



BME
Budapesti Műszaki és Gazdaságtudományi Egyetem

HAUT
Közlekedésautomatikai Tanszék



Járműfedélzeti rendszerek II.

7. előadás

Dr. Aradi Szilárd

A CAN hálózat

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- Az első szabványos autóiipari kommunikációs hálózat
- Bosch fejlesztés, 1986 SAE (Society of Automotive Engineers) congress
- 1991 CAN 2.0 specification

A CAN hálózat

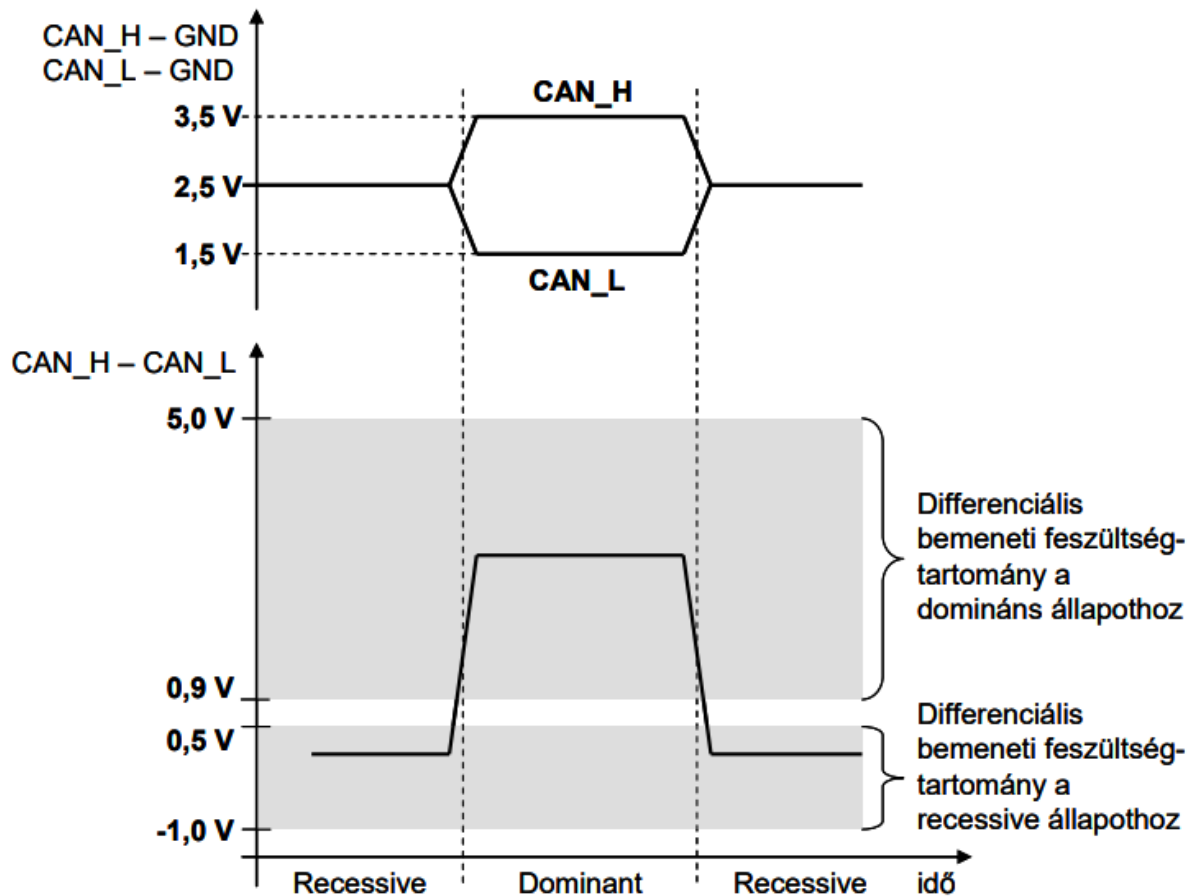
- A CAN hálózatot leíró szabványok a fizikai és az adatkapcsolati réteget írják le, erős hangsúlyt fektetve ez utóbbira, melyet további alrétegekre bontanak:
 - MAC (Media Access Control: Közeg hozzáférés vezérlés)
 - LLC (Logical Link Control: Logikai kapcsolatvezérlés)

