

Járműinformatika

Autóipari kommunikációs rendszerek

2021/2022. tanév, II. félév

Dr. Kovács Szilveszter

E-mail: szkovacs@iit.uni-miskolc.hu

Informatika Intézet 107/a.

Tel: (46) 565-111 / 21-07

Ajánlott és felhasznált irodalom

Az előadás anyag forrása:

- Pannon Egyetem, Mérnöki Kar, Gépészmérnöki Intézet
Járműmechanika Intézeti Tanszék
- **Dr. Fodor Dénes, Speiser Ferenc:**
Autóipari beágyazott rendszerek
- http://moodle.autolab.uni-pannon.hu/Mecha_tananyag/autoipari_kommunikacios_rendszerek/index.html



Dr. Fodor Dénes intézetigazgató, tanszékvezető egyetemi docens
szoba: C218

telefon: 88/624-000/6082, 88/624-776

e-mail: fodor@almos.uni-pannon.hu

Speiser Ferenc egyetemi

tanársegéd

szoba: C217

telefon: 88/624-000/6083

e-mail: speiserf@almos.uni-pannon.hu

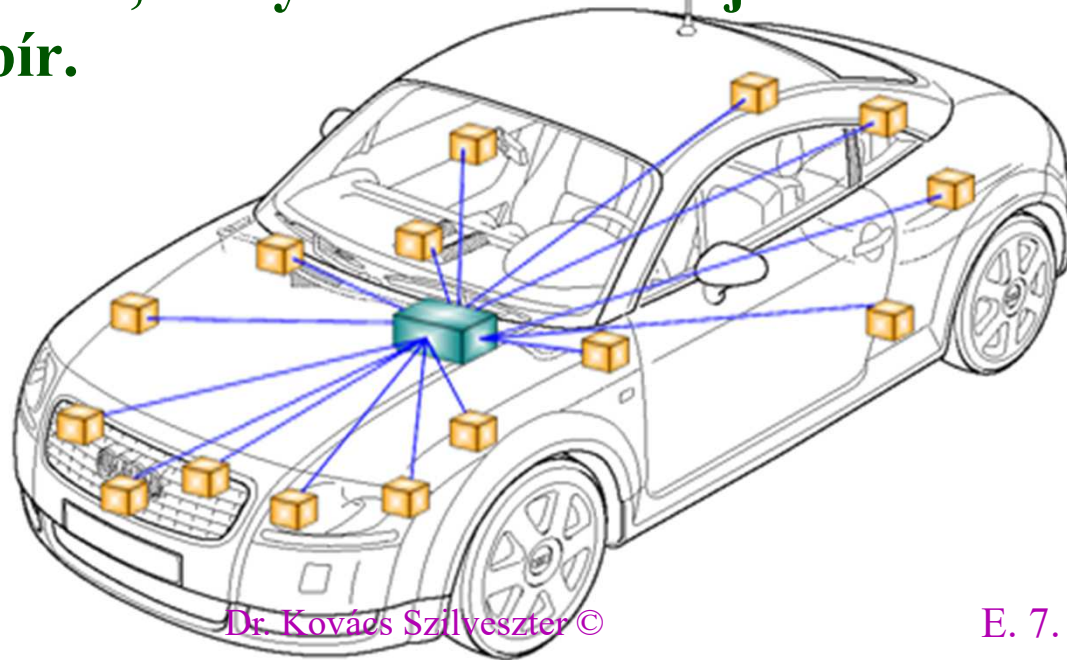


Bevezetés

- A közelmúltban a mikrokontrollerek egyre hatékonyabbá és egyre olcsóbbá váltak, amely lehetővé tette, hogy a gyártók távoli I/O eszközökbe, nyomógombokba, szenzorokba és egyéb komponensekbe ágyazzák őket, olyan „intelligens eszközöket” létrehozva, amelyek önállóan is képesek a szabályozási feladatuk ellátására.
- Így a '70-es, '80-as években domináló *központosított szabályozó rendszerek* (centralized control systems) helyett egyre inkább elterjedhettek az úgynevezett *elosztott rendszerek* (distributed/decentralized control systems), melyet csak kommunikációs rendszerekkel lehet megoldani.

Központosított szabályozó rendszer

- A rendszert alkotó egységek hagyományos módon **egy központi vezérlő egységhez csatlakoznak**, amelynek feladata az egész rendszer koordinálása. A központi vezérlő (*master*) ciklikusan lekérdezi a többi eszköz (*slave*) üzeneteit. Így bár determinisztikus, hogy egy eszköznek maximum mennyit kell várnia az átviteli közeg használatára, az ilyen modell több jelentős hátránnyal is bír.

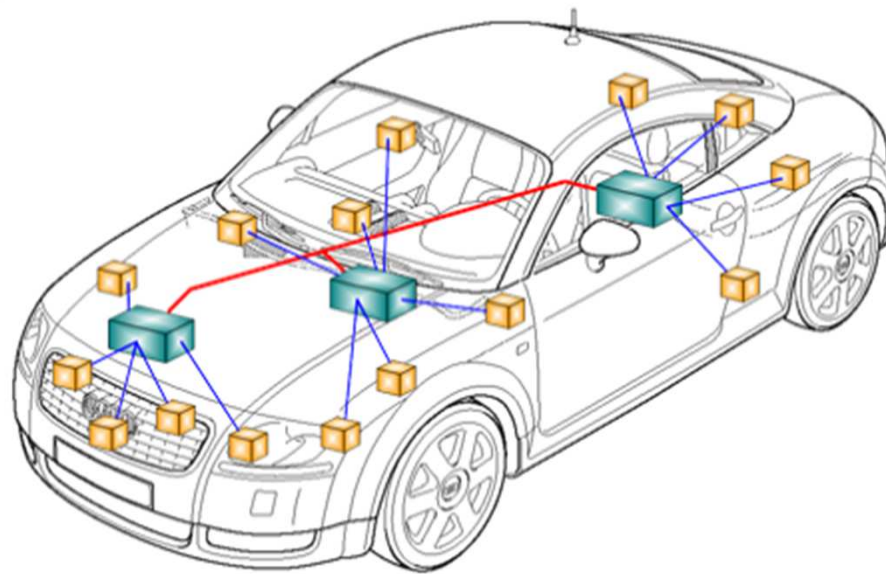


Központosított rendszer hátrányai

- A különböző egységek más-más típusú csatlakozókkal rendelkezhetnek, így **nagy számú vezetékre lehet szükség a központhoz kapcsolásukhoz.**
- A rendszer komplexitásának növekedésével a huzalok száma és a csatlakozók mérete is növekszik.
- Az ilyen *mester-szolga* (master-slave) rendszerben keletkező hibák felderítése bonyolult, és **a központi egység (CPU – Central Processing Unit) leállításával a teljes rendszer működésképtelenné válik.**
- **Új eszközök hozzáadásakor újabb problémák merülhetnek fel, például egy speciális csatlakozóval rendelkező egység integrálása egy már létező rendszerbe költség és munkaigényes feladat.**

Elosztott szabályozó rendszer

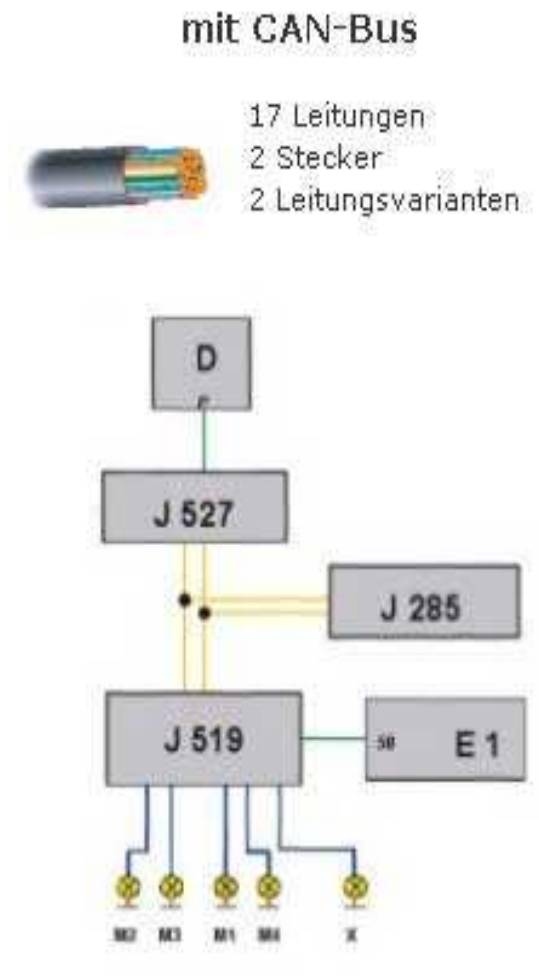
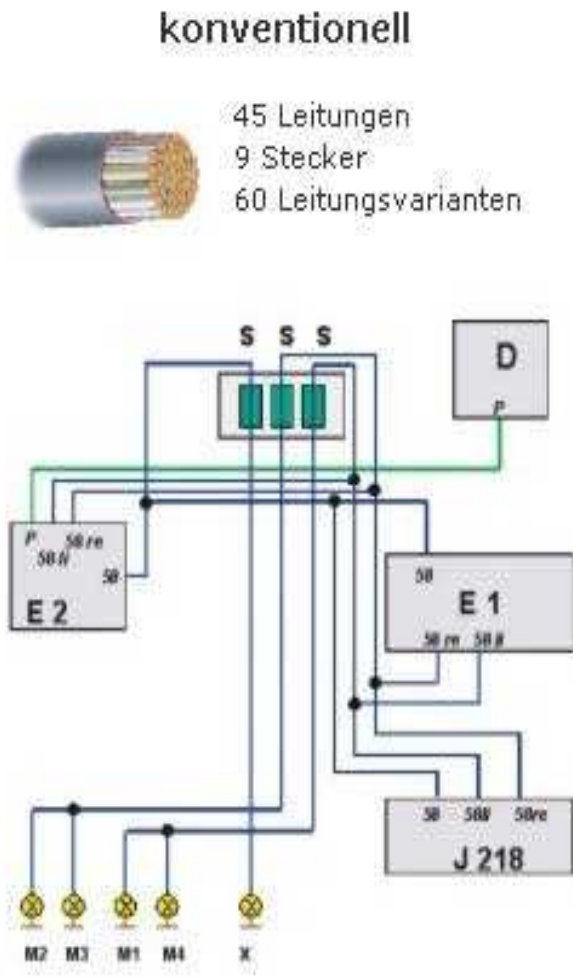
- Az ilyen rendszerek alapelve, hogy egy közös kommunikációs vonalra (*buszra*) kötik az összes egységet. Az ily módon egy hálózatba kapcsolt egységek immár önállóan kommunikálnak egymással. A hálózat használata új szabályozási koncepciót eredményezett, az úgynevezett *elosztott szabályozást*.



Elosztott rendszerek előnyei

- **Egy vezetékkötegre van szükség, amely gyakran már az energia ellátását is biztosítja a részegységeknek, ezzel is csökkentve a fizikai csatlakozók számát. A kevesebb vezeték nemcsak megbízhatóbbá teszi a rendszert, de egyszerűbbé és főként olcsóbbá is.**
- **Ezzel a megoldással lehetővé válik a rendszer folyamatos bővítése, mivel csupán egyfajta, szabványosított csatlakozóra van szükség, így lehetséges akár különböző gyártók eszközeinek közös rendszerbe integrálása is.**

Elosztott rendszerek előnyei



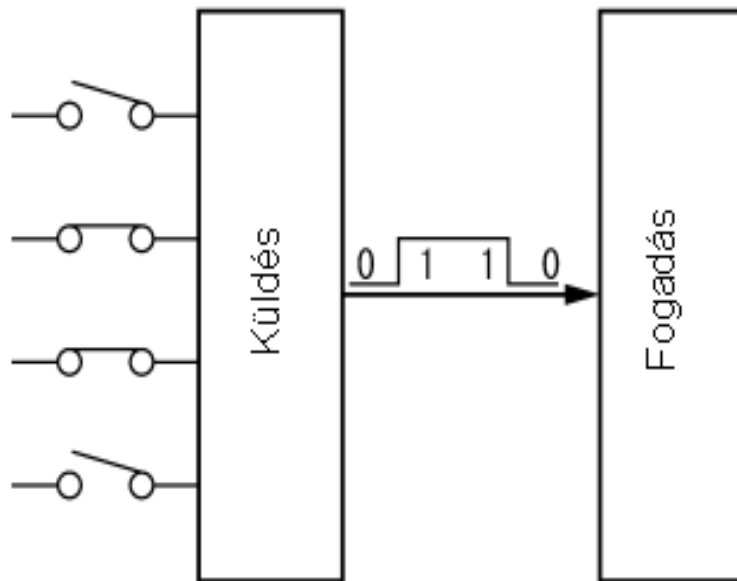
Elosztott rendszerek járművekben

- Az előzőek alapján kijelenthető, hogy az autóiparban az egyes járművekbe beépített elektronikai eszközök túlnyomó többsége **elosztott rendszert/rendszereket alkotnak**, és őket **különféle csoportokba lehet besorolni** úgy, hogy az egyes területek között nincs átfedés.

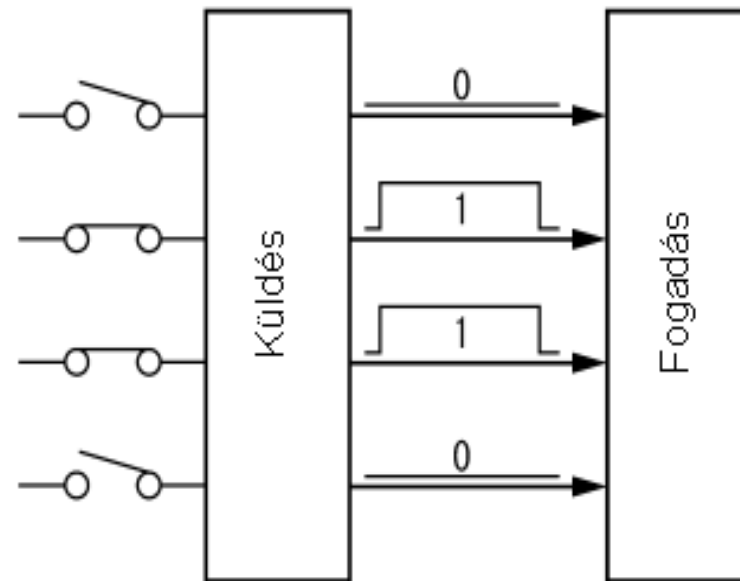
E területek az alábbiak:

- **Motorvezérlő (Engine Control) elektronika**
- **Sebességváltó elektronikai eszközei (Transmission Electronics)**
- **Karosszéria elektronikai eszközei (Chassis Electronics)**
- **Aktív biztonságért felelős elektronikai eszközök (Active Safety)**
- **Vezetőt támogató rendszer elektronikai eszközei (Driver Assistance)**
- **Kényelmi, vagy komfort elektronika (Passenger Comfort)**
- **Fedélzeti tájékoztató és szórakoztató elektronika (Infotainment Electronics)**

Soros/párhuzamos átvitel

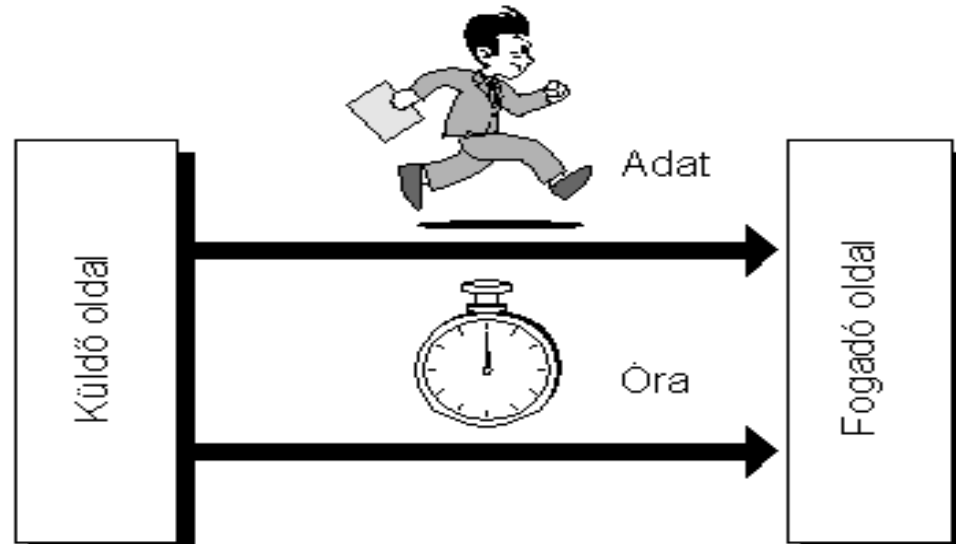


Soros kommunikáció

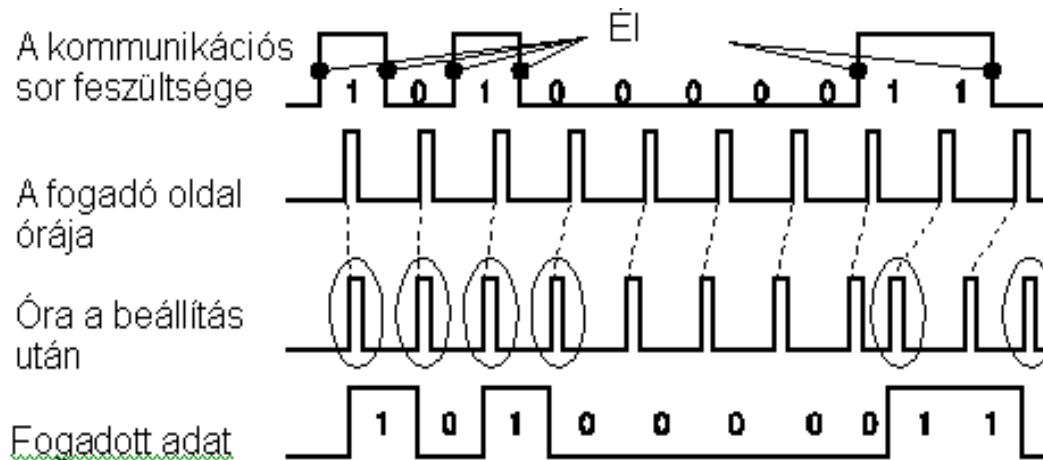


Párhuzamos kommunikáció

Szinkron/aszinkron átvitel

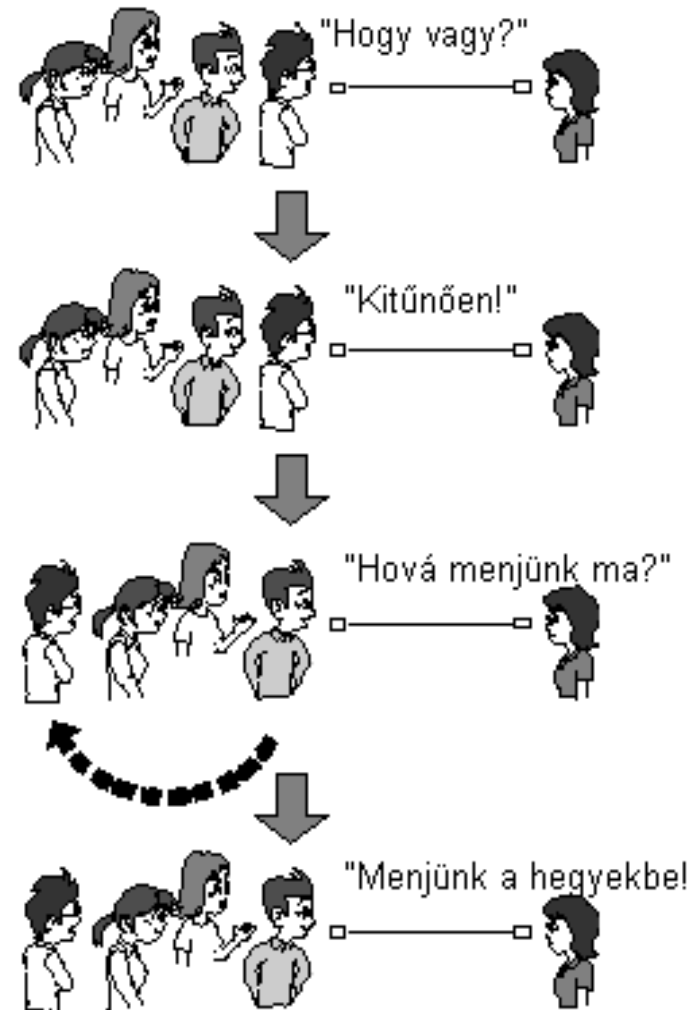


Van külön óra vezeték



Szinkronizációs módszerek aszinkron

- Ha egyszerre több ember beszél, akkor nem értjük, hogy mit mondanak. Ha a beszédjüket egy szabály határozza meg, akkor mindegyik mondanivalóját meg lehet hallgatni. Műszaki értelemben **multiplex kommunikációról** beszélünk, ha egy szabály szerint **viszünk véghez sok párbeszédet.** (multiplex kommunikáció)



A multiplex kommunikáció megvalósítása

- **A multiplex kommunikáció, azaz a megfelelően szabályozott információcsere megvalósítása többféle módon is lehetséges, például:**
 - **Frekvenciaosztásos többszörös hozzáférés (FDMA) jellemző pl. a kábeltévé hálózatokon**
 - **Időosztásos többszörös hozzáférés (TDMA, Time Division Multiple Access) rendszerben működik pl. a vezetékes telefon hálózat (E1, E2 stb...)**
 - **Vivőérzékeléses többszörös hozzáférés, ütközés detektálással (CSMA/CD) rendszerben működnek a helyi adathálózatok (pl. Ethernet, CAN)**

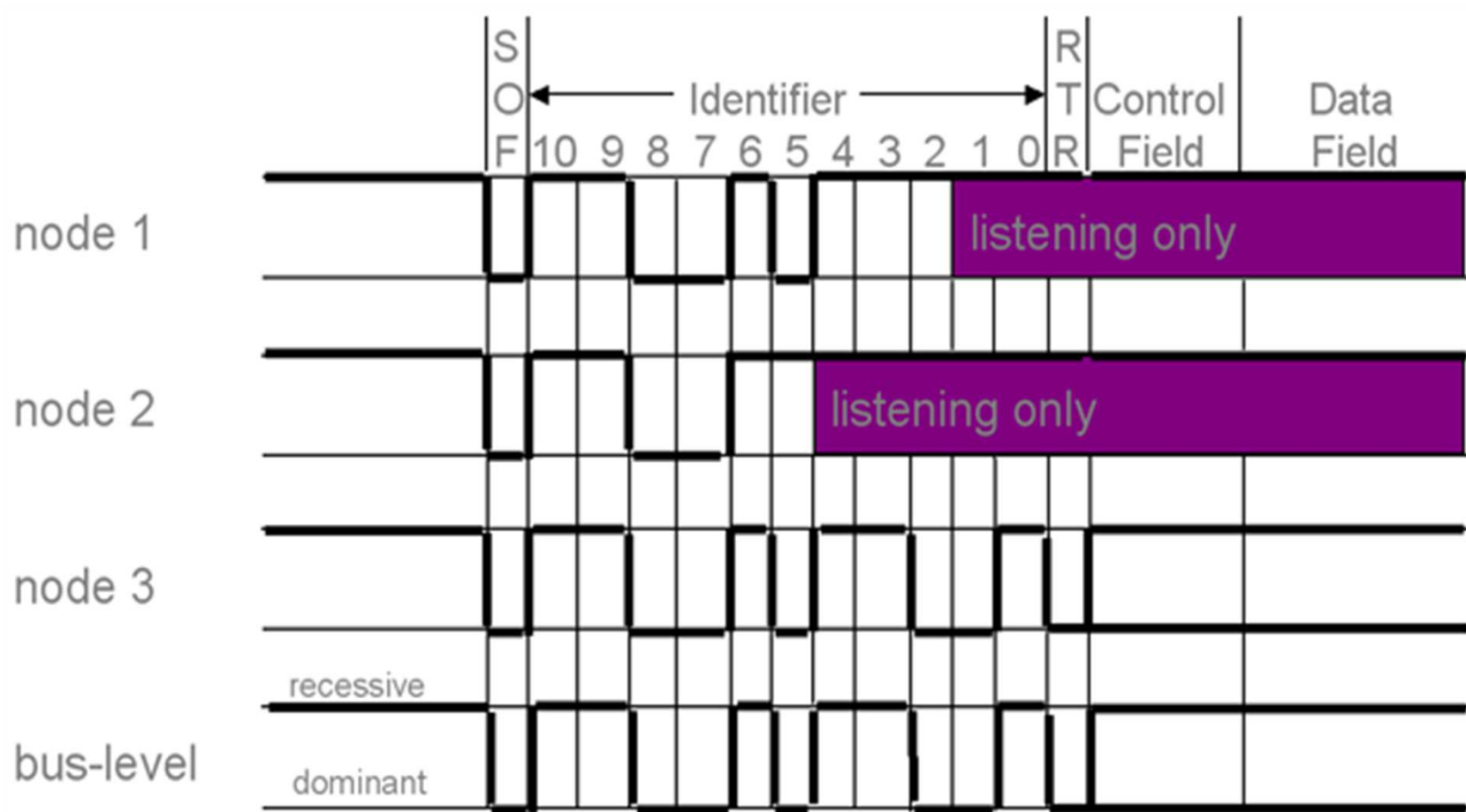
Hálózati átviteli módok

Hozzáférési pontok	irányok	adatforgalom jellege	példa
pont-pont,	1	szimplex	LVSD (Low-voltage differential signaling)
pont-pont,	2	fél- vagy teljes-duplex	UART
1 adó több vevő	1	üzenetszórás (broadcast)	PLC egyes fajtái
1 mester több szolga	2	lekérdezéses	SPI, LIN
1 gyűjtő, több forrás	1	adatgyűjtés	SENT (Single Edge Nibble Transmission)
változó mester	2	osztott közeghozzáférés	CAN

Busz arbitráció (versengés)

- Az autóiparban több helyen is alkalmazott **CSMA/CD + AMP** (Carrier Sense Multiple Access with Collision Detection and Arbitration on Message Priority) jellegű protokollok esetén a fizikai jelszintek kialakítása olyan, hogy ha egy időben több egység is megpróbál adni a buszon, a több különböző jelszintből mindig az un. domináns jelszint érvényesül.
 - A keret elején elhelyezkedő azonosító alapján, a legnagyobb prioritású (pl. CAN esetén, ahol a domináns jelszint a 0, a legalacsonyabb azonosítóval rendelkező) ECU jele nem sérül, viszont az összes többié igen, amelyek ennek észrevétele után az adást felfüggesztik. Ez a folyamat a **bitenkénti arbitráció**.
- Az arbitrációban **nyertes eszköz zavartalanul befejezi a kommunikációt, a vesztes eszközök pedig újra megpróbálkoznak az adással, miután az ismét felszabadul.**

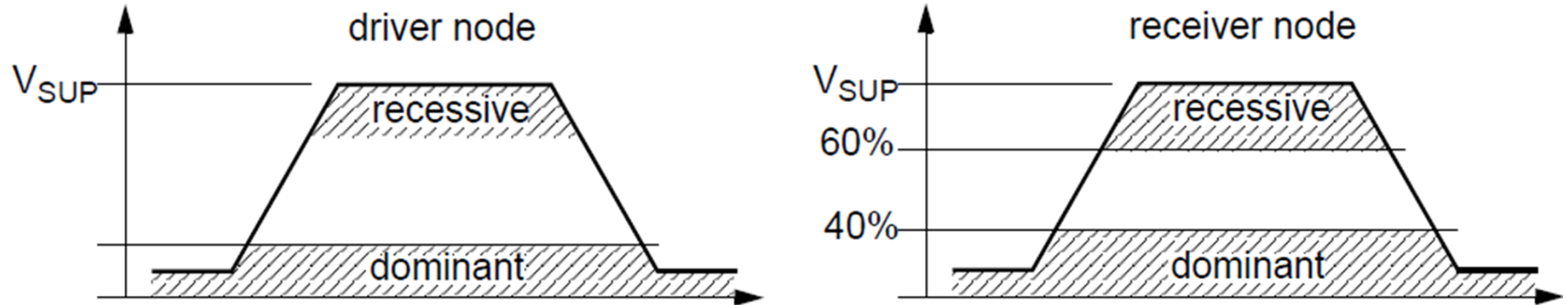
PI: CAN arbitráció 3 résztvevővel



Node 3 wins arbitration and transmits his data.

