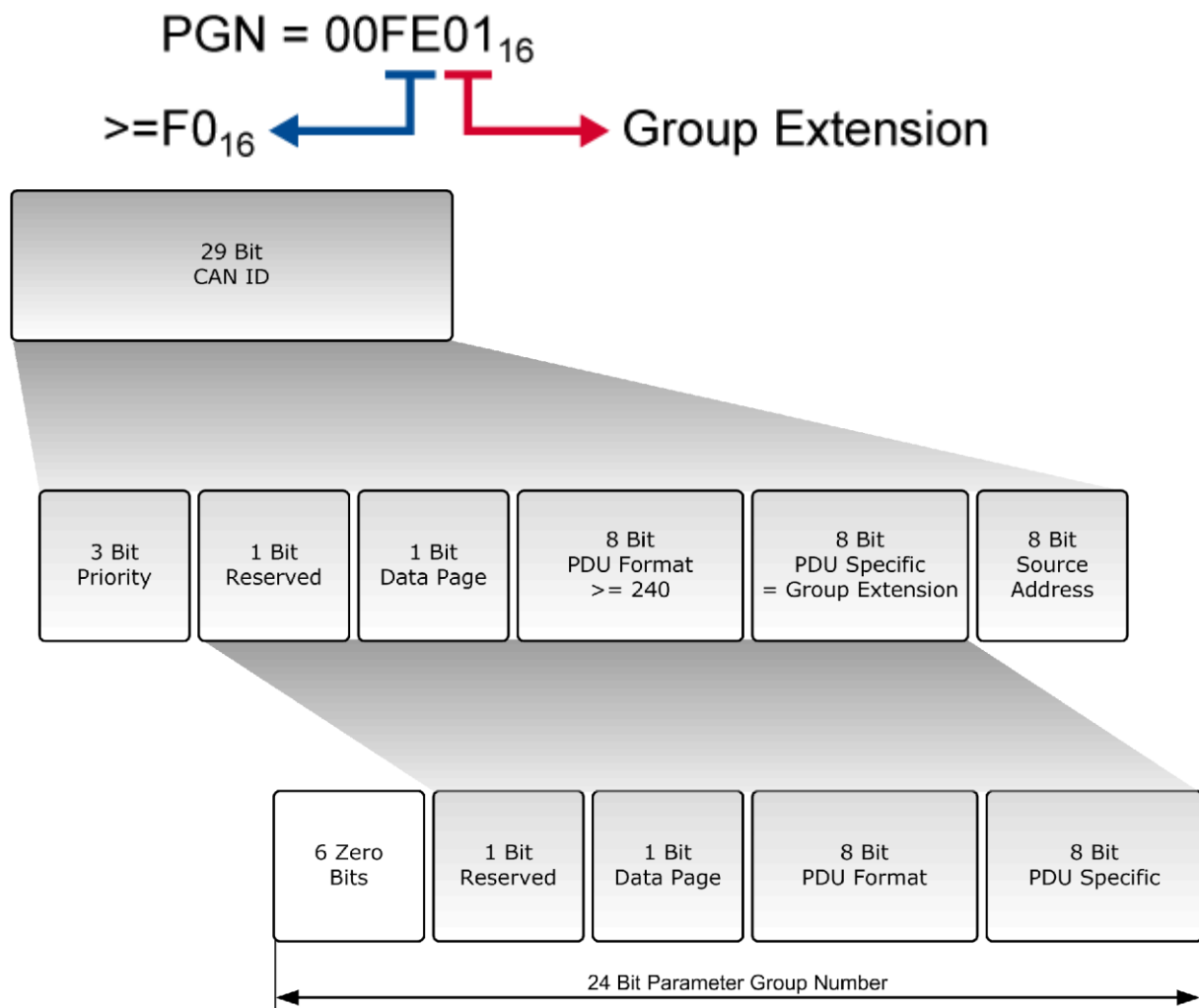
A PDU Format mező határozza meg, hogy az üzenet peer-to-peer vagy broadcast, a következőképpen:

- Ha a PDU Format kisebb 240-nél ($< 0xF0$), akkor peer-to-peer az üzenet és a PDU Specific-ben a célállomás címe (Destination Address) van. A Global (255) is felhasználható cél címnek, ekkor a paramétercsoport célja az összes eszköz.
- Ha a PDU Format nagyobb vagy egyenlő 240-nél ($\geq 0xF0$), akkor broadcast az üzenet és a PDU Specific a csoport kiterjesztése (Group Extension).

Minden paramétercsoportot egy egyedi szám révén címeztek meg, ez az egyedi szám a PGN. A PGN-hez egy 24 bites értéket használnak, ami 6 nulla értékű bitből, a PDU Format-ból (8 bit), a PDU Specific-ből (8 bit), a Data Page-ből (1 bit) és az Extended Data Page-ből (1 bit) tevődik össze.

Két típusa van a PGN számnak:

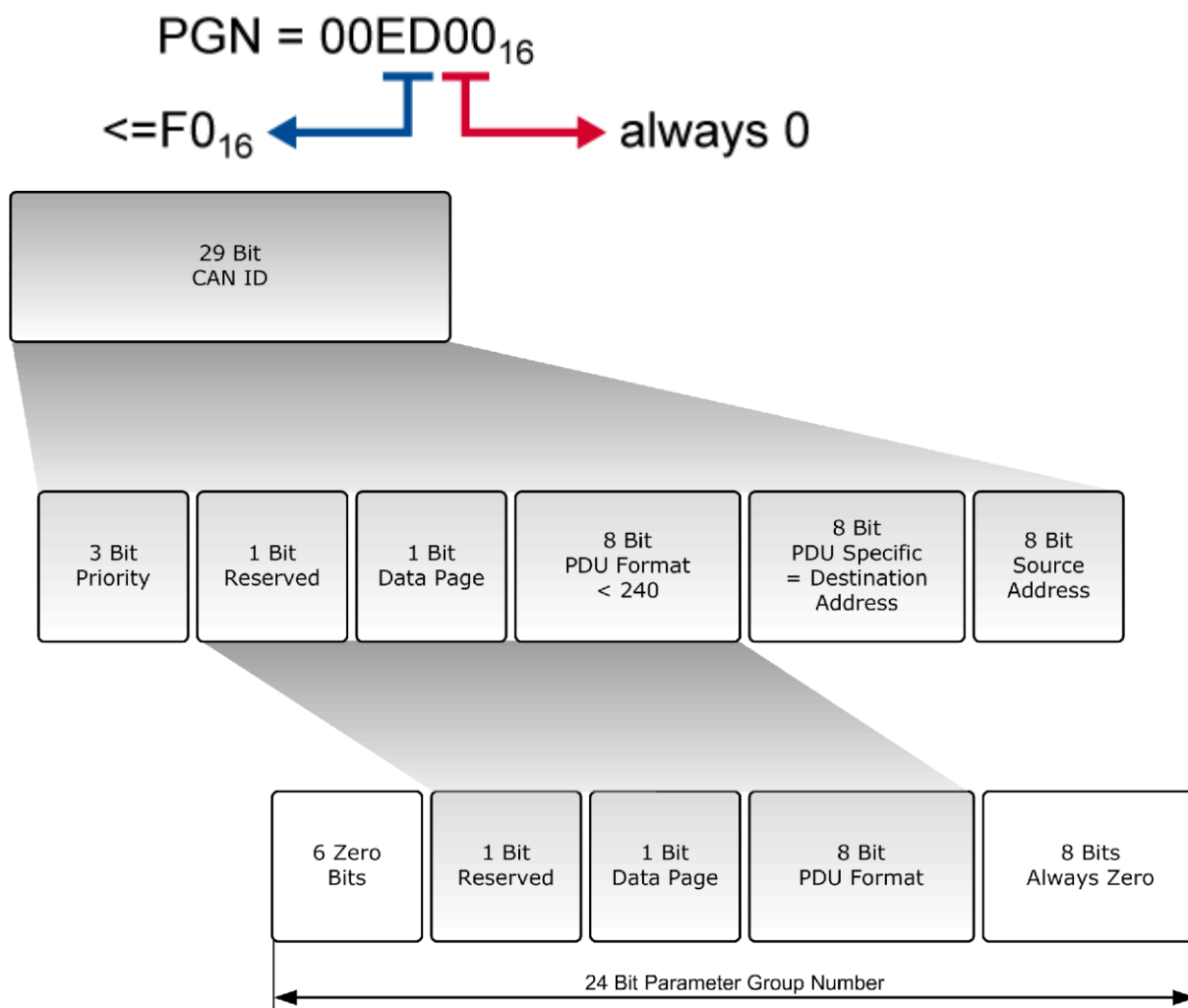
- Globális PGN-ek azonosítják azokat a paramétercsoportokat, amelyeket mindenkinek küldenek (broadcast). Itt a PDU Format, a PDU Specific, a Data Page és az Extended Data Page is használt a paramétercsoport megfelelő azonosításához. A globális PGN-eknél a PDU Format 240 vagy nagyobb és a PDU Specific mező a Group Extension.



3.3. ábra: A paramétercsoport száma a PDU1 szerint

- Egyedi PGN-eket csak a meghatározott eszközökhöz küldött paramétercsoportok alkotják (peer-to-peer). Itt csak a PDU Format, a Data

Page és a Extended Data Page használt a paramétercsoport megfelelő azonosításához. A PDU Format 240-nél kisebb és az utolsó 8 bit értéke 0.



3.4. ábra: A paramétercsoport száma a PDU2 szerint

Ezt a két csoportot szokás PDU1 és PDU2-nek nevezni, a PDU1 a peer-to-peer üzenet, a PDU2 a broadcast. Ezzel a PGN felbontással $240 + (16 * 256) = 4336$ különböző paramétercsoport lehetséges minden adatlapon belül.

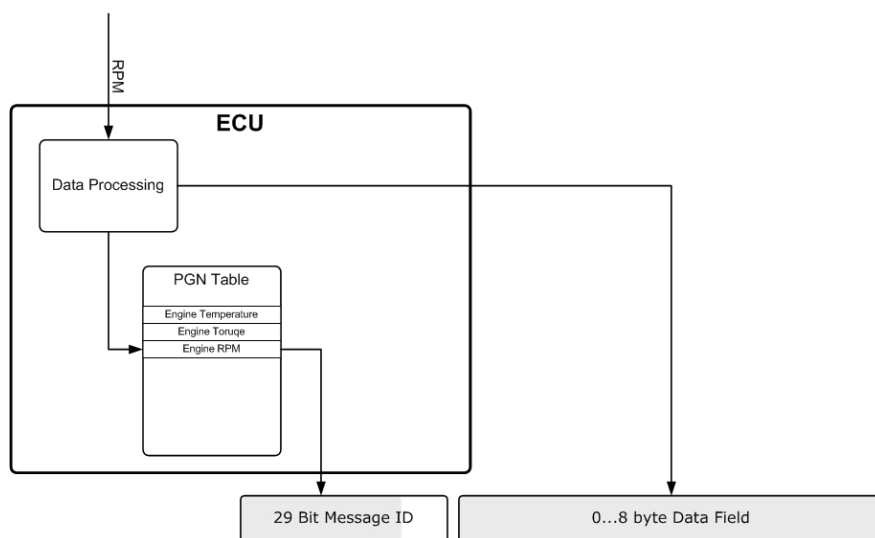
A Data Page bit és a Extended Data Page bit segítségével 4 különböző adatlapból lehet választani, ezt mutatja az alábbi táblázat.

