

# Vhodno-izhodne naprave (VIN)

Predavanja

## 10. Povezovalni standardi

RS-232, RS-485, CANBUS, Modbus, USB, ...

Robert Rozman

[rozman@fri.uni-lj.si](mailto:rozman@fri.uni-lj.si)

# Vsebina

---

## □ Povezovalni standardi

1. EIA/TIA-232 (RS-232)
2. EIA/TIA-485 (RS-485)
3. CANBUS
  - ▶ Opis protokola in fizičnega nivoja
  - ▶ Primer uporabe:
    - Integra BM: HW moduli in IEX-2 odprta protokolska nadgradnja, ...
    - Vsebina LAB vaje : CANBUS, STM32 in IEX-2 protokol
4. Modbus:
  - ▶ Opis protokola in fizičnih nivojev
  - ▶ Primeri Modbus komunikacij:
    - STM32 (MB master) <-> PC (MB slave)
    - Cybro kot MB Slave (višje nivojsko omrežje)
5. USB - Universal Serial Bus
6. PCI Express (PCIe)

# 10. Povezovalni standardi - uvod

- ❑ **Standard** je dokumentiran dogovor, ki vsebuje tehnične specifikacije ali druge natančno določene kriterije. Lahko ga imenujemo tudi **priporočilo**.
- ❑ Uporablja se kot pravila, smernice ali definicije lastnosti, da bi materiali, izdelki ali storitve ustrezali svojim namenom.
- ❑ Zakaj **standardi**?
  - Omogočajo **pravilno komunikacijo med programi ali programom in strojno opremo** (slike, tekst, zvok, telefonski pogovori).
  - Neuporaba lahko onemogoči komunikacijo, ali pa vodi k napačni interpretaciji informacij.
- ❑ Kako pride do njihove uveljavitve?
  - **Neformalno sprejemanje** standardov  
Primer: TTL napetostni nivoji (0V – low, 5V – high) so pri V/I lastnostih logičnih vezij postali standard zaradi možnosti povezovanja čipov različnih proizvajalcev.
  - **Formalno sprejemanje** standardov - običajno jih definirajo združenja proizvajalcev ali organizacije:
    - IEEE – Institute of Electrical and Electronic Engineers
    - EIA – Electronic Industries Association
    - TIA – Telecommunications Industry Association

- ❑ Za povezavo dveh sistemov, dveh naprav, dveh modulov, dveh tiskanih vezij želimo uporabiti serijski prenos podatkov.
- ❑ Kako lahko to izvedemo?
- ❑ Kaj potrebujemo?
  - Prenosni medij
  - Konektor
  - Združljivost

- **RS-232 – RS-232**



- **USB – RS-232**



- **USB – RS-485**



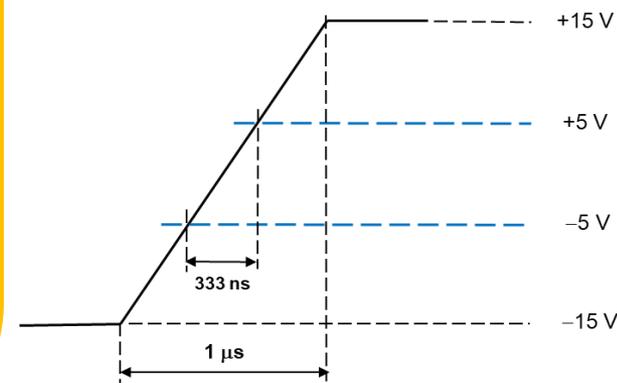
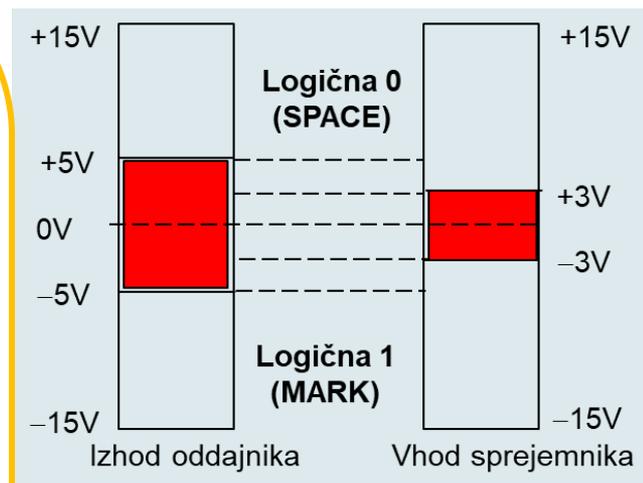
## 10.1 EIA/TIA-232 (RS-232)

- ❑ Standard RS 232 določa:
  - napetostne in signalne nivoje ter časovne lastnosti
  - pomen in vrste signalov in
  - razpored pinov na konektorjih in konfiguracije povezav
  - minimalno količino krmilnih informacij med gostiteljskim in perifernim sistemom.
  
- ❑ Začetek in razvoj standarda:
  - RS-232 – leta 1962 (organizacija EIA)
  - RS-232 C leta 1969
  - RS-232 D leta 1986
  - TIA-232 F leta 1997 (še vedno v veljavi)
  
- ❑ Standard RS-232 omogoča široko združljivost in zanesljivo povezovanje V/I naprav.
  
- ❑ Vir: <https://www.virtual-serial-port.org/article/what-is-serial-port/rs232-pinout/>

# RS-232 na kratko



- asimetrični prenos (ang. unbalanced) podatkov,
- s skupnim povratnim vodnikom za vse signale,
- in napetostno nezaključeno napajanje linije
  - $R_{IZH} < R_0$  in  $R_{VH} > R_0$
- Šumna imuniteta:  $5\text{ V} - 3\text{ V} = 2\text{ V}$
- za omejevanje odbojev in presluha je uporabljeno razmerje  $\frac{t_r}{\tau}$ .
- hitrost spreminjanja signala je omejena na  $30\text{ V}/\mu\text{s}$  (ang. slew rate)
- ❑ Asinhrona komunikacija – ni urinega signala
- ❑ Format podatka
  - Prenos: Start,  $b_0, \dots, b_7$ , Stop; Ni paritetnega bita.
- ❑ Standardne baudne (bitne) hitrosti
  - 1200, 2400, 4800, 19200, 38400, 57600, 115200 bps