Fizikai réteg

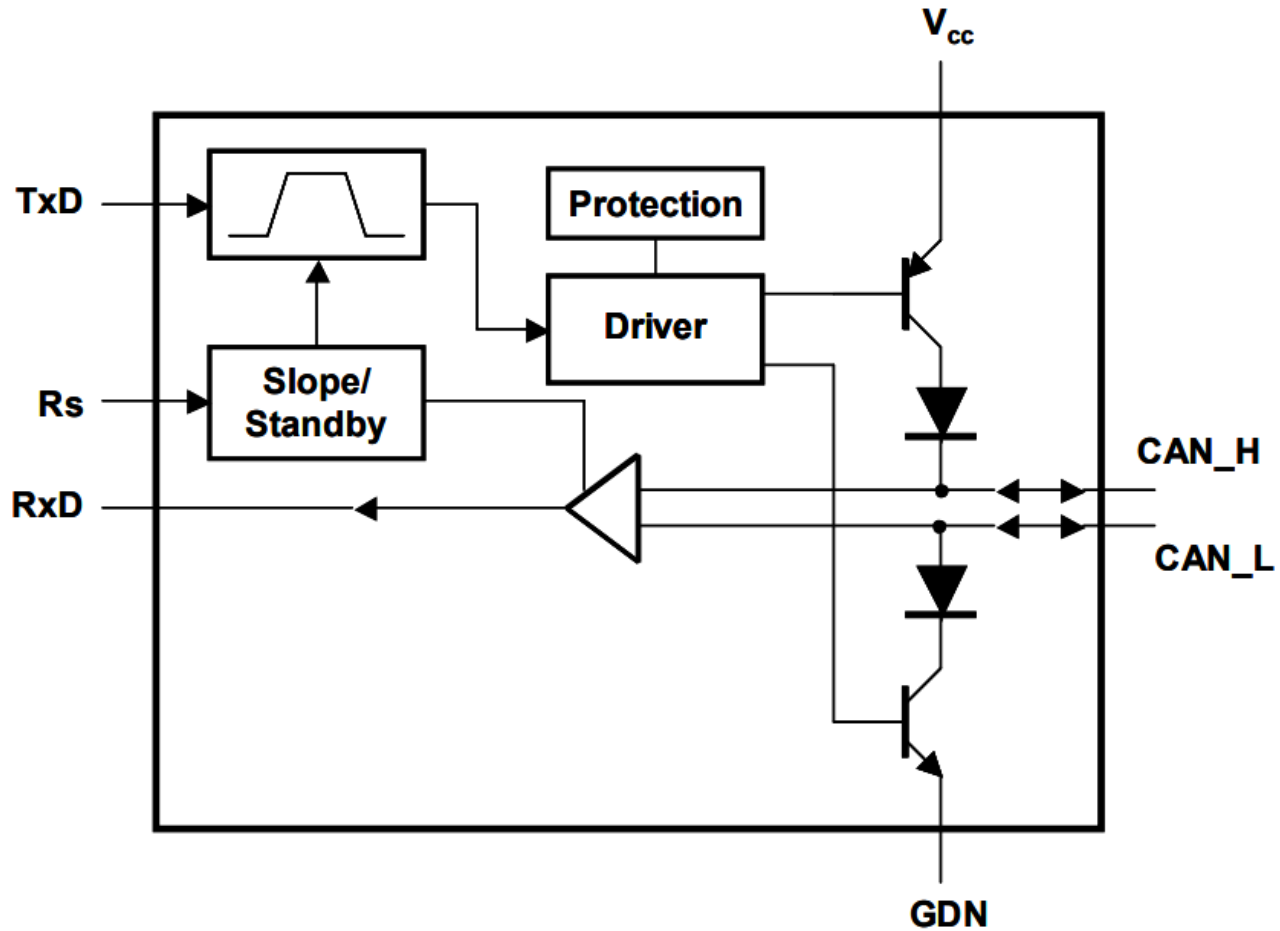
- Kétvezetékes szimmetrikus busz
- A vezeték sodort érpár, mely lehet árnyékolt vagy árnyékolatlan is
- A CAN kontrollert egy adóvevő (transceiver) áramkör illeszti a buszra, mely a controller TTL szintű jeleit alakítja át a busz differenciális jeleire
- A buszon lévő állomások között huzalozott ÉS kapcsolat áll fent