Extended Data Page bit	Data Page bit	Leírás
0	0	SAE J1939 Page 0 Parameter Groups
0	1	SAE J1939 Page 1 Parameter Groups (NMEA2000®)
1	0	SAE J1939-nek fenntartott
1	1	ISO 15765-3 definiálja

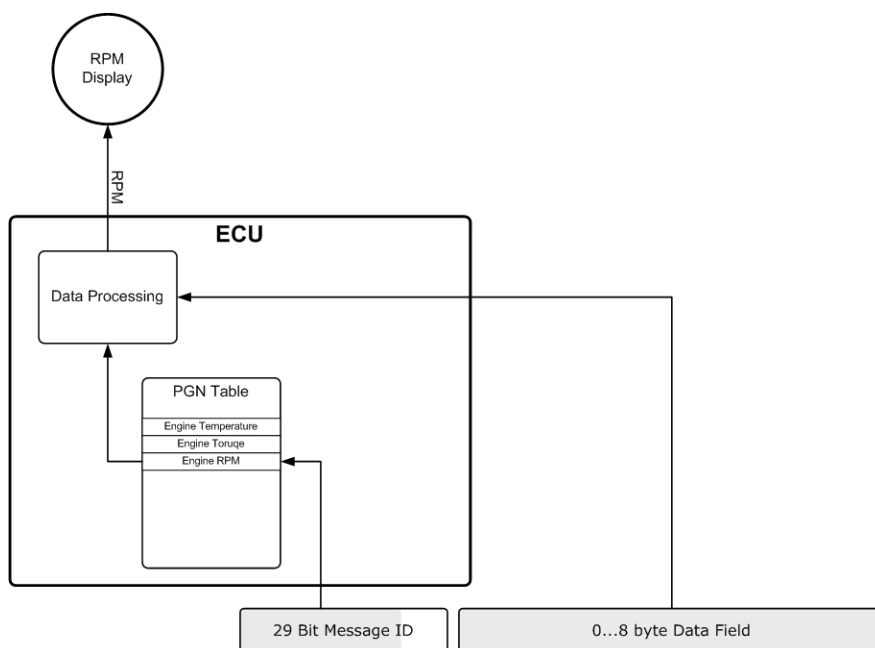
Az Extended Data Page a jelenlegi járművekben továbbított üzenetek esetén mindig 0, ezt a bitet későbbi célokra tartják fenn.

Az azonosító utolsó 8 bitje a forrás cím, az adó ECU (node) címe. 254 cím lehetséges a hálózaton és minden címnek egyedinek kell lennie. Az ECU-k nem tudják megosztani a címüket. A PGN-ek függetlenek a forrás címtől. Minden ECU adhat üzenetet. Itt jegyzem meg, hogy a CAN szabvány önmagában nem támogatja a node (ECU) címeket, csak az üzenet azonosítókat.

A 3.5. és 3.6. ábrákon egy-egy üzenet küldése és fogadása látható.



3.5. ábra: Példa az RPM küldésére



3.5. ábra: Példa a RPM fogadására

Suspect Parameter Number (SPN)

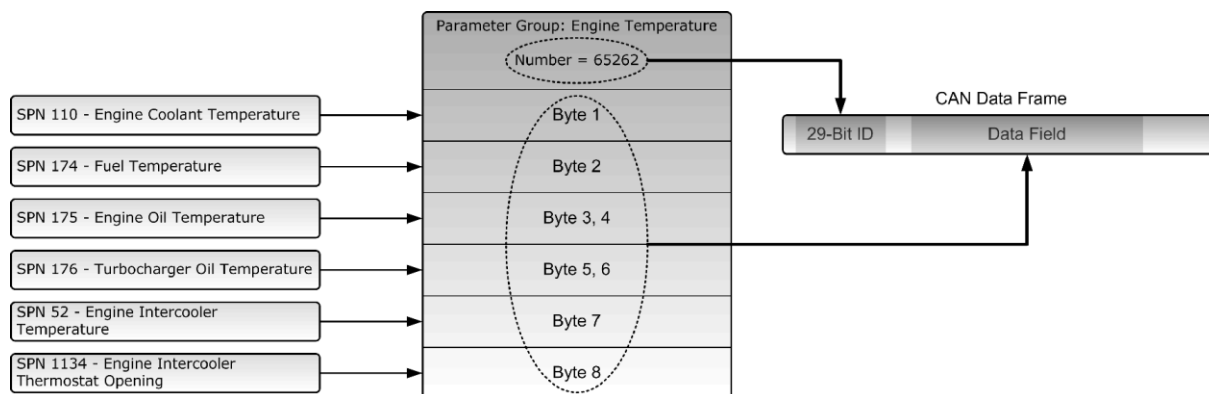
Egy paramétercsoportnak vagy összetevőnek minden egyes paraméteréhez hozzárendelnek egy SPN számot. Ezt diagnosztikai célból alkalmazzák egy vezérlő alkalmazás (CA – Controller Application) rendellenes működésének azonosítására és jelentésére. Az SPN egy 19 bites szám, ami 0-tól 524287-ig terjedhet. A gyártói paramétereknek 520192-től 524287-ig terjedő tartomány van fenntartva.

Egy paramétercsoport megadási példa:

Név	Motor hőmérséklet (Engine temperature 1- ET1)
Átviteli sebesség	1s
Adat hossz	8 bájt
Extended Data Page	0
Data Page	0
PDU Format	254
PDU Specific	238
Alap prioritás	6
PGN	65 262 (00FEEHex)

Az adat leírása:

Bájt	Leírás	SPN
1	Motor hűtőfolyadék hőmérséklet	110
2	Motor üzemanyag hőmérséklet	174
3,4	Motor olaj hőmérséklet	175
5,6	Motor turbofeltöltő olaj hőmérséklet	176
7	Motor intercooler hőmérséklet	52
8	Motor intercooler termosztát nyitva	1134



3.7. ábra: Az SPN, PGN és a CAN üzenetkeret kapcsolata

Egy SPN megadási példa:

SPN 110	Motor hűtőfolyadék hőmérséklet
Adat hossz	1 bájt
Resolution	1 °C/bit
Offset	-40 °C
Adattartomány	-40 – 210 °C
Típus	Mért
Reference	65 262 (00FEEE _{hex})

PGN tartományok

DP	PGN tartomány (hex)	PGN-ek száma	SAE vagy gyártó	Kommunikáció
0	000000 – 00EE00	239	SAE	PDU1
0	00EF00	1	GY	PDU1
0	00F000 – 00FEFF	3840	SAE	PDU2
0	00FF00 – 00FFFF	256	GY	PDU2
1	010000 – 01EE00	239	SAE	PDU1
1	01EF00	1	GY	PDU1
1	01F000 – 01FEFF	3840	SAE	PDU2
1	01FF00 – 01FFFF	256	GY	PDU2

SAE – SAE által kijelölt

GY – Gyártó specifikus, tulajdonosi üzenetek

Speciális paramétercsoportok

A SAE J1939-21 meghatároz néhány paramétercsoportot az adatkapcsolati rétegen:

a) Kérés paramétercsoport

A kérés paramétercsoport (RQST, PGN 00EA00_{hex}) elküldhető az összes vagy csak egy bizonyos CA-nak, hogy kérjen egy megadott paramétercsoportot. A RQST tartalmazza a kérni kívánt paramétercsoport PGN számát. Ha az adott kérelemnek a címzettje nem tud válaszolni, akkor küldenie kell egy negatív visszaigazolást. A RQST adat hossza 3 bájt és ez az egyetlen paramétercsoport 8 bájtnál kevesebb adathosszúsággal.

b) Visszaigazolás paramétercsoport

A visszaigazolás paramétercsoportot (ACKM, PGN 00E800_{hex}) arra lehet használni, hogy küldjön egy negatív vagy pozitív visszaigazolást, azaz választ a kérelemre.

c) Parancsolt cím paramétercsoport

A parancsolt cím paramétercsoportot (CA, PGN 00FED8_{hex}) a CA címének megváltoztatására lehet használni. Az adat mezőben meg kell adni az eszköz nevét és az új címét, ez összesen 9 bájt, amit egy üzenetben nem lehet elküldeni, ezért ez mindig átviteli protokollt használ.

d) Átviteli protokoll paramétercsoport

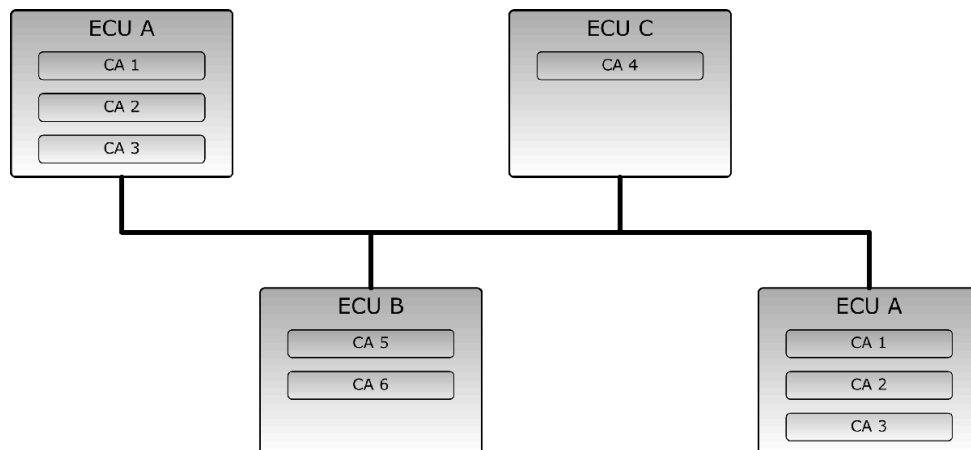
Az átviteli protokoll paramétercsoportot (TP-CM, PGN 00EC00_{hex} és TP-DT, PGN 00EB00_{hex}) használják, ha 8-nál több adat bájtot kell átvinni a paramétercsoportoknak. Működését később részletezem.

e) Gyártó specifikus paramétercsoportok

PDU1-et vagy PDU2-t is használhatnak, azaz lehet peer-to-peer vagy broadcast üzenet is. (Proprietary A, PGN 00EF00_{hex} és Proprietary B, PGN 00FF00-00FFFF_{hex})

Hálózat menedzsment

Egy Electronic Control Unit (ECU) szoftvere a Controller Application (CA). Egy ECU tartalmazhat egy vagy több CA-t is.



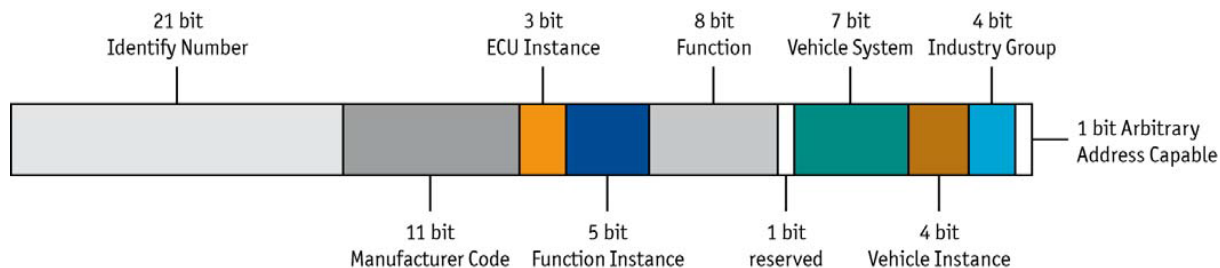
3.8. ábra: Az ECU-k és a CA-k kapcsolata

Minden egyes CA-nak van egy egyedülálló címe és egy kapcsolódó eszköz neve. Valamennyi üzenet, amit a CA elküld, tartalmazza ezt a forrás címet. A lehetséges címek a következők:

- 0..253 Érvényes forrás címek CA-knak.
- 0..127 és 248..253 Használható CA-knak előnyben részesített címmel és definiált függvényekkel.
- 128..247 Elérhető minden CA-nak.
- 254 Null.
- 255 Globális.

A legtöbb CA-nak, mint a motor, váltó, stb. van egy preferált címe. Mielőtt egy CA használhatna egy címet, be kell regisztrálnia magát a buszon. Ezt az eljárást nevezik address claiming-nek (cím igénylésnek). Eszerint az eszköz küld egy cím igénylő paramétercsoportot (ACL, PGN 00EE00_{hex}) a kívánt forrás címmel. Ez a paramétercsoport tartalmaz egy 64 bites eszköz nevet is.