## 10.1.1 Električni del standarda

### □ Napetostna in logična nivoja

**Logična 0** : pozitiven napetostni nivo

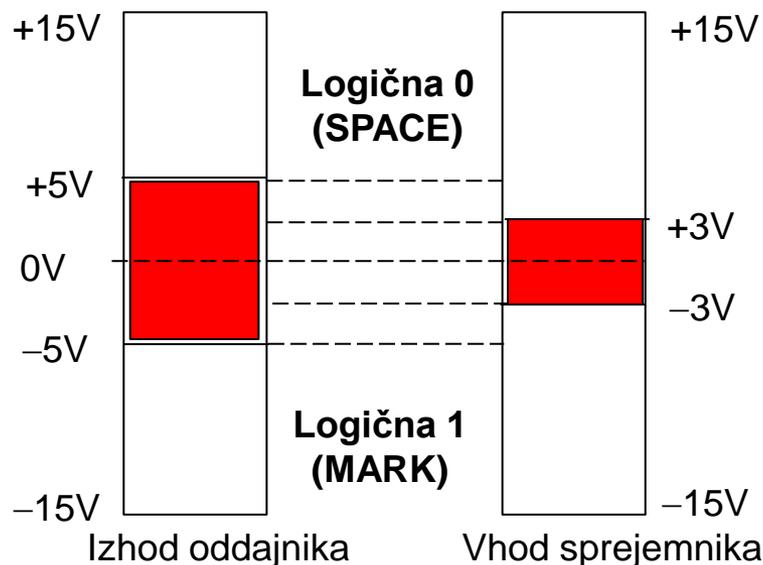
Oddajnik: od +5V do +15V

Sprejemnik: od +3V do +15V

**Logična 1** : negativen napetostni nivo

Oddajnik: od -5V do -15V

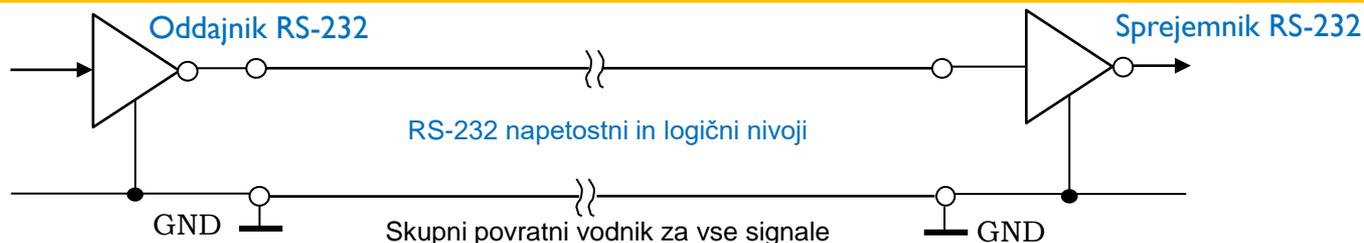
Sprejemnik: od -3V do -15V



- **Šumna imuniteta** je napetostna razlika med izhodom oddajnika in vhodom sprejemnika, ki pri neugodnih razmerah določa odpornost signalov na šum ( $5\text{ V} - 3\text{ V} = 2\text{ V}$ ).

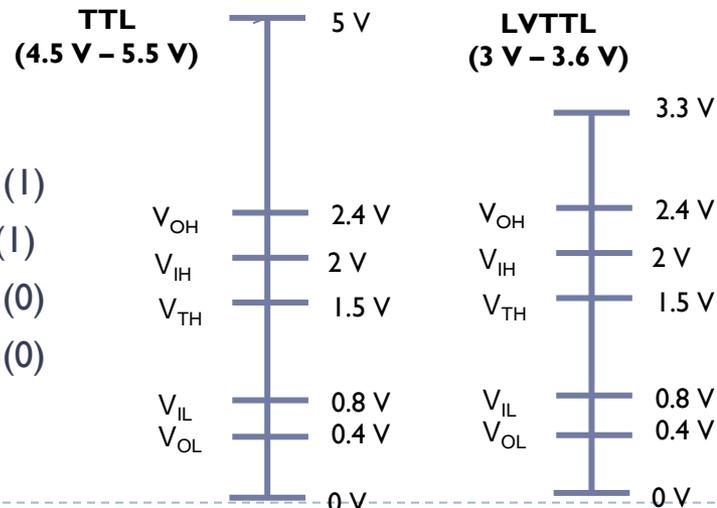
## □ V RS-232 je uporabljen

- asimetrični prenos (ang. unbalanced) podatkov,
- s skupim povratnim vodnikom za vse signale,
- in napetostno nezaključeno napajanje linije ( $R_{IZH} < R_0$  in  $R_{VH} > R_0$ )



### Napetostni nivoji za TTL in LVTTTL:

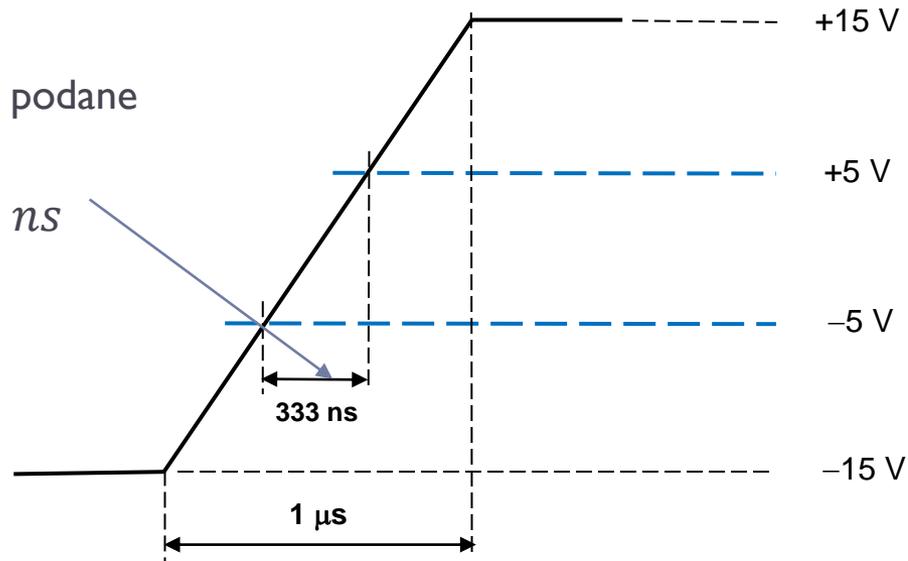
- $V_{OH}$  = Minimalna izhodna napetost pri visokem nivoju (1)
- $V_{IH}$  = Minimalna vhodna napetost pri visokem nivoju (1)
- $V_{IL}$  = Maksimalna vhodna napetost pri nizkem nivoju (0)
- $V_{OL}$  = Maksimalna izhodna napetost pri nizkem nivoju (0)
- $V_{TH}$  = Napetost preklopa (preklopni prag)



- Za omejevanje odbojev in presluha je uporabljeno razmerje  $\frac{t_r}{\tau}$ .
- Hitrost spreminjanja kateregakoli signala je omejena na  $30 \text{ V}/\mu\text{s}$  (ang. slew rate).