- Csak ha a hálózat mérete kicsi a bitidő alatt a jel által megtett úthoz képest

Recesszív és domináns bitnek vett feszültség szintek

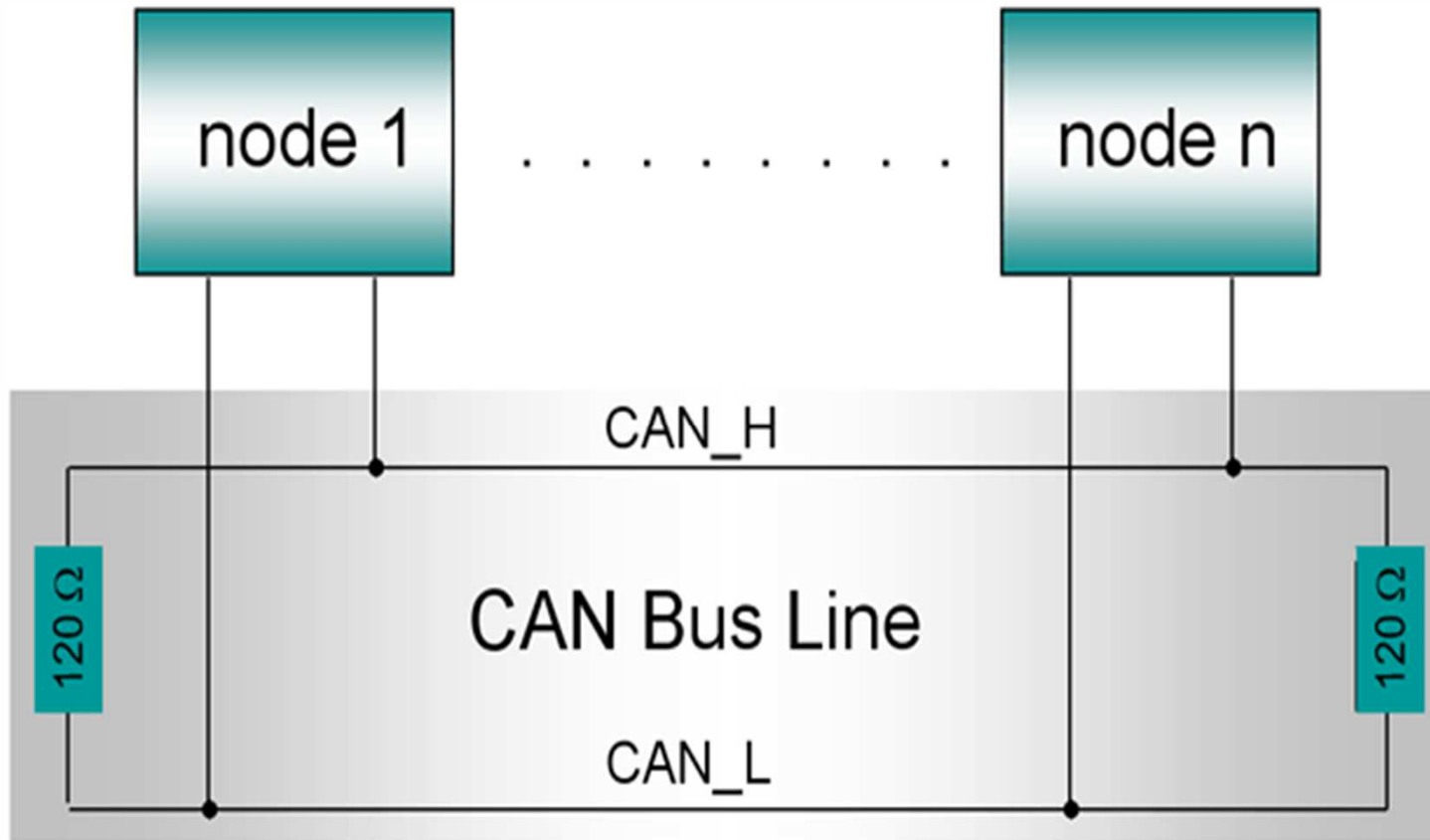


Vezetékes adatátvitel jellemzői

- **Sávszélesség**
 - A szükséges átviteli sebesség és átviteli távolság függvényében változik a vezetéktől megkövetelt minőség, aminek az elsődleges paramétere a csillapítás.
- **Jelzéssebesség (Baudrate), adatátviteli sebesség (bitrate)**
 - Sok helyen a bit/s és baud (=szimbólum/másodperc) mennyiségeket egymás alternatíváiként használják, ez elvileg hibás, de sajnos elterjedt gyakorlat, érdemes tudni a különbségükről a félreértések elkerülésére.
- **Lezárás**
 - Ha az átviteli út hosszúsága (l) összemérhető a jel bitidő (t_{bit}) alatt megtett távolságával (nagyjából $l > c \cdot t_{\text{bit}} / (2 \cdot 10)$, ahol c a vezetéken mért terjedési sebesség), akkor mindenképp le kell zárni a vezetéket a hullámimpedanciának megfelelő ellenállással, legalább az egyik végen, hogy elkerüljük a többszörös reflexiót.

Lezáró ellenállások CAN buszon

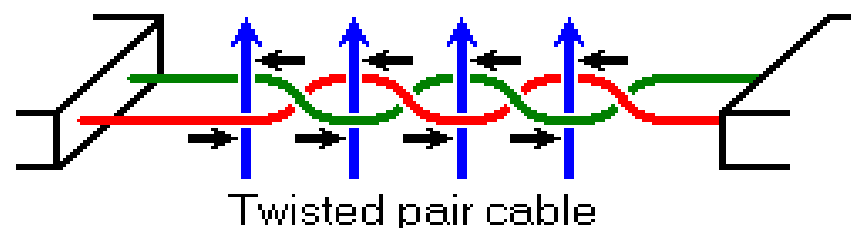
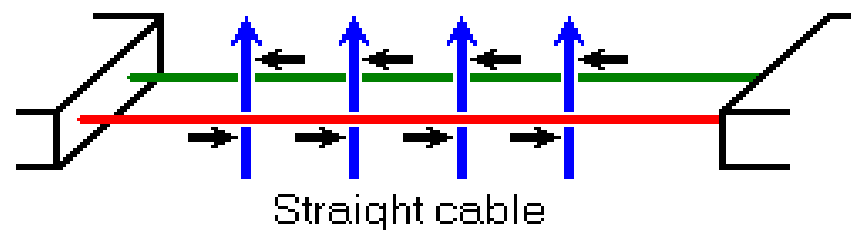
- A lezárás azonos a kábel hullám-impedanciájával



Vezetékes adatátvitel jellemzői

- **Zavarvédelem**

- A jelátvivő vezetékek ki vannak téve külső zavaró elektromos és mágneses tereknek, igen gyakran más vezetékekkel együtt futnak, vagy teljes hosszúságban, vagy csak rövid távolságon. A többi vezetékben folyó áram vagy a rajtuk mérhető feszültség változása induktív vagy kapacitív csatoláson keresztül megváltoztatja a hasznos jelet, zavarjelet ad hozzá. A differenciál módusú zavarás ellen a kábel szimmetrikus kialakításával hatékonyan lehet védekezni.



➡ **Magnetic field**
➡ **Induced noise current**

Kommunikációs hálózatok alkalmazása

- **Az autóiipari kommunikációs hálózatok alkalmazási területei a hálózat kiterjedése alapján három főbb csoportra bonthatók:**
 - készüléken (ECU) belüli,
 - készülékek közötti, de járművön belüli,
 - külvilághoz való kapcsolódódás.

Készüléken (ECU) belüli

- A készüléken (ECU) belüli összeköttetés esetén az átviteli csatorna meglehetősen **biztonságos**, sem külső zavarás, sem lehallgatás veszélye nem merül fel jelentős mértékben, az **adatátviteli távolság is limitált**, ezért meglehetősen **olcsó** eszközökkel megvalósítható.

Készülékek közötti, de járművön belüli

- Készülékek (ECU) közötti összeköttetés a gépjárművekben ma elsősorban **vezetékes átviteli közegben** valósul meg. A lehallgatás esélyét, habár létezik, lényegtelennek tekintik, viszont **a járművön belüli kölcsönös zavarás létező és fontos veszély**, részben a gyújtás és egyéb teljesítmény-elektronika impulzusai, részben maguk a kommunikációs hálózatok, részben a környezet (RF adók, villámok, elektrosztatikus kisülés) által okozott zavarjelek ellen kell védeni a hasznos jelet. A **készülékek közötti távolság már jelentősebb** (néhányszor 10 méter) és topológia szempontok is szerepet játszanak a hálózatok megbízható kialakítása során, ezért ez jelentősen **magasabb áron** kivitelezhető a készüléken belüli kommunikációhoz képest.

Külvilághoz való kapcsolódás

- Az autó külvilággal való kapcsolatát majdnem kizárólag **vezeték nélküli** kapcsolattal oldják meg (kivéve pl. **diagnosztika**), ezek a kapcsolatok jellemzően jobban ki vannak téve **zavaró hatásoknak**, illetve a **lehallgatás veszélye** is gyakran felmerül. További kihívást jelent, hogy a különböző, előre meg nem határozott gyártók termékeinek együttműködését nehéz garantálni. Ezért igen nagy szerepe van a szabványos, minősített készülékeknek. Ilyen, kívülről hozzáférhető rendszerek segítségével a jármű üzembiztonságát befolyásoló információk megváltoztatása nem engedélyezett, strukturálisan kell úgy kialakítani, hogy ne legyen lehetséges kívülről befolyásolni (speciális kivételektől eltekintve).

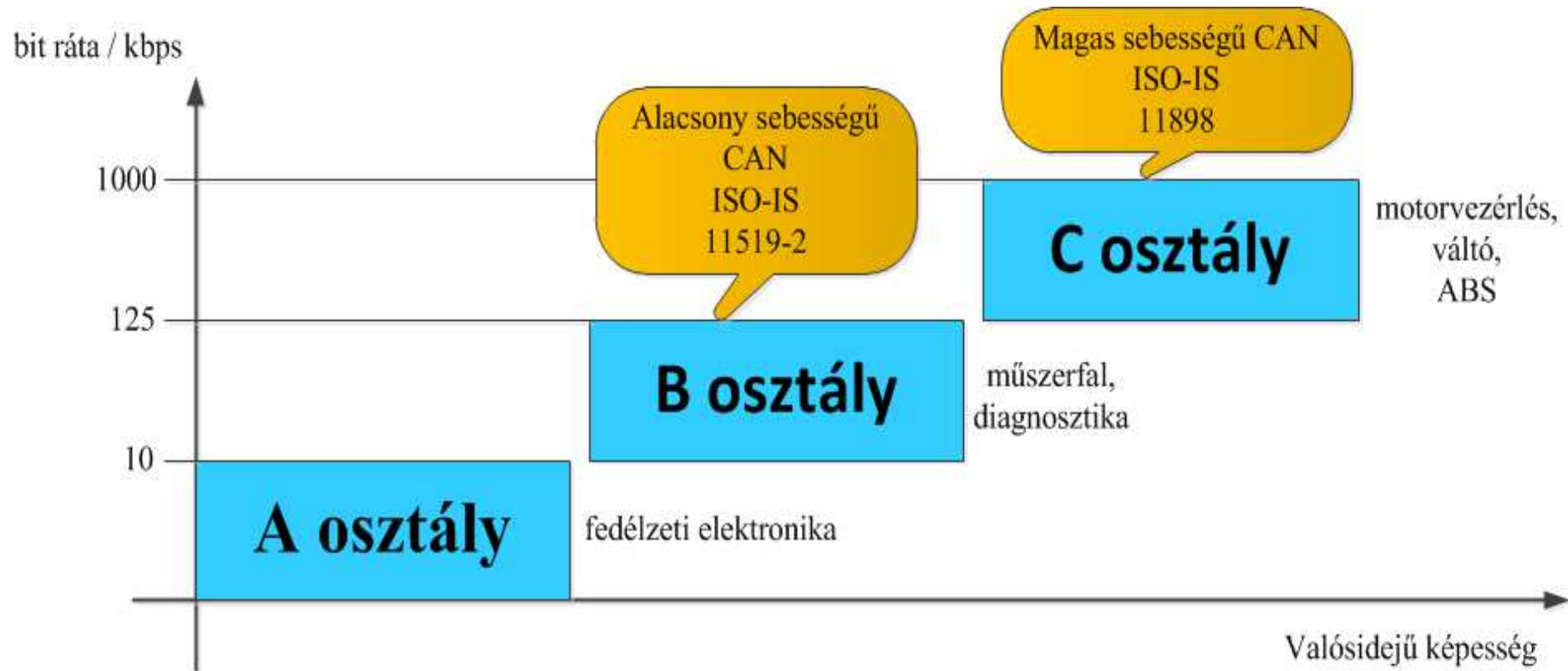
Kommunikációs hálózatok a használat jellege szerint

	készüléken belüli	járművön belül, készülékek között	járművön kívüli kapcsolat
Vezetékes (wired)	I2C, SPI, LVDS	CAN, J1708, J1850, Flexray, LIN, MOST, Ethernet (OPEN, RTPGE), USB, SENT, LVDS, PLC	diagnosztika (K vonal, J1850, CAN, J1708), RS232
Vezeték nélküli (wireless)	-	Bluetooth, WiFi, IRDA	WiFi Hot Spot-ok, mobiltelefon hálózat (GSM/GPRS/UMTS), műholdas rendszerek (GPS), közlekedési információk (RDS)

SAE autóipari alkalmazás osztályok

- **A osztályú alkalmazások:** a karosszéria-elektronika olyan kevésbé intelligens eszközeinek kommunikációja, mint pl. kapcsolók, fényszórók, tükörbeállítás, ülés pozicionálás, elektromos ablakemelő, központi zár. ($\leq 10\text{Kbit/s}$)
- **B osztályú alkalmazások:** magasabb szintű információk cseréjére szolgálnak, pl. információátvitel a műszerfal, vagy a légkondicionáló vezérlése felé. ($\leq 125\text{Kbit/s}$)
- **C osztályú alkalmazások:** valós idejű (real time critical) információátvitel, 1-10ms-os ciklusidővel, és 1ms-nál rövidebb késleltetéssel. ($\leq 1\text{Mbit/s}$)
- **D osztályú alkalmazások:** ebben az esetben nagyobb mennyiségű adat továbbítására van szükség, az adatblokkok mérete elérheti a néhány kbyte-ot is, mint pl. a hangrendszer, telefon vagy GPS (Global Positioning System) kommunikációja során. ($1\text{Mbit/s}–10\text{Mbit/s}$)

Az autóiipari alkalmazások SAE szerint

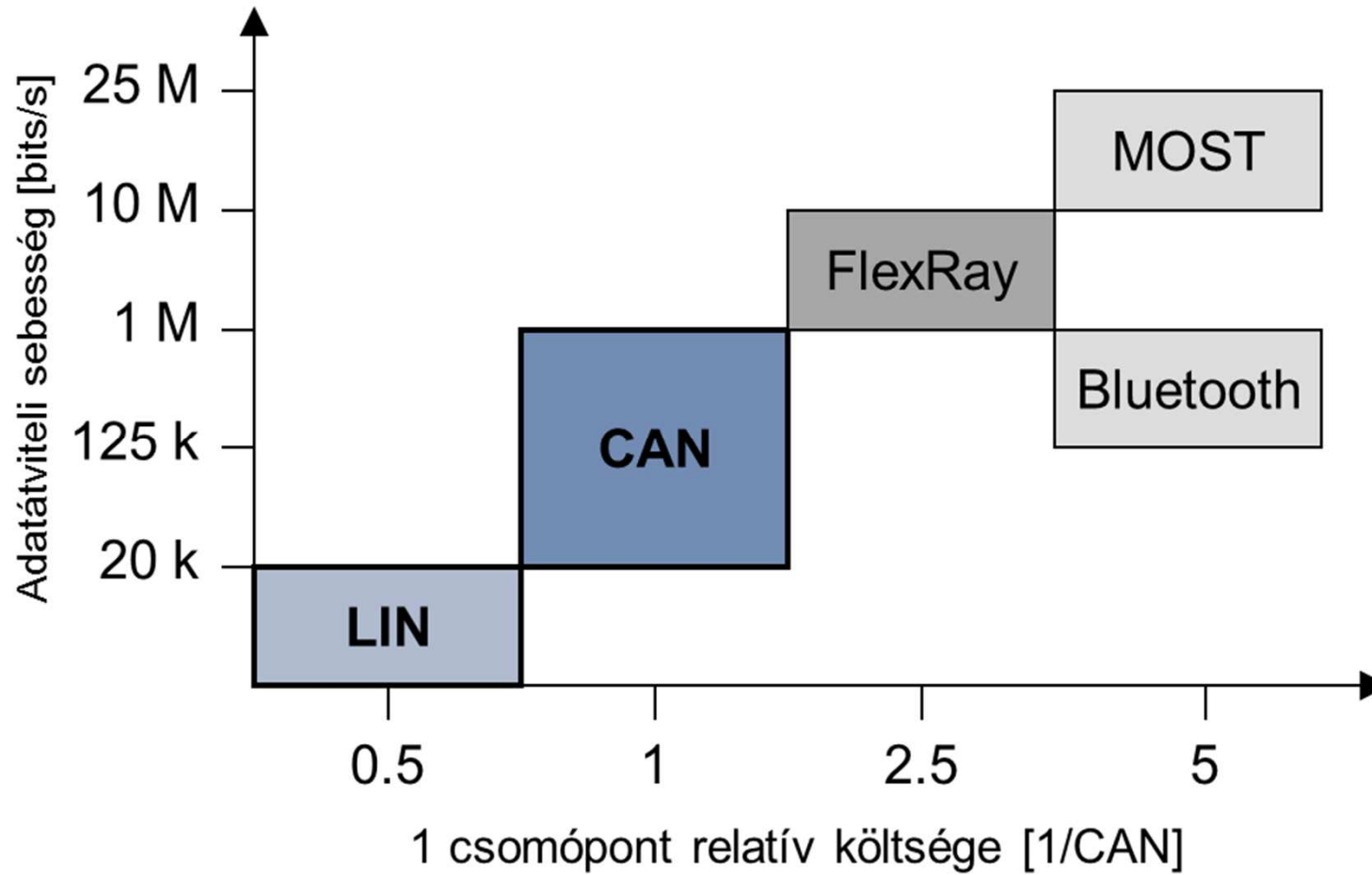


- **SAE - Society of Automotive Engineers**

ISO szerinti osztályozás

- **Alacsony sebességű kommunikáció: 125kb/s alatt.**
- **Nagysebességű kommunikáció: 125kb/s felett.**

Jelentősebb autóiipari kommunikációs protokollok



Jelentősebb autóiipari kommunikációs protokollok

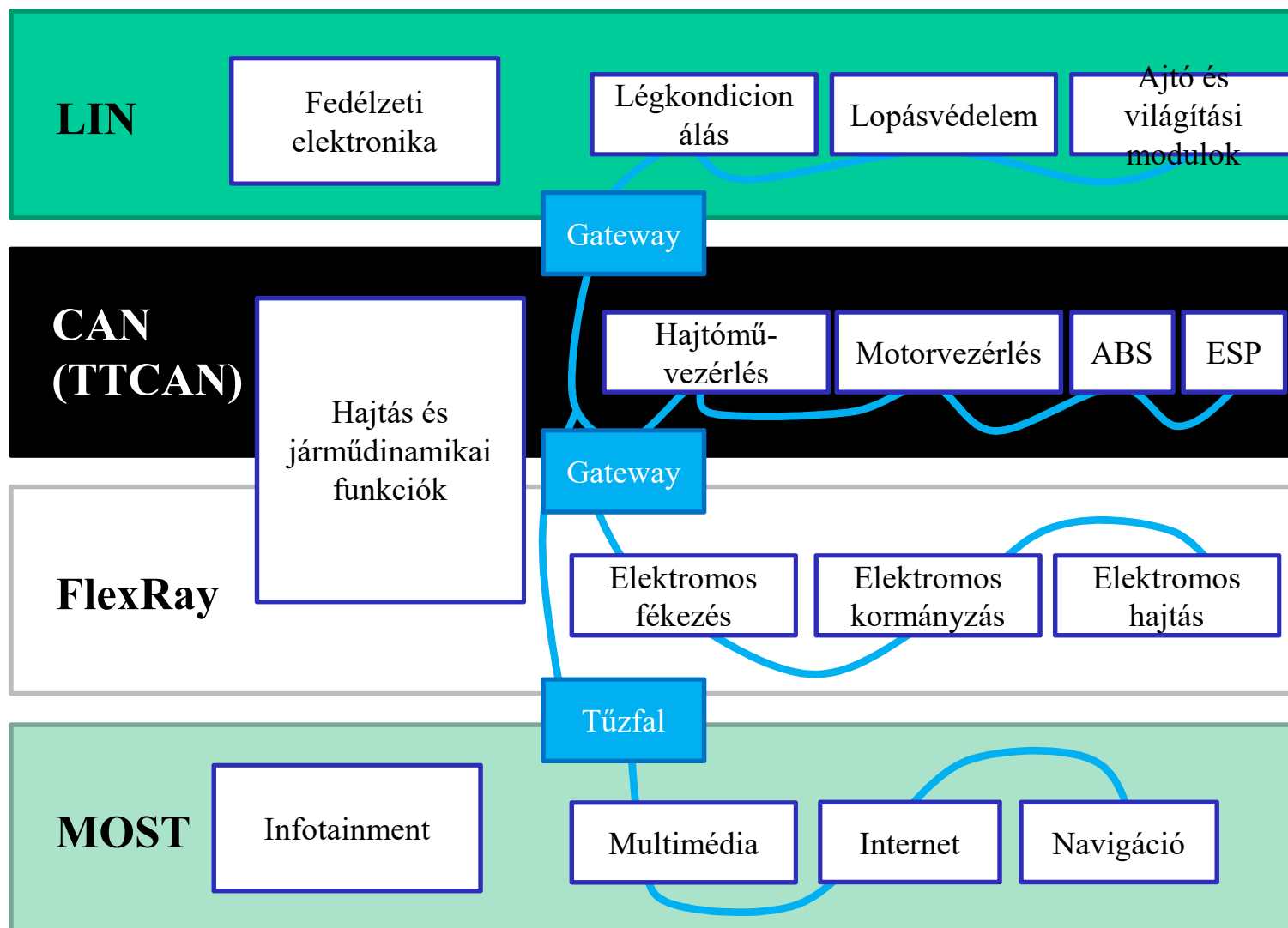
- **LIN:** Local Interconnect Network
- **CAN:** Controller Area Network (Vezérlőterületi Hálózat)
- **FlexRay:** Egy skálázható flexibilis nagy sebességű kommunikációs rendszer. A FlexRay busz az X-by-wire-
lehetőségekhez kifejlesztett nagysebességű, real-time
adatátvitelt biztosító rendszer.
- **MOST:** Media-Oriented Systems Transport. A MOST
protokoll egy multimédiás optikai hálózat.
- **Bluetooth:** Interfész specifikáció mobiltelefonok,
számítógépek, stb. rövid távú vezeték nélküli
összekapcsolására.

Buszrendszerek összehasonlító táblázata



Specifikáció	1999	1983	1999	2009
Rel. Ár/Csomópont	Kicsi (~1€)	Közepes(~2€)	Magas (~5€)	Közepes(~3€)
Medium	Egy vezetékes	Két vezetékes	Optikai	2x Két vezetékes
Átviteli sebesség	20 Kbit/s	1 Mbit/s	22,5 Mbit/s	2x10 Mbit/s
Adatmennyiség	Kicsi	Közepes	Magas	Magas
Busz-kezelés	Master/Slave	Multi-Master	Master/Slave	Multi-Master
Buszhozzáférés	Aszinkron	Aszinkron	Szinkron / Aszinkron	Szinkron / Aszinkron
Résztevők száma	Master + 16 Slaves	Javasolt: Max. 32	Max. 64	Nem definiált

Buszrendszerek egy járműben



Az RS-232-C interfész

- Számítógép és terminál, vagy modem közötti interfész.
- Szabványos elnevezések:
 - Számítógép v. terminál: **adatvégberendezés: DTE** (Data Terminal Equipment);
 - Modem v. **adatátviteli berendezés: DCE** (Data Circuit-Terminating Equipment).
- Fizikai réteg protokoll:
 - mechanikai
 - villamos
 - funkcionális
 - eljárás } interfészek meghatározása.