Az eszköz név tartalmaz információkat a CA-ról és leírja a fő funkcióját. A gyártó kódot a SAE-től kell kérni. A mezők értékeit a SAE J1939-81 határozza meg.

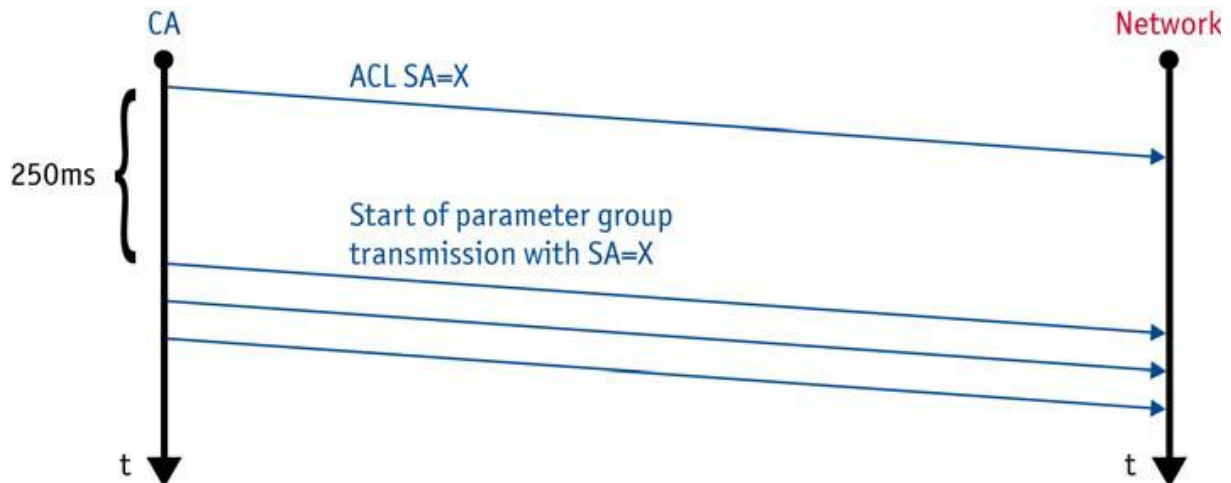


3.9. ábra: A cím igénylő paramétercsoport eszköz név adata

Kétféle cím igénylő folyamat van:

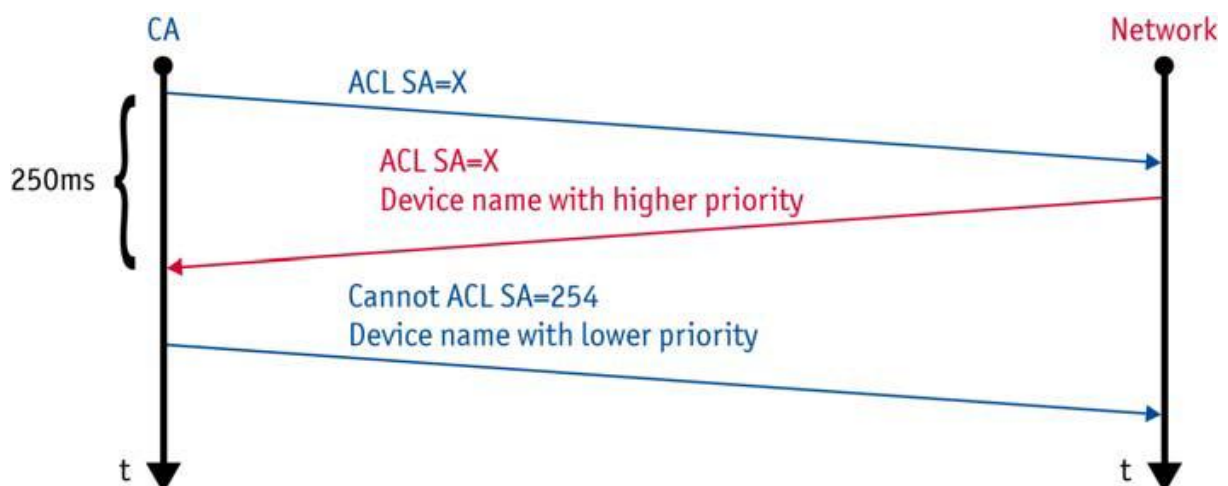
a) Cím igénylő üzenet küldése

Gyakori helyzet, hogy a CA küld egy cím igénylő paramétercsoportot az induláskor és vár egy meghatározott ideig. Ha nem érzékel cím ütközést, akkor egy megadott idő után el tudja indítani a normális kommunikációt. Az ECU-k fogadják a cím igénylést és feljegyzik azt a belső cím táblájukba.



3.10. ábra: A cím igénylő üzenet küldésének folyamata

Abban az esetben, ha egy másik CA már használja azt a címet, cím ütközés következik be. Amelyik CA-nak nagyobb prioritású az eszköz neve az kapja meg a címet. A másik CA-nak kell küldenie egy „Cannot Claim Address” paramétercsoportot, nullás (254) forrás címmel.



3.11. ábra: Cím igénylő folyamat cím ütközéssel

A CA képességétől függ, hogy hogyan jár el, ha egy cím nem szerezhető meg.

b) Kérelem cím igényléshez

Egy CA képes érzékelni más CA-kat a hálózaton a kérelmező ACL paramétercsoport által. Itt engedélyezett a Null cím (254) használata, mint forrás cím, mielőtt a CA elvégezné a cím igénylő folyamatot. Ha a kérelem címzettje a Global (255) cím, akkor a hálózaton minden CA-nak válaszolnia kell az ACL paramétercsoporttal (beleértve a saját CA-t is, ha egy címet igényelt már). Ez egy szükséges eljárás, ha egy ECU később kapcsolódik be (például diagnosztikai eszköz). Ezzel meghatározható, hogy mely címek szabadok és hogy milyen ECU-k vannak jelenleg a hálózaton.

Request for Address Claimed, PGN 59904



3.12. ábra: Kérelem cím igénylésre

Cím képesség

A SAE J1939-81 meghatározza a következő képesség típusokat egy cím megszerzéséhez:

a) Önkényes címzésre képes CA

Egy belső algoritmus által választ címet magának. Az ilyen CA-k képesek arra is, hogy új címet válasszanak egy cím ütközés során. Ezt a képességet az Arbitrary Address Capable mező jelzi az eszköz nevében.

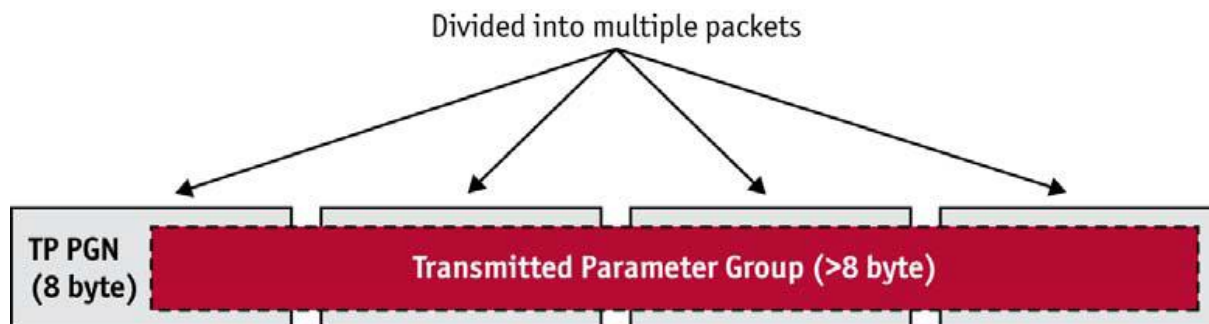
b) Egyszerű címzésre képes CA

Ezek a típusú CA-k csak egy címet használhatnak. Cím ütközéseknél nem tudnak másik címet választani. Ezek támogatják a parancsolt cím paramétercsoportot vagy

tulajdonosi mechanizmusokat a cím megváltoztatására. Az Arbitrary Address Capable mező ezeknél a CA-knál nincs beállítva.

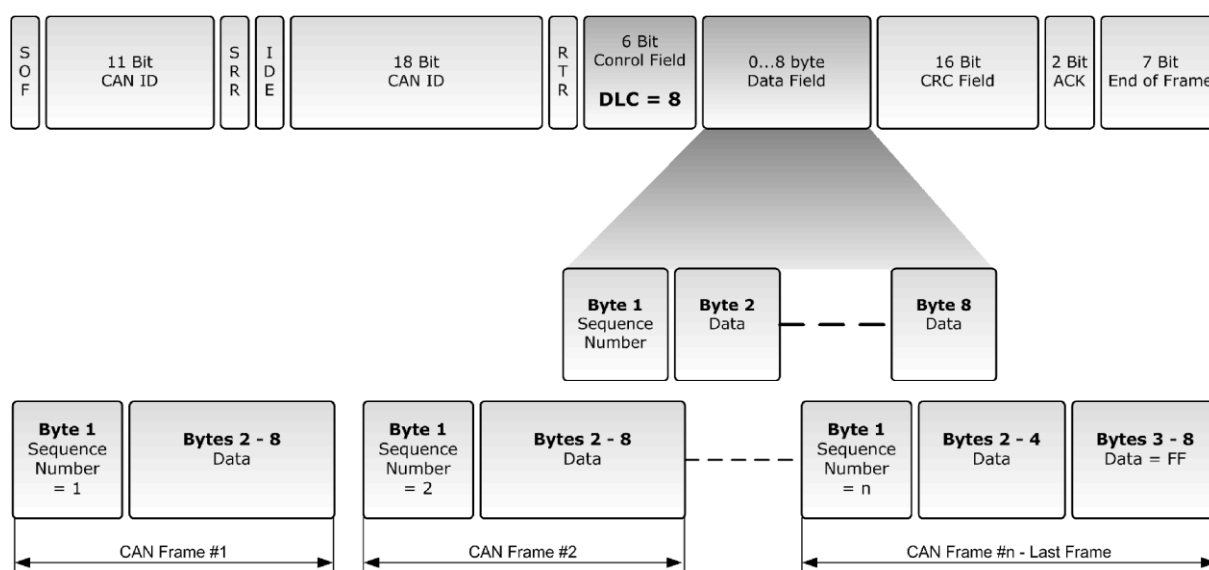
Átviteli protollok

Több mint 8 adat bájtt továbbítása átviteli protokoll segítségével történik.



3.13. ábra: Átviteli protokoll

A peer-to-peer és broadcast kommunikációhoz két különböző protokoll van. Az átviteli protollok (Transport Protocol – TP) felhasználnak két speciális paramétercsoportot, amelyek a kapcsolatmenedzsment (Connection Management – TP.CM) és az adat továbbítás (Data Transmission – TP.DT).