Čas vzpona  $t_r$  signala je za podane napetostne omejitve

$$t_r \geq 333 \text{ ns}$$

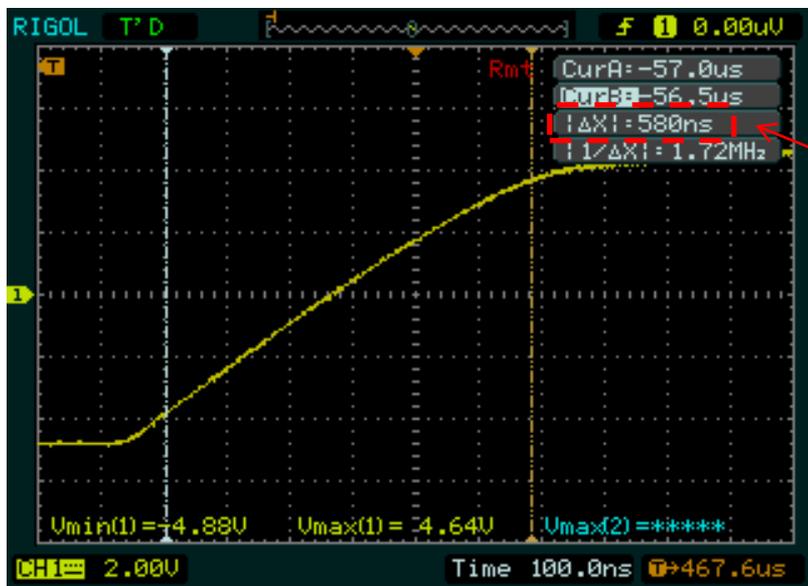


- Pri upoštevanju pogoja za omejevanje odbojev  $t_r \geq (3 \div 5)\tau$ , standard RS-232 priporoča **največjo dolžino povezave 15 m**.

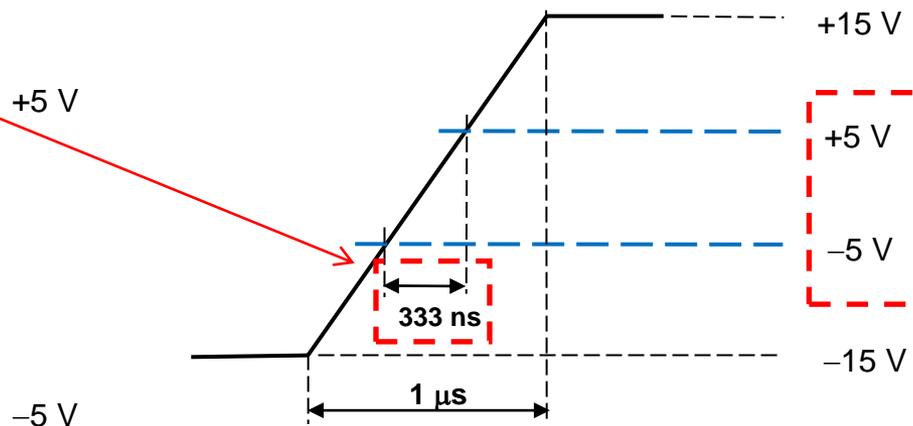
Izračun:  $t_r \geq 4\tau$  in  $\delta = 5,6 \left[ \frac{\text{ns}}{\text{m}} \right]$

$$\text{čas vzpona } t_r \geq 4\tau = 4 \cdot l \cdot \delta \rightarrow l \leq \frac{t_r}{4 \cdot \delta} = \frac{333 \text{ [ns]}}{4 \cdot 5,6 \text{ [ns/m]}} = 15 \text{ [m]}$$

- Primer meritve časa vzpona (Slew Rate) – VIN LAB vaja



580 ns



580 ns

□ RS-232 povezava dveh naprav

- Data Terminal Equipment (DTE)
- Data Communication Equipment (DCE)

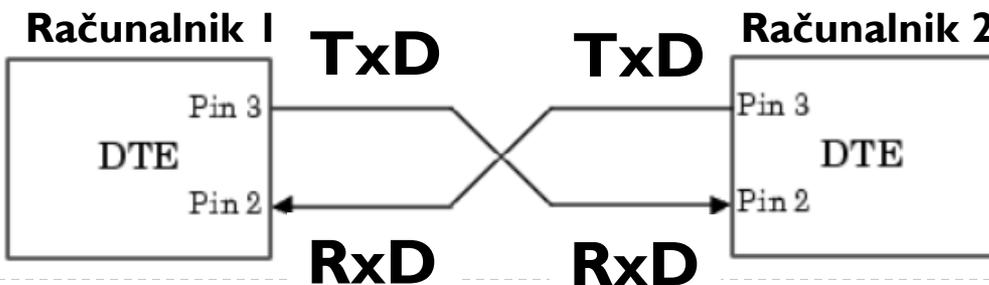
DB9 Pins	DTE	In/Out		In/Out	DCE	DB9 Pins
1	DCD	In	↔	In	DCD	1
2	RXD	In	←	Out	TXD	2
3	TXD	Out	→	In	RXD	3
4	DTR	Out	→	In	DSR	4
5	GND	--	--	--	GND	5
6	DSR	In	←	Out	DTR	6
7	RTS	Out	→	In	CTS	7
8	CTS	In	←	Out	RTS	8
9	RI	In	↔	In	RI	9

Prenos podatkov:

- Povezava naprav DTE in DCE (direktna povezava)



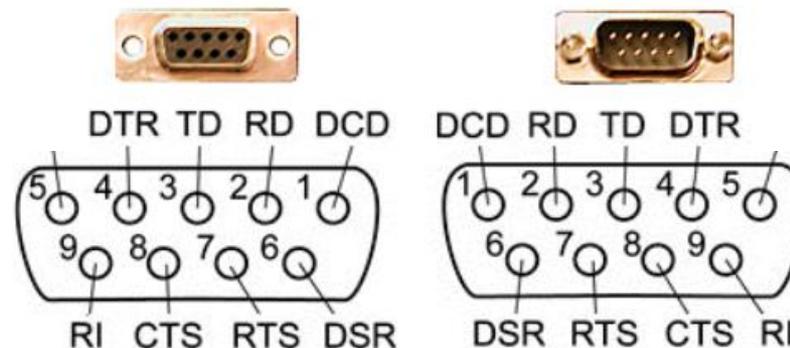
- Povezava dveh naprav DTE (navzkrižna povezava)