Az RS-232-C interfész

- **Az RS-232-C az RS-232 harmadik, javított változata**
 - **Alkotója: Electronic Industries Association**
→ EIA RS-232-C
 - **CCITT változata: V.24 (nagyon hasonló) (ENSZ szervezet a CCITT)**

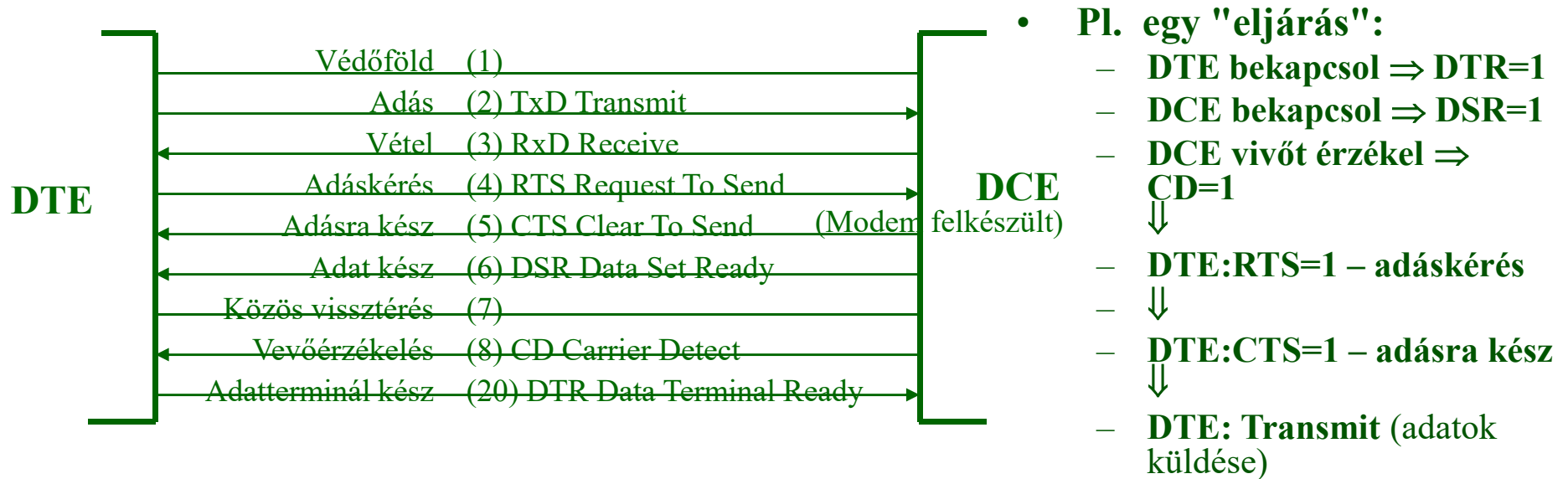
CCITT: Comite Consultatic International de Télégraphique et Téléphonique

Az RS-232-C interfész

- **A mechanikai meghatározás:**
 - **47,04±0,13 mm szélességű 25 tűkés (D-CANNON) csatlakozó**
 - **A felső sor tűi 1-13-ig balról jobbra, az alsó 14-25-ig számozva**
- **A villamos specifikáció**
 - **-3 V-nál kisebb feszültség jelent bináris 1-et,**
 - **+4 V-nál nagyobb: bináris 0-át.**
 - **Legfeljebb 15 m kábelek,**
 - **20 Kbps sebesség a megengedett.**

Az RS-232-C interfész

- **Funkcionális specifikáció:**
a tűkhöz tartozó áramkörök kijelölése és azok leírása

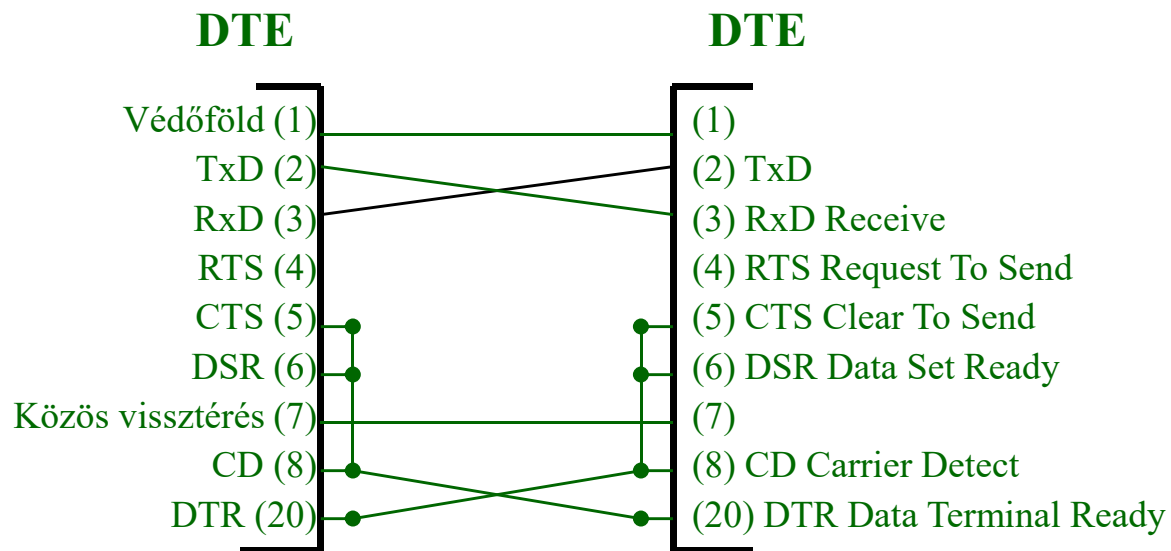


A fel nem tüntetett áramkörök a szinkron átvitelhez tartozó áramkörök, illetve egyéb jelek pl. csengető jel érzékelése

- **Az eljárás specifikáció:**
protokoll, mely az események érvényes sorrendjeit határozza meg.
Akció-reakció eseménypárokra alapul.

A null-modem

- **DTE-DTE kapcsolatra alkalmas kábel:**
 - **R - T/T - R csere és néhány átkötés (és visszahurkolás)**

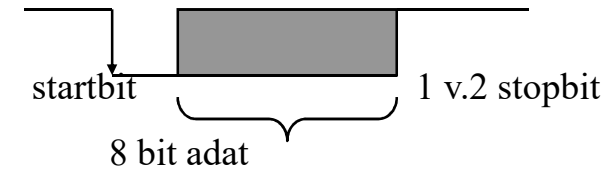


A bitidők behatárolása

- **Aszinkron eset: start-stop bites**

megoldás

- olyan pontos időzítés kell, hogy az átvitt adatok alatt ne essen ki a bitidőből
- szabványos sebességek kellene, amiben a DTE és DCE megegyeznek (vagy kézzel konfigurálnak):
150, 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 38400, 57600, 115200 bit/s.
- A start-stop bites: veszteség (rossz csatornahasználat)



- **Szinkron eset: külön bitidőzítés túske**

- A DCE (modem) diktálja a vétel-adás sebességét
- a DTE-ben nem kell sebességet konfigurálni
- A kerethatárokhoz speciális bitminta (szinkron karakter, vagy flag).

UART: Universal Asynchronous Receiver Transmitter

- **Hasonló az RS232-höz, csak itt szabványos TTL (Tranzisztor - Tranzisztor Logika) jelszinteknek, azaz a**
 - logikai 0: 0 V – 0,8 V
 - logikai 1: 2,4 V – 5 V
- **Alapértelmezetten három vezeték:**
 - RXD: vevő
 - TXD: adó
 - GND közös referencia (földelés)

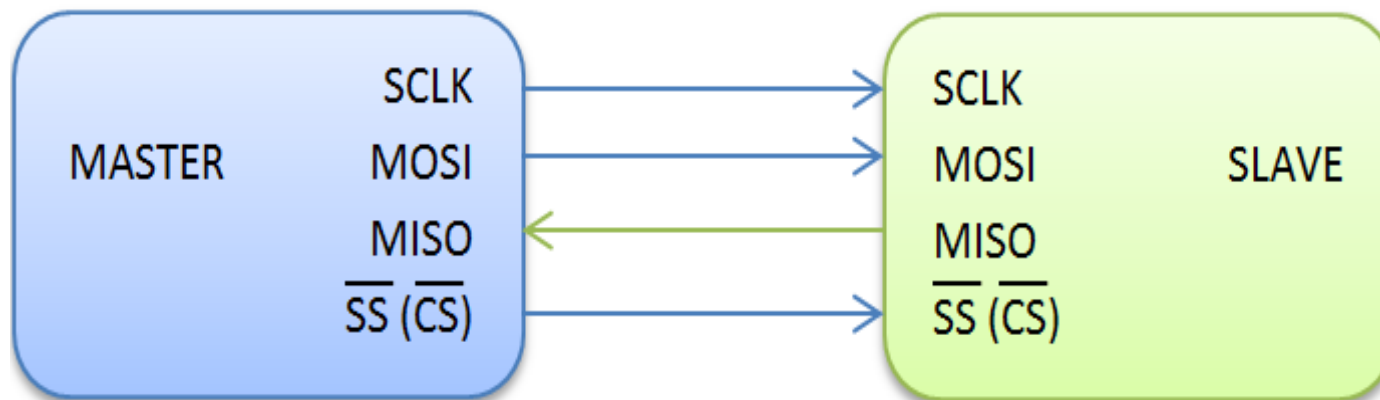
SPI: Serial Peripheral Interface

- **SPI – soros adatátviteli interfész**
- **Full-duplex** (egyidejűleg kétirányú komm.)
- **Szinkron soros** (közös órajel)
- **Master / slave** (egy master - több slave)
- **Rendkívül gyors** (órajel: pár kHz - 100 MHz)

- **Nincs hivatalos szabványa, a Motorola a névadó**

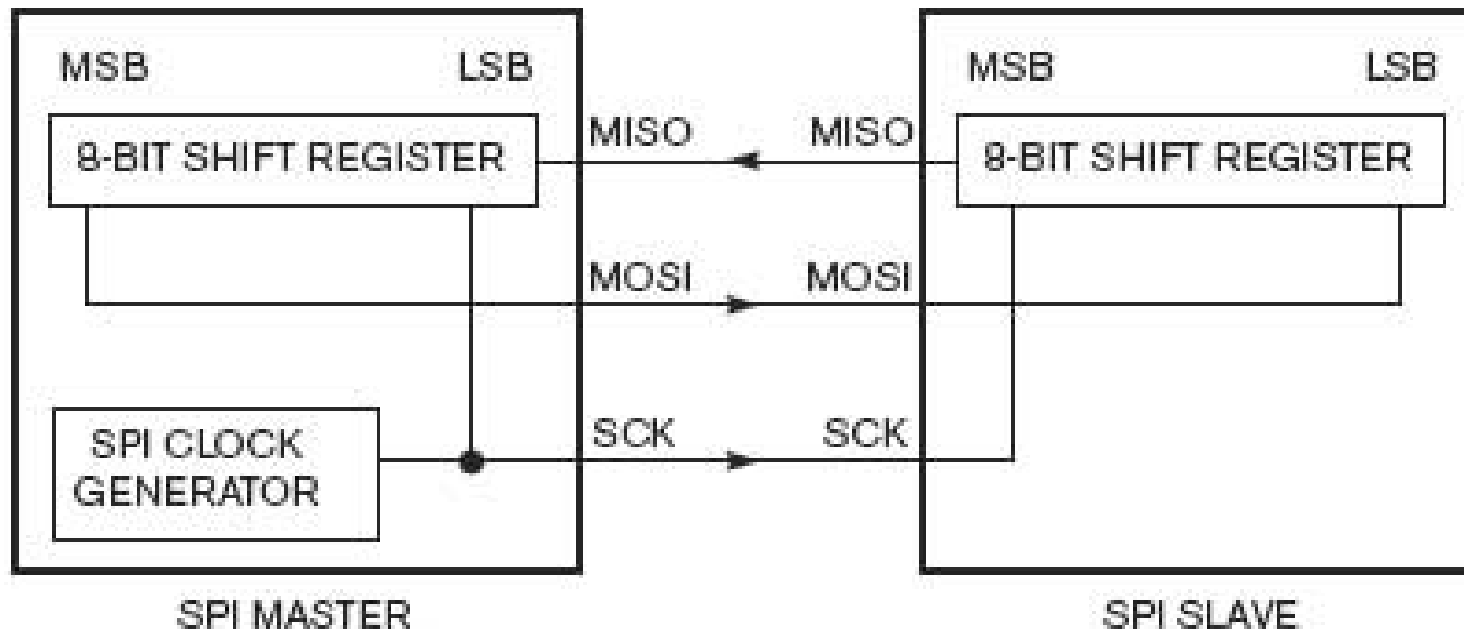
A fizikai kapcsolat

- **SCLK** (serial clock) – órajel
- **MOSI** (master output slave input) – adatátvitel
- **MISO** (master input slave output) – adatátvitel
- **SS** (slave select) vagy **CS** – vezérlés

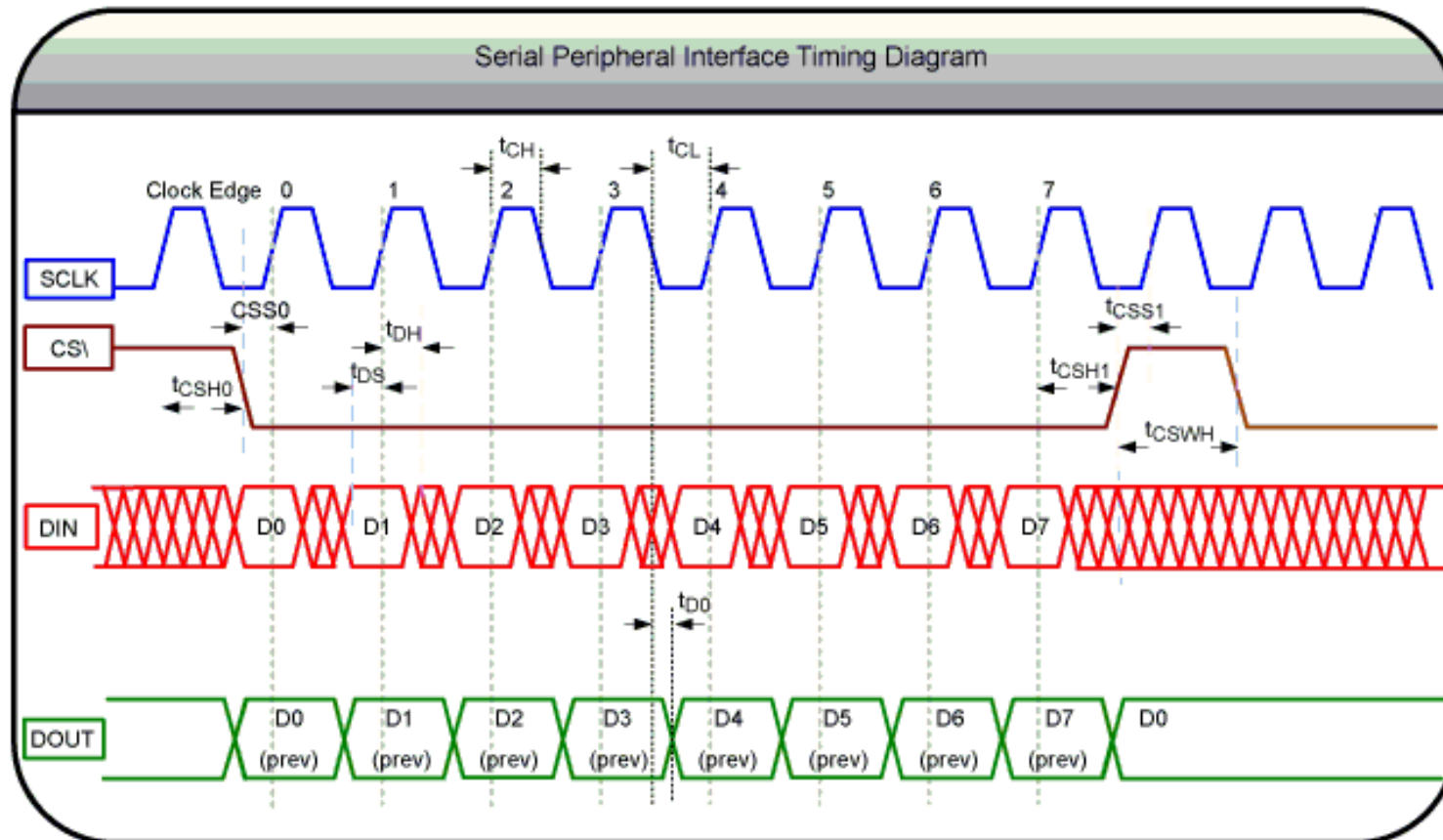


Működése

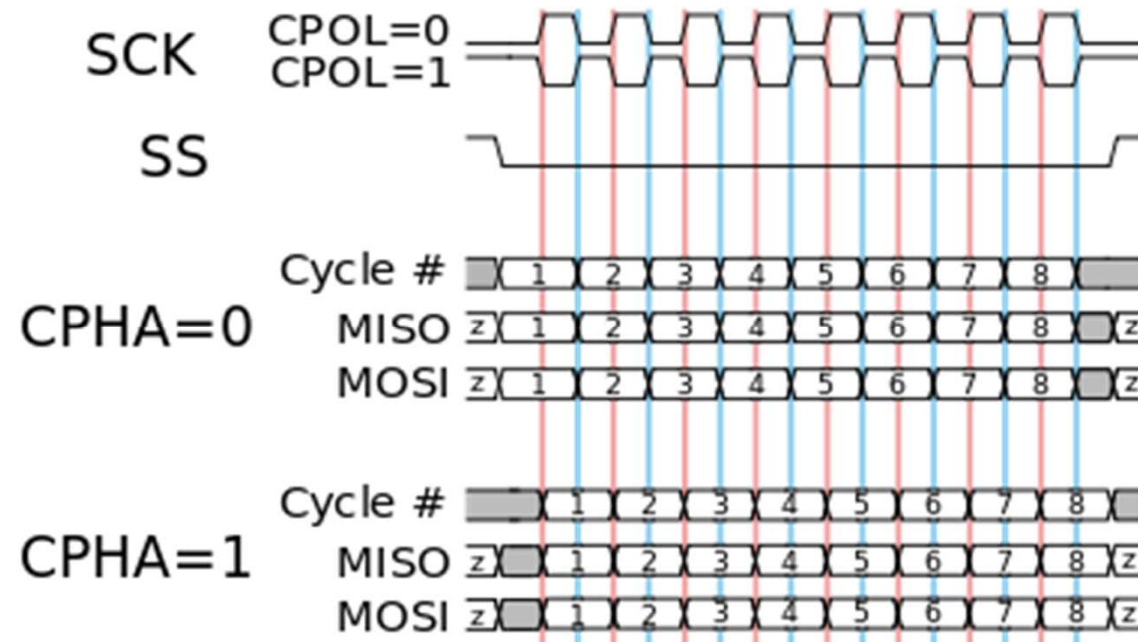
- Konfigurációt a master végzi (frekvencia, CPHA, CPOL, késleltetések), de a slave-hez igazodik
- Master kiválasztja a slave-et: Slave select (SS)
- Master szolgáltatja az órajelet
- Master kilépteti a regiszteréből az adatbiteket, miközben fogadja a slave-től érkezőket



Idődiagram



Idődiagram



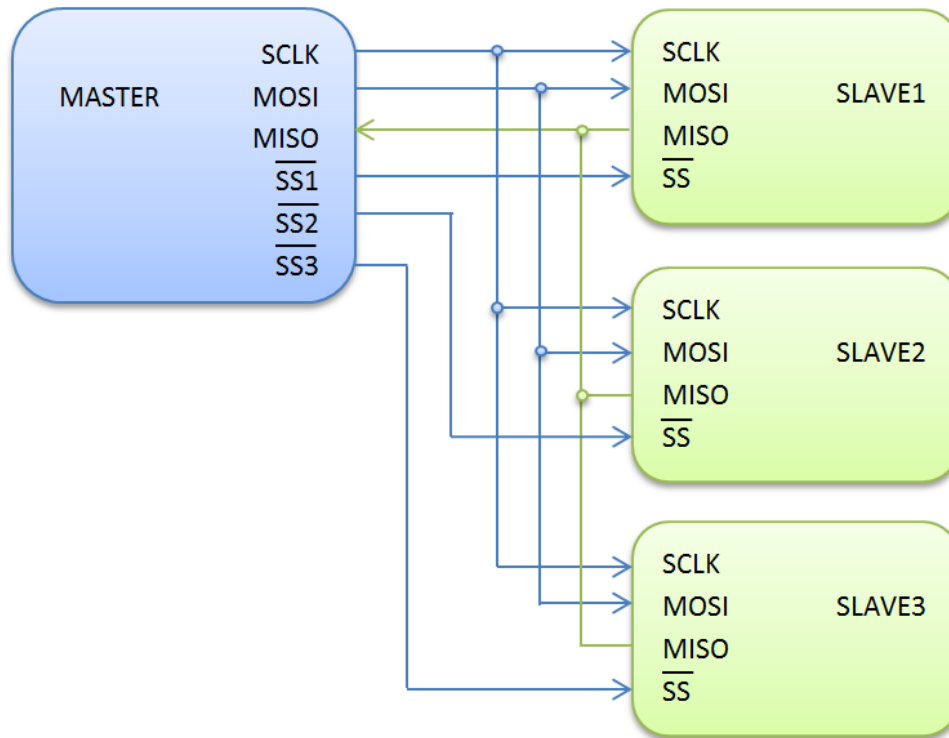
CPOL: Clock Polarity

CPHA: Clock Phase

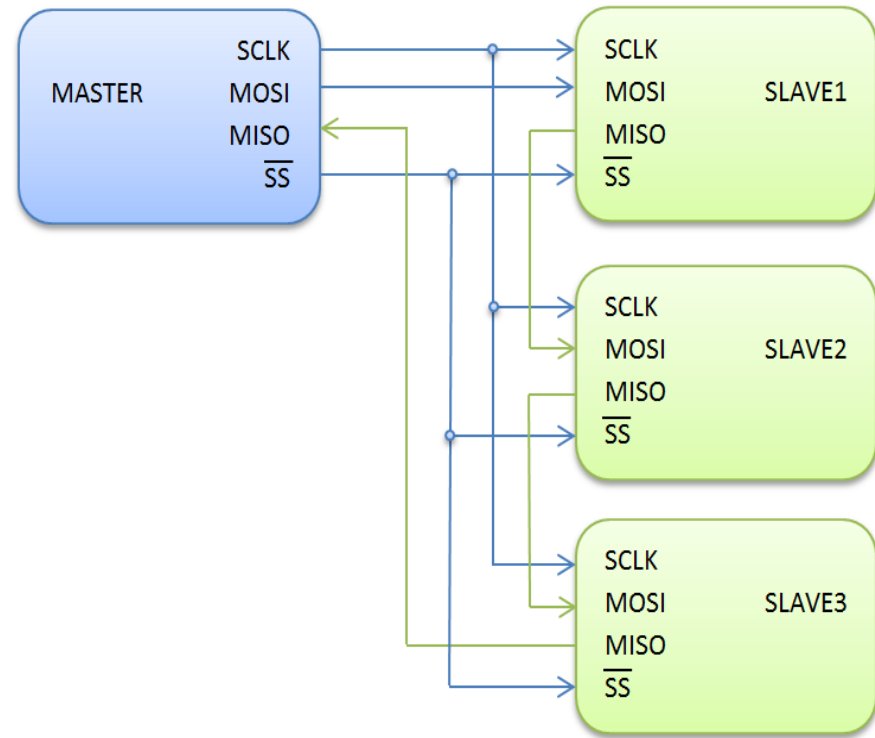
A master konfigurálja a slave-hez igazodva

Topológiák

Normál



Daisy-chained



Előnyök

- **full-duplex kommunikáció**
- **gyors, nagy átviteli sebesség**
- **rugalmas protokoll bitek küldéséhez**
- **nincsenek limitálva az üzenetszavak**
- **egyszerű hardver interfész**
- **nincs szükség transceiver-re**
- **kis teljesítmény felvétel**
- **nincs arbitráció vagy összetett hibamód vizsgálat**
- **nincs szükség precíziós oszcillátorokra, az órajelet a master szolgáltatja**
- **nincs szükség egyedi azonosítóra a slave-eknek**
- **a jelek egyirányúak, így könnyű a galvanikus leválasztás**

Hátrányok

- **egyetlen master lehet csak**
- **nincs hibadetektálás – biztonságkritikus rendszereknél elengedhetetlen lenne**
- **nincs nyugtázás a slave részéről**
- **csak rövid távolságok mellett működik**
- **nincs zajok, zavarok elleni védelem**
- **több vezeték kell, mint a 2 vagy 3 vonalas kommunikációs protokolloknál**

Alkalmazási területe

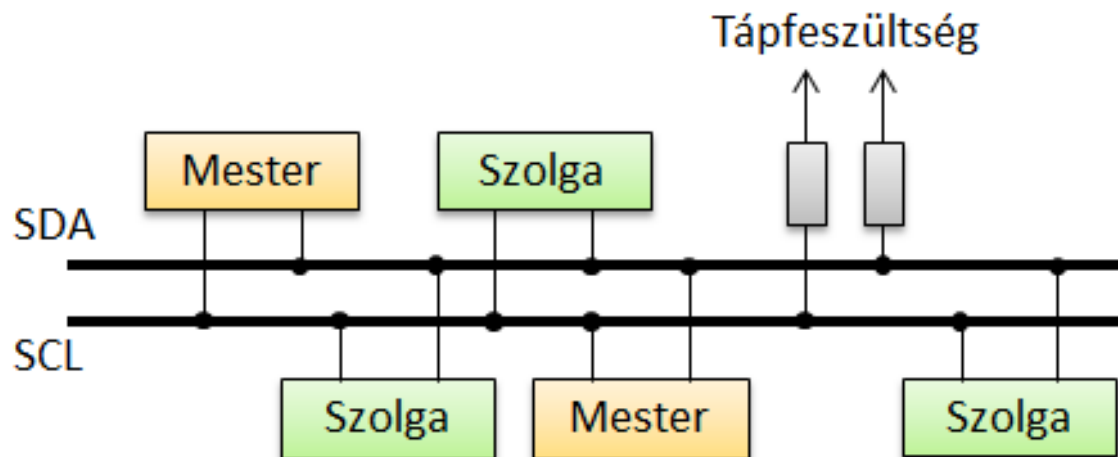
- **konverterek**: ADC (pl. Maxim MAX11200, MAX186), DAC (pl. Maxim MAX5732);
- RTC-k - real-time clock (pl. Maxim DS3234);
- **memóriák**: EEPROM (pl. Atmel AT25512) és Flash (pl. STMicroelectronics M25P32, Atmel AT25FS040);
- **szenzorok**: hőmérséklet (pl. Microchip TC72, Maxim MAX6628), nyomás (pl. MPL115A1); mágneses tér (pl. MLX90363), gyorsulás (pl. Bosch SMB365, Analog Devices ADIS16210);
- **kommunikáció**: UART/RS232 – SPI interfész (pl. MAX3110), CAN controller (pl. MCP2515), Ethernet, USB-SPI interfész (pl. Diolan DLN-4S);
- **egyéb**:
 - digitális potenciométer (pl. Maxim MAX5481);
 - audio mixer (pl. AD1893);
 - audio kódek (pl. ADAV801);
 - érintőképernyő vezérlő (pl. Texas Instruments ADS7846);
 - kapcsolók - SPST, audio/video, video mux (pl. Maxim MAX395, MAX4571, MAX4589);
 - LCD kijelző (pl. NHD-0216K3Z-FL-GBW);
 - erősítő (pl. MAX9939); stb.