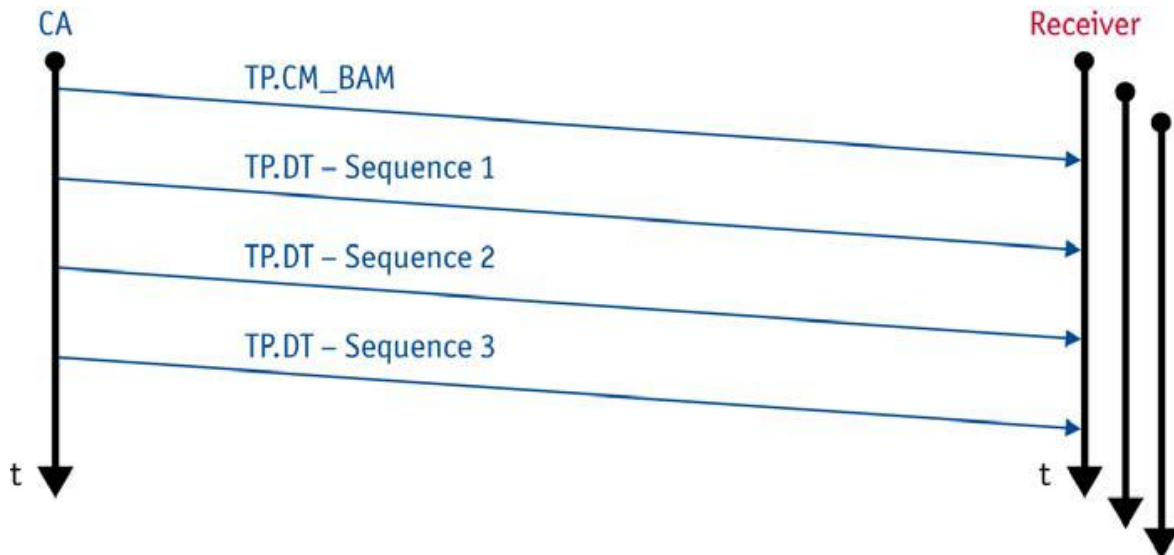
3.14. ábra: A több mint 8 bájtt adat felbontása

Broadcast továbbításhoz a BAM (Broadcast Announce Message) protokollt használják. Itt egy BAM paramétercsoport után, az adó küldi az összes adatot legalább 50 ms időközönként. A BAM üzenet a következő komponenseket tartalmazza:

- a multi-packet üzenet PGN száma,
- a multi-packet üzenet mérete,

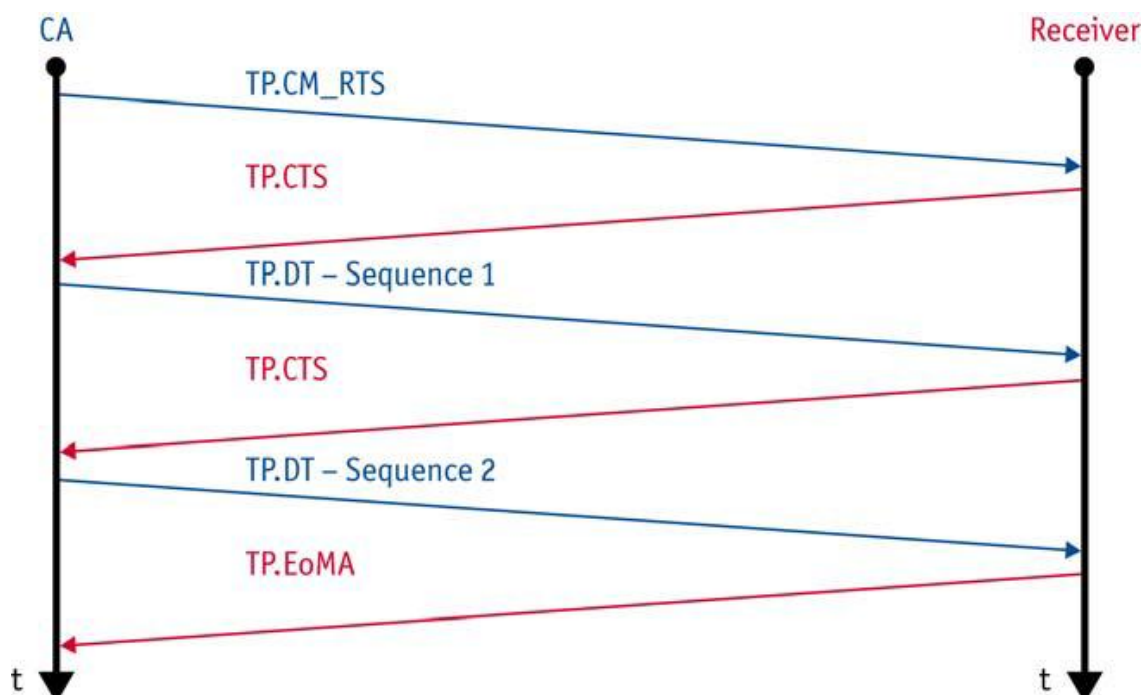
- a csomagok száma.

Az első üzenet a kapcsolatmenedzsment paramétercsoport (TP.CM) szerint küldi, majd az aktuális adatok az adat továbbítás (TP.DT) szerint.



3.15. ábra: BAM továbbítás

Peer-to-peer átvitelnél az adó kezdeményezi a kapcsolatot egy „Request To Send” üzenettel. Ezután a vevő vezérli az átviteli protokollt a „Clear To Send” üzenettel és végül visszaigazolja azt az „End of Message Acknowledge” üzenettel. Ezek az üzenetek a kapcsolatmenedzsment paramétercsoportba tartoznak.



3.16. ábra: RTS/CTS átvitel

Diagnosztika

A SAE J1939 diagnosztikai funkciója a következő szolgáltatásokat tartalmazza:

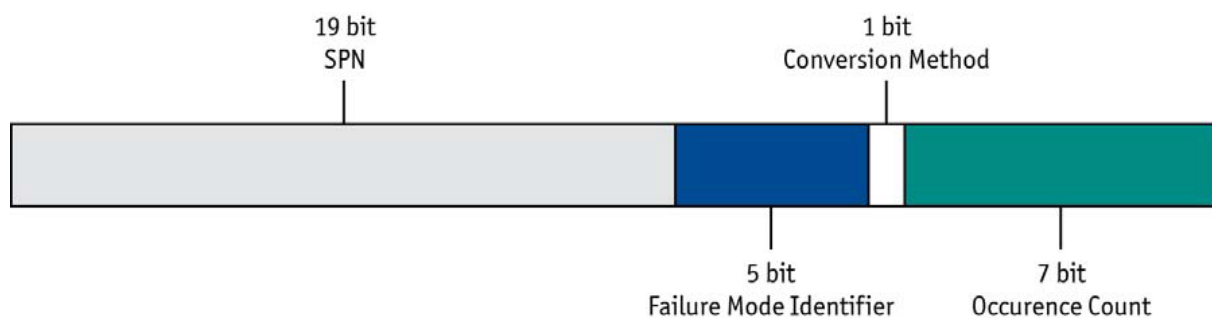
- A rendellenes működés jelentése és azonosítása.
- Memória hozzáférés.
- Felügyeleti tesztek.

Fontos paramétercsoport a diagnosztikai üzenetek 1 (DM1, PGN FECA₁₆). Ha ez támogatott, akkor minden egyes CA ciklikusan elküldi a saját állapotát. A paramétercsoport tartalmaz állapotokat a következő lámpákhoz:

- Üzemzavar jelző lámpa.
- Piros féklámpa.
- Sárga figyelmeztető lámpa.
- Védelmező lámpa.

A műszeregységek arra használhatják ezt az információt, hogy jelentsék a rendszer állapotát a vezető számára.

Továbbá a paramétercsoport tartalmazza a diagnosztikai hibakódokat (DTC). Együtt a feladó címével a hibásan működő komponenseket is lehet azonosítani.



3.17. ábra: Diagnosztikai hibakód

A DTC 4 bájtot tartalmaz, amely magában foglalja az SPN-t, a Failure Mode Identifier-t (FMI) és egy Occurrence Count-ot. Ha a DM1 egynél több hibakódot tartalmaz, akkor átviteli protokollt használ.

A diagnosztikai csatlakozót a J1939/13 szabványrész határozza meg. Ez a 9 tűs, kerek csatlakozó az előző fejezet 2.4. ábrájának jobb oldalán látható.

A csatlakozó lábkiosztása [6]:

A – Akkumulátor negatív

B – Akkumulátor pozitív

C – J1939 CAN-H (+)

D – J1939 CAN-L (-)

E – CAN árnyékolás

F – J1587 high (+)

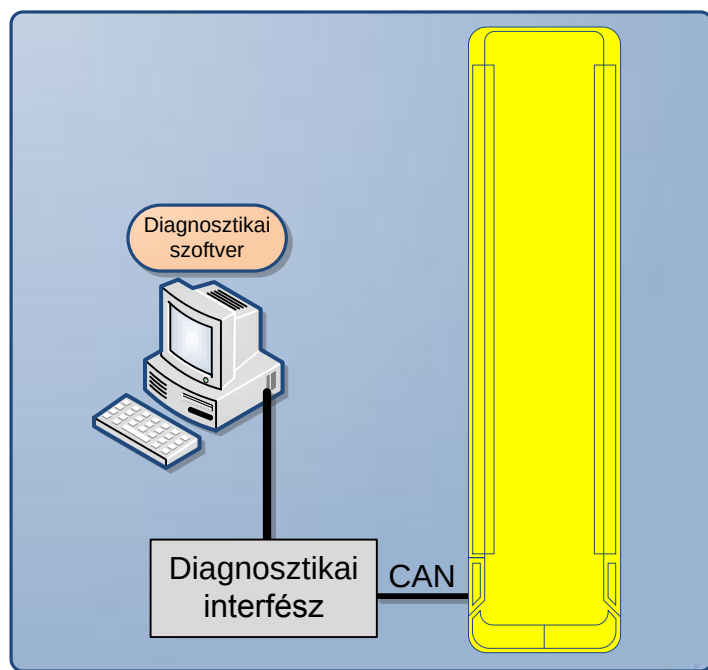
G – J1587 low (-)

A paramétercsoportok leírása, a CAN azonosító séma és a hálózat menedzsment együtt a vészérlőegységek a gyártókra kiterjedő együttműködését biztosítja.

A J1939 az itt bemutatott mechanizmusok mellett leírja a fizikai tulajdonságokat és a buszon lévő szegmensek használatát is.

4. Autóbuszokon végzett mérések

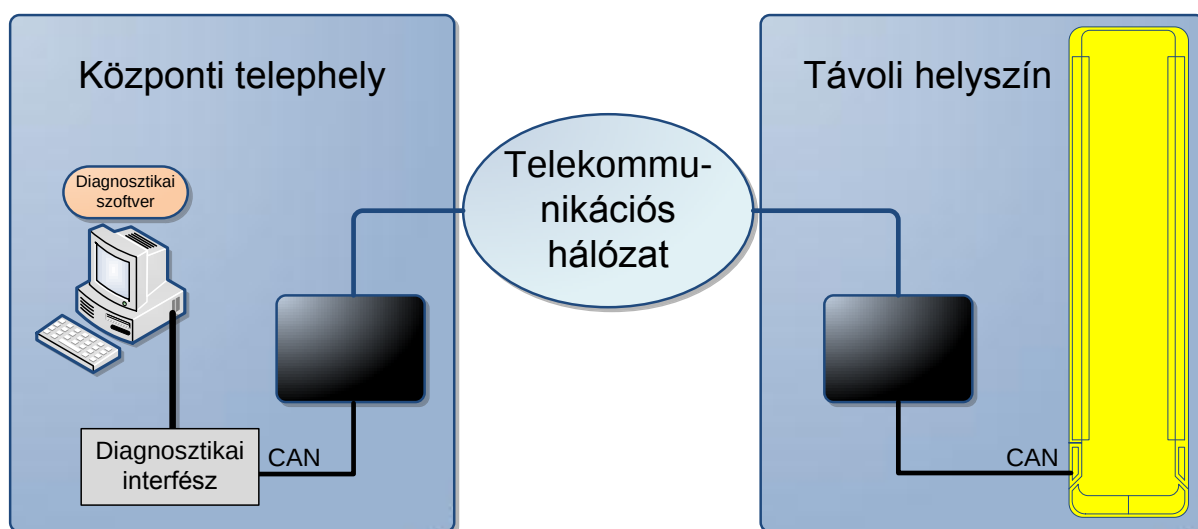
Először a diagnosztizálás folyamatával kell tisztába lenni, ezt már gyakorlott szakemberek mutatták meg nekem. A diagnosztizáláskor a jármű belső hálózatára csatlakozik egy diagnosztikai interfész. Az általam vizsgált autóbuszok esetében a sofőr ülése mellett helyezkedik el a diagnosztikai csatlakozó egy lecsavarozható védőpanel mögött. A diagnosztikai interfész egy számítógéphez csatlakozik, régebbi eszközök a soros port-ra, de már vannak újabb USB-s eszközök is. A számítógépen egy diagnosztikai szoftver fut, amely fogadja az adatokat a diagnosztikai eszköztől a megfelelő port-on keresztül. Nem ritka, hogy az USB-n is virtuális soros port-ot hoznak létre, ugyanis ilyenkor a diagnosztikai szoftvert nem kell lecserélni.