□ Vrste in razpored signalov na 9-pinskem konektorju (DB9 Male):

- RD (ang. receive data) - sprejem podatka
- TD (ang. transmitt data) – oddaja podatka
- RTS (ang. request to send) - zahteva za pošiljanje podatka
- CTS (ang. clear to send) – pripravljenost za sprejem podatka
- DCD (ang. data carrier detect) – sprejem prenosa iz oddaljene lokacije
- DSR (ang. data set ready) – pripravljenost za prenos podatka
- DTR (ang. data terminal ready) – pripravljenost za sprejem zahteve
- RI (ang. ring indicator) – zazna prihajajoči signal iz linije
- GND (ang. ground) – ozemljitev

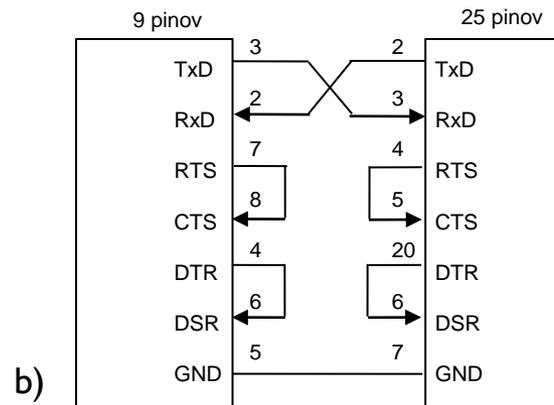
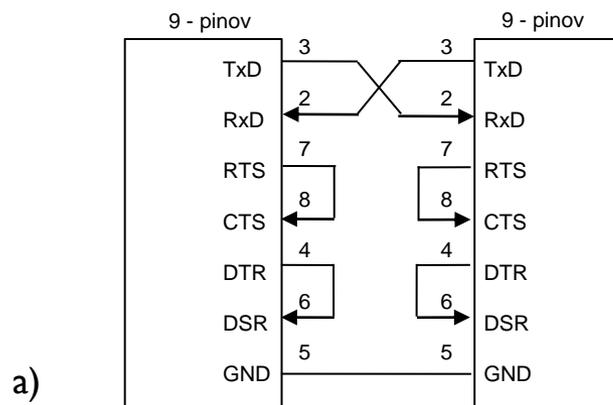
- Konektorja DB9 Female in DB9 Male



## 10.1.2 Načini delovanja RS-232

### a) Enostavno **ne-usklajeno delovanje** (ang. simple no-handshaking)

- Komunikacija deluje s predpostavko, da **sprejemnik lahko bere podatke iz vmesnega pomnilnika** (ang. buffer) predno je sprejet nov znak.
- Podatek je poslan iz oddajnika (pin TxD) in sprejet na sprejemniku (pin RxD).
- Prikaz povezave med dvema vozliščema:
  - a) 9-pinski konektor – 9 pinski konektor
  - b) 9-pinski konektor – 25 pinski konektor

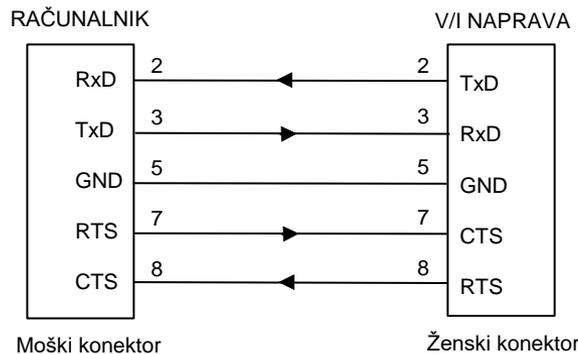


### b) **Programsko usklajeno delovanje** (ang. software handshaking)

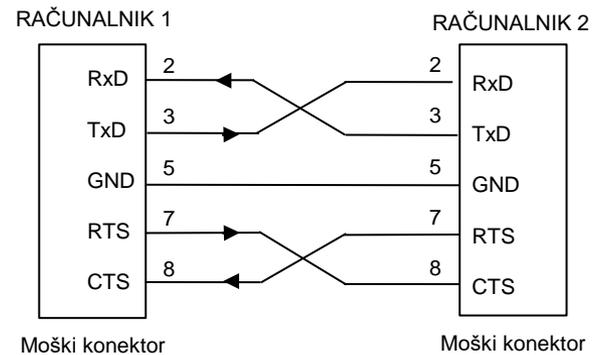
- Hitro procesiranje tako na oddajniku, kot sprejemniku
- Znaka ASCII sta uporabljena za začetek (start - Ctrl-S) in konec prenosa (stop - Ctrl-Q)

c) **Signali so uporabljeni za usklajevalno delovanje (ang. hardware handshaking)**