I2C: Inter Integrated Circuit

- **Soros, szinkron, fél-duplex kommunikációs protokoll.**
- **Az SPI-hoz hasonlóan megtalálható a legtöbb mikrokontrollerben.**
- **Több mester és több szolga** egységet is kezelni tudó protokoll.
- **A kommunikációhoz két vezetékes:**
- **SCL: szinkron órajel vezeték**
- **SDA: adatjel vezeték**
- **Átviteli sebesség: 10 kbit/s - 3,4 Mbit/s közötti (minden órajel impulzusnál egy bitet továbbít).**

I2C: Inter Integrated Circuit

- A részvevők a két vezetékre nyitott draines, illetve nyitott kollektoros kimenettel csatlakoznak.
- A vezetékek elengedett állapotban magas logikai szinten (felhúzó ellenállások).
- Logikai ÉS kapcsolat a vonalakon.

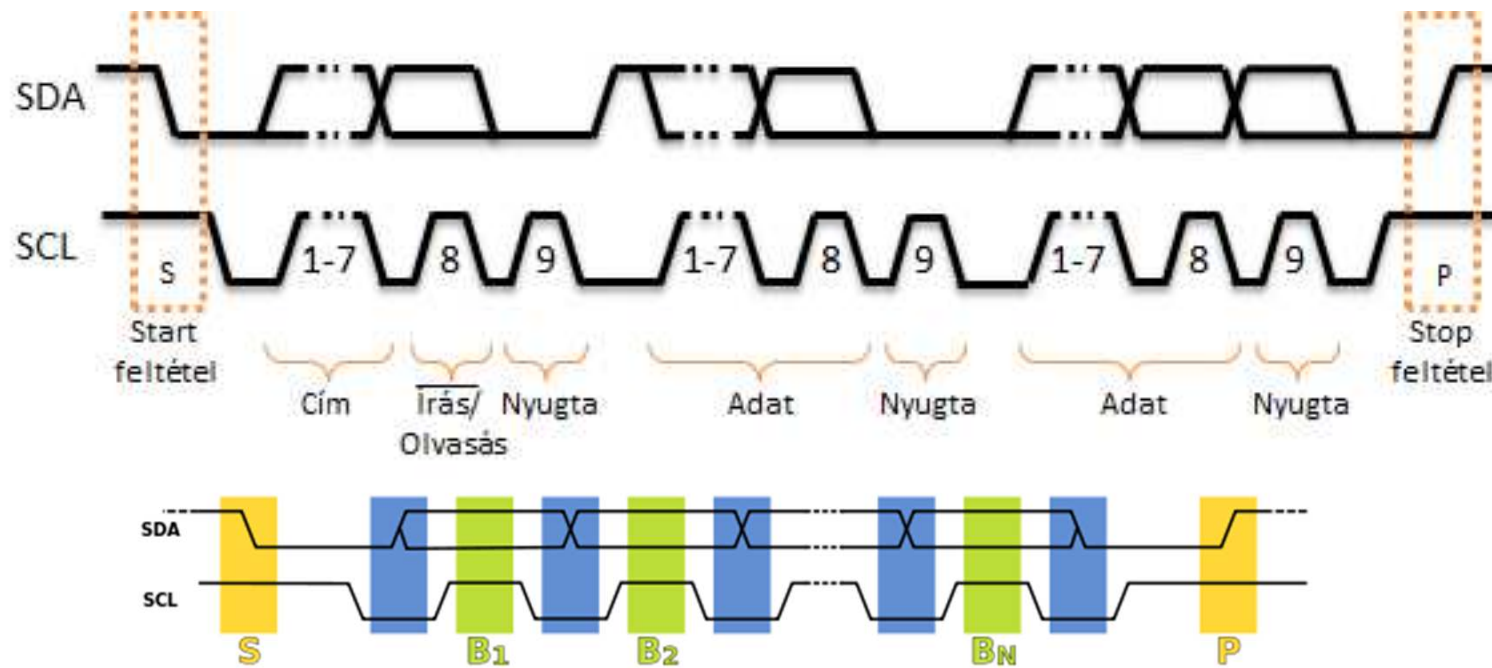


I2C: Inter Integrated Circuit

- **Cím alapú kommunikáció.**
- Minden eszköznek van egy 7 (vagy 10) bites címe.
- A mester vezérli a kommunikációt és szolgáltatja az órajelet.
- **Ha a szolga nem kész a kommunikációra, akkor lent tarthatja az órajelet, késleltetve ezzel a mestert.**
- **Mind a mester, mind a szolga eszközök lehetnek adók illetve vevők is.**
- **Az SDA adatvezeték az adatbitek átvitele alatt mindig az aktuális küldő (adatkivitelnél a mester, adatbeolvasásnál a szolga), a nyugtázás ideje alatt pedig a fogadó egység vezérli.**

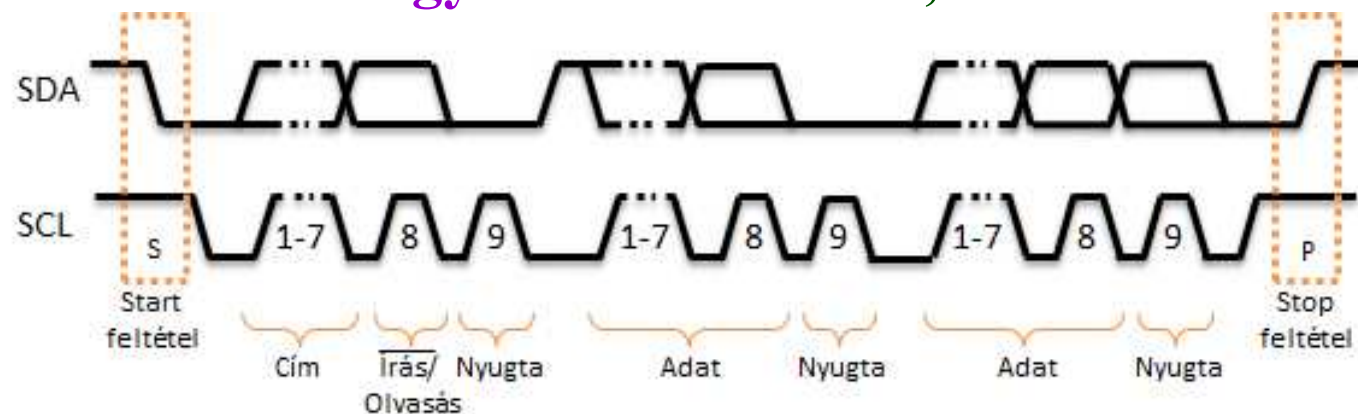
I2C keretek

- Az átvitel keretekben történik.
- Keret start, stop jelzése: az SCL órajel magas értéke alatt vált az SDA állapotot (egyébként SCL magas alatt az SDA stabil).



I2C kommunikáció

- **Mester adatot küld az egyik szolga eszköznek:**
 - a mester megcímzi a céleszközt (lehet mindenkinek szóló cím is) majd
 - adatokat küld, és végül
 - lezárja az átvitelt.
- **Mester olvas a szolga eszköztől:**
 - a mester megcímzi a céleszközt, majd
 - adatokat fogad a szolga felől, és végül
 - lezárja az átvitelt.
- **Azt, hogy a mester írni vagy olvasni szeretne, a cím utáni első bittel jelzi.**



I2C órajel szinkronizáció

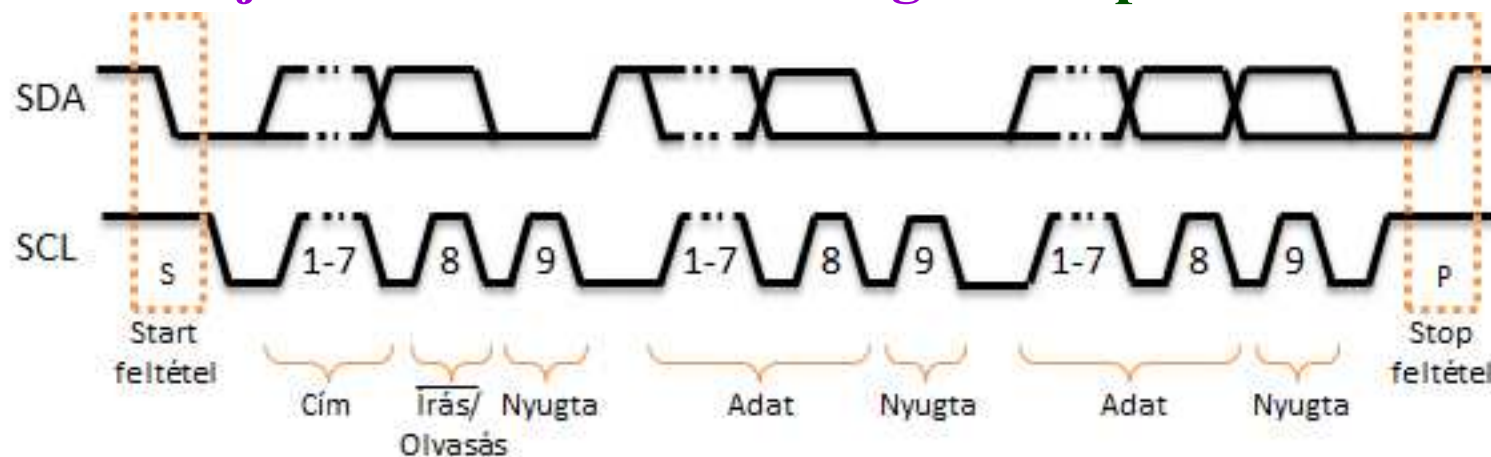
- **Több mester is lehet.**
- **Gond: órajel szinkronizáció a mesterek között:**
 - **Az SCL vezeték a leghosszabb alacsony periódusú eszköz tartja alacsonyan.**
 - **Amikor minden érintett eszköz az alacsony periódusa végére ért, az órajel vezeték felszabadul, és magas logikai állapotba kerül.**
 - **Ekkor minden eszköz elkezdi kiszámolni a magas periódusát. Az első eszköz, amely végzett a magas periódusával, ismét alacsonyra húzza az SCL vezetékét.**
- **Az SCL vezetéken egy *szinkronizált órajel* áll elő, melynek az alacsony periódusát a leghosszabb alacsony periódusú órajel, a magas periódusát az egyik legrövidebb magas periódusú órajel határozza meg.**

I2C mester arbitráció

- **Versenyhelyzet: ha kettő vagy több mester próbál információt küldeni, azaz a start feltétel minimális tartási idején belül több mester is generál egy start feltételt.**
- **Az arbitráció úgy zajlik, hogy az első olyan mester elveszti az arbitrációt, amelyik logikai 1-et akar küldeni, miközben a többiek 0-át.**
- **Ha a címbitek összehasonlításánál nincs különbség, akkor az arbitráció az adatbitekkel folytatódik a további részeken, így az arbitráció során nincs információvesztés.**

I2C kommunikáció

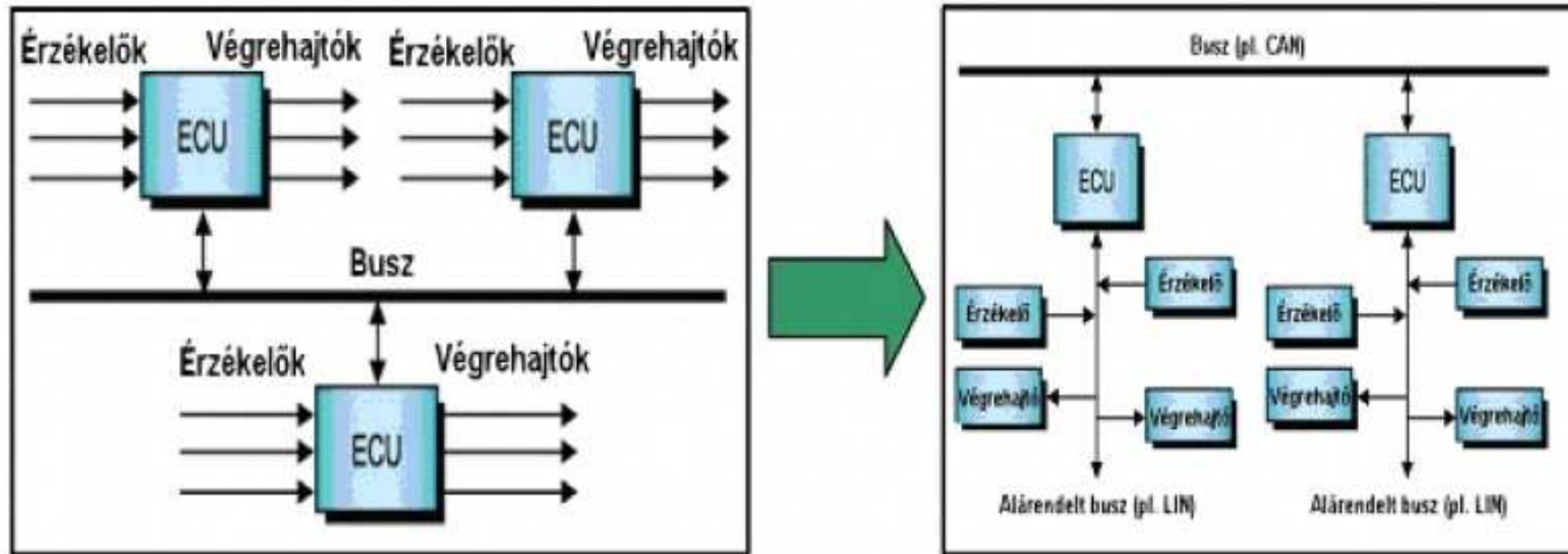
- A mester kezdeményezi az adatátvitelt a start fázissal.
- Kiírja a címet az adatvonalra.
- A cím utáni egy bites vezérlő jellel megadja, hogy olvasni vagy írni szeretne-e az adott szolgáltól.
- A szolga nyugtázza a címének vételét (0:ACK, 1:NACK).
- Ezután következik az írási vagy olvasási ciklus.
- A mester jelzi az adattranszfer végét a stop fázissal.



LIN: Local Interconnect Network

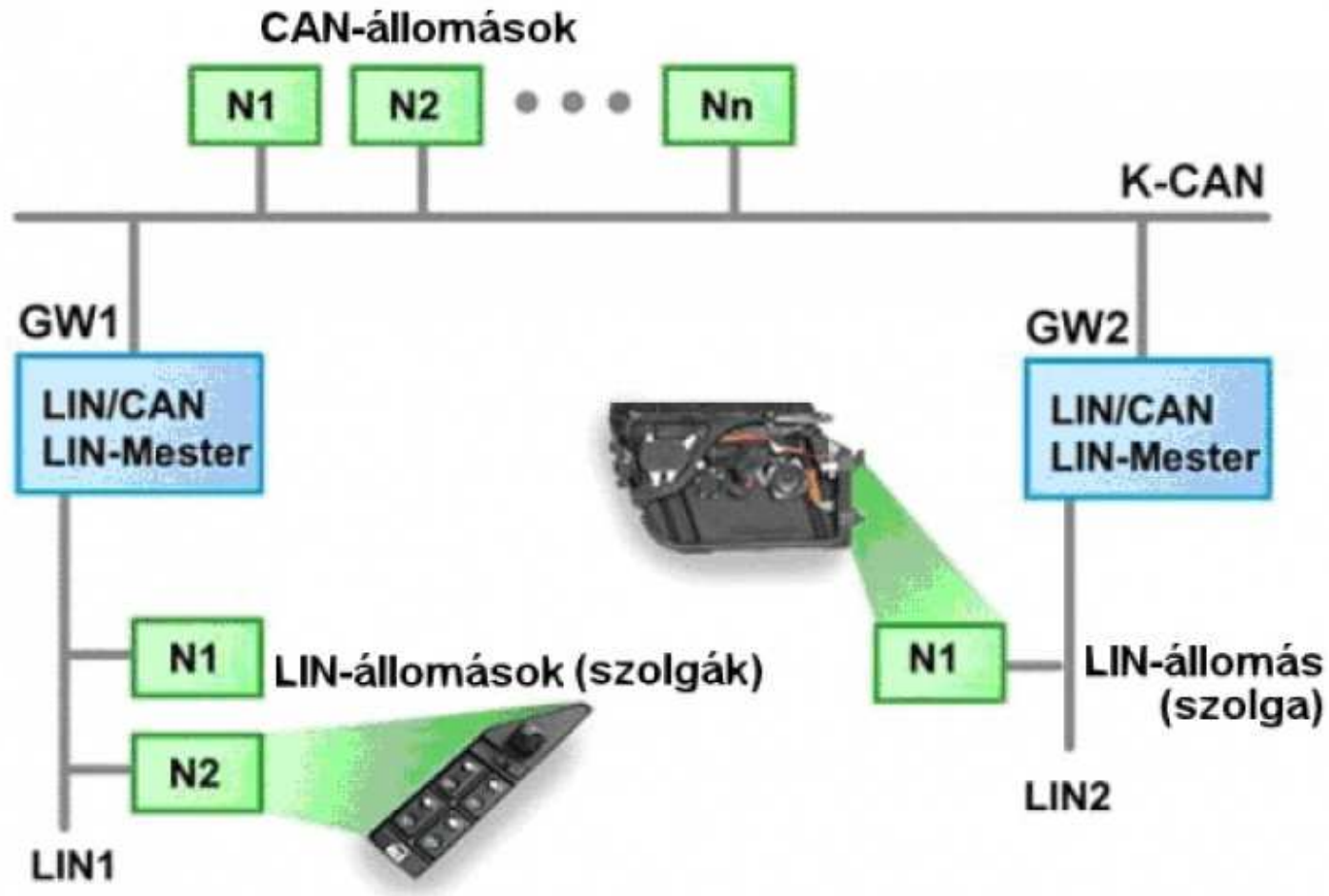
- **Az 1999-es évben néhány VLITE buszt használó autóiipari vállalat hatására került sor a LIN (Local Interconnect Network) első változatának (LIN 1.0) kidolgozására. A LIN kommunikációs protokoll szabványosításával az akkor választékosan sokrétű megoldást alkalmazó autóiipari kis hálózatoknál a fejlesztés, a termék előállítás, a javítás és a logisztika erőforrásigényei kisebbek lettek. A LIN alkalmazása a mai napig az autóiiparban leginkább a hierarchikusan felépülő elektronikai eszközök rendszerében alul elhelyezkedő, kis csomópontszámmal rendelkező elemi (al)hálózatokra terjed ki.**

Mi az a LIN?



Bár a CAN busrendszer megbízható, gyors, és lényegesen olcsóbb a külön kábelekkal történő kommunikációnál, még hagyott réseket a felhasználásban. A jármű azon részén, ahol nincs szükség feltétlen a gyors kommunikációra, további kábelhosszokat lehet megspórolni egy, a CAN-nek alárendelt, az érzékelők és beavatkozók kommunikációját buszrendszerben egyesítő megoldással. Ez a buszrendszer a LIN.

A LIN szerepe



A LIN alapvetően a komfort oldalon kerül használatra, ahol nincs szükség a gyors kommunikációra, és nem kerül veszélybe az utasok épsége egy esetleges buszrendszer leállás következtében fellépő irányíthatatlanság miatt.

A LIN rendszerek összeköttetésben állnak a többi vezérlőegység CAN hálózatával, de többnyire önállóan látja el feladatát, a CAN rendszeren csak közli a többi egységgel a számukra fontos információkat.

A LIN fő ismérvei

- Egyvonalas busz (ISO 9141), mehet akár a tápvezetéken is (drasztikus érszám csökkenés)
- Definiált meredekségű jelváltozás ($2V/\mu s$) - EMC
- Korlátolt sebesség (20kBit/s), vonalhossz (40m) és állomásszám (16db)
- RC oszcillátor (olcsó)
- Master-Slave kapcsolat (Single Master)
- Licenstdíj mentes (a CAN licenstdíjas)
- Időrendi táblázat

A LIN alkalmazási területei

- A LIN szempontjából a **Kényelmi/Komfort elektronika** a legfontosabb felhasználási terület. Ennek két fő oka, hogy e szinten lévő eszközöknél megengedhető az **emberi reakcióidővel összemérhető válaszidő**, működési tartomány, valamint ezen eszközök meghibásodása, vagy egy-egy – vagy akár az összes – alhálózat teljes kiesése sem veszélyezteti a jármű menetbiztonságát, nem eredményezi a jármű mozgásképeségének megszűnését.
- A LIN jellemzően az alábbi funkcióknál kerül alkalmazásra:
 - Elektromos ablakemelő
 - Ajtózár, központi zár
 - Fűtésrendszer vezérlés
 - Elektronikusan állítható visszapillantó tükör

Szabványosítás

- **Az 1999-es megjelenése után a LIN szabványon a 2000-es évben kétszer is módosítottak, így a LIN 1.2 2000 Novemberében látott napvilágot. Két évvel később, 2002-ben a LIN Konzorcium megjelentette a LIN 1.3 szabványt, melyben a változtatások a LIN *Fizikai réteget* célozták meg a csomópontok összeférhetősége érdekében.**
- **A gyakorlati tapasztalatok vezettek a következő lépéshez, melyet a 2006-os LIN 2.1 jelentett, ahol a korábbi verziókkal való kompatibilitás érdekében az egyes alkotórészek pontos funkcionalitása került letisztázásra, megkötéseket vezettek be és egyes elemeket kivettek a specifikációból.**

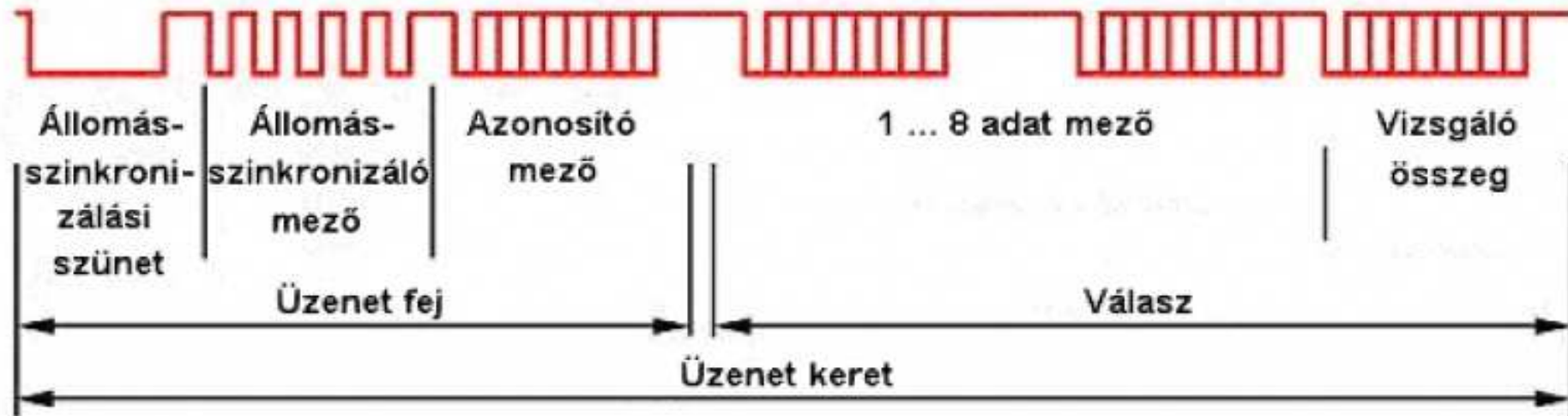
A legújabb, LIN 2.2 szabvány

- A LIN 2.2 szabványban megfogalmazott *mester* csomópont képes kezelni mind a LIN 1.3 *szolgákat* és a LIN 2.2 *szolgákat*. A LIN 2.2 *mester* így a LIN 1.3 szabvány szerint működő *szolga* csomópontoknál bizonyos funkciókat nem vár el:
 - Ellenőrző összeg (checksum) megnövelése,
 - Újrakonfigurálás és diagnosztika,
 - Automatikus átviteli sebesség (baud rate) felismerése,
 - „Hiba a válaszrésznel” állapot megfigyelése.
- A LIN 2.2 szabvány szerint működő *szolga* csomópont azonban nem képes a LIN 1.3 *szolga* csomópont kezelésére (mivel a LIN 2.2-nél a LIN 1.3-hoz képest megnövelték az *ellenőrzőösszeg* nagyságát).