4.1. ábra: A diagnosztizálás folyamata

Az ábrán szemléltetem a hagyományos diagnosztizálást. Ezt telekommunikációs eszközökkel támogatva kapjuk a távdiagnosztizálást. A menet közbeni diagnosztizálási lehetőség mellett erre azért van szükség, mert az autóbuszok nem mindig egy telephelyre érkeznek vissza és nincs minden telephelyen diagnosztikai eszköz. A távdiagnosztizálással egy központi telephelyről monitorozható lenne az összes jármű, ezzel a karbantartási lehetőségek növekednének, és a hibás működés megelőzhető.

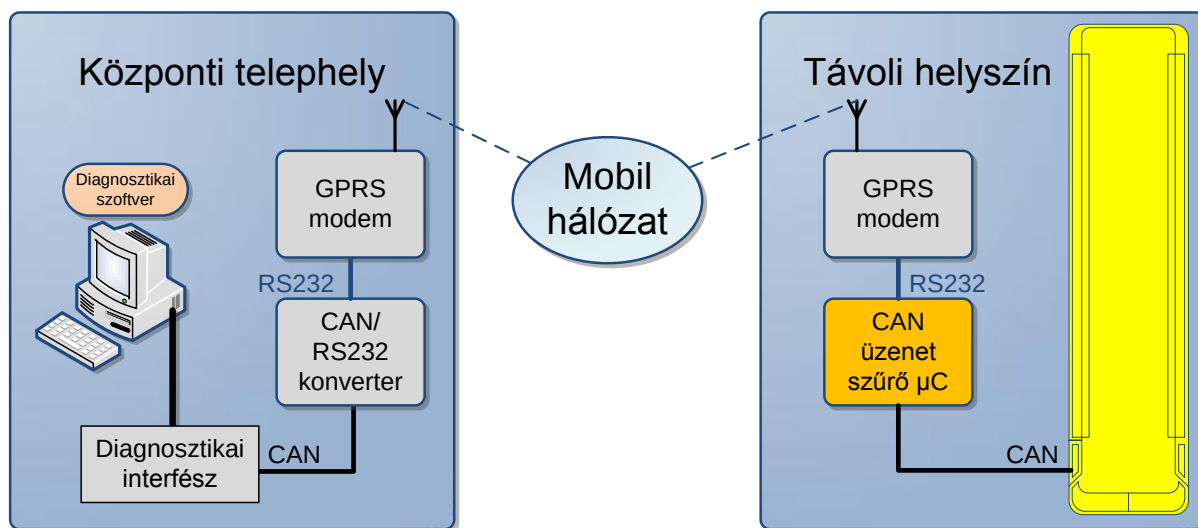
lenne. Az esetleges meghibásodást is egyből érzékelni tudná a rendszer, így a legközelebbi szervizközpontot értesítve gyorsabb lenne a javítási művelet.



4.2. ábra: Távdiagnosztika

Az előző ábrán is látható, hogy az eredeti diagnosztikai interfész és szoftver használatával szeretném megoldani a távdiagnosztizálást, ugyanis ezek az eszközök és szoftverek elég drágák. A távdiagnosztizáláshoz használt kommunikációt jelölik az ábrán látható fekete téglalapok. A telekommunikációs hálózatnak mindenképpen vezeték nélkülinek kell lennie. A legfontosabb követelmény a lefedettség biztosítása a közlekedésben, ennek már több hálózat nem tesz eleget. Teljes lefedettséget biztosít a műholdas adatátvitel, ami nagy távolságot és sebességet biztosít, ez azonban több szempontból sem alkalmazható távdiagnosztizáláshoz. A kiépítése költséges, jelentős eszköz és szerelés igényel jár, például külső antenna felszerelését követeli meg. Emellett hátránya még, hogy a kapcsolat minősége kevésbé kiszámítható, erősen függ az időjárástól is. Hazánkban csak a GSM hálózat felel meg az elvárásoknak, ami 99 százalékos lefedettséget biztosít. GSM hálózaton adatátvitelt adathívással és GPRS technológiával lehet megoldani. Az adathívás alacsony átviteli sebessége miatt nem alkalmazható. A GPRS már magasabb sebességet biztosít, de ez még mindig jóval kisebb, mint amire a jármű CAN hálózata képes. Ez alapján belátható, hogy nem minden adat vihető át, mindenképpen valahogy csökkenteni kell az adatmennyiséget. Napjainkban rohamosan fejlődnek a szélessávú mobilinternet szolgáltatások (HSPA, LTE), ám ezek még csak a nagyvárosokban és azok környékén érhetőek el. Ezen

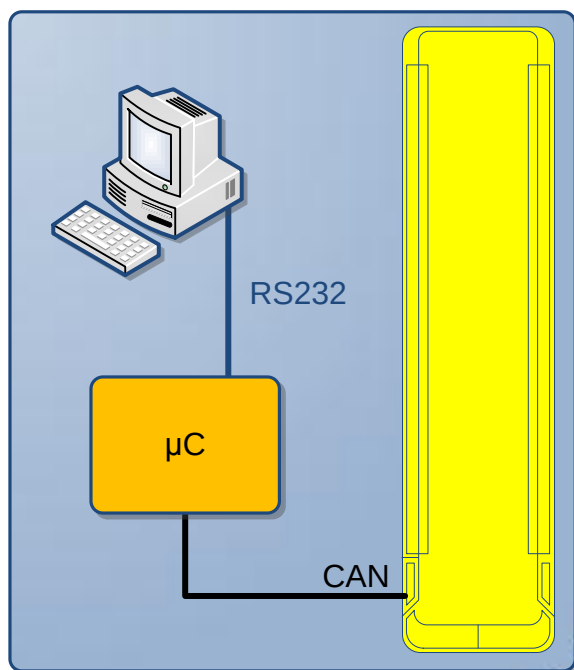
hálózatok teljes kiépítése után lehetséges, hogy minden CAN adatot át tudnánk küldeni rajtuk, de mivel ezeken a hálózatokon adatkorlát van, így ekkor is célszerű lenne csökkenteni a nagy adatmennyiséget. A három legnagyobb magyarországi mobil szolgáltató GPRS és 3G-s lefedettség térképe a 8.1. sz. Mellékletben található.



4.3. ábra: GSM hálózaton keresztüli távdiagnosztizálás

Az előző ábrán szemléltetem az általam tervezett távdiagnosztikai rendszert. Látható, hogy két GPRS modem segítségével kommunikálna egy jármű és a diagnosztikai eszköz. Ez azt jelenti, hogy az adatokat transzparensen átviszi a két modem. A további vizsgálatok során nem veszem bele a GPRS kapcsolatot, ugyanis ha RS-232-n keresztül működik a diagnosztizálás, akkor már csak a két GPRS modemet kell felkonfigurálni. Szintén észrevehető az ábrán, hogy a központi telephely oldalán, azaz ahol a diagnosztikai eszköz van, ott egy CAN/RS-232-es konverter alakítja át az RS-232-es adatokat CAN adatokká és visszafelé. A diagnosztikai eszköz nem küld olyan sok adatot (ezt a megállapítást már a mérések elvégzése után jelenthetem ki), hogy azt csökkenteni kellene, ezért itt elég egy átalakító. A jármű oldalán található mikrokontrollernek kell elvégeznie az adatok szűrését és úgy kiküldeni azokat, hogy a CAN/RS232-es konverter azt tudja értelmezni és továbbítani. Belátható, hogy ha ismerném a diagnosztikai interfész pontos működését, akkor nem kellene visszaalakítani az RS-232-es adatokat CAN adatokká, hanem egyből küldhető lenne a diagnosztizálást végző számítógépnek.

Az adatmennyiség csökkentése érdekében szűrési eljárásokat fejlesztettem ki. Ezek elkészítése előtt mindenképpen meg kellett ismerni alaposan a jármű hálózatának a működését. A megismeréshez méréseket végeztem több autóbussen az alábbi elrendezésben.



4.4. ábra: Az elvégzett mérések elrendezése

Mivel a jármű hálózatának sebessége nagyobb, mint amit az RS-232 segítségével el tudunk érni, ezért az a lehetőség nem megoldható, hogy minden adatot átküldünk a számítógépnek és csak ott dolgozzuk fel. Mindenképpen valamilyen programozható vezérlőt kell használni, amely képes a CAN adatokat kezelni. A mérésekhez egy olyan mikrokontrollert használtam, amely támogatja a CAN 2.0A-t és 2.0B-t is, valamint a CAN szabványos funkcióit, az újraküldést, a busz arbitrációt, illetve a hibafelismerést is. Továbbá az eszköz rendelkezik RS-232, RS-485 és Ethernet interfészekkel, így a programozhatóságának köszönhetően számos ipari kommunikációs probléma oldható meg a segítségével. Az eszközben található egy beépített 120 Ω -os lezáró ellenállás, melyet szükség esetén ki-be lehet kapcsolni egy jumper segítségével. Ennek a bekapcsolására például akkor volt szükség, amikor labor körülmények között létrehozott CAN hálózaton teszteltem a megírt programot. Viszont amikor kint a terepen próbáltam ki az eszközt, akkor ott egy már létező CAN-buszra csatlakozott,

amelyet már elláttak a szükséges lezáró ellenállásokkal. Ilyenkor értelemszerűen ki kell kapcsolni az ellenállást, ellenkező esetben nagy eséllyel hibás működés tapasztalható.

Az általam választott eszközön egy beágyazott operációs rendszer fut, ennek egyik nagy előnye, hogy nem szükséges alacsony szinten programozni az eszközt, hanem a már hozzá elkészített függvénykönyvtárak használatával, magasabb szinten tehetjük meg azt.

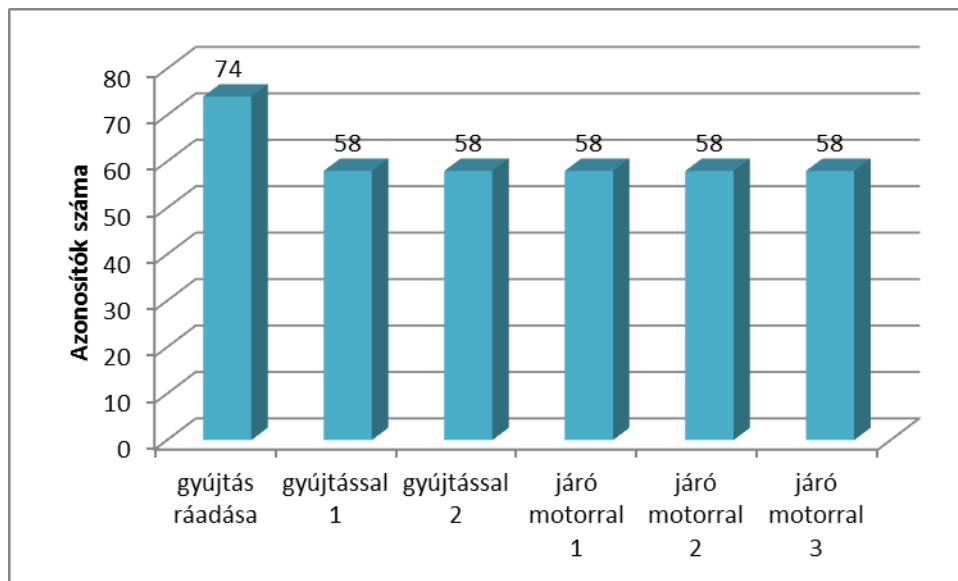
Az első mérési funkció, amelyet elkészítettem az azonosító lista. Ez az érkező üzenetek azonosítóit vizsgálja meg. Minden egyes új üzenet beérkezésekor megvizsgálja, hogy az azonosító szerepel-e a listában, ha nem akkor beteszi azt és az azonosító darabszámát 1-re állítja. Ha az azonosító benne van a listában, akkor a darabszámát növeli eggyel. A funkció elindulása előtt meg kell adni, hogy hány másodpercig fogadja az üzeneteket, így azonos idejű mérések végezhetők. Ez a funkció megadja, hogy milyen típusú üzenetek milyen gyakorisággal továbbítódnak a buszon. Ha minden azonosító darabszámát összeadjuk, megkapjuk a megadott idő alatt küldött üzenetek összes darabszámát, ezzel az üzenetek egységnyi időre vonatkoztatott számát számolhatjuk ki, ami hasznos különböző járműveken végzett mérések összehasonlításánál.