Fizikai réteg (Jelszintek)

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék



Fizikai réteg (Transceiver)

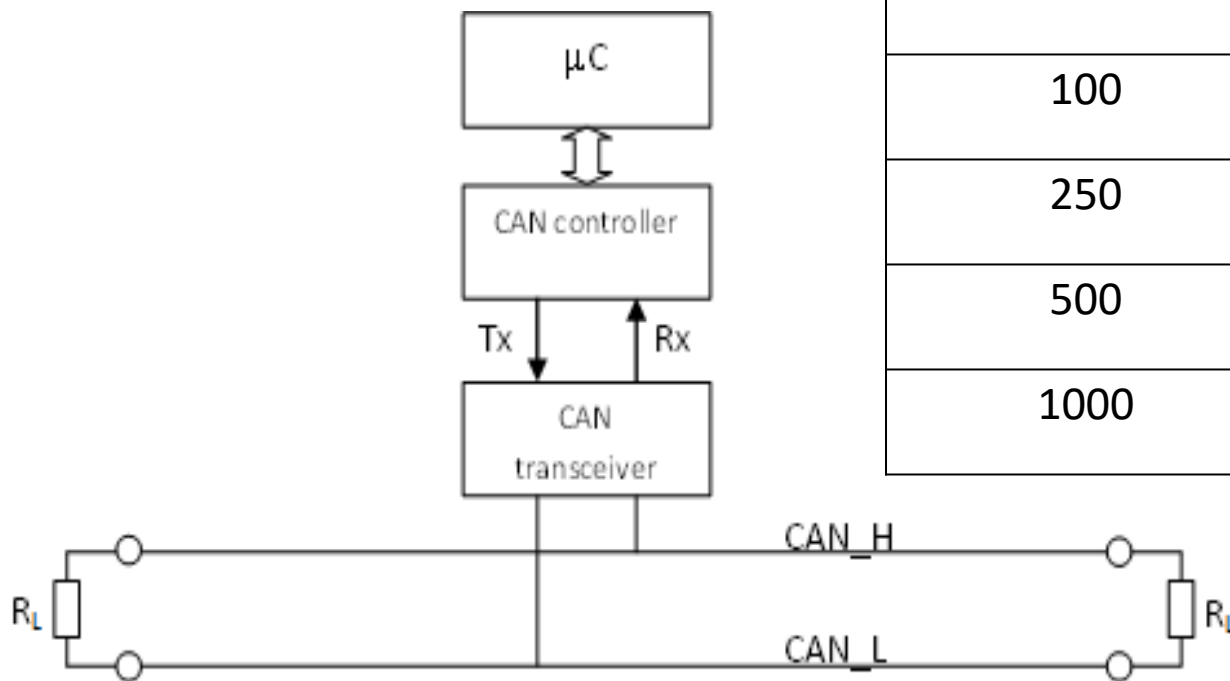


Fizikai réteg (Lezárás)

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- A busz impedancia-illesztéséről lezárásokkal kell gondoskodni
- Ennek szabványos értéke $R_L = 120\Omega$
- Ez a fizikai felépítés legfeljebb 1Mbit/s adatátviteli sebességet tesz lehetővé
- Az adott rendszer sebességét természetesen korlátozza a busz hossza és a transceiverek késleltetése
- ISO11898 specifikálja

Fizikai réteg (Architektúra)



Kábelhossz (m)

Adatátviteli
sebesség (kbit/s)