LIN szabványok összefoglalása

Specifikáció	Megjelenés ideje	Leírás
LIN 1.0	1999-07-01	A specifikáció kezdeti verziója.
LIN 1.1	2000-03-06	
LIN 1.2	2000-11-17	
LIN 1.3	2002-12-13	
LIN 2.0	2003-09-16	Fő áttekintő lépés
LIN 2.1	2006-11-24	Funkcionális tisztázás, konfigurációs módosítások, Szállítási réteg növelése és diagnosztikai funkciók bevétele
LIN 2.2	2010-12-31	Frissítve a LIN 2.1 hibajegyzéke (errata) alapján, bit mintavételezésének megkötésein lazítások
LIN 2.2A	2010-12-31	A felébresztő jel javítása (LIN 2.2 2.6.2-es fejezetében)

Az üzenet felépítése



Üzenet fej (a Master küldi):

-Első rész: állomások figyelmeztetése, hogy üzenet fog érkezni: 1 Start bit (domináns), 13 domináns bit, 1 Stop bit (recesszív)

-Második rész: ütemfrekvencia szinkronizálás, a pontatlan RC oszcillátorok belövése a Master kvarc oszcillátorának frekvenciájára (Start – 0x55 – Stop)

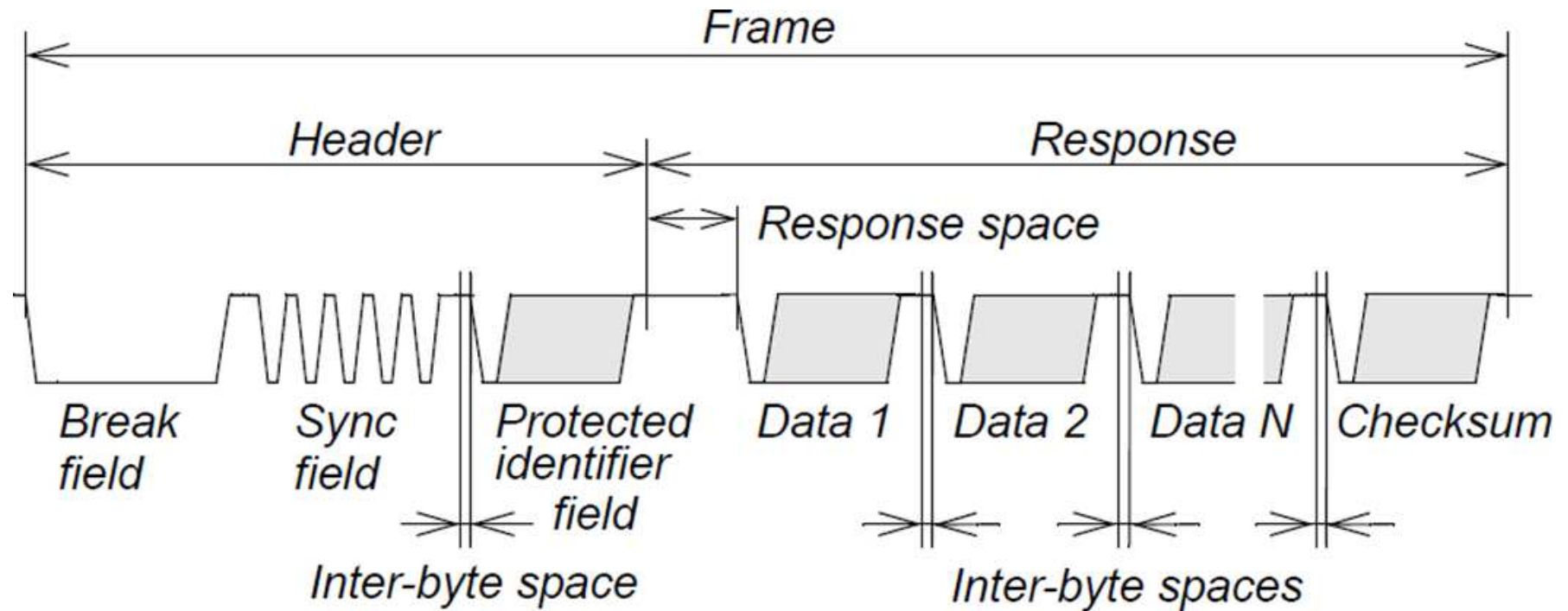
-Harmadik rész: azonosító mező, itt adja meg a master, hogy melyik állomástól vár és milyen információt (az üzenet típusát jelöli, nem az állomást választja ki (“mit”, nem “kinek”)), valamint a paritásbiteket is itt közli

Válasz (vagy a Master adja ezt is, vagy az egyik állomás küldi):

-2, 4, vagy 8 bájt adat

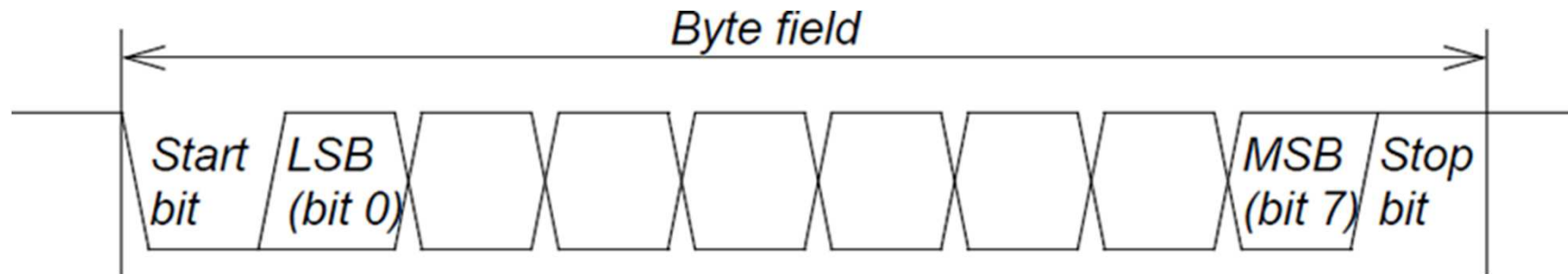
-Ellenőrző összeg: ellenőrzi, hogy az adatok megfelelően lettek-e átküldve

A LIN üzenet felépítése

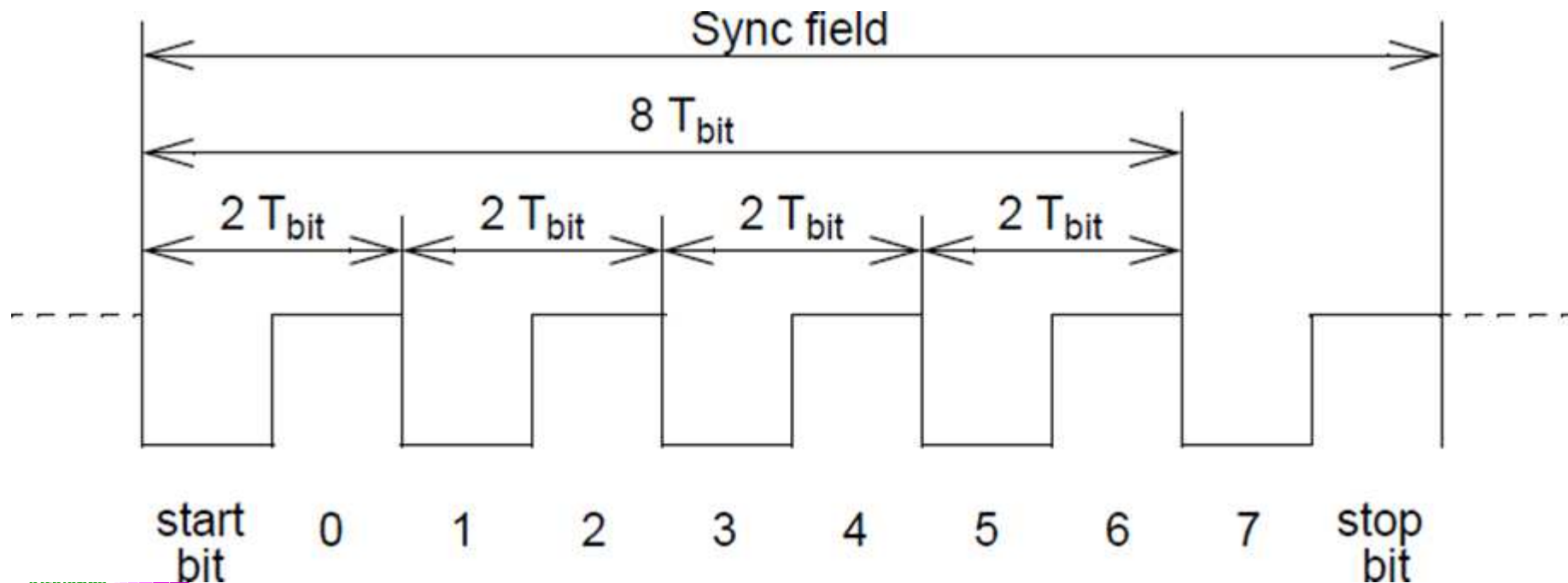


A LIN üzenet felépítése

- A bájmezők szerkezete:

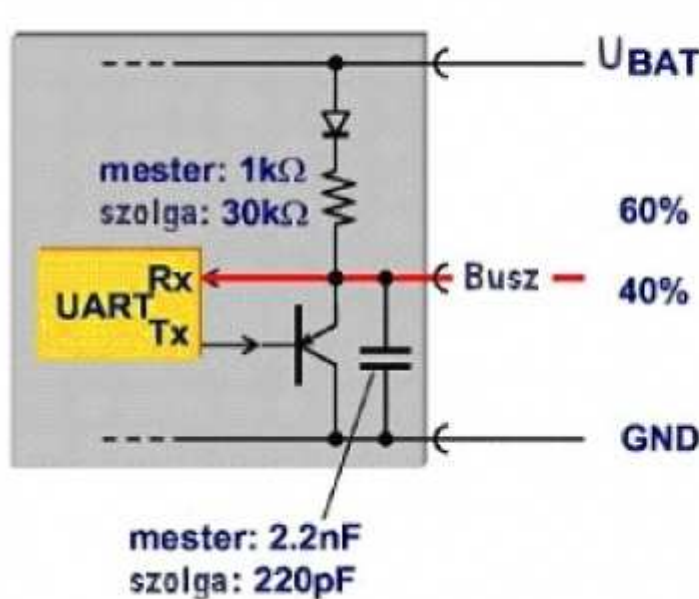


- A szinkronizációs mező struktúrája: '0x55'

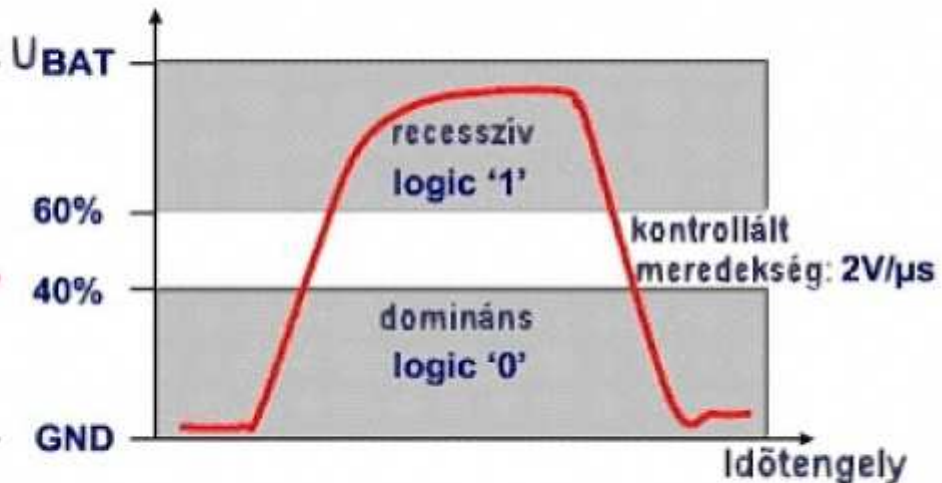


Fizikai felépítés

Vonalmeghajtó (Transceiver)



Busz-feszültség



Transceiver: üzenetek küldése és fogadása

LIN Controller: felhasználja a kapott üzeneteket, kiadja az elküldendő üzeneteket

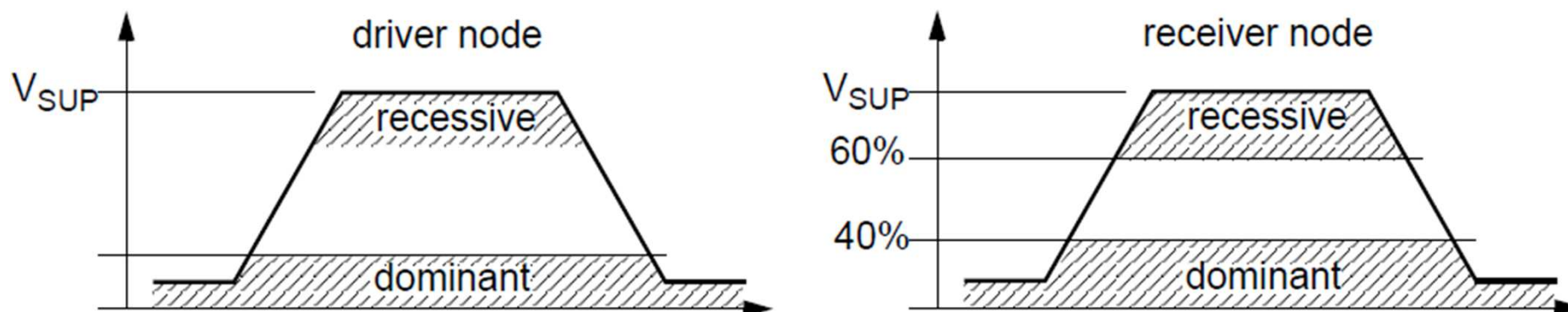
A jelszintek 0V és 12V közti értékek. A logikai 0 és 1 bitek a jel küldésekkor 9,6V-12V és 0V-2,4V közti értékek, fogadásakor 7,2-12V és 0-4,8 V közt minősíthetők elfogadhatónak.

A jelszint váltáskor a feszültségváltozás meredeksége 1-3V/us, de többnyire 2V/us értékű, ezzel csökkentve a kisugárzott elektromágneses tér energiáját, ami ha túl magas értékű, zavarja a környezet kommunikációját.

A Transceiver nyitott kollektoros. Az ellenállás értéke a Master esetén 1kOhm, a Slave-eknél 30kOhm.

A párhuzamosan kapacitás Master esetén 2,2-2,5nF míg a Slave-eknél 220-250pF.

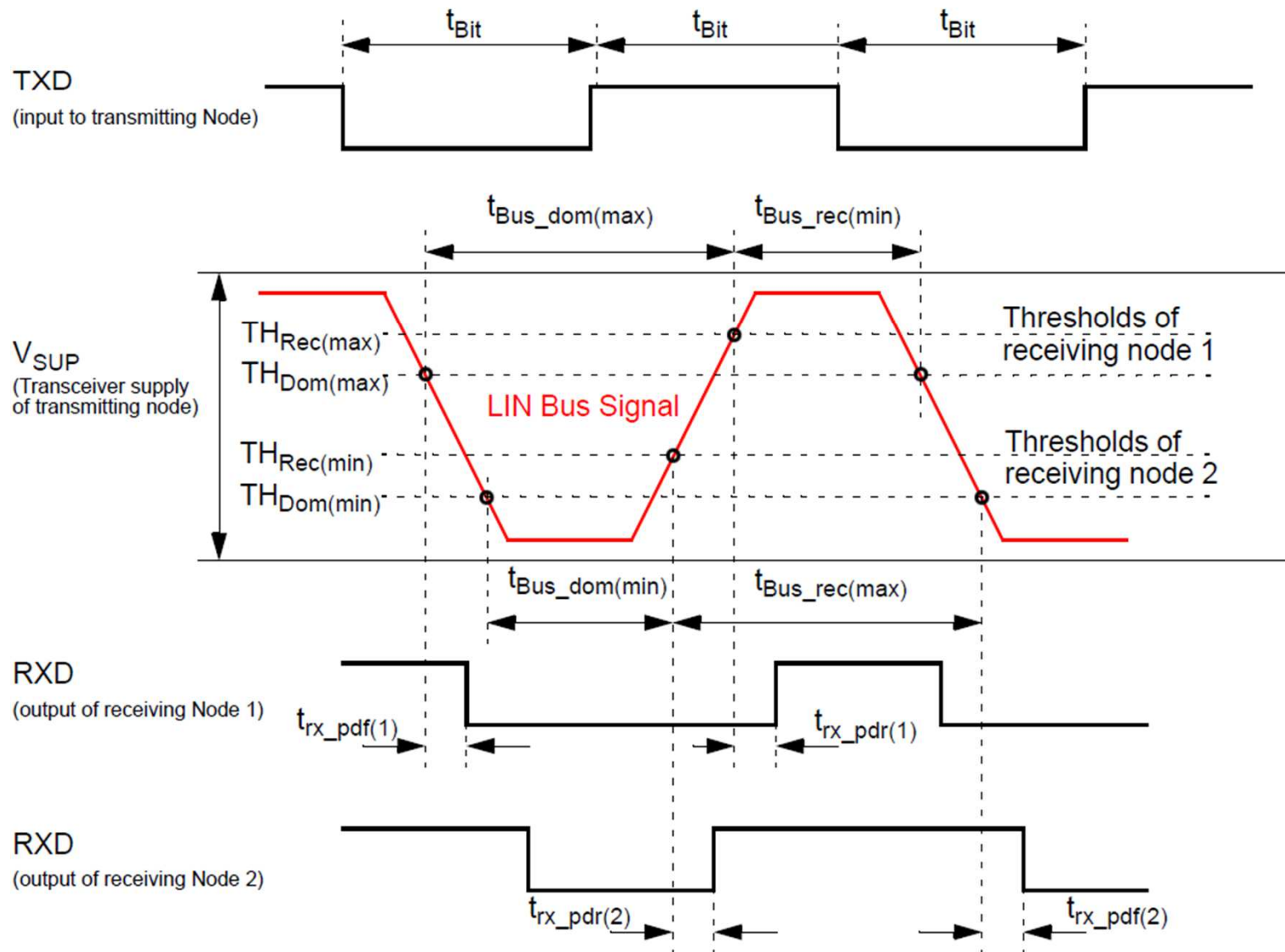
Recesszív és domináns bitnek vett feszültség szintek



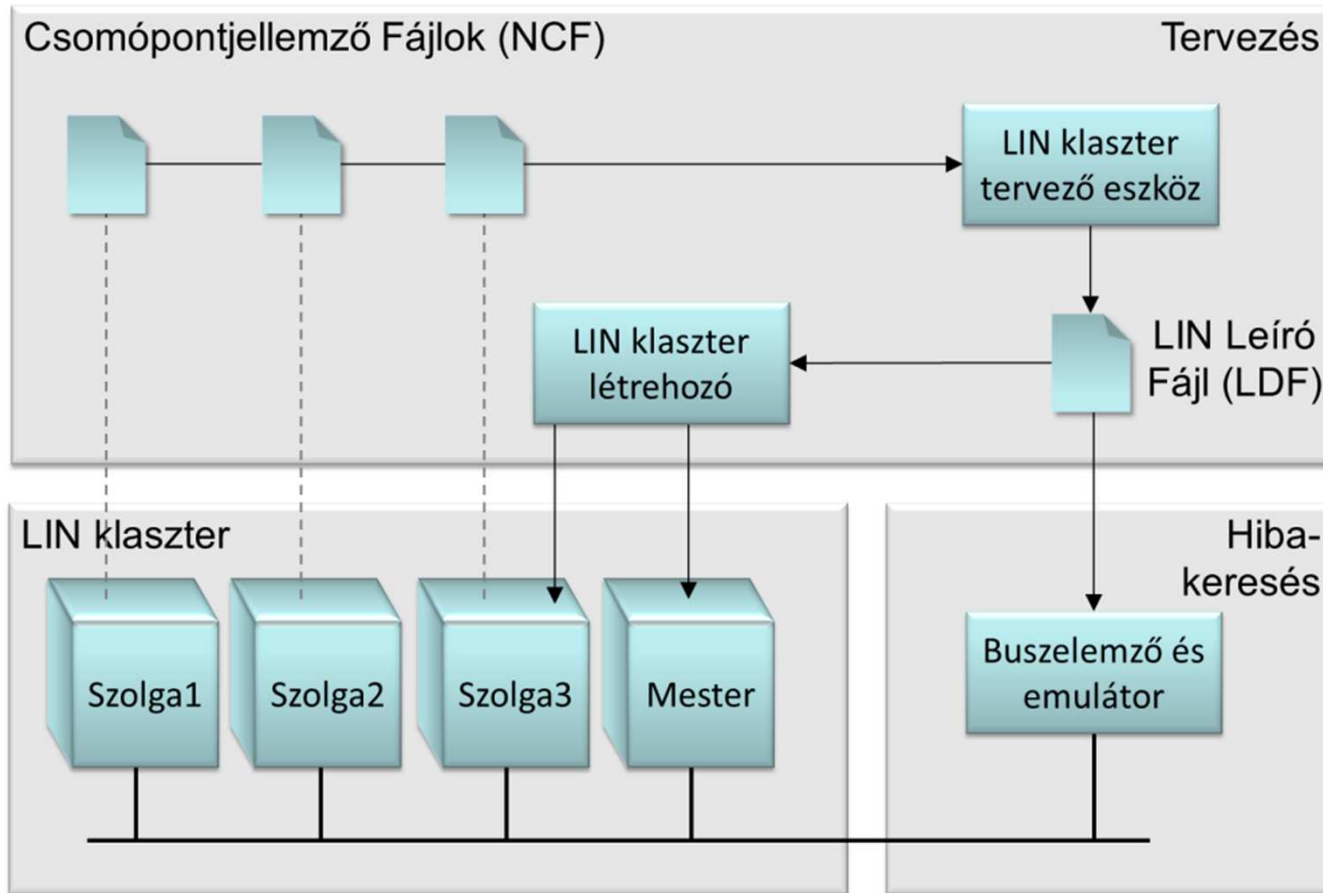
Egy bit megfelelő küldése és fogadása érdekében a fogadó fél mintavételezésének időpontjában elengedhetetlen a jel megfelelő feszültség szinten (domináns vagy recesszív) tartása.

Ezért figyelembe kell venni a földpont eltolódását, a tápfeszültség feszültségcsökkenését, a gyakran/rendszeresen előforduló hibákat és a *terjedési késleltetést*.

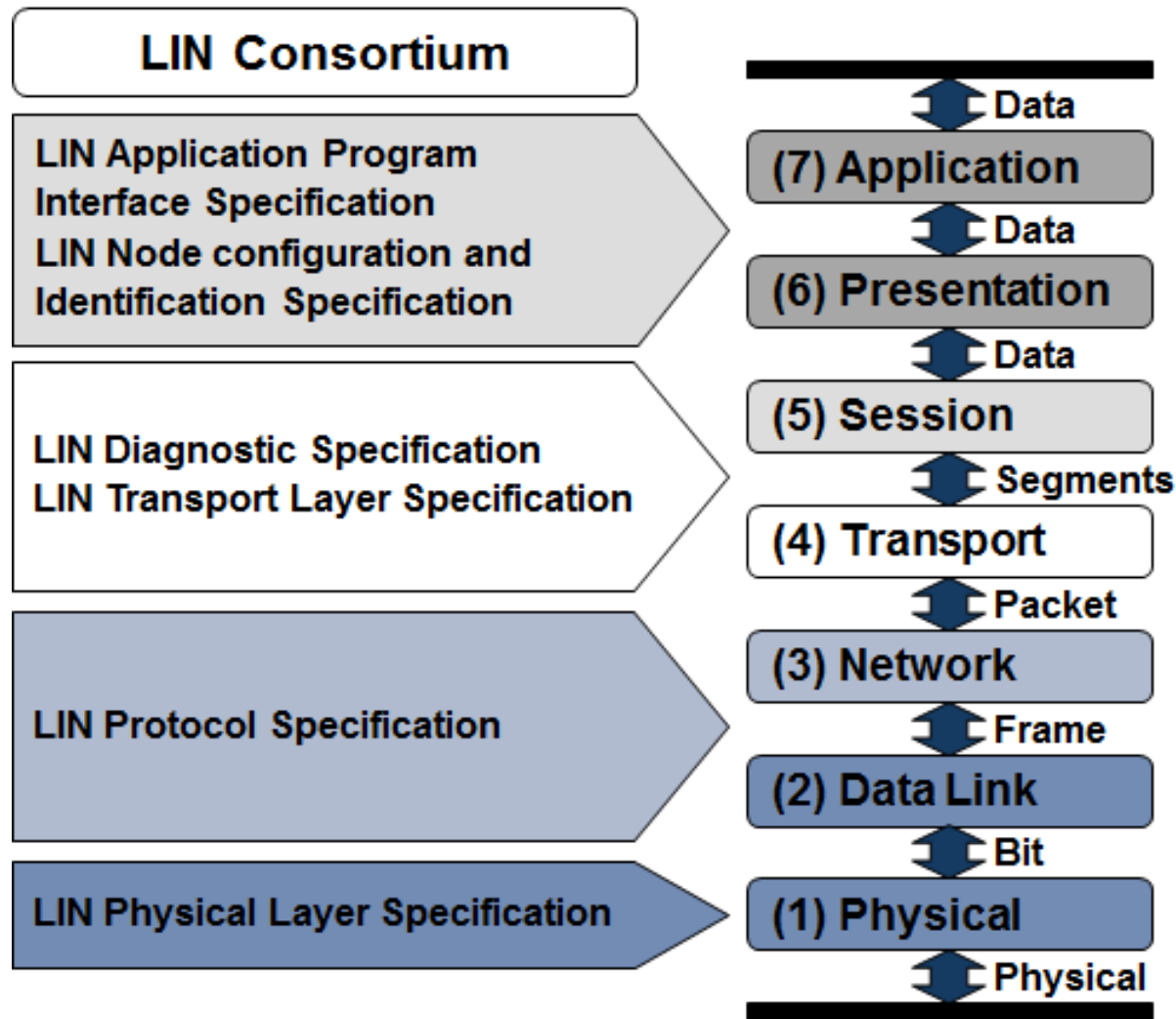
„Időzítési diagram” - busz időzítés



A LIN klaszter tervezési folyamata



A LIN kommunikációs modellje

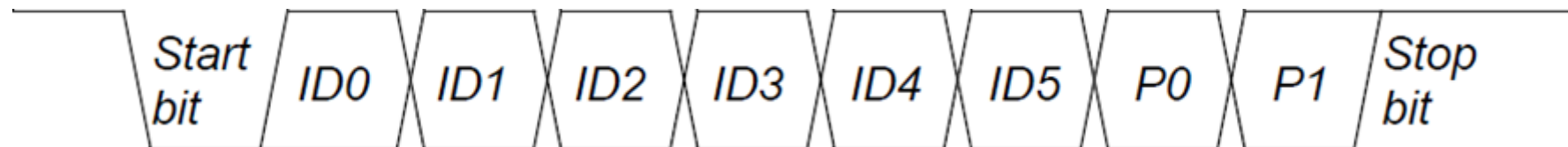


LIN ütemező táblázatok

- **A LIN kommunikációs protokoll kulcstulajdonsága az ütemező táblázatok (schedule tables) használata. E táblázatok biztosítják, hogy a busz túlterhelése soha sem következik be, valamint kulcsösszetevőként garantálják a jelek periodikusságát.**
- **A mester csomópont feladata, hogy egy működési módban fontos összes üzenetnél rendelkezésre álljon a továbbításukhoz szükséges idő.**
- **A mester az ütemező táblázat alapján állítja a védett azonosító mezőket, létrehozva egy üzenethelyet (frame slot).**
- **A fejlécre a válaszrész mindig az üzenet közzétevő csomópontja biztosítja.**
- **Az üzenet összes feliratkozó csomópontjának fogadnia kell az üzenetet.**

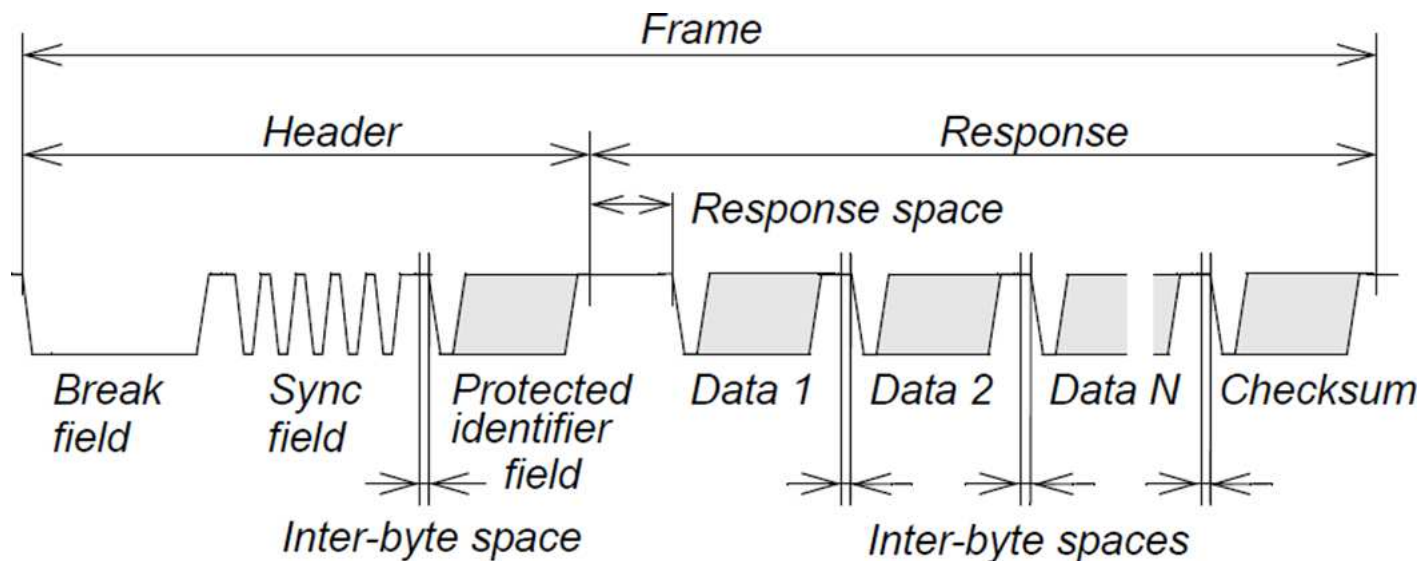
LIN védett azonosító mező

- A védett azonosító mező két kisebb részre osztható: az üzenet azonosítót (frame identifier) és a paritás bitek (parity bits). A 8 bit hosszú védett azonosító mezőből a 0-5 bitek tartalmazzák a 6 bit hosszú üzenet azonosítót, míg a 6. és 7. bitek a paritás bitek.
- Az üzenet azonosító 6 bitje $2^6=64$ darab különböző üzenet megkülönböztetését engedi egy LIN hálózaton. Ezen azonosító alapján az üzeneteket három csoportba lehet sorolni:
- 0-59 (0x00 – 0x3B) közötti értékek a jelhordozó üzeneteknek. 60 (0x3C) és 61 (0x3D) értékek a diagnosztikai és konfigurációs adatok jelzésére szolgálnak. A 60-as a mester kérőüzenet, míg a 61-es szolga válaszüzenet jelöli. 62 (0x3E) és 63 (0x3F) értékek a jövőbeli felhasználás érdekében foglalt azonosítók.