A második mérési funkció az adat szűrés, ekkor csak a megadott azonosítót tartalmazó üzeneteket küldi tovább az eszköz a számítógépnek. A monitorozni kívánt azonosítókat egymás után hexadecimális formátumban kell megadni, ezután elindul a mérés. Ez a funkció azért hasznos, mert menetközben tudjuk figyelni az egyes paraméterek változását. A vezérlőtől kapott adatokat a számítógépen egyszerűbben ki lehet értékelni.

Több autóbuszon végeztem mérést, azok különféle állapotaiban. Három állapotot különböztettem meg:

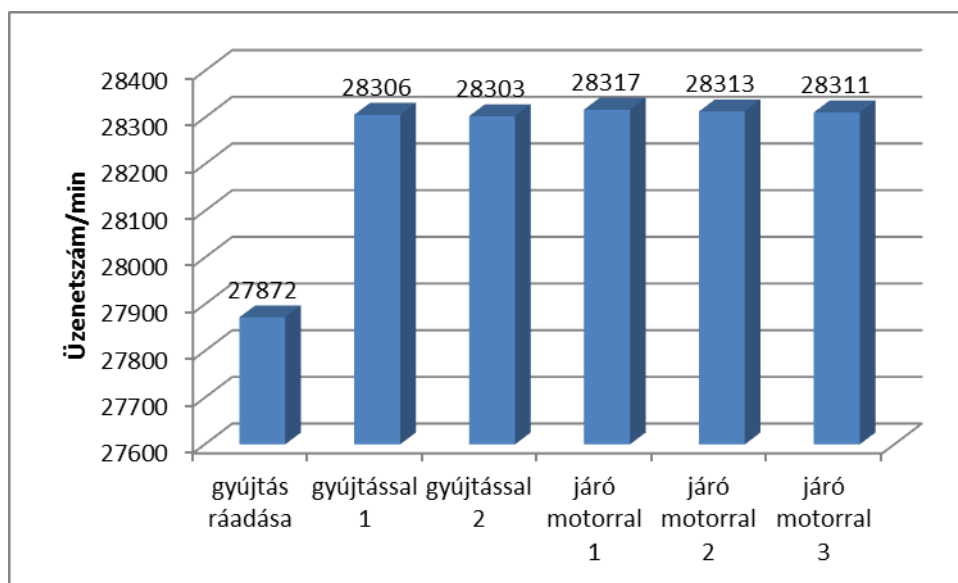
- a) Gyűjtással, állandósult állapotban, azaz amikor a járművön már körülbelül fél perce rajta van a gyűjtás, akkor futtattam le az azonosító lista funkciót,
- b) járó motorral, azaz amikor a motort be is indítjuk,

c) gyújtás ráadása előtt elindított mérés. Ez azért fontos, mert ekkor láthatjuk a cím igénylő üzeneteket is.



4.5. ábra: Az első autóbuszon végzett hat datab azonosító lista mérés

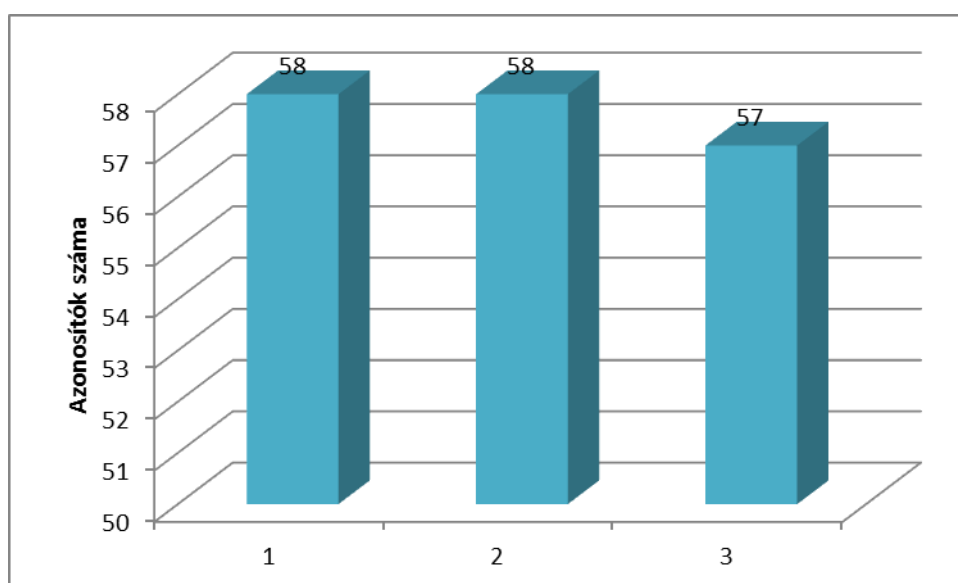
Már az első autóbuszon elvégzett mérések után észrevehető, hogy az a) és b) állapot között nincs különbség, azaz a motor beindítása nem generál több üzenetet és másféle azonosítók sem keletkeznek. A továbbiakban ezt alapállapotnak nevezem. Az öt mérésnél 15 üzenet volt a legnagyobb különbség a percenkénti üzenetszámok között, ez a kis számú üzenetszám változás abból adódik, hogy az egyes üzenetek más-más időközönként továbbítódnak a jármű hálózatán és így nem lehetséges két ugyanolyan üzenetszámú mérést végezni.



4.6. ábra: Hat darab egy perces mérés eredményei az első autóbussen

A diagramokon látható első értékek kilógnak a sorból. Ezeket a fentebb ismertetett c) típusú mérésnél tapasztaltam. Az azonosítók száma a cím igénylő üzenetek miatt több, mint amit az alapállapotban észleltem. Az üzenetszám nem mérhető ennél a mérésnél, ugyanis a mérés elindítása után adtam rá a gyújtást az autóbussen és a vezérlő egy ideig nem kapott üzeneteket. Egy időben ráadni a gyújtást és elindítani a mérést kézzel nem lehetséges, mert ezek az üzenetek 10 – 100 ms-onként továbbítódnak.

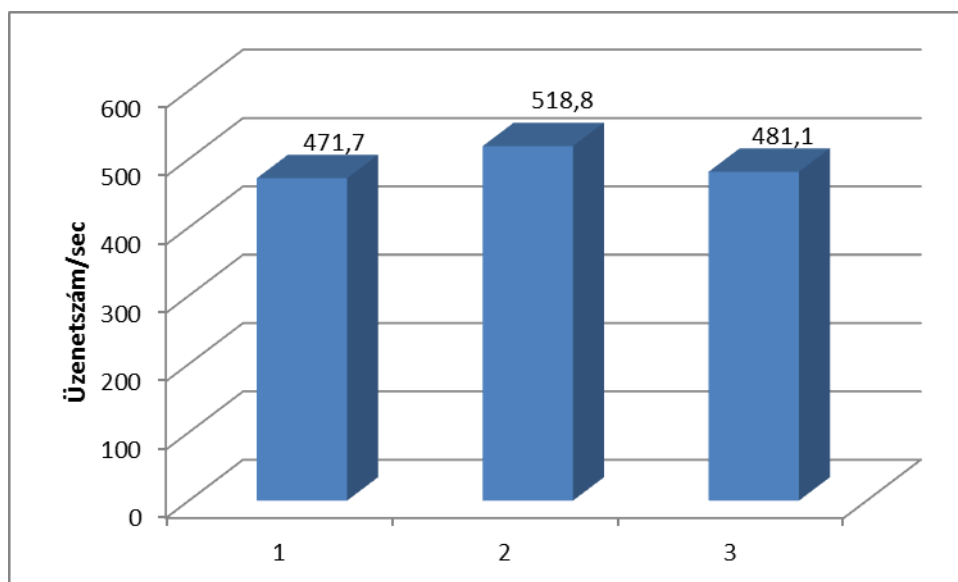
Az alábbi diagramon három különböző autóbussen végzett alapállapotú mérési eredmény látható.



4.7. ábra: Három különböző autóbussen az azonosítók száma

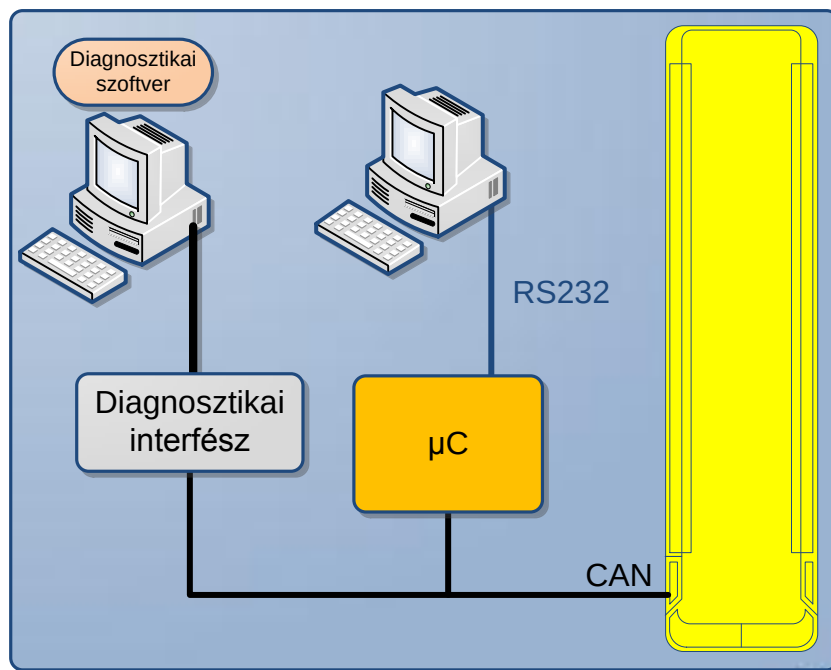
Az elvégzett mérések alapján megállapítható, hogy az egyes autóbuszok között azonos állapotban hasonló eredményeket értem el. Az autóbuszok közötti különbségek az eltérő paraméterekből és az esetleg eltérő ECU-kból adódik. Előfordulhat, hogy egy meghibásodott ECU-t kicseréltek egy újabbra és így az más paramétercsoportokat küld a hálózatra. Az általam vizsgált autóbuszok valamelyikén aktív hibakódok is voltak, így azok is eredményezhetik a változást. A diagramon látható, hogy az első és második autóbuszon egyaránt 58 féle azonosító fordult elő. Ez nem azt jelenti, hogy ezeken ugyanolyan kommunikáció van, mert az 58 azonosítóból csak 57 ugyanolyan, volt 1-1 azonosító, amelyek csak az egyikben fordultak elő.

Különböző ideig tartó méréseket végeztem, így az egyes autóbuszok összehasonlításához kiszámoltam az egy másodperc alatt érkező üzenetek számát.



4.8. ábra: Három különböző autóbuszon a másodpercenkénti üzenetszám

Ez az üzenetszám eltérés is abból származik, hogy különböző azonosítókat használnak az egyes járművek. Vannak olyan azonosítót tartalmazó üzenetek, amelyek mindig a szabvány által meghatározott időközönként továbbítódnak, viszont vannak olyanok is, amelyeket az adott járműn található ECU határoz meg. Így az egyes üzenetek küldési gyakorisága eltérő, ezáltal változik az egyes járművek esetében az üzenetszám értéke. A következő méréseknél a CAN busz topológiáját kihasználva egyszerre csatlakoztattam a diagnosztikai interfészt és a méréshez használt eszközt az autóbusz CAN hálózatára.