30

1000

100

500

250

250

500

125

1000

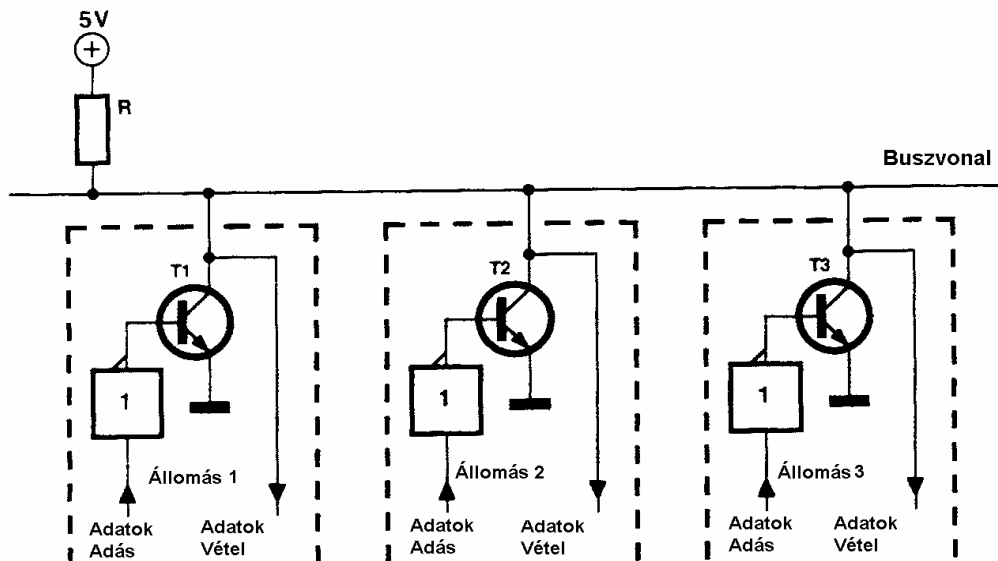
62,5

Fizikai réteg

(Logikai ÉS megvalósítás)

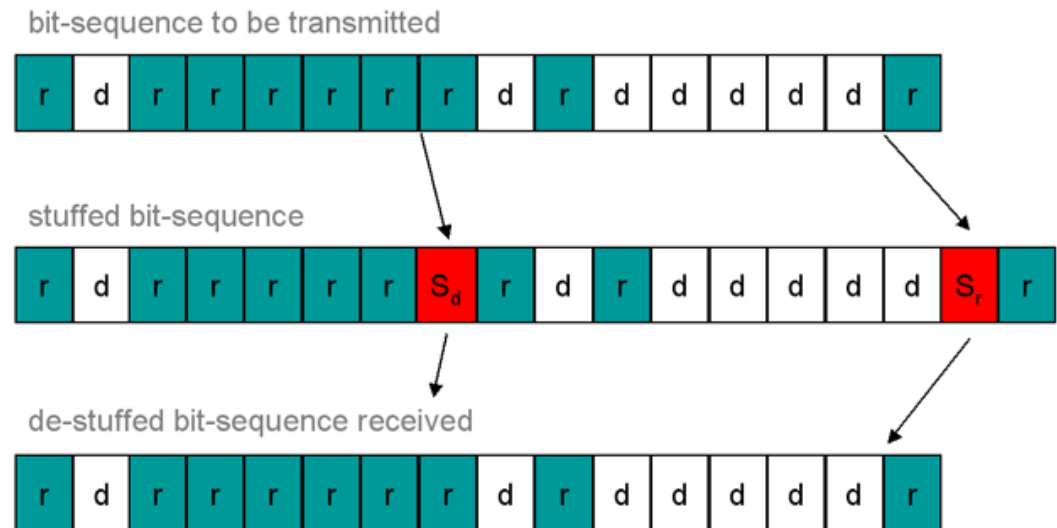
Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- A buszvonal **recesszív** szintje 1
- Ez bármelyik állomás által átírható a **domináns 0-ra**



NRZ, Bit Stuffing

- „Non-return to Zero” kódolás, a jel teljes tartamában a megadott szinten van
- Nincs jelenkénti él, ami a szinkronizációt nehezíti
- Öt azonos jelszintenként egy ellentétes jel beszúrása (stuffing)
- A vevő ezt kiveszi (destuffing)
- Teljes üzenetre érvényes (kivéve EOF)

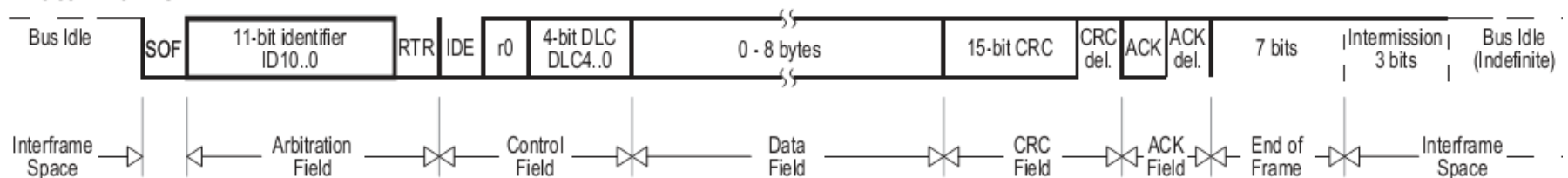


Üzenetformák és működés

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- BUS Idle
- SOF – Start of Frame
- Arbitration Field
- Control Field
- Data Field
- CRC Field
- ACK Field
- EOF End of Frame
- Interframe Space