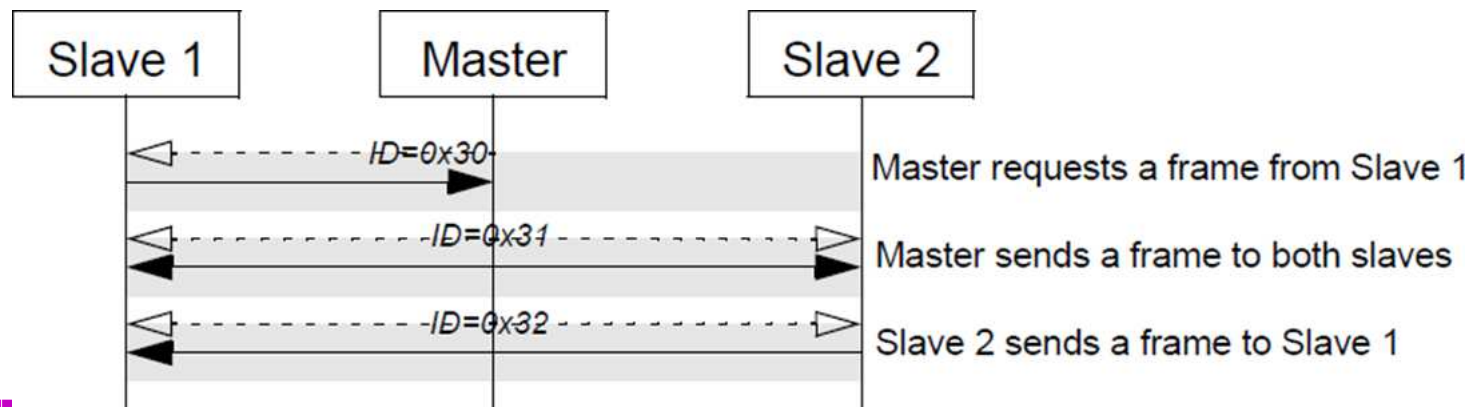
LIN adatmezők

- **Egyetlen egy üzenet 1-től 8 bájtig terjedő adat közvetítésére képes. Az, hogy egy adott azonosítóval rendelkező üzenet mennyi adatot tartalmaz, le van rögzítve a közzétevő, és az összes feliratkozó csomópontnál.**
- **Az üzenet utolsó, váró mezője az ellenőrzőösszeg mező.**



LIN általános üzenet küldése

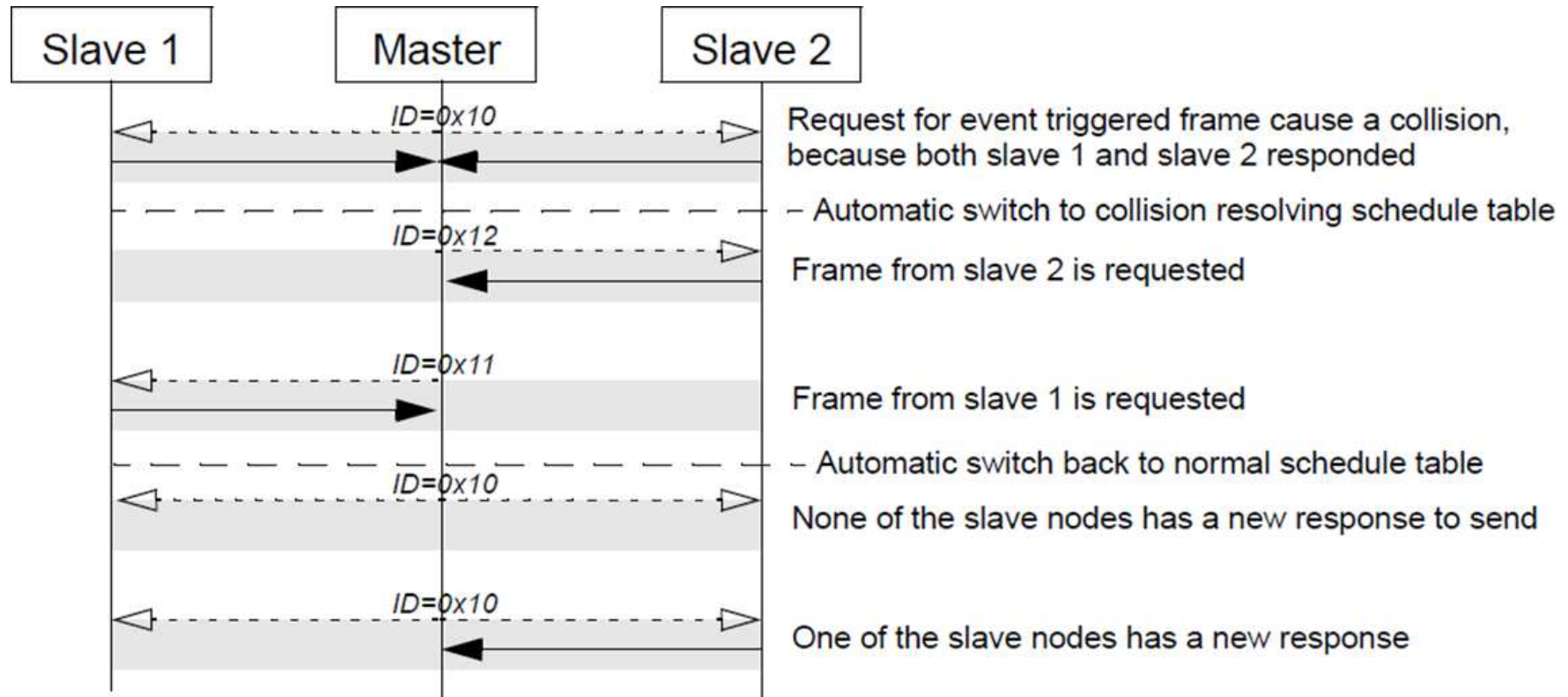
- Egy üzenet akkor minősül általános üzenetnek (unconditional frame), ha az azonosítójának értéke 0 és 59 (0x00 – 0x3B) között van.
- Egy általános üzenet fejléce mindig akkor kerül továbbításra, amikor a mester folyamat megkezdje a megfelelő üzenethely (frame slot) feldolgozását.
- A fejlécre a válaszrészt mindig az általános üzenet közzétevője (mindig a szolga folyamat) biztosítja.
- Az általános üzenet összes feliratkozójának fogadnia kell az üzenetet és azt elérhetővé kell tennie az alkalmazása számára (feltételezve, hogy nem következett be hiba a kommunikáció során).



LIN eseményvezérelt üzenet küldése

- Egy eseményvezérelt üzenet fejléce akkor kerül elküldésre, amikor az eseményvezérelt üzenet számára fenntartott üzenethely (frame slot) feldolgozása megkezdődik.
- Ha a szolga csomópontok egyike sem válaszol a fejlécre, akkor a fennmaradó üzenethely üresen marad (hallgatnak a csomópontok, pl. nincs frissítendő jel egyik csomópontnál sem).
- Ha több mint egy szolga csomópont válaszol a fejlécre ugyanabban a kijelölt üzenethelyen, akkor ütközés fog bekövetkezni.
- Ütközés megoldása: A mester csomópontnak át kell váltania egy ütközésmegoldó ütemező táblázatra (collision resolving schedule table), ezért minden egyes eseményvezérelt üzenetnek van egy társított ütemező táblázata. Majd annak végeztével vissza kell váltania az előző ütemező táblázatra.

LIN eseményvezérelt üzenet küldése



LIN Network management

- **Elalvás**

A mester adhatja ki az utasítást

Csak a Lin driver alszik, az ECU nem

- **Felébresztés**

Bármelyik node küldheti a jelzést

Meghatározott jel csomaggal

Ismételhető

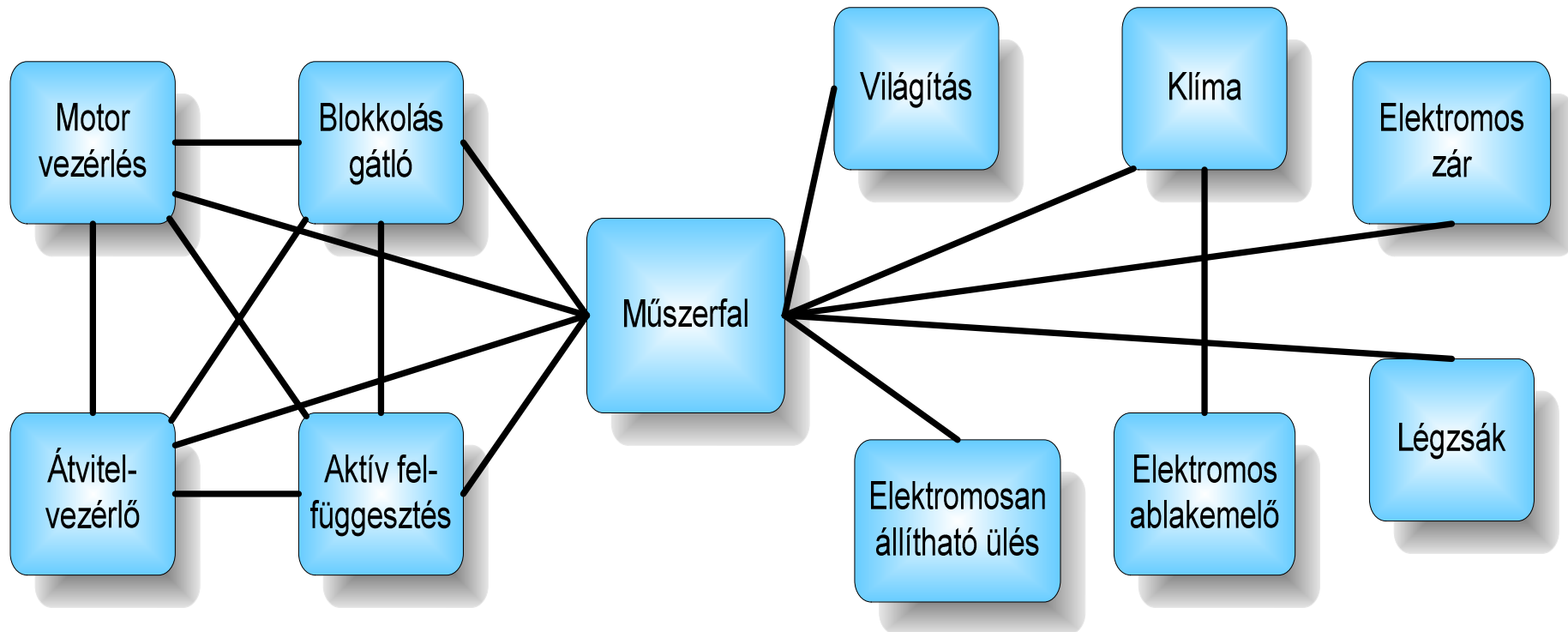
- **Üresjárat**

A node-nak érzékelnie kell, ha az üresjárat túl hosszú a buszon, miután ez megtörtént, alvó állapotba kell váltania

CAN: Controller Area Network

- A '80-as évek elején a *Bosch* mérnökei új buszrendszer tervezését kezdték meg.
- Fő céljuk a vezetékek számának csökkentése és a biztonság növelése volt, melyeket az 1986-os megszületésekor a *CAN* (Controller Area Network) protokoll sikeresen alkalmazott.
- Napjainkra a CAN széles körben elterjedt kommunikációs protokollá vált az ipar számtalan területén. Ennek köszönhetően nagyon sok cég gyárt és forgalmaz hardver és szoftver eszközöket a CAN-hoz kapcsolódóan.
- 1986-ban a detroiti SAE kongresszuson Automotive Serial Controller Area Network néven mutatták be először.

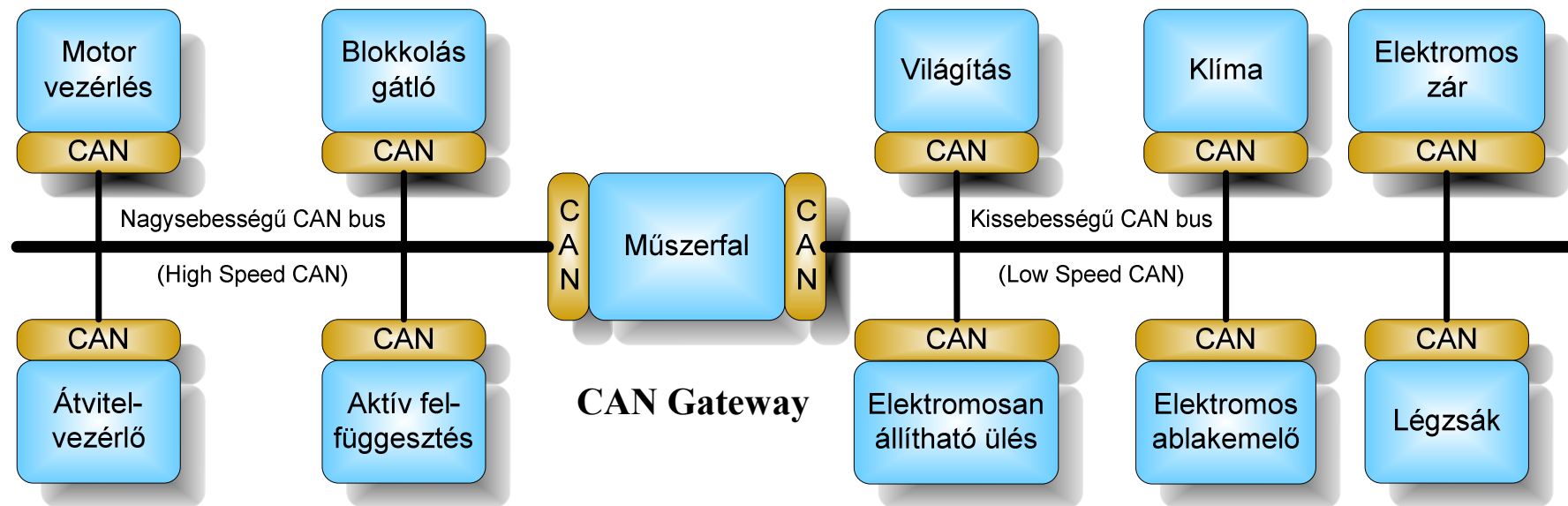
A CAN előtt alkalmazott rendszerstruktúra



A CAN protokoll jellemzői és felépítése

- A *CAN* olyan hálózatot reprezentál, amelyben a vezérlőeszközöket (pl.: mikrokontrollerek, DSP) egy soros buszrendszer köti össze, ezáltal alkalmas *elosztott irányító rendszerek* megvalósítására. A vezérlőeszközök a hálózat *csomópontjai* (node), amelyek *üzenetkeretekkel/üzenetekkel* (message frame) *kommunikálnak* egymással. Az *üzenetkeret* *fajtaik* szerint többféle felépítéssel és különféle *mezőkkel* rendelkezhetnek

CAN alkalmazásával előálló felépítés



A CAN protokoll jellemzői

- **Több-mester (Multimaster)**
- **Üzenetközpontúság**
- **Nem-destruktív arbitrációs mechanizmus**
- **Üzenetszórás (Broadcast)**
- **Eseményvezérelt**
- **Távoli válaszkérés**
- **Rugalmasság**
- **Valós idejű megvalósíthatósági lehetőség**
- **Alacsony költség**
- **Megbízhatóság**
- **Hiba detektálás a kommunikációs médium szintjén**
- **Nyugtázás**
- **Teljes rendszerre nézve konzisztens üzenetátvitel**
- **Csomópontok közötti szinkronizáció**
- **Széles eszköztáraszték**

Több-mester (Multimaster)

- Nincs kiválasztott *busz-vezérlő* (**bus master**), minden *csomópont* teljesen egyenrangú, képes az üzeneteit önállóan, bármely másik *csomópont* segítségével nélkül továbbítani az *adatbuszon* (**data bus**), ha az felszabadult. Egy *csomópont* leállása esetén az egész rendszer nem válik működésképtelenné, **csak a teljesítőképesége csökken.**

Üzenetközpontúság

- **Az üzenetek azonosítása nem a küldő vagy a fogadó *csomópont* címe alapján történik (mint általában a többi buszrendszerénél), hanem egyedi *azonosító* (identifier) alapján, amit az üzenetek a hordozott információ fontossága szerint kapnak. Így az üzenet *azonosítója* (*Azonosító mezeje*) határozza meg az üzenet prioritását, valamint közvetlenül szerepet játszik a buszért való versengés eldöntésében is. E versengési folyamat az *arbitráció* (arbitration)**

Nem-destruktív arbitrációs mechanizmus

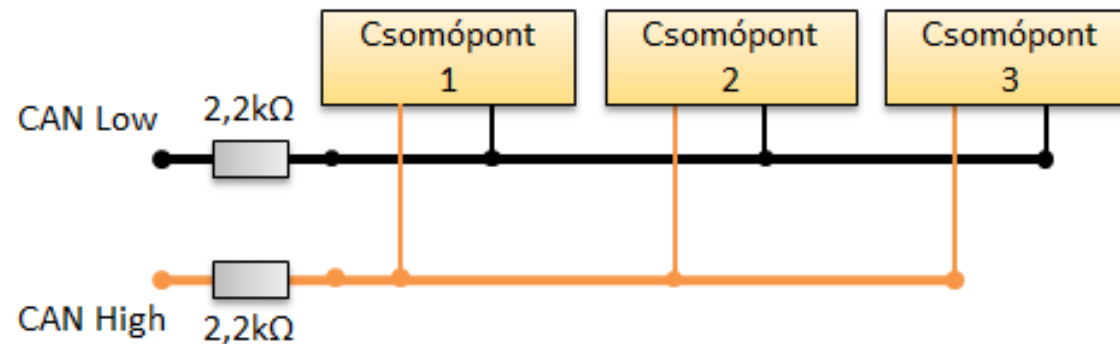
- A *CAN* ún. prioritásos *CSMA/CD+CR* médiaelérési technikát használ. Az *adatbuszt* elérni kívánó *csomópontok* várnak a busz felszabadulásáig, majd megkezdik a kommunikációt, amely egy speciális *Üzenet kezdete bittel* indul és egyben *szinkronizálja az összes kommunikációs partnert*. Ezután történik az *üzenetazonosító* továbbítása. Több partner egyidejű adási szándéka esetén ebben a szakaszban történik az ütközés feloldása, bitszintű *arbitrációval*. Ezt a technikát *nem-destruktív arbitrációs mechanizmusnak* (non-destructive arbitration) nevezzük, mivel a „vesztes” *csomópont* úgy mond le buszigényéről, hogy emiatt az átvitt magasabb prioritású üzenet nem sérül. Ez annyit jelent, hogy *mindennemű késleltetés nélkül a legmagasabb prioritású üzenet továbbítódik a buszon*. Vívőjel érzékeléses többszörös hozzáférés ütközésérzékeléssel, (Carrier Sense, Multiple Access/Collision Detection + Contention-Resolution).

Üzenetszórás (Broadcast)

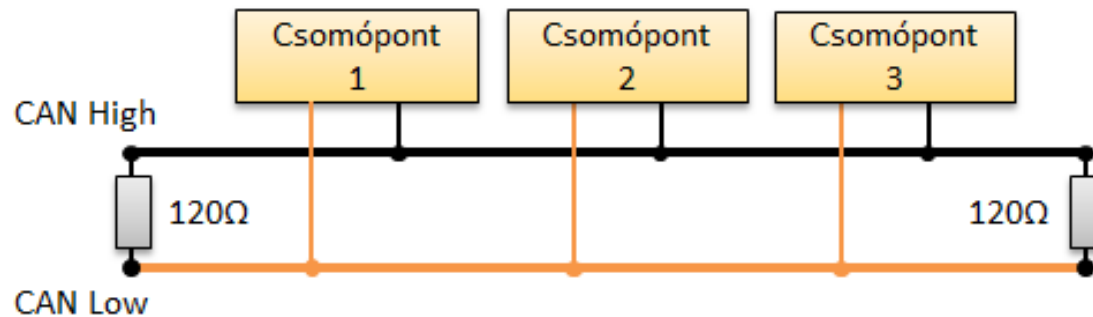
- A *CAN buszon* telefonkonferencia szerű kommunikáció zajlik. Ahogy a telefonkonferencia bármely résztvevője szabadon elmondhatja a mondanivalóját, amelyet minden más résztvevő hall, éppúgy bármelyik *csomópont* elküldheti az üzenetét a *CAN buszon*, és azt minden egyes, ugyanarra a buszra csatlakozó *csomópont* megkapja. Azonban a „bemondott” információ csak akkor értelmezhető, ha egyidőben csak egy résztvevő beszél. Hogy éppen melyik konferenciátagé ez a kiváltság, azaz melyik *csomópont* használhatja a buszt a saját üzenete elküldésére, azt az *arbitráció* folyamata hivatott eldönteni.
- A csomópontok *az üzenetazonosítók* alapján döntenek el, hogy pufferelik-e azt későbbi kiértékelés céljából, vagy figyelmen kívül hagyják *üzenetszűrés* (message filtering) által. Az *üzenetszűrőt* a felhasználói alkalmazás állítja be.

CAN hálózat fizikai réteg

- **Alacsony sebességű 10 - 125 kbps CAN hálózat**

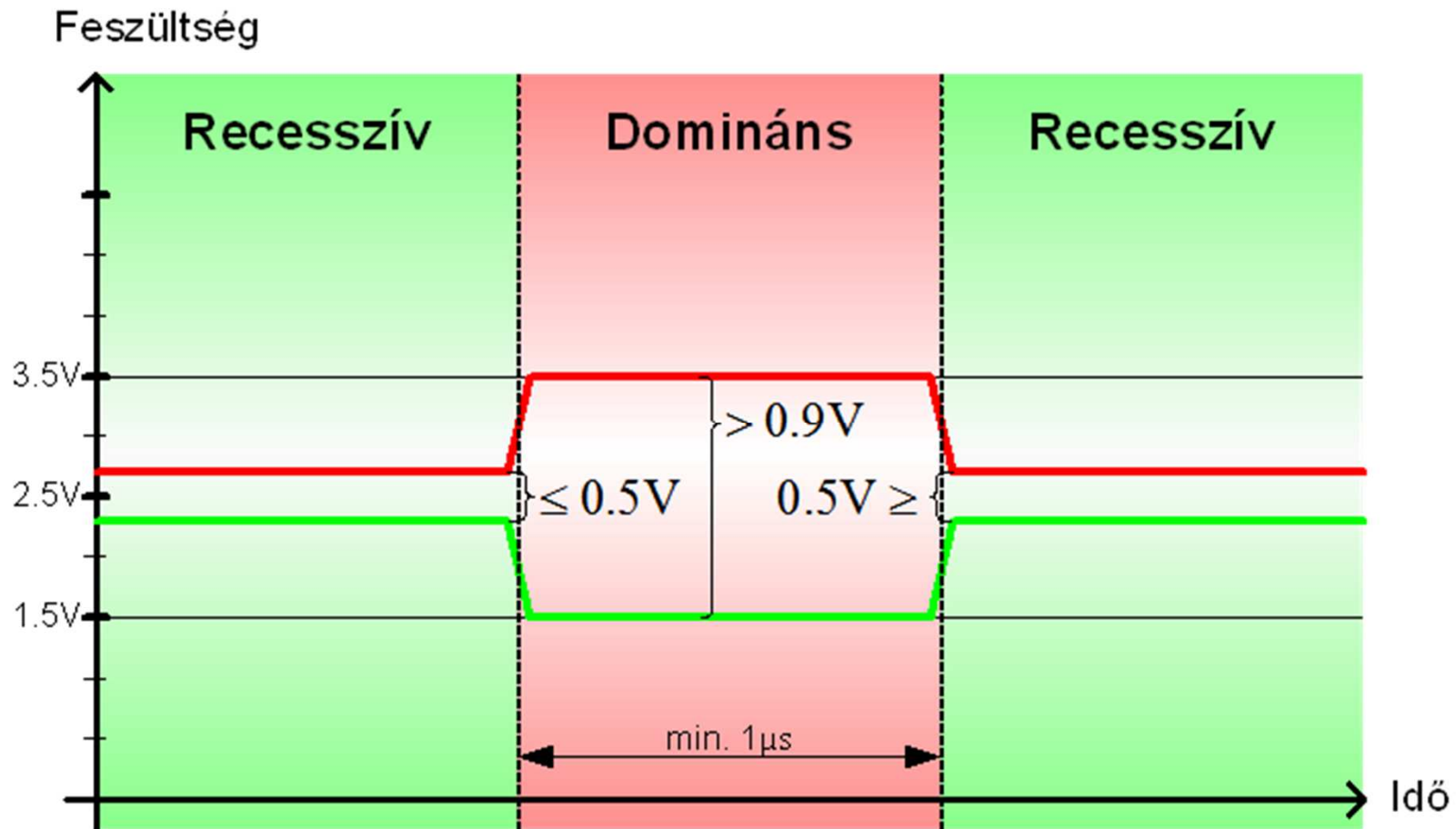


- **Nagysebességű 125 kbps - 1 Mbps CAN hálózat**



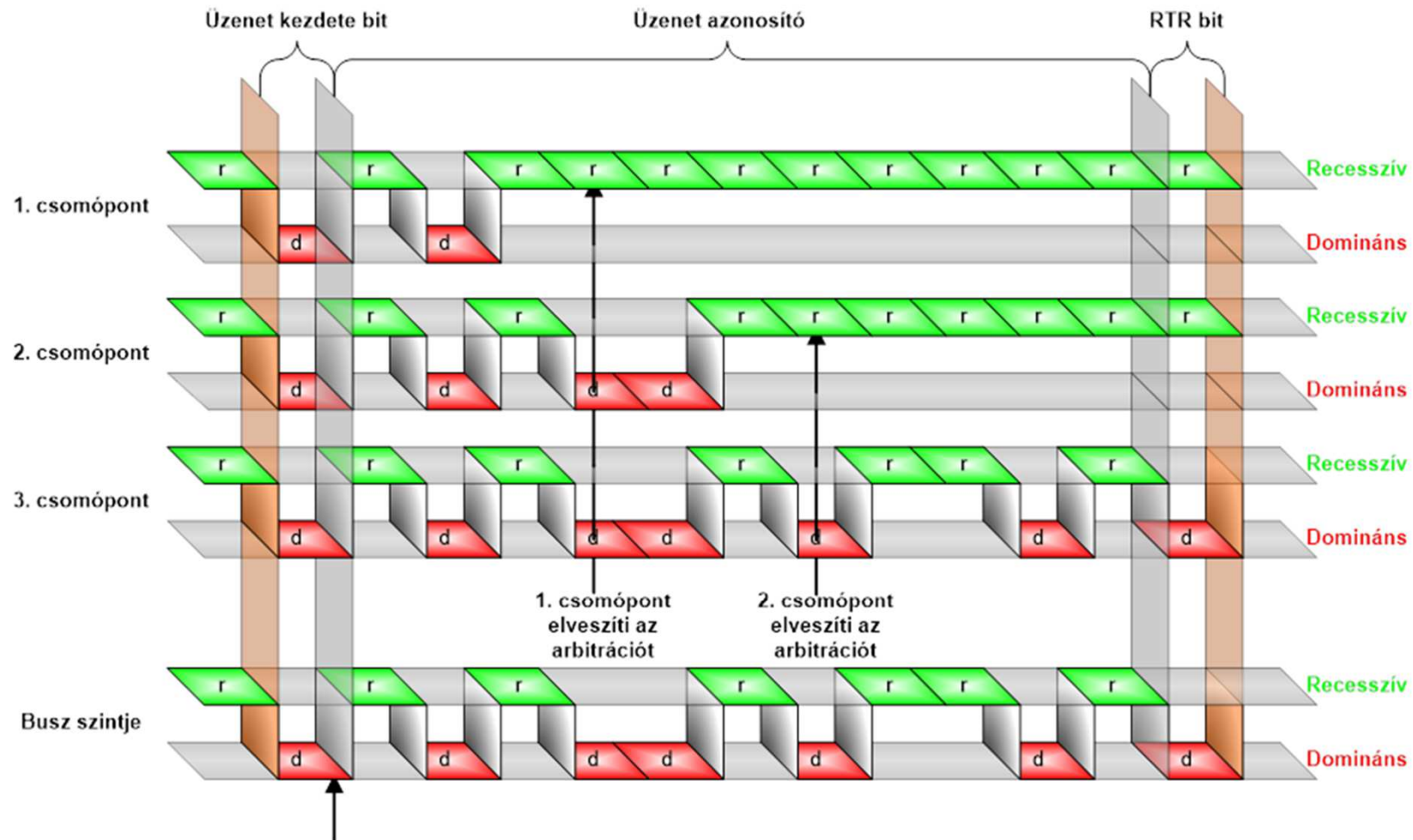
Bitrepresentáció a CAN-on

- **A bitszint meghatározása:** a *CAN_magas* és a *CAN_alacsony* vezetékek feszültség-szint-különbségei alapján történik