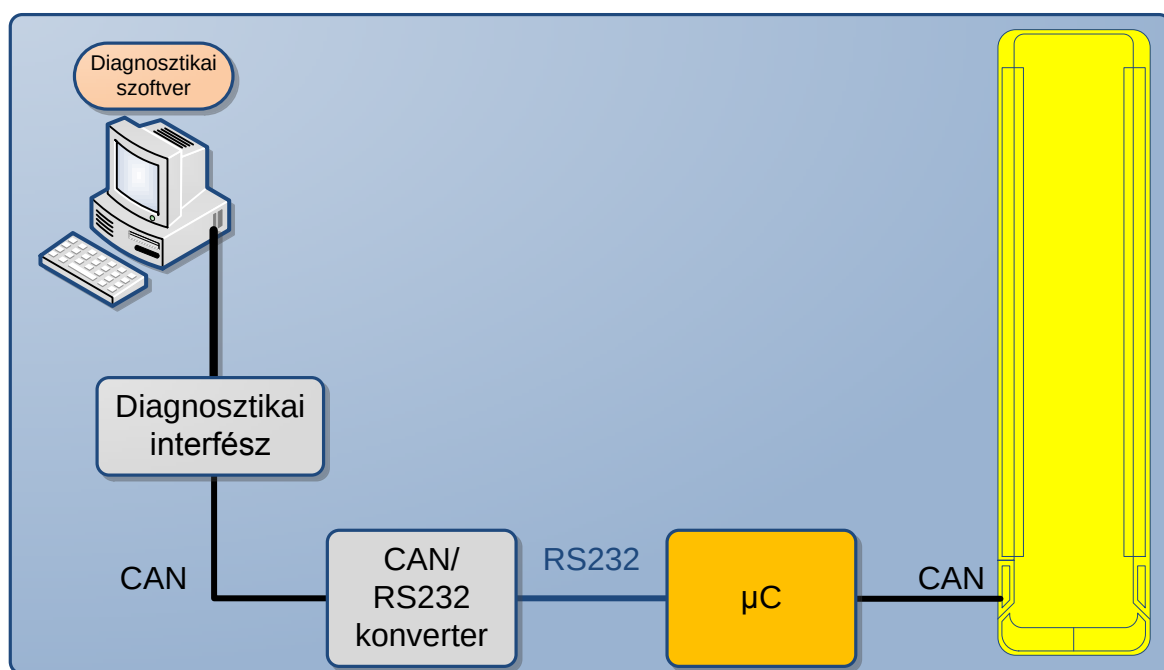
4.9. ábra: A diagnosztikai eszköz és a mérőrendszer is csatlakozik az autóbusz hálózatára

Ezeknél a méréseknél egyszerre láthattam a jármű hálózatának forgalmát és a diagnosztikai eszközzel való kommunikációját. Az azonosítók száma és az üzenetszám is megnőtt az alapállapothoz képest. A diagnosztikai eszköz igényel egy címet, hogy tudjon kommunikálni a többi ECU-val.

Egy hibakód törlése közben is végeztem mérést, viszont ilyenkor a járműről egy megadott ideig le kell venni a gyújtást, majd újra ráadni, így a mérési eredmények nem mérvadóak, de megfigyelhető ekkor is az új cím igénylés és az ECU-k közötti kommunikáció.

5. CAN üzenetforgalom szimulációja

Az előző fejezetekben bemutatott mérésekre és SAE J1939 szabványra alapozva készítettem el többféle szűrési eljárást, ezeknek a bemutatása egy általam is készített korábbi cikkben található [7]. A szűrőalgoritmusokat kipróbálva valós körülmények között volt, amelyikkel működött a diagnosztizálás. Működés közben megállapítottam, hogy instabil volt a kapcsolat a diagnosztikai szoftver és a jármű hálózata között, így mindenképpen fontosnak tartottam a szűrési eljárások labori körülmények melletti tesztelését és fejlesztését.

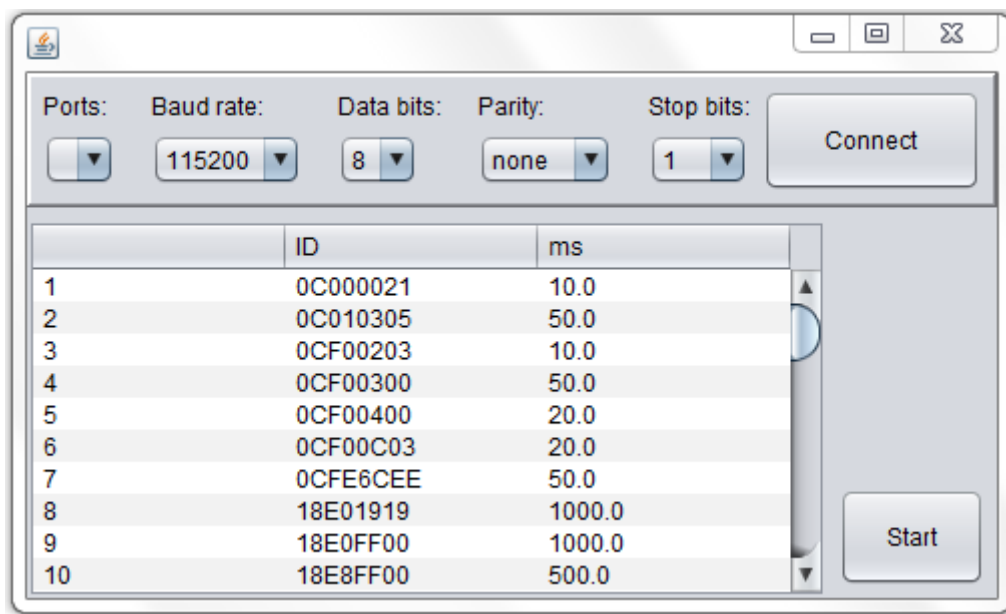


5.1. ábra: A szűrőalgoritmusok tesztelése éles körülmények között

A fenti ábrán látható elrendezésben teszteltem a szűrő eljárásokat, ennek a szimulációját kellett megvalósítanom labor körülmények között. Az ötletem, hogy egy számítógép a soros port-ján (ha nincs, akkor USB-n egy virtuális soros port-on) keresztül küldi az adatokat. Ezek az adatok a jármű hálózatán lévő azonosítókat szimbolizálják. Úgy kell kialakítani a küldést, hogy az adatmennyiség tükrözze a mérésnél tapasztaltakat. Ezeket az adatokat egy CAN/RS-232-es átalakítás után a mikrokontroller kapja meg és szűri. A kontroller által küldött adatok elemezhetőek, ezekből tudhatjuk meg, hogy mit kapna meg az adott szűrés után a diagnosztikai eszköz. Az eddig általam készített szűrési eljárások úgy működtek, hogy egyes

azonosítókat nem engedtek át, így csökkentve a továbbítandó adatmennyiséget, de ezek az eljárások nem vezettek célra, ezért mindenképpen olyan szűrő algoritmusra van szükség, amely minden azonosítót átenged.

A CAN üzenetforgalom szimuláló programot Java nyelven készítettem, a felhasználói felülete a következő ábrán látható.



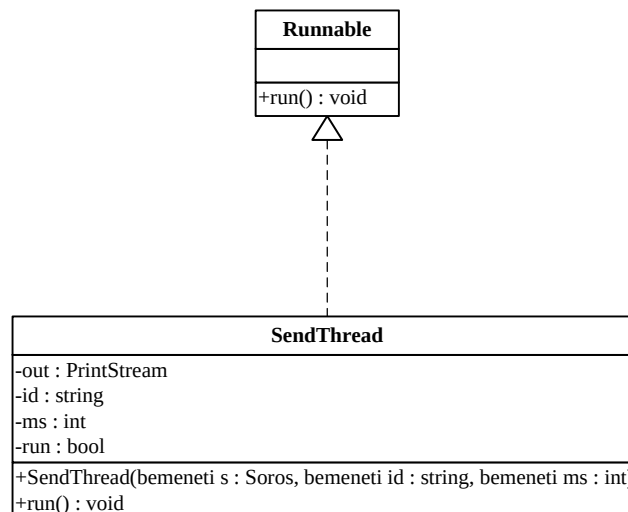
5.2. ábra: Az elkészített program felhasználói felülete

A program elindításakor megvizsgálja a soros port-okat a számítógépen, ezután ki lehet választani azt a Ports feliratú lenyíló menüből. Az RS-232 paramétereit a mellette található menükből lehet kiválasztani. Ezek a paraméterek attól függenek, hogy milyen eszközhöz csatlakozunk. A Baud rate alatt a soros kommunikáció átviteli sebességét lehet kiválasztani 600-tól 115200 bps-ig terjedő tartományból (csak a szabványos és leggyakoribb értékeket tartalmazza). A következő menüből lehet meghatározni, hogy 5, 6, 7 vagy 8 adatbitet (Data bits) használunk egy keretben. A negyedik lenyíló menü a paritás ellenőrzés módját tartalmazza, ez lehet páros (even), páratlan (odd), mark, space vagy none, amikor nem használunk paritásbitet. A stopbitek számát az utolsó menüből lehet kiválasztani. A megfelelő értékek kiválasztása után lehet kapcsolódni a Connect gombbal.

Soros
-port : CommPortIdentifier -sport : SerialPort -ports : Vector
+Soros() +connect(bemeneti portName : string, bemeneti baud : int, bemeneti bits : int, bemeneti stop : int, bemeneti parity : int) : void +output() : PrintStream +input() : DataInputStream +disconnect() : void +listPorts() : void +getPorts() : Vector +getPortTypeName(bemeneti portType : int) : string

5.3. ábra: Az RS-232-es kapcsolatot kezelő osztály struktúrája

A program elindulásakor még betölt egy Excel fájlt, amely tartalmazza azokat az azonosítókat, amelyeket küldeni szeretnénk. Azért döntöttem .xls fájlformátum mellett, mert ilyen fájlokban tárolom a mérési eredményeket és Excelben létrehoztam egy automatikus kiértékelőt is, amely a beadott azonosítókat felbontja. Az ID oszlopban az azonosítók szerepelnek, mellettük a „ms” oszlopban pedig a sebességük, például ha ez az érték 50, akkor a program 50 milliszekundumonként küld egy üzenetet az adott azonosítóval. A program folyamatábrája a 8.2. sz. Mellékletben látható.



5.4. ábra: Az üzenetküldő osztály

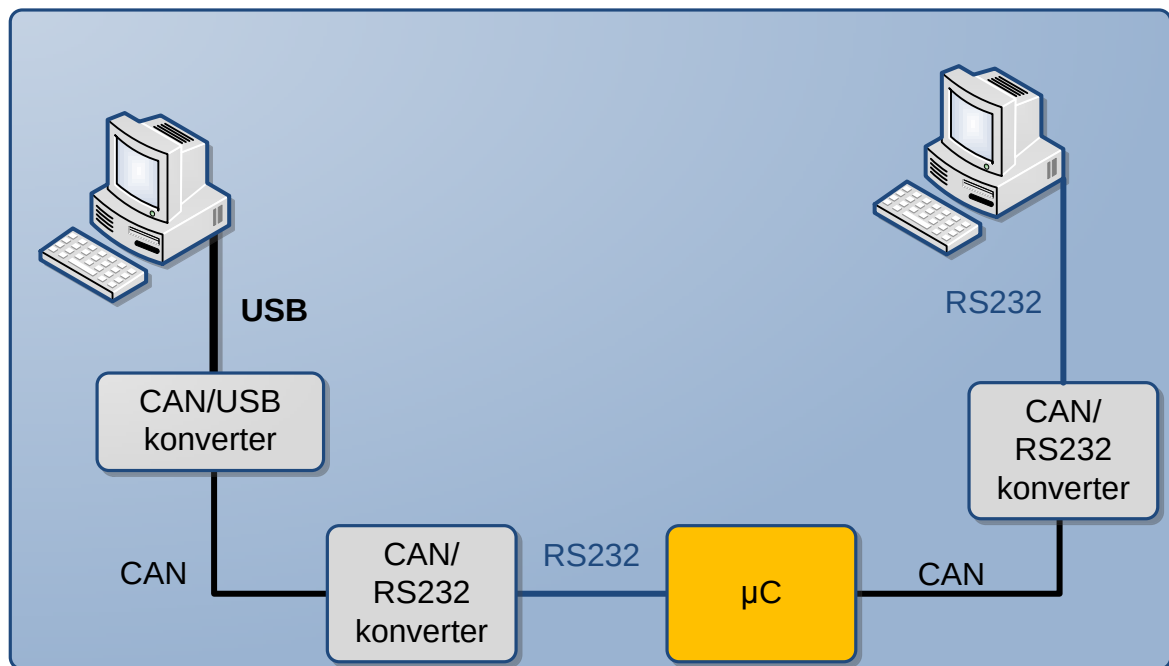
Minden azonosító küldéséhez egy új szálhozok létre a programban, hogy az időzítés megfelelő legyen, így egy adott szál csak egyfajta üzenetet küld és várakozik a megadott ideig. Az üzenetküldő osztály implementálja a java.lang.Runnable interfészt és felülírja annak a run metódusát, ez látható az 5.4. ábrán is.