Data Frame



SOF

Start of Frame

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- A busz alaphelyzetben magas (1) szinten van
- Amennyiben valamelyik résztvevő használni kívánja, a vonalat alacsony szintre húzza
- Erre a többiek elkezdik figyelni az adást, és szinkronizálják az órájukat a lefutó élre. (Az órajelet mindenki magának állítja elő, az üzenet olvasás közben folyamatos szinkronizálás folyik)

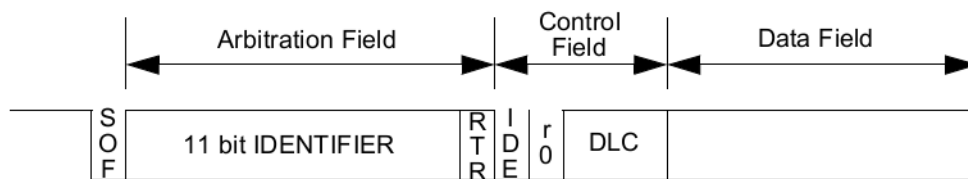
Arbitration Field

- Mivel a buszon **ütközésfigyeléses, osztott közeghozzáférés** vezérlés van előfordulhat, hogy egyszerre több szereplő próbál kommunikálni.
- Prioritása nem a résztvevőknek, hanem az üzeneteknek van.
- A hozzáférés eldöntésére szolgál az arbitrációs mező

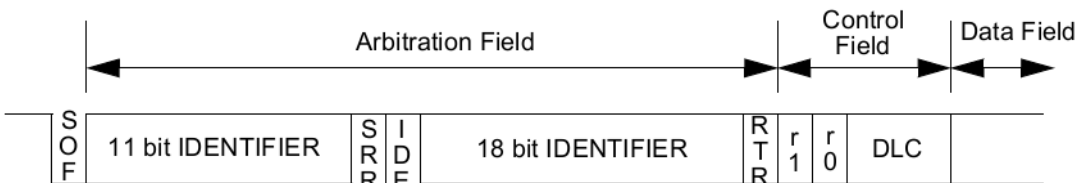
Arbitration Field

- 11+1 bit (CAN 2.0A), vagy 11+2+18+1 bit (CAN 2.0B) értéke dönti el a jogosultságot,
- minél kisebb ez a számérték, annál nagyobb a buszhasználati prioritás