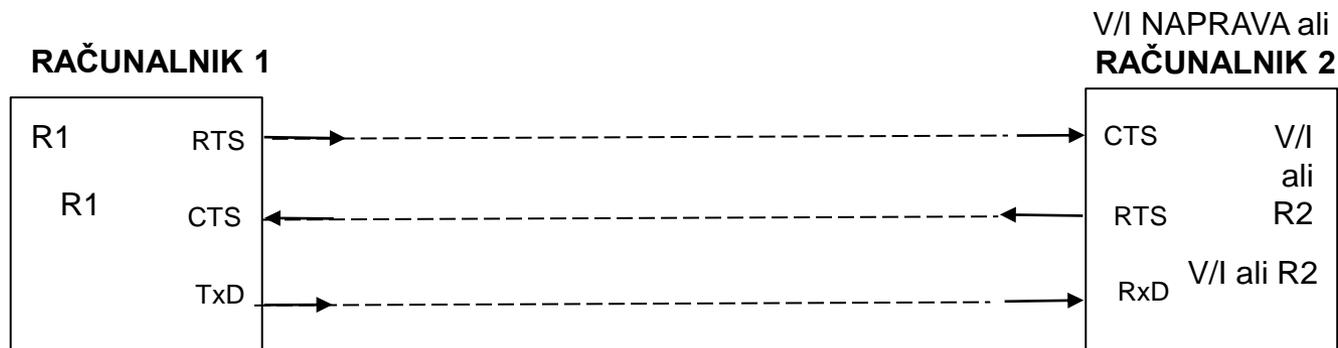
Računalnik – V/I naprava

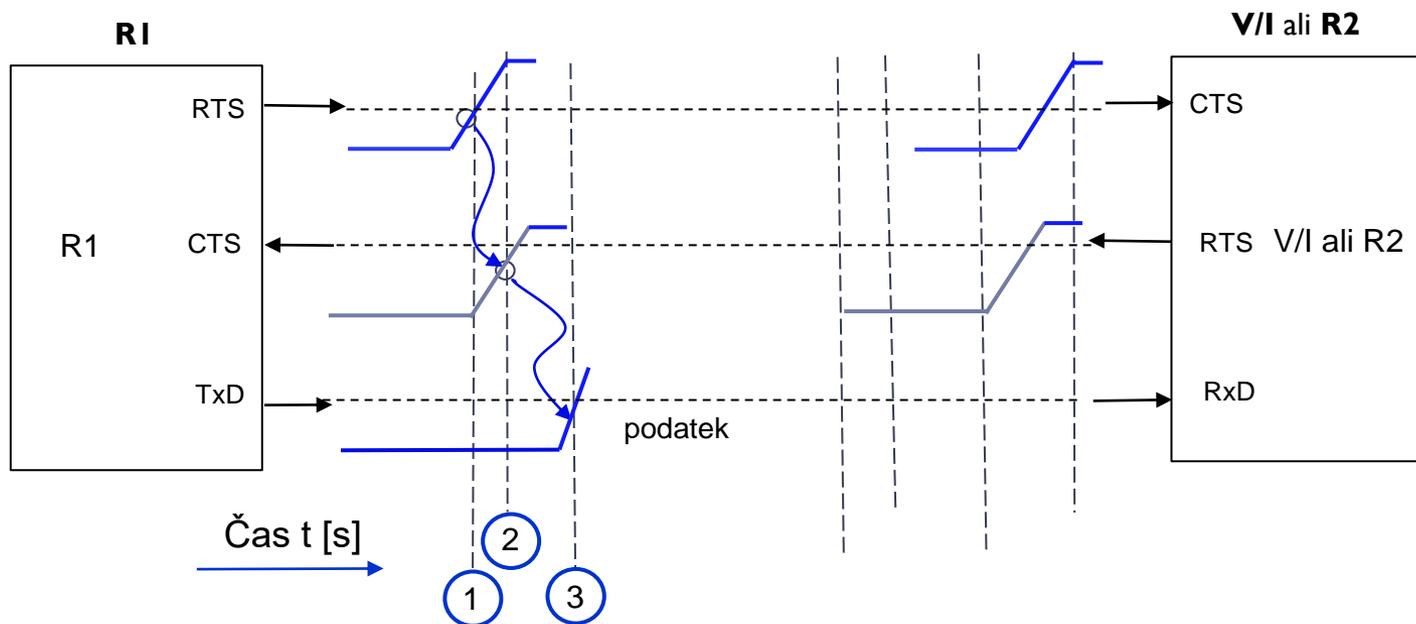


Računalnik 1 – Računalnik 2

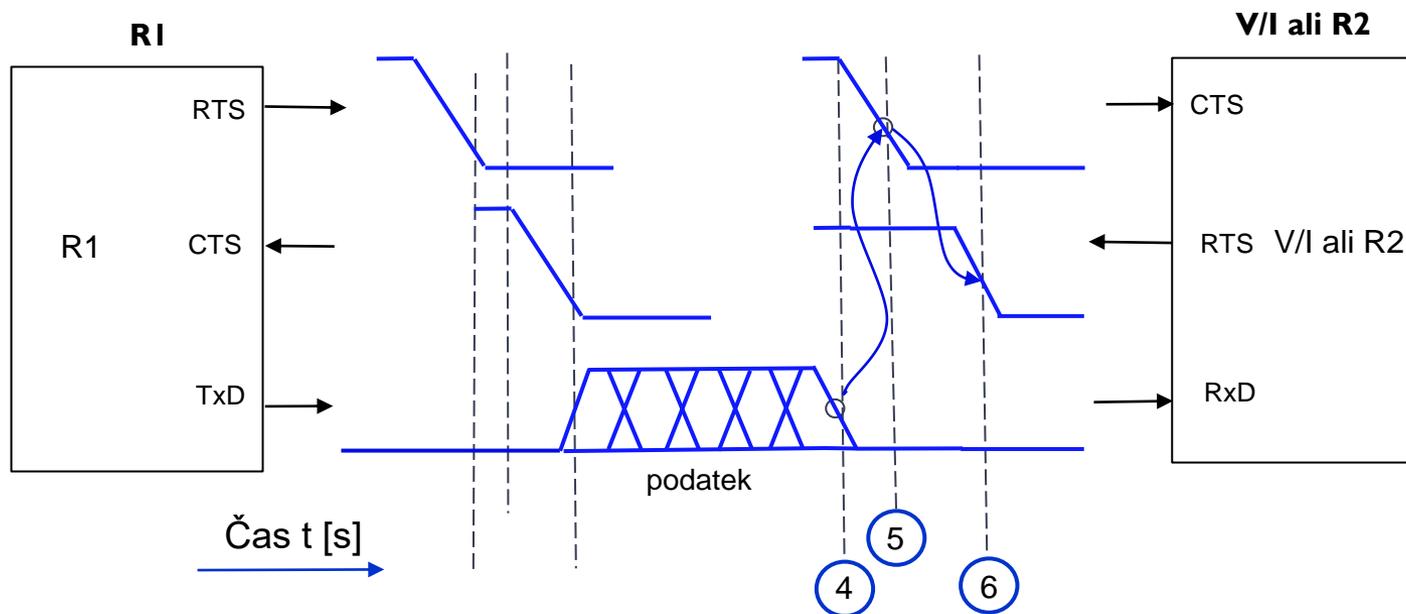


□ Primer za prenos podatka: RAČUNALNIK 1 → V/I NAPRAVA ali RAČUNALNIK 2

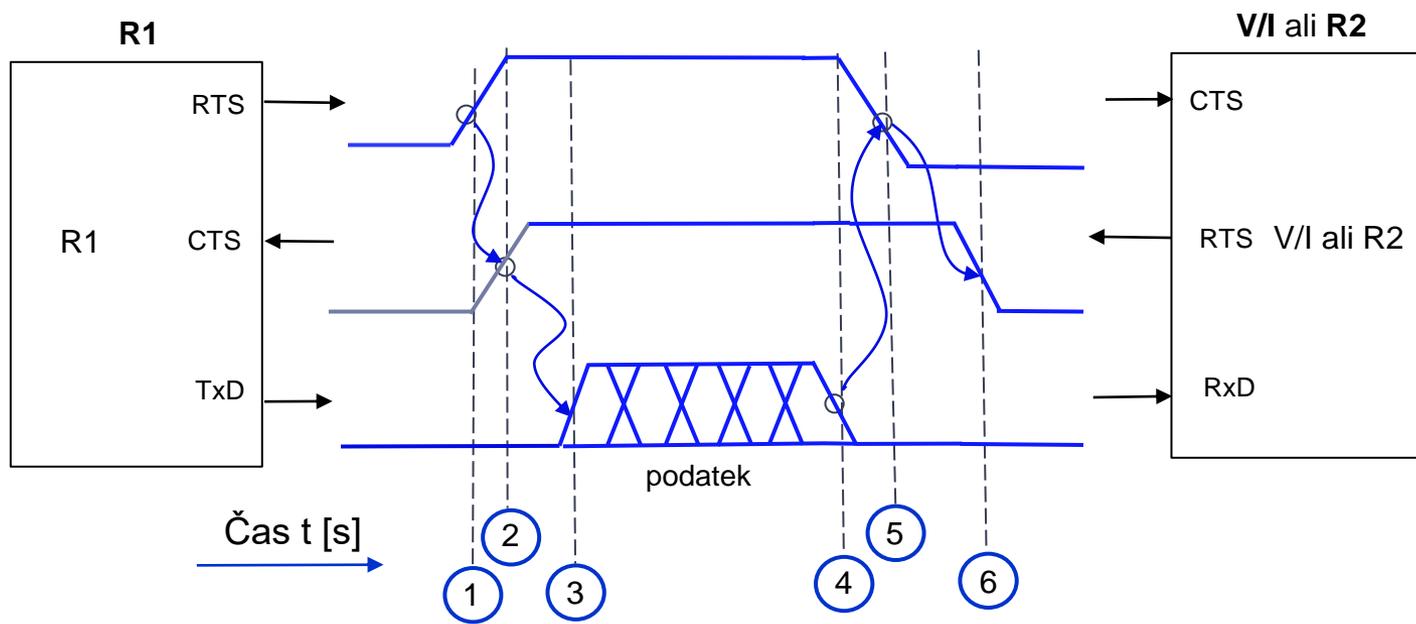




- (1) R1 aktivira signal RTS, ko želi poslati podatek
- (2) Ko V/I ali R2 sprejme signal RTS na svoj CTS vhod, odgovori s signalom RTS, ki ga R1 sprejme na vhod CTS, kar pomeni da V/I ali R2 lahko sprejme podatek.
- (3) R1 prične s pošiljanjem podatka



- (4) R1 konča pošiljanje
- (5) R1 deaktivira signal RTS
- (6) V/I ali R2 odgovori - deaktivira svoj RTS