Nem-destruktív arbitráció folyamat

- **Vivőjel érzékeléses többszörös hozzáférés ütközésérzékeléssel, (Carrier Sense, Multiple Access/Collision Detection + Contention-Resolution)**



A csomópontok ugyanabban az időpillanatban kezdik meg az átvitelt

Az adatküldésre szolgáló üzenetek felépítése

CAN 2.0A (Hagyományos)

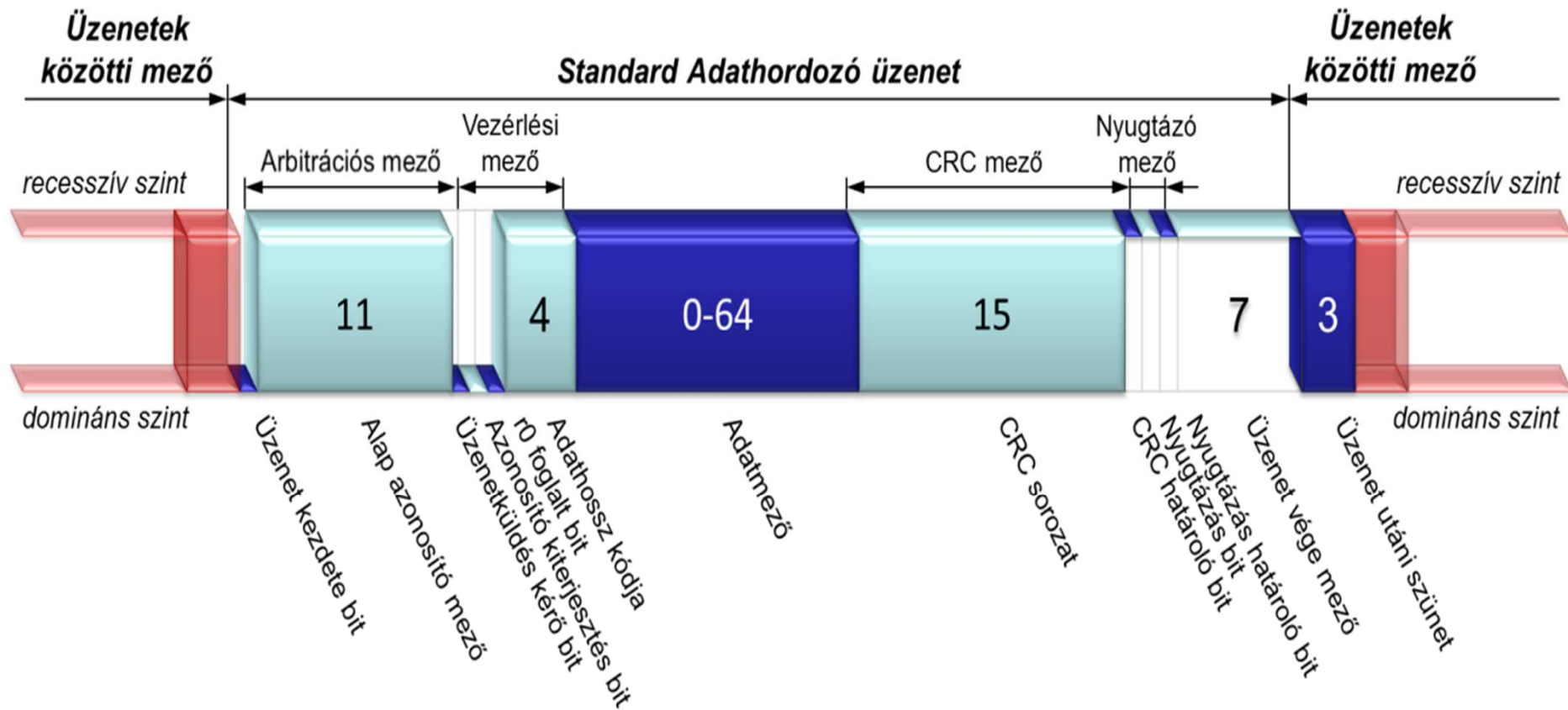
SOF	ID	RTR	IDE	R0	DLC	DATA	CRC	ACK	EOF	IFS
1	11	1	1	1	4	0..8	16	2	7	3...

CAN 2.0B (Kiterjesztett)

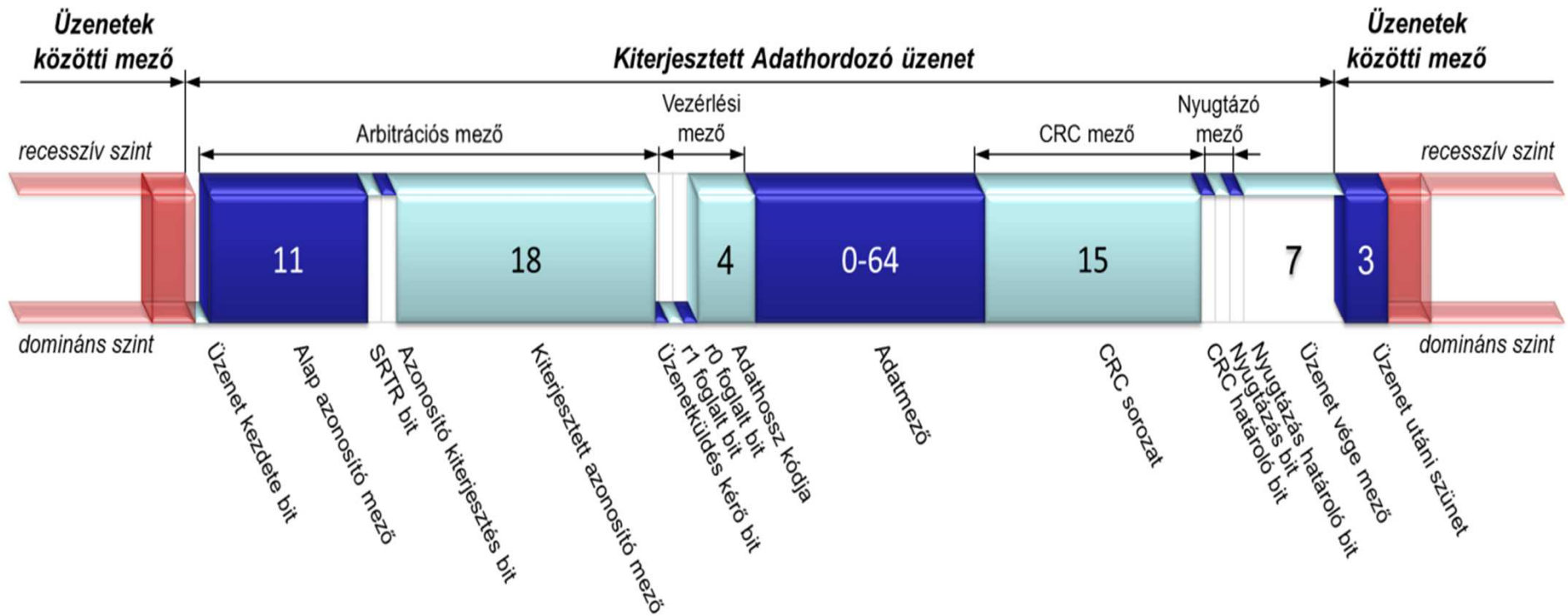
SOF	IDA	SRR	IDE	ID B	RTR	R1	R0	DLC	DATA	CRC	ACK	EOF	IFS
1	11	1	1	18	1	1	1	4	0..8	16	2	7	3...

- **SOF (start-of-frame), RTR (remote transmission request)**
- **Üzenet azonosító (IDA): hagyományos üzenetek esetében 11 bit, kiterjesztett üzenetekenél 29 bit**
- **IDE (identifier extension). Hagományos üzenet: 0**
- **DLC (data length code), Adat: 0-8 Byte**
- **15 bit CRC (cyclic redundancy check)**
- **ACK nyugta mező, első bit valaki vette (domináns), a második recesszív elválasztó bit. EOF (end-of-frame), majd legalább három további recesszív bit (IFS (interframe space)).**

Adathordozó üzenet (standard formátum)



Adathordozó üzenet (kiterjesztett formátum)



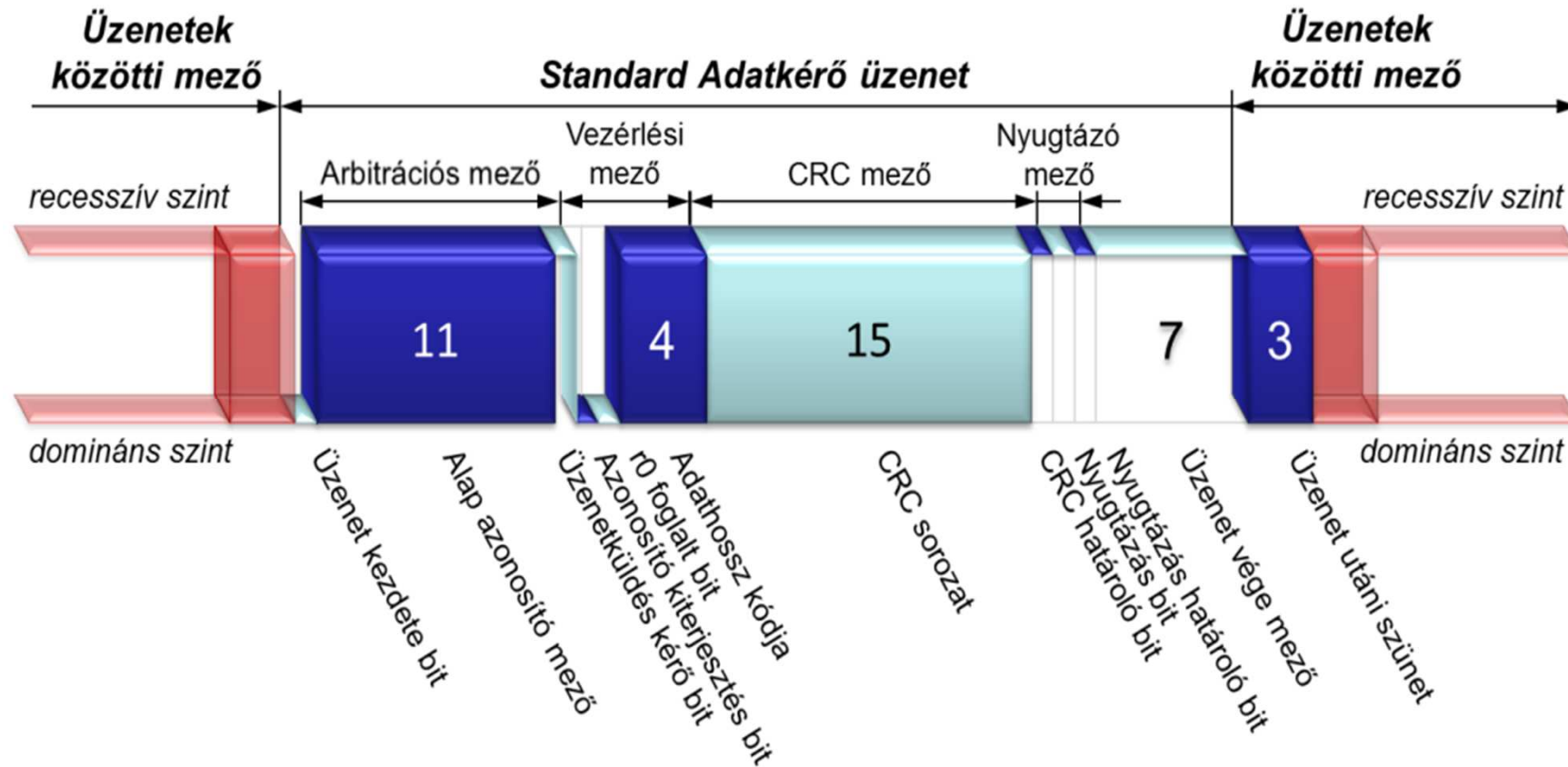
Eseményvezérelt

- A kommunikáció adott esemény bekövetkezésének (új információ generálódott egy *csomópontban*) hatására kezdődik el. Az új üzenettel rendelkező *csomópont* maga kezdi meg az átvitelt. Így jelentős kommunikációs időt takarít meg például azokhoz a rendszerekhez képest, amelyekben a *csomópontok* minden ciklusban adott időszellel rendelkeznek, melyben az új információjukat elküldhetik. Ugyanis ez esetben, ha nincs új információja egy *csomópontnak*, akkor ez az időszak kárba vész, míg esetlegesen egy másik, új információval rendelkező eszköznek várnia kell, amíg sorra kerül.
- Lehetőség van ciklikus információcserére is, ekkor belső óra, vagy egy másik *csomópont* kezdeményezi a kommunikációt (ISO 11898-4).

Távoli válaszkérés

- *Az eseményvezérelt kommunikációt kiegészítve a CAN lehetőséget biztosít ún. „távoli válaszkérő üzenetek” küldésére. Ezek segítségével egy fogadó csomópont kérheti a számára szükséges információ elküldését a megfelelő küldő csomóponttól. A kérés, és a válasz külön üzenetet képez. Főleg a csomópontok állapotának (aktív/inaktív) lekérdezésére használják ezt a technikát.*

Adatkérő üzenet (standard formátum)

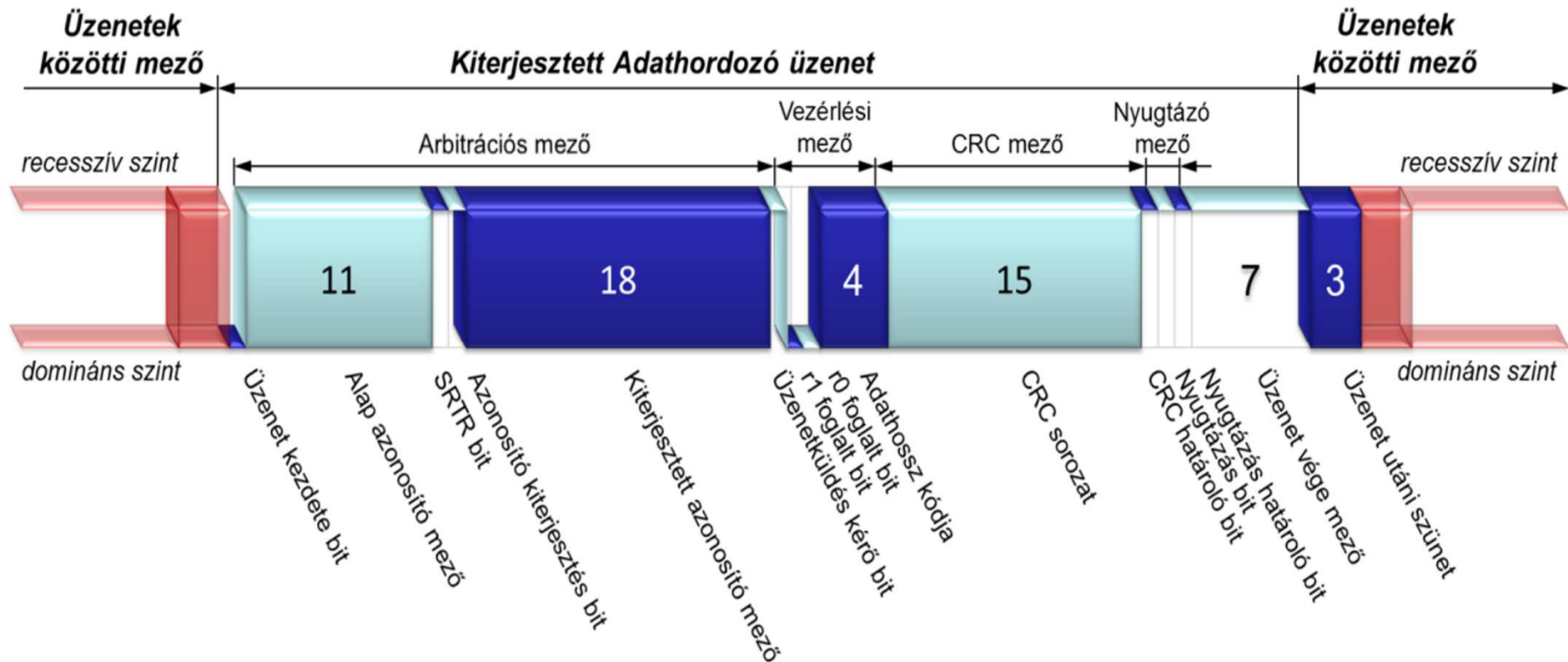


Az Üzenetküldés kérő bit mindig recesszív, hogy az arbitrációt az Adathordozó üzenet nyerje.

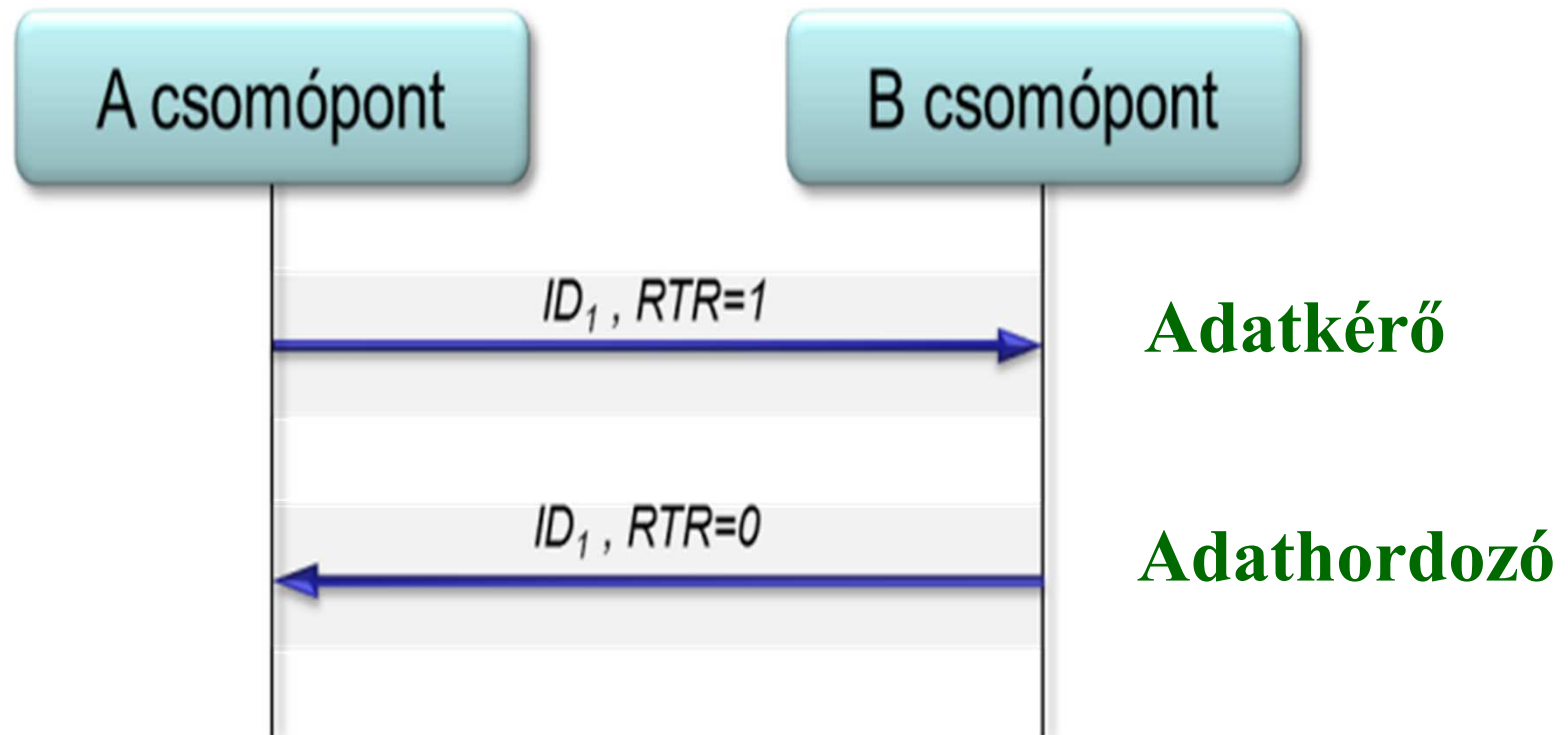
Az adathossz-kód a kért üzenet Adatmezőjére vonatkozik.

Az Adatkérő üzenetnél az Adatmező üres

Adatkérő üzenet (kiterjesztett formátum)



Adatkérési ciklus



Rugalmasság

- **A csomópontokat** dinamikusan rákapcsolhatjuk, illetve leválaszthatjuk a buszról anélkül, hogy a többi *csomópont* kommunikációját zavarnánk, így a rendszer rugalmasan alakítható.
 - Egy rendszeren belül **32 csomópont** lehet szabványos buszmeghajtók esetén, **64-128** darab lehet alkalmazás-specifikus meghajtók esetén.
 - **Üzenetek száma a rendszerben:** *Standard üzenetformátum* esetén 2^{11} (= 2048), *Kiterjesztett üzenetformátum* esetén 2^{29} (=536 870 912).
 - **Adatmennyiség üzenetenként:** 0-8 bájt. A rövid üzenetek elegendők a járművekben valamint beágyazott illetve automatizált gyártó rendszerekben történő kommunikációhoz (rövid *buszelérési idő*, összeütközés kisebb valószínűsége).
 - **Maximális üzenethossz:** beszúrt bitekkel együtt **117 bit** *standard üzenetformátum* esetén, **136 bit** *kiterjesztett üzenetformátum* esetén.
 - **Bitráta:** 5kbit/s és 1Mbit/s között programozható (a buszhossztól függően).

Valós idejű megvalósíthatósági lehetőség

- Az adattovábbítás maximális sebessége **1Mbit/s** (40m-es buszhossznál), az üzenetek rövidek, a *késleltetési idő* maximálva van, az *arbitráció* pedig gyors. Az utóbbi tulajdonságok alkalmassá teszik a *CAN rendszert* a valós idejű események (pl.: ABS, motorvezérlés) irányítására.

Alacsony költség

- A kivitelezéshez szükséges eszközökre nagy igény van az ipar különböző területein, ezért a sorozatgyártás alacsony árat és kedvező teljesítmény-ár viszonyt eredményez. A **csavart érpár**, amelyet a *CAN rendszereknél* a leggyakrabban használnak, szintén olcsó, mert ez az egyik leggyakoribb buszfajta. A rendszer üzemeltetési költségének csökkentése érdekében a *csomópontok* átállhatnak ún. „**alvó állapotba**” (sleep mode), amely azt jelenti, hogy belső aktivitásuk megszűnik és lekapcsolják a buszmeghajtókat, ezáltal **csökkentve a rendszer áramfogyasztását**. Az „**alvó állapot**” követi az ún. „**ébredési fázist-t**” (wake-up), aminek következtében a belső aktivitás újra indul. A rendszernek **lehetősége van arra**, hogy egy speciális azonosítóval rendelkező **üzenet elküldésével aktiváljon egy csomópontot**.