Standard Format



Extended Format

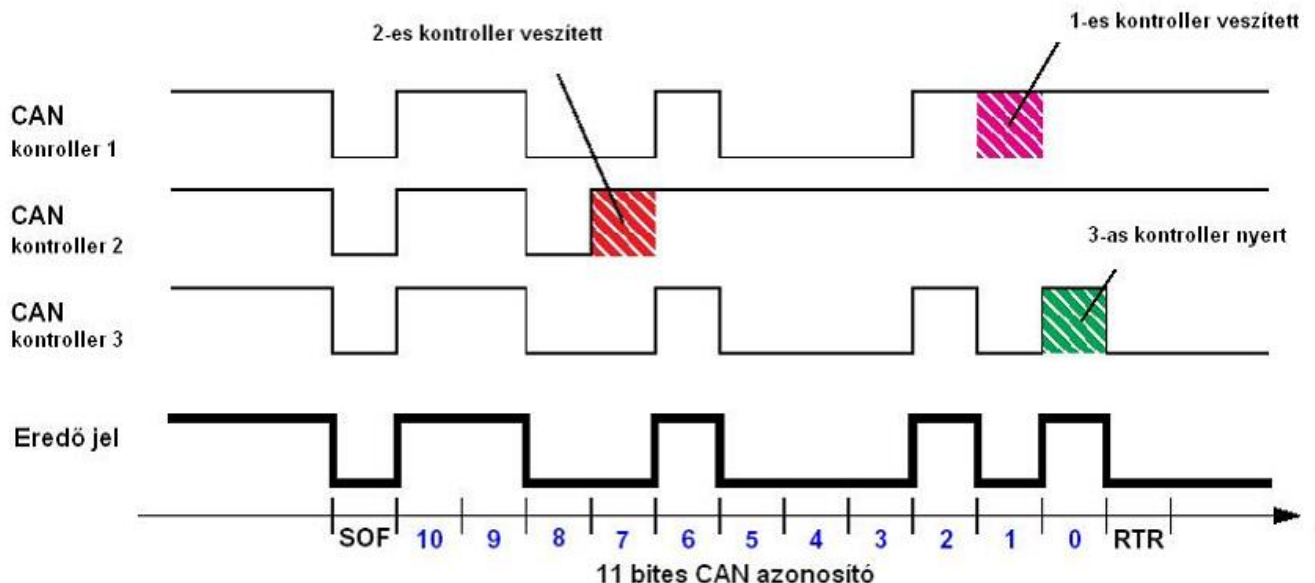


- **RTR** - Remote Transmission Request
- **IDE** - Identifier Extension
- **r0,r1** - Reserved (0,1)
- **SRR** – Substitute Remote request

Az arbitráció folyamata

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- Három egyidejűleg adni kívánó controller esetén
- A legkisebb azonosító nyer (korábban megjelenő domináns jelszint)



Control Field

- 6 bit – [IDE, r0, 4*DLC], vagy [r0,r1,4*DLC]
- Az első bit alapján eldönthető, hogy extended, vagy standard mező.
- A DLC (Data Length Counter) megmondja, hogy milyen hosszú adatmező követi.
- Szerepe van a hibafelismerésben.

Data Field, CRC, EOF, Interframe

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- Adatok 0-8 byte
- CRC 15 (15 bit+1 bit lezáró)
- 6 Hamming távolság
- 5 bit hibát észlel
- Nem hibajavító
- EOF – 7 recesszív bit (bitstuffing nélkül)
- Interframe Space (legalább 3 bit)

Keret Típusok

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- Data Frame
- Remote Frame
- Error Frame
- Overload frame

Data Frame/Remote frame

CAN Standard Frames

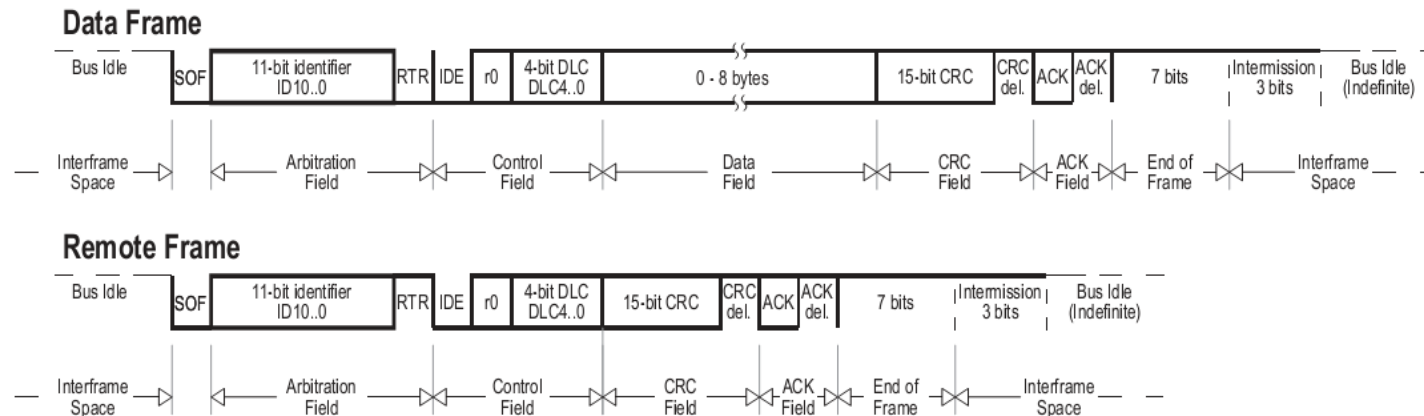
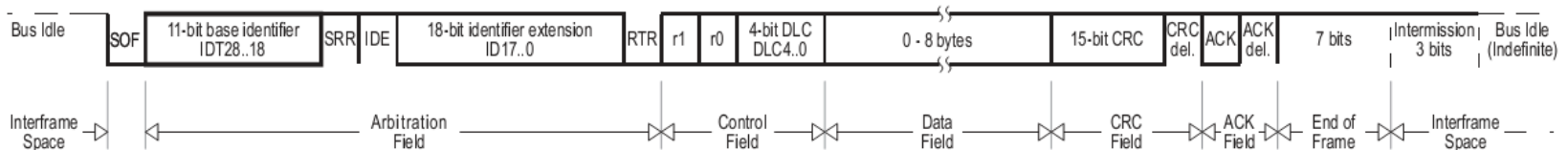
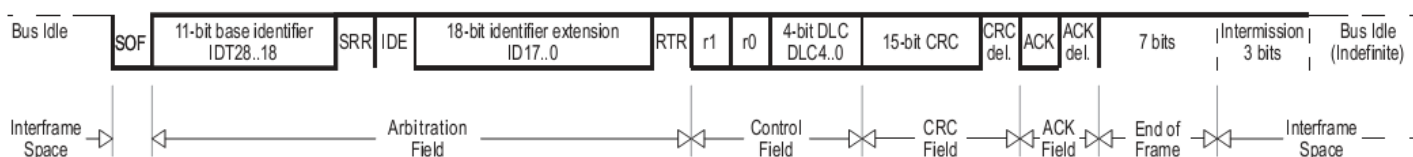