- (1) **RI aktivira signal RTS**, ko želi poslati podatek
- (2) Ko V/I ali R2 sprejme signal RTS na svoj CTS vhod, **odgovori s signalom RTS**, ki ga R1 sprejme na vhod CTS, kar pomeni da V/I ali R2 lahko sprejme podatek.
- (3) **RI prične s pošiljanjem** podatka
- (4) **RI konča pošiljanje**
- (5) **RI deaktivira signal RTS**
- (6) V/I ali R2 odgovori - **deaktivira svoj RTS**

## 10.1.3 Lastnosti RS-232

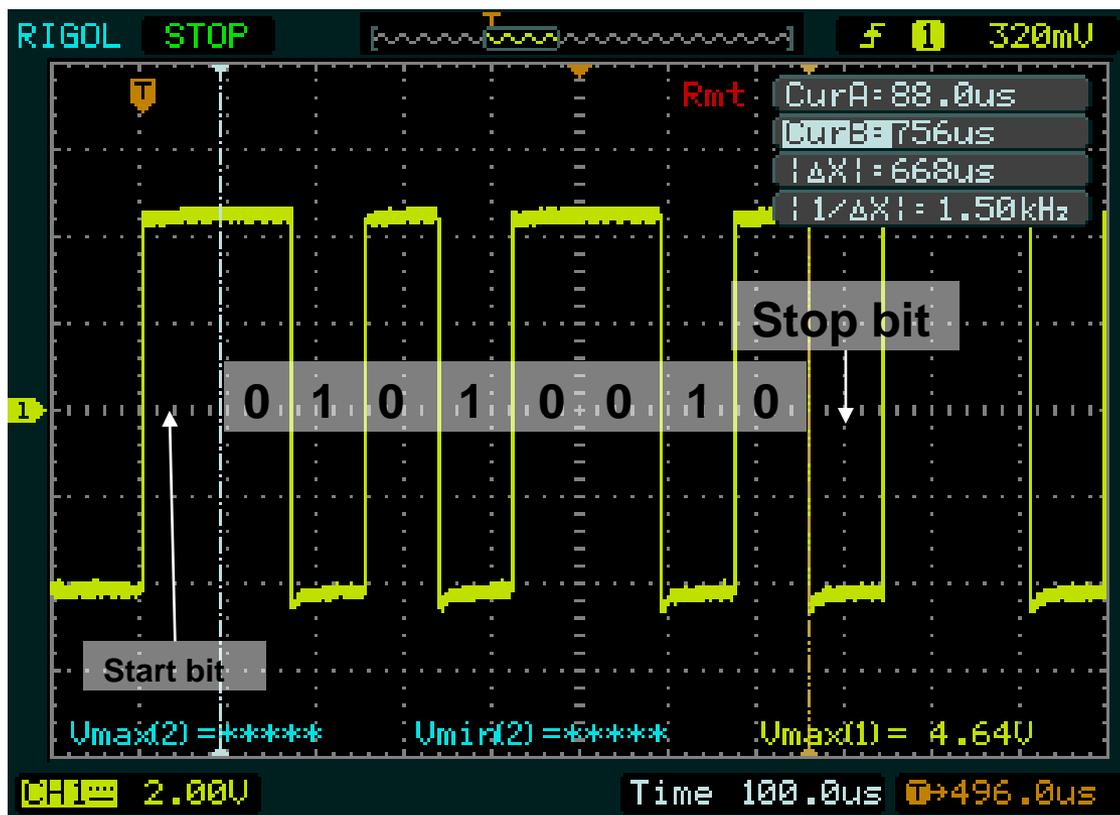
- ❑ Napetostni nivoji:
  - Logična 1: -3V do -15V; -5V do -15V
  - Logična 0: +3V do +15V; +5V do +15V
- ❑ Asinhrona komunikacija – ni urinega signala
- ❑ Format podatka
  - Prenos: Start,  $b_0, \dots, b_7$ , Stop; Ni paritetnega bita.