A program main osztályának az adattagjai és metódusai a következő ábrán láthatók.

Ablak
-s : Soros -ms : int = 0 -n : int = 0 -in : DataInputStream -out : PrintStream -id : string -st : Thread -tmodel : DefaultTableModel -table : JTable -scrollPane : JScrollPane -boxBaud : JComboBox -boxBits : JComboBox -boxParity : JComboBox -boxPort : JComboBox -boxStop : JComboBox -buttonConnect : JButton -buttonStart : JButton -baudRate : JLabel -dataBits : JLabel -parity : JLabel -ports : JLabel -stopBits : JLabel -panel : JPanel
+Ablak() - initComponents() : void - buttonConnectActionPerformed(bemeneti evt : ActionEvent) : void - buttonStartActionPerformed(bemeneti evt : ActionEvent) : void +main() : void

5.5. ábra: A főosztály adattagjai és metódusai

A felhasználói felület elkészítéséhez a javax.swing csomagjában található elemeket használtam fel, a gombok eseménykezeléséhez pedig a java.awt.event-ben található ActionEvent típust.



5.6. ábra: Az elkészített CAN üzenetforgalom szimuláló rendszer

Az előző ábrán látható az elkészített rendszer. Az 5.1. ábrával szemben itt a jármű oldalán található számítógép szimulálja annak a CAN üzenetforgalmát az általam készített szoftverrel. Ezeket a CAN üzeneteket fogadja a mikrokontroller, majd szűrés után továbbítja azt az RS-232-es port-on. Az RS-232/CAN átalakítás után azokat az adatokat kell kapnunk, amelyeket a diagnosztikai eszköznek szeretnénk küldeni, ezt CAN/USB konvertálás után ellenőriztem egy számítógépen. Az ábrán látható két számítógép lehet ugyanaz is, ilyenkor az egyik programmal szimuláljuk az üzeneteket, a másikkal pedig monitorozhatjuk a szűrés eljárás hatékonyságát.

6. Összefoglalás

Dolgozatom célja az autóbuszok CAN üzenetforgalmának szimulálása távdiagnosztikai szűrőalgoritmusok fejlesztése céljából. A járművek hibás működése a közlekedésben késéshez vagy akár leálláshoz vezet, amelynek költségvonzata nem elhanyagolható. Ennek elkerülése érdekében a járművek folyamatos felügyelete szükséges, lehetővé téve a zavarok megelőzését vagy időben elhárítását. A távdiagnosztika feladata a szükséges diagnosztikai információk továbbítása egy felügyeleti központ felé, ahol szakértő azonosítja a működési paramétereket, felméri az eltéréseket és előrejelzést ad. Járművek távdiagnosztizálásához mindenképpen vezeték nélküli kommunikációs hálózatot kell használni. Ma hazánkban csak a GSM hálózat felel meg a követelményeknek, viszont az alacsony átviteli sebessége miatt nem lehet egyszerűen megoldani a távdiagnosztizálást.

Egy rövid bevezető után ismertettem a járművekben használt CAN buszrendszert általánosan, majd még ebben a fejezetben kitértem a járművek két csoportjára és azok összehasonlítására a diagnosztikai szabványok tekintetében. Ez az összehasonlítás fontos, hogy tisztában legyünk a járművek közötti különbségekkel.

Dolgozatom elkészítése során az autóbuszok távdiagnosztizálásával foglalkoztam, ezek a közepes és nagyteljesítményű járművek csoportjába tartoznak, a teherautók, kamionok és egyéb közúton használt szállítójárművek mellett. Ezek a járművek a CAN felett egy magasabb szintű protokollt is alkalmaznak, a SAE J1939-es szabványt. Ennek a szabványnak a bemutatása a harmadik fejezetben található.

A GSM-en keresztüli távdiagnosztizáláshoz csökkenteni kell a CAN adatok mennyiségét. A szűrőalgoritmusok fejlesztése előtt tisztában kell lenni a járműveken zajló kommunikációval, ennek érdekében autóbuszokon végeztem méréseket. A mérési eredményeket a negyedik fejezetben értékeltem ki.

Az ötödik fejezetben az általam készített CAN üzenetforgalom szimuláló programot mutattam be. Itt tartom fontosnak megemlíteni, hogy a dolgozatomban részletezett SAE J1939-es szabvány ismerete nélkül nem sikerülhetett volna létrehoznom egy ilyen

programot és a szűrőalgoritmusok fejlesztéséhez is elengedhetetlen a szabvány megismerése.

Az elkészített rendszer segítségével teszteltem a korábbi szűrőalgoritmusokat és már az első teszt sikeresnek mondható, ugyanis rájöttem az egyes szűrőalgoritmusok hiányaira. A hibák kiküszöbölése és új szűrési eljárások készítése folyamatban van.

A jövőbeli célom még a program univerzálissá tétele, azaz hogy ne csak az autóbuszok CAN üzenetforgalmát tudja szimulálni, hanem bármilyen CAN üzenetet tudjunk vele küldeni, akár 2.0A vagy 2.0B szabványnak megfelelőt. Ezáltal a program alkalmas lenne az oktatásba történő bevonásra, az Ipari kommunikáció tantárgy gyakorlatain be lehetne mutatni. Így a hallgatók könnyebben megismerkednének a CAN buszrendszerrel.

„A bemutatott kutató munka a TÁMOP-4.2.1.B-10/2/KONV-2010-0001 jelű projekt részeként az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg.”

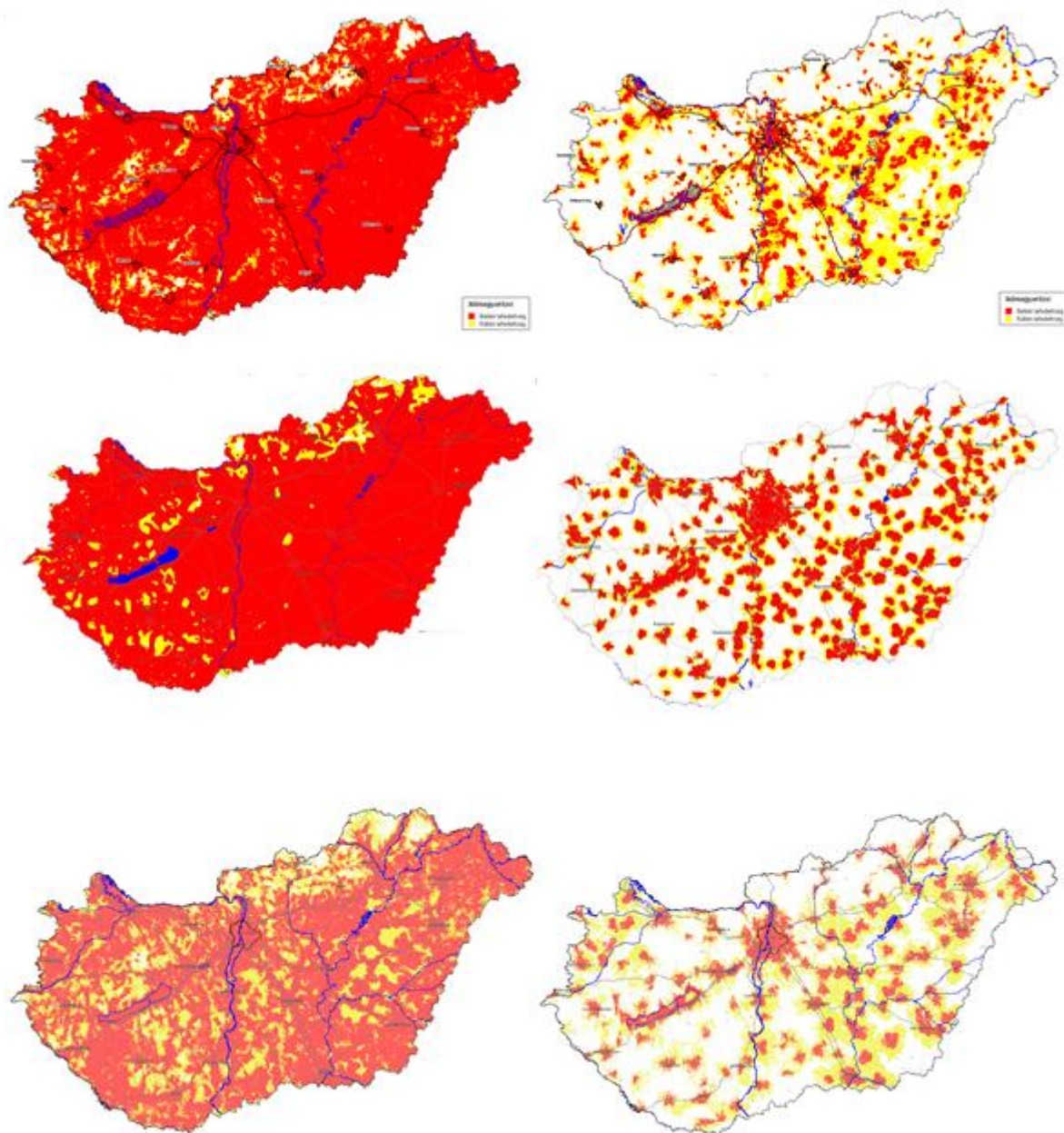
7. Irodalomjegyzék

- [1] Dominique Paret: Multiplexed Networks for Embedded Systems: CAN, LIN, FlexRay, Safe-by-Wire..., John Wiley & Sons, 2007.
- [2] Ajtonyi István: Ipari Kommunikációs Rendszerek I., AUT-INFO Kft., Miskolc 2008.
- [3] Jeffrey Craig (Vector CANtech, Inc.): A Comparison of J1939 and ISO15031, SAE On-Board Diagnostics Symposium, 2009.
- [4] Markus Junger: Introduction to J1939, Vector Informatik GmbH, 2010.
- [5] Wilfred Voss: SAE J1939 – Serial Control and Communications Vehicle Network, esd electronics, Inc., 2012.
- [6] Sean Bennett: Heavy Duty Truck Systems, Cengage Learning, 2010.
- [7] Biró Zoltán, Trohák Attila: Szűrési eljárások kutatása járművek GSM alapú távdiagnosztikai rendszerének kifejlesztése céljából, FMTÜ XVII. nemzetközi tudományos konferencia, 2012.

8. Mellékletek

8.1. sz. Melléklet

A három legnagyobb magyarországi mobil szolgáltató GPRS és 3G-s lefedettség térképe (balra a GPRS, jobbra a 3G, piros színnel a beltéri, sárga színnel pedig a kültéri lefedettség)



Forrás: Nemzeti Média- és Hírközlési Hatóság (www.nmhh.hu)

8.2. sz. Melléklet

Az elkészített CAN üzenetforgalom szimuláló program folyamatábrája.