Figure 19-2. CAN Extended Frames

Data Frame



Remote Frame

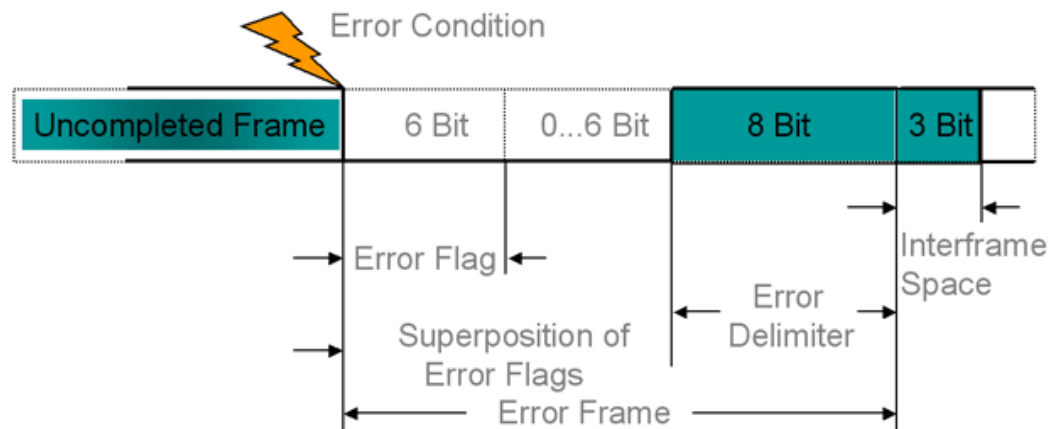


Data Frame/Remote Frame

- Az üzenetek adat keretekben közlekednek, de előfordulhat, hogy az egyik fél adatokat kér egy másiktól, ekkor ún. Remote Frame-et küld, aminek azonosítója megegyezik a Data Frame-el.
- Az „ütközést” ilyenkor az RTR bit küszöböli ki:
 - 0 domináns data frame-ben
 - 1 recesszív remote frame-ben
- Ütközés esetén tehát a remote frame küldő csomópont érzékeli, hogy azonos dataframe „jön”.

Error frame

- Hiba észlelése esetén az érzékelő csomópont
 - 6 domináns bitet küld a hálózatra (bit stuffing nélkül)
 - Ezt a vevők érzékelik, így tovább 0-6 bitnyi (domináns) error flag keletkezik a hálózaton
 - Ezt 8 bitnyi recesszív hiba lezáró bit követi