- ❑ Asimetrična povezava
  - Dvosmerna povezava (ang. full duplex)
  - Povezava dveh naprav na isto maso (GND).
  - Večja občutljivost na šum.
  - Omejitev dolžine povezave je 15 m.
- ❑ Baudna hitrost
  - Običajna - 9600 bps
  - Druge standardne možnosti: 1200, 2400, 4800, 19200, 38400, 57600, 115200 bps

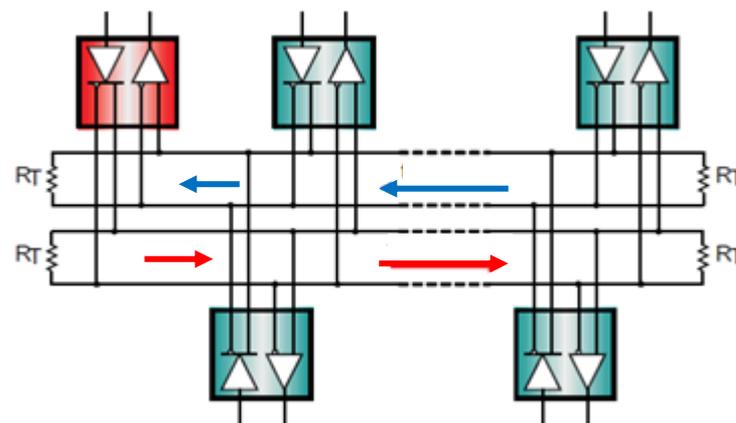
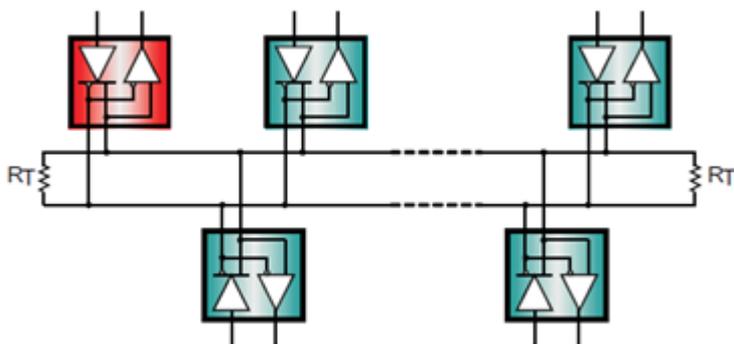
## 10.1.4 RS232 na osciloskopu (VIN LAB)

- ❑ Start bit , 8 podatk. bitov (obratni vrstni red), Stop bit (mirovno stanje)
- ❑ Poslan znak na sliki je  $0b01001010 = 0x4a = 'j'$



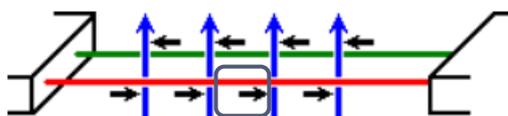
## 10.2 EIA/TIA-485 (RS-485)

- Standard RS-485 definira **električne lastnosti** oddajnikov, linije in sprejemnikov za simetrično povezavo v obliki vodila.
  - Določa simetrični, **diferencialni prenos v obe smeri** (half-duplex ali full-duplex).
  - Priporoča uporabo **parice** in **zaključitev linije** za omejevanje odbojev.
  - Omogoča **hitrosti do 35 Mb/s** in več ter dolžino **povezave do 1200 m**, vendar ne hkrati (višja kot je hitrost  $\Leftrightarrow$  krajša je povezava in obratno)
  - Na vodilu je tako **število oddajnikov, kot število sprejemnikov do 32**.
  - Če obremenitev ni presežena, je lahko sprejemnikov tudi več, do 256.
  - Primer: half-duplex in full-duplex struktura vodila v RS-485

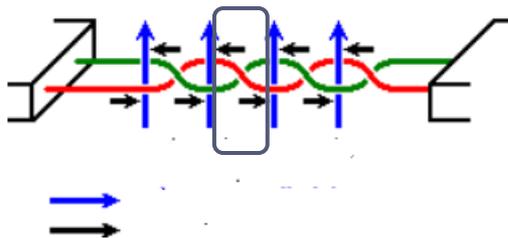


- ❑ Standard ne določa posebnega protokola za prenos podatkov, obratno pa veliko protokolov uporablja RS-485 kot električni standard povezave (npr. Modbus)
- ❑ Povezave po standardu RS-485 so zaradi simetrične povezave in uporabe paric zelo uporabne v industrijskih okoljih:
  - Neobčutljive za šum in premik potenciala ozemljitve
  - Ne povzročajo motenj navzven (presluh na sosednje povezave)
  - Možne so višje hitrosti prenosa
  - Kjer ni zahtevana velika hitrost, so povezave lahko zelo dolge (npr. v procesni industriji)
  - Primer za raven kabel in sukano parico

Raven kabel

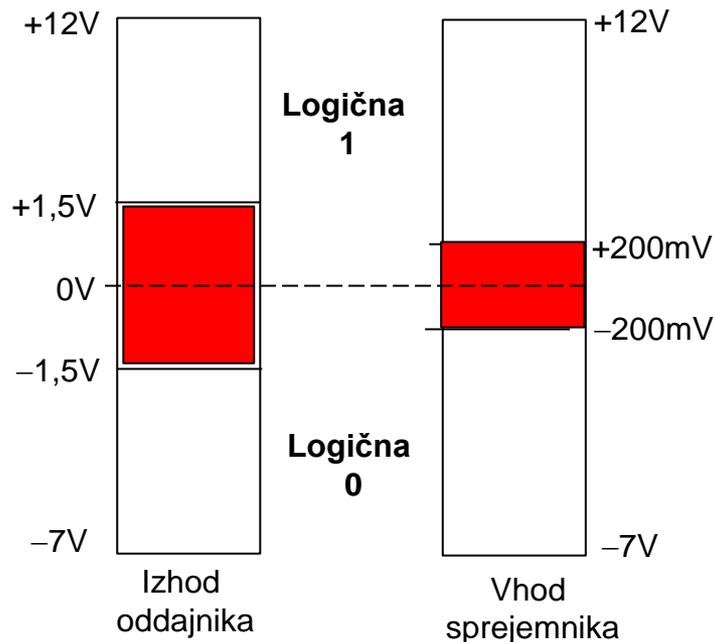


Sukana parica

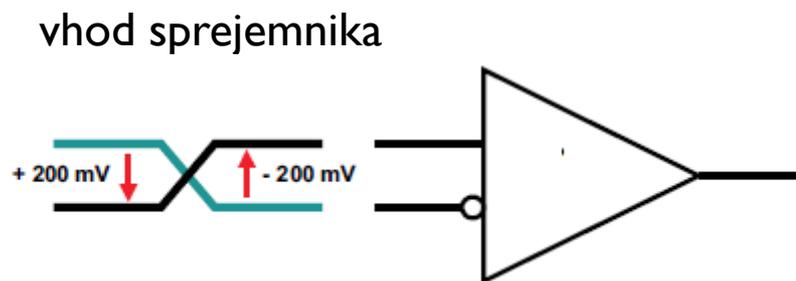
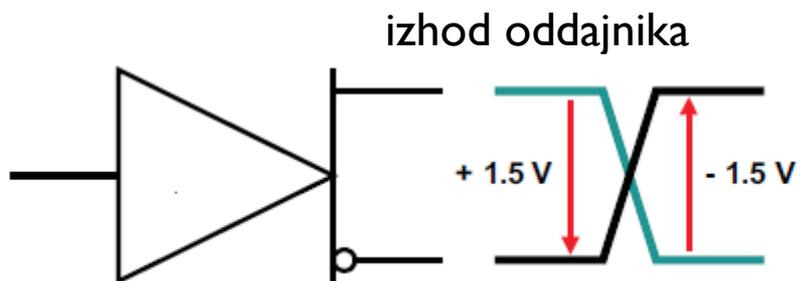