Megbízhatóság

- **Kifinomult hibadetektáló és hibakezelő mechanizmusokkal rendelkezik, mint például:**
 - **15 bites, 6-os Hamming-távolságú CRC-vel (Cyclic Redundancy Check), amely 5 hibás bit felismerését teszi lehetővé üzenetenként.**
 - **Nem rendszeres hibák helyreállítása a hibás üzenetek automatikus újraküldésével.**
 - **Ismétlődő hibák kiküszöbölése a hibás csomópont kikapcsolásával, ami determinisztikussá teszi a rendszer esetleges hibák utáni helyreállításának idejét.**
 - **Az elektromágneses interferenciákra alacsony az érzékenysége.**
 - **A rendszer garantálja, hogy a küldő-csomópont által elküldött adatok megegyeznek a fogadó-csomópontok által fogadott adatokkal. Átlagos terhelés mellett 1000 év alatt egy olyan hiba fordul elő, amelyet a rendszer nem észlel.**

Hiba detektálás a kommunikációs közegben

- A *CAN vezérlők* (CAN controller) sok fajta vezetékhibát ismernek, és definiálnak: szakadás, testzárlat, egyéb zárlatok. A protokoll nem írja le, hogy mi a teendő a fenti hibák esetén, de az újabb *CAN vezérlőkben* legalább a fenti esetek egyikének kezelése implementálva van.

Nyugtázás

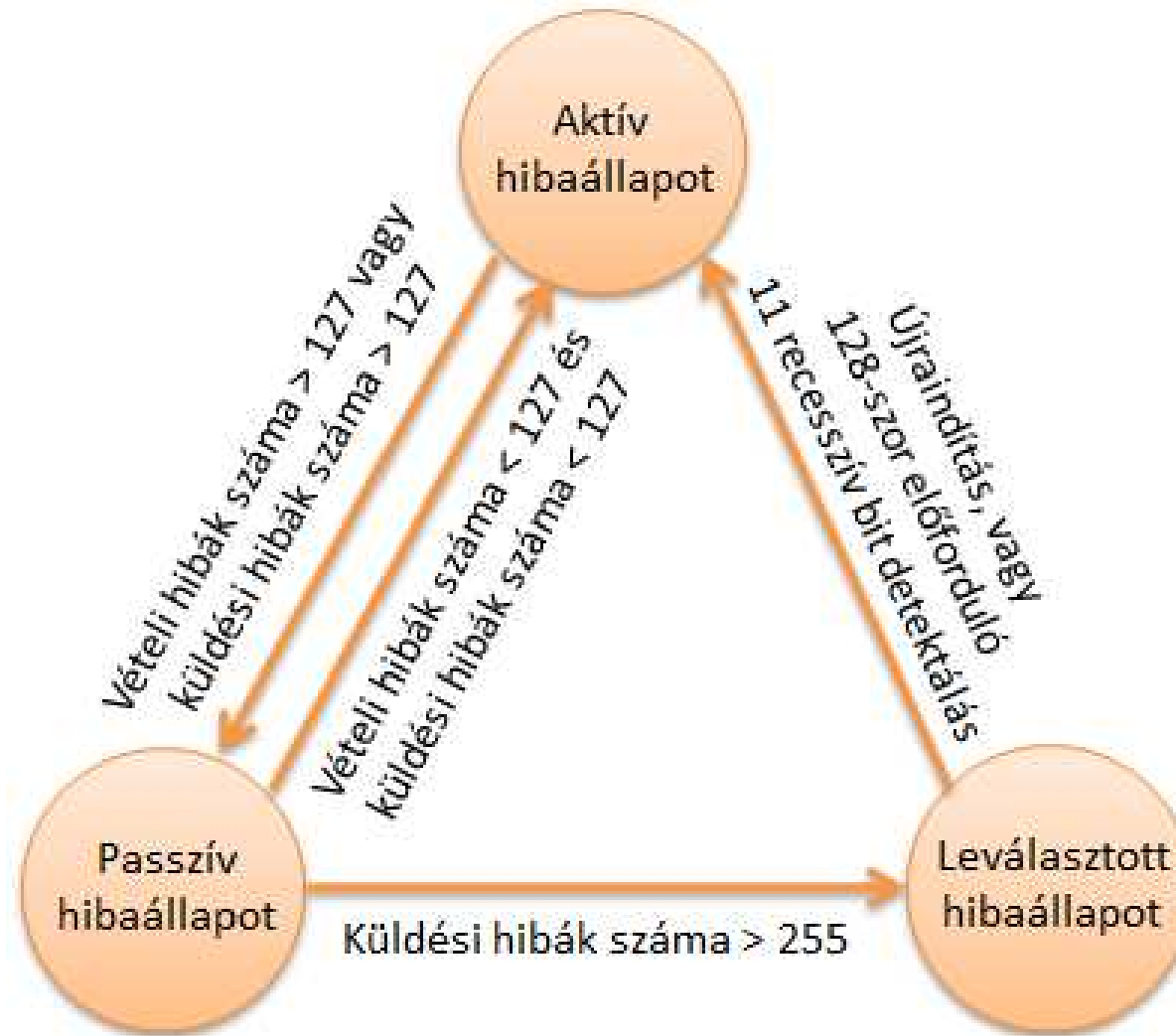
- Az üzenetek globális *Nyugtázó mezővel* (Acknowledgement field) rendelkeznek, amely jelzi a küldő *csomópontnak*, hogy legalább egy kommunikációs partnerhez hibátlanul megérkezett az üzenet. Így a küldő információt kap arról, hogy még a buszhoz van-e csatlakoztatva, vagy sem. Az üzenetszórás-jellegű üzenettovábbítás következtében minden *csomópont* nyugtázó jellel válaszol, ha nem észleltek hibát.

Konzisztens üzenetátvitel

- A rendszer minden egyes elküldött üzenetre garantálja, hogy azt vagy minden *csomópont* elfogadja, vagy minden *csomópont* elutasítja. Ha legalább egy vevő hibát észlel a fogadás során, akkor egy *Hibaüzenettel* (Error frame) rögtön megszakítja az átvitelt, és jelzi a többi fogadó állomásnak, hogy hagyják figyelmen kívül az üzenetet, a küldőnek pedig, hogy küldje el ismét azt. Ez eredményezi a teljes üzenet-konzisztenciát a rendszerben, azaz vagy minden egyes *csomópont* megkapja ugyanazt az információt ugyanabban az időpillanatban, vagy egyik sem. Ez az *elosztott rendszerekben* igen fontos tulajdonság, mivel így garantálható, hogy a különböző mikrokontrollerek ne dolgozzanak ugyanahhoz a változóhoz tartozó eltérő adatokon egyidőben.

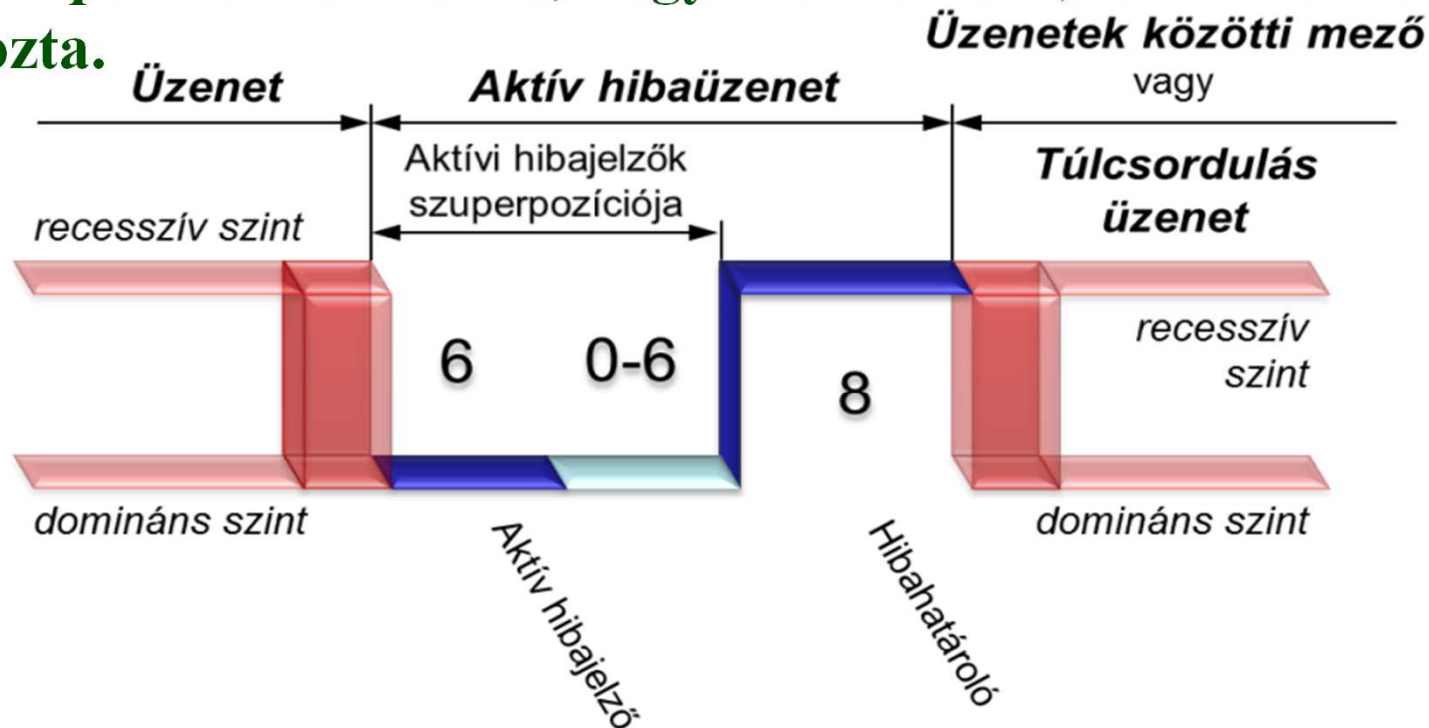
Megbízhatóság

- CAN csomópont hibaállapotok közötti átmenete



Aktív hibaüzenet

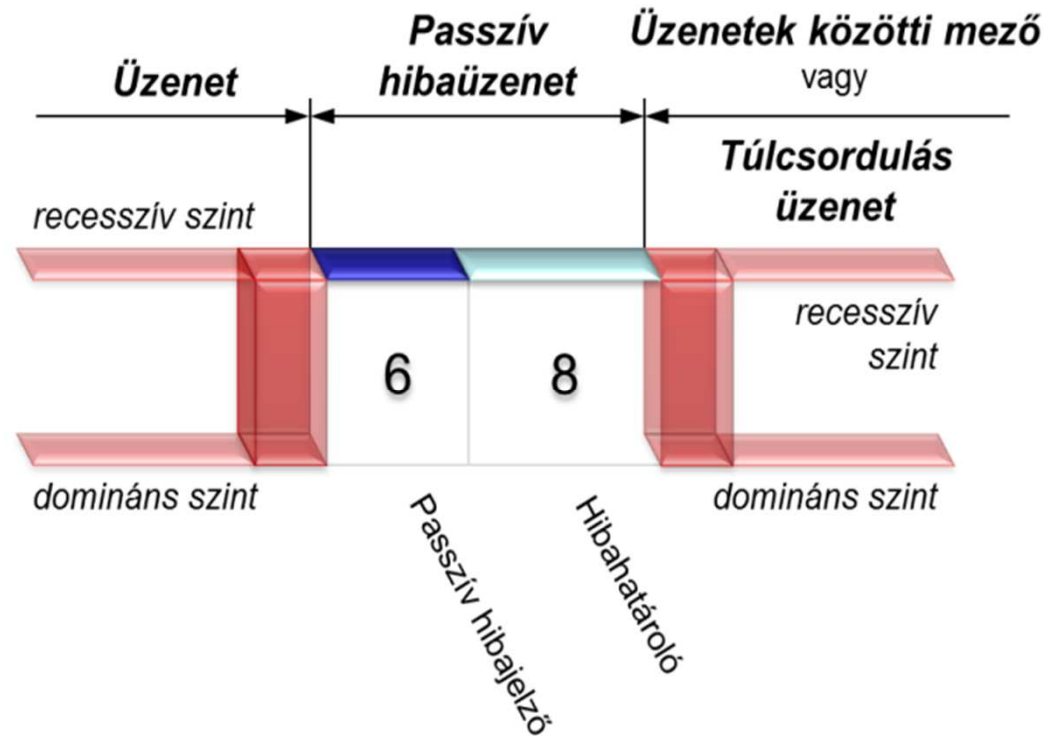
- Egy csomópont *Hiba aktív állapotban* van, ha a saját *hibaszámlálója* a meghatározott érték alatt van. Hibát észlelve *Aktív hibaüzenet* küldésére képes. Ekkor a *csomópont* biztos benne, hogy hiba történt, és azt nem ő okozta.



A kezdeti hiba észlelésekor (egy vagy több) *Hiba aktív állapotú csomópont* azonnal megszakítja a kommunikációt
CRC hiba: csak a nyugtázás mező után kezd

Passzív hibaüzenet

- Egy csomópont *Hiba passzív* állapotban van, ha átlépte a kijelölt *hibaszámláló* értéket – és valószínűleg helyi meghibásodása van –, de még nem olyan súlyos a helyzet, hogy le kelljen válnia a buszról. E *csomópont* hiba észlelése során *Passzív hibaüzenet* küldésére képes.



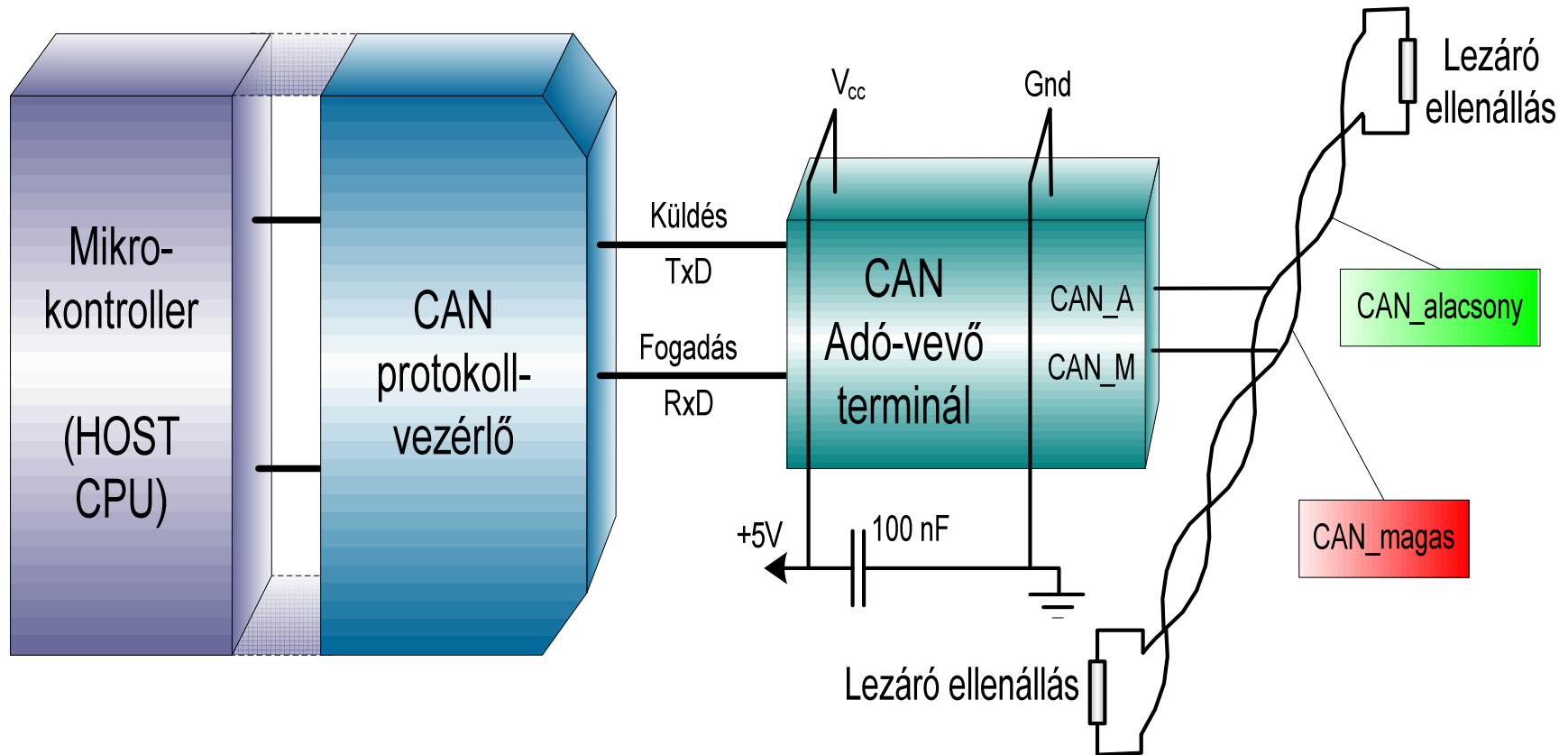
Csomópontok közötti szinkronizáció

- **Üzenetek sikeres küldése és fogadása után az összes résztvevő *csomópontban* egy megszakítás (interrupt) generálódik, amit fel lehet használni az *elosztott vezérlőrendszer* óráinak beállításához. Az órák szinkronizáltsága elengedhetetlen feltétele az elosztott valós idejű alkalmazások működésének.**

Széles eszköztár

- Manapság a legtöbb mikrokontroller gyártó (a legnevesebbek pl. Intel, Motorola, Siemens, Philips) kínálatában megtalálhatók *CAN chipek* is, amelyek a nagy választék és az árverseny miatt meglehetősen olcsón beszerezhetők, és így gyorsan elterjednek.

CAN csomópont felépítése



CAN protokollvezérlők csoportosítása

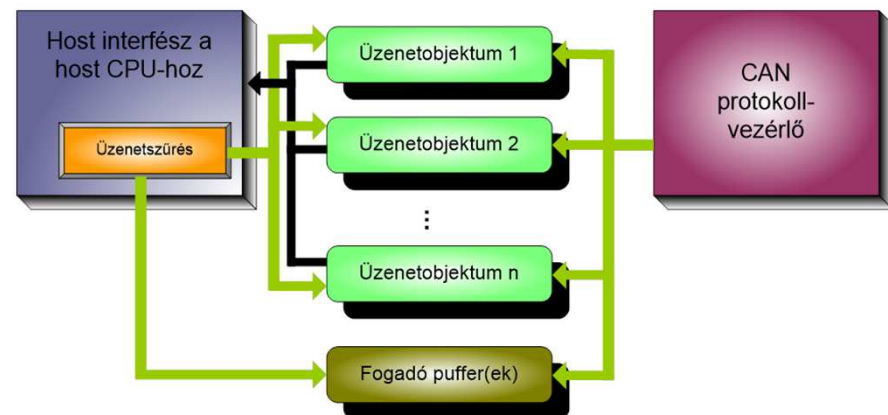
Csoportosítás üzenetszűrés szerint

- **BasicCAN**

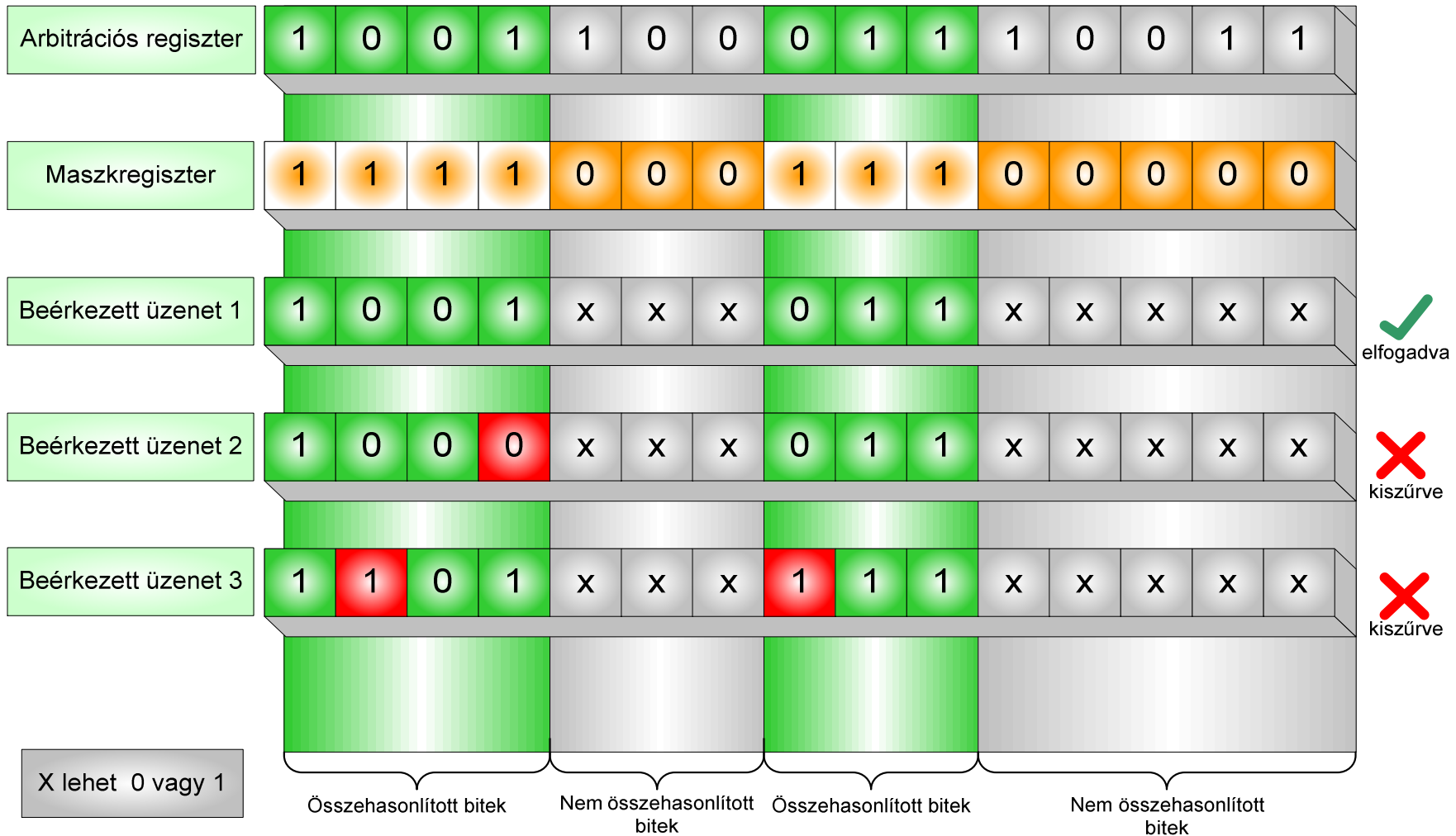
- 1-3 küldő puffer
- 1-2 fogadó puffer
- Arbitrációs regiszter
- Maszk regiszter

- **FullCAN**

- minden egyes elfogadási mintának csak egy üzenet felel meg
- nincs szükség utólagos szűrésre



Az üzenetek szűrése



A CAN alkalmazási területei

- **A CAN ára alacsony, a kiépített rendszer megbízható, és valós idejű környezetben is alkalmazható, flexibilis protokoll.**
 - orvosi elektronika,
 - háztartási eszközök,
 - épületautomatizálás,
 - irodai automatizálás,
 - vasúti rendszerek,
 - hajózás,
 - mezőgazdasági gépek,
 - repülőgép-elektronika,
 - *PLC*,
 - robotvezérlés,
 - intelligens motorvezérlés,
 - űrtechnológia

Szabványosítás (ISO 11898)

- Három évvel az első *CAN vezérlő chipek* megjelenése után, **1990-ben**, a Bosch féle **CAN specifikációt** nemzetközi szabványosításra nyújtották be.
- A különböző megoldások egységesítéséhez, valamint a *CAN* további technikai fejlődésének biztosításához szükség volt egy – felhasználókból és gyártókból álló – semleges platformra. **1992** márciusában hivatalosan is megalakult a *CAN in Automation (CiA)* nemzetközi felhasználói és gyártói csoport.
- **1993-ra** megjelent az **ISO 11898-as CAN szabvány**, amely a protokoll 11 bites azonosítójú, *standard formátumú üzenetein* túl a *Fizikai réteget* is definiálja, 1Mbit/s-os átviteli sebességig (CAN Specification 1.2).

Kiterjesztett formátumú üzenetek

- Az üzenetek fajtáinak növekedésével szükségessé vált a 29 bites azonosítójú, *kiterjesztett formátumú üzenetek* specifikálása, melyet a CAN Specification 2.0 definiál, amelyet az **ISO 11898** kiegészítéseként rögzítettek.
 - *CAN Specifikáció 2.0 „A fejezet” (Part A)*, amely csak a *standard formátumú üzeneteket* definiálja. Ez magába foglalja a korábbi CAN Specifikáció 1.2-t (CAN Specification 1.2).
 - *CAN Specifikáció 2.0 „B fejezet” (Part B)*, pedig a *standard – és a kiterjesztett formátumú üzeneteket* együttesen specifikálja.
 - *CAN Specifikáció 2.0 Függelék* útmutatást ad arra vonatkozólag, hogy hogyan érdemes megvalósítani a *CAN protokollt* úgy, hogy megfeleljen a szabvány *A* vagy *B fejezetében* leírtaknak.

Az átdolgozott CAN specifikációk szabványosítása

- ISO 11898-1: a *CAN Adatkapcsolati rétegének* és a fizikai jelterjedésnek (physical signaling) a leírása.
- ISO 11898-2: a *CAN* nagysebességű *Fizikai rétegét* jellemzi, amely leginkább az autóiparban és ipari vezérléseknél használatos (two-wire balanced signaling).
- ISO 11898-3: a *CAN* alacsony sebességű, hibatűrő *Fizikai rétegét* rögzíti.
- ISO 11898-4: a *CAN idővezérelt kommunikációja* (TTCAN = Time-Triggered CAN), ahol a *CAN Adatkapcsolati rétegében* található rendszeróra ütemezi az üzeneteket (messages).
- ISO 11898-5: a *CAN* nagysebességű *Fizikai rétegének* leírása *alacsony energiaszintű módban* (Low-Power Mode).
- ISO 11898-6: a *CAN* nagysebességű *Fizikai rétegének* leírása *szelektív felébresztő funkció* (selective wake-up function) esetén.
- ISO 11992-1: a teherautókra, trélerekre szabott *CAN* protokoll leírása.
- ISO 11783-2: 250kbit/s bitsebességű mezőgazdasági szabvány
- SAE J1939-11: 250kbit/s bitsebességű *árnyékolt csavart érpár* (STP – Shielded Twisted Pair) közeg leírása.

Ajánlott és felhasznált irodalom

Az előadás anyag forrása:

- Pannon Egyetem, Mérnöki Kar, Gépészmérnöki Intézet
Járműmechanika Intézeti Tanszék
- **Dr. Fodor Dénes, Speiser Ferenc:**
Autóipari beágyazott rendszerek
- http://moodle.autolab.uni-pannon.hu/Mecha_tananyag/autoipari_kommunikacios_rendszerek/index.html



Dr. Fodor Dénes intézetigazgató, tanszékvezető egyetemi docens
szoba: C218

telefon: 88/624-000/6082, 88/624-776

e-mail: fodor@almos.uni-pannon.hu

Speiser Ferenc egyetemi

tanársegéd

szoba: C217

telefon: 88/624-000/6083

e-mail: speiserf@almos.uni-pannon.hu