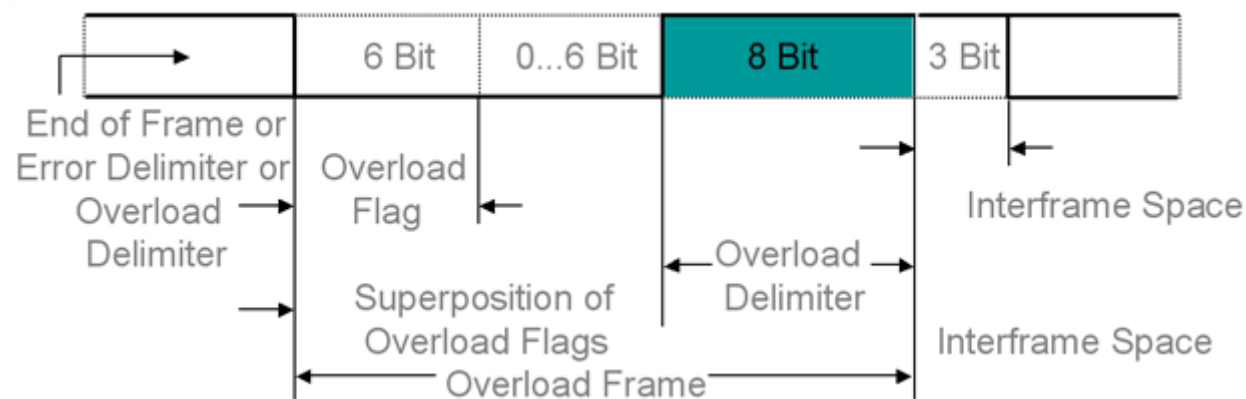


Hibafajták

- **BERR:** Bit error – a kiküldött bit nem egyezik a visszaolvasottal
 - Kivétel: Arbitráció, error frame
- **SERR:** Stuff Error – Több mint 5 azonos bit észlelése
- **CERR:** CRC Error – (RX)
- **FERR:** Form error – (CRC del., ACK del., EOF, Error del., overload del.)
- **AERR:** Acknowledgment error

Overload frame

- Megegyezik az error frame-el, de csak az Interframe space-ben küldhető
- Késlelteti a következő üzenetet



A DBC file rövid ismertetése (Data Base for CAN)

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

- Általánosan használt CAN hálózat leíró
- Üzenetek leírása négyféle sortípussal:
 - BO_ - Message/Üzenet leíró,
 - SG_ - Signal/Jel leíró és
 - CM_ Comment leíró sorok.
 - SIG_VALTYPE_ - speciális signal
- Teljesség igénye nélkül egyéb sortípusok:
 - BU_ - Hálózati csomópontok felsorolása,
 - VAL_ - Értéktábla
 - BA – Attribútum sor

Csomópontok definiálása

A Hálózati csomópontok definiálása a BU_ sorral történhet meg, a csomópontok neveinek felsorolásával:

BU_: #Node1 #Node2 #Noden

Például:

BU_ Node1 Node2

Az egyes elemek tartalma a következő:

Elem	Formátum	Tartalom
#NodeX	String	Csomópont azonosító

Üzenet definiálása

Egy Message leíró sor formátuma a következő:

BO_ #Canid #Name: #DLC #Node

Például:

BO_ 2298429457 New_Message_7: 7 Vector_XXX

Az egyes elemek tartalma a következő:

Elem	Formátum	Tartalom
#Canid	U32	CAN ID azonosító
#Name	String	Az üzenet neve
#DLC	U8	DLC
#Node	String	Csomópont azonosítója (‘Vector___XXX’- unassigned)

Jelek definiálása

A message leíró sor után következnek az üzenethez tartozó „Signalok” felsorolása.

Egy Signal formátuma a következő:

SG_ #Name : #Startbit|#Length@#Byteorder#Valuetype (#Factor,#Offset) [#Min|#Max] "#Unit" Vector_XXX

Például

SG_ New_Signal_13 : 32|1@1- (1.1,0.1) [-1|0] "" Vector_XXX

Az egyes elemek tartalma a következő:

Elem	Formátum	Tartalom
#Name	String	A Signal neve
#Startbit	U8	A Signal Kezdő bit pozíciója
#Length	U8	A Signal hossza bitben
#Byteorder	(0/1)	Endianness (1-Intel, 0-Motorola)
#Valuetype	(+/-)	Előjel (+ előjel nélküli, - előjeles)
#Factor	Float	Erősítés
#Offset	Float	Eltolás
#Min	Float	Minimumérték
#Max	Float	Maximumérték
#Unit	String	Mértékegység

Vége

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedésautomatikai Tanszék

Köszönöm a figyelmet!