## Logični nivo – napetostni nivo

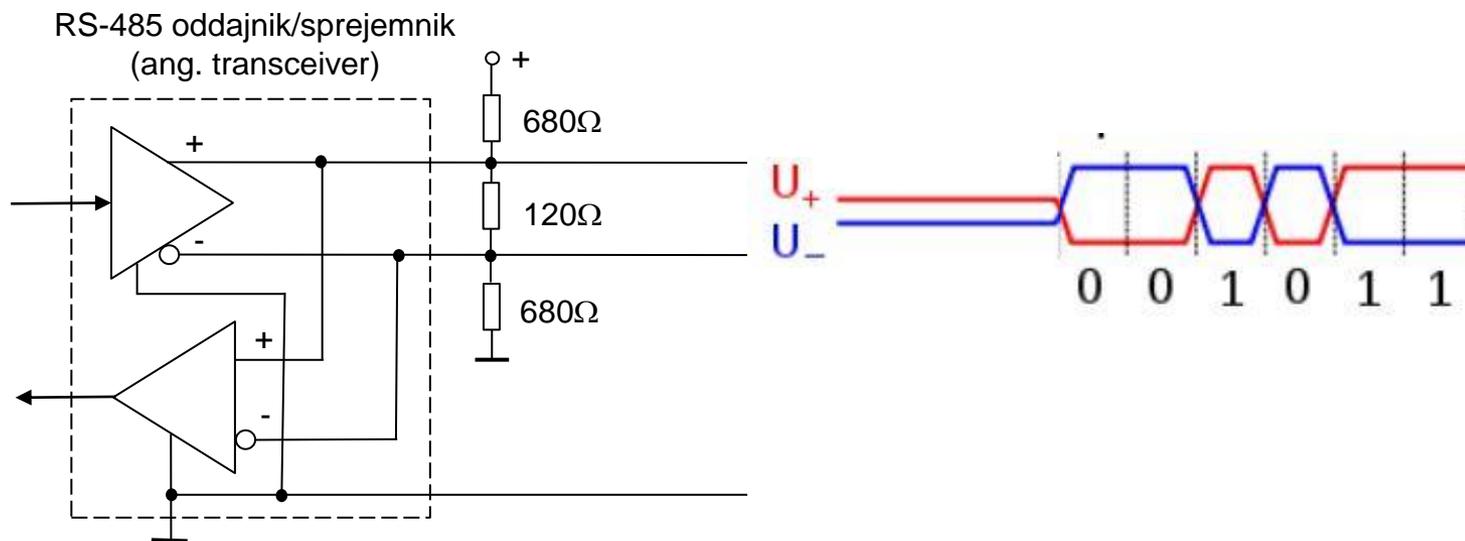
- Logična 0 : negativen napetostni nivo
  - Oddajnik: od -1,5V do -7V
  - Sprejemnik: od -200 mV do -7V
- Logična 1 : pozitiven napetostni nivo
  - Oddajnik: od +1,5V do +12V
  - Sprejemnik: od +200 mV do +12V



## Minimalne napetosti



- Za omejevanje odbojev se uporablja **zaključitev linije** na obeh koncih ( $120\Omega$ ), upornost zaključitvenega upora je odvisna od karakteristične impedance linije ( $Z_0$ ), zato je standard ne specificira.
- Primer:



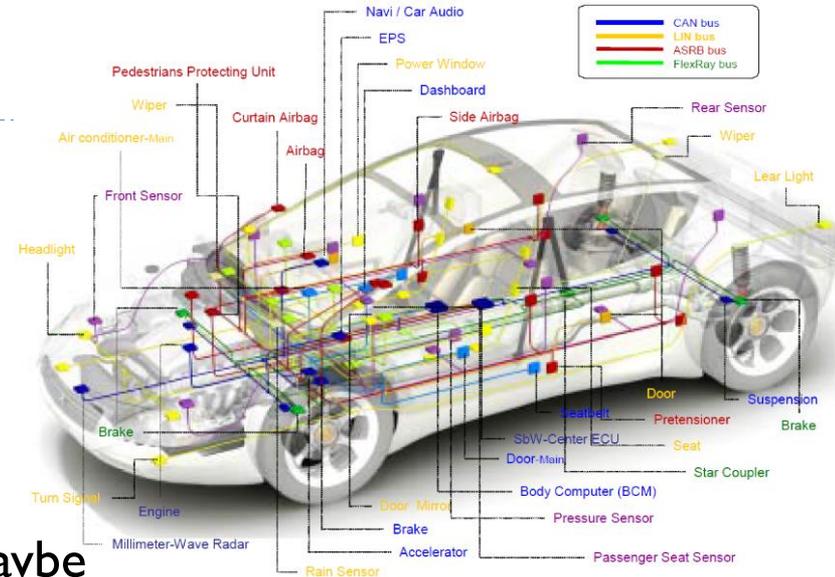
□ Povzetek standardov RS-232 in RS-485

	RS232	RS485
Differential	no	yes
Max number of drivers	1	32
Max number of receivers	1	32
Modes of operation	half duplex full duplex	half duplex
Network topology	point-to-point	multipoint
Max distance (acc. standard)	15 m	1200 m
Max speed at 12 m	20 kbs	35 Mbs
Max speed at 1200 m	(1 kbs)	100 kbs
Max slew rate	30 V/ $\mu$ s	n/a
Receiver input resistance	3..7 k $\Omega$	$\geq$ 12 k $\Omega$
Driver load impedance	3..7 k $\Omega$	54 $\Omega$
Receiver input sensitivity	$\pm$ 3 V	$\pm$ 200 mV
Receiver input range	$\pm$ 15 V	-7..12 V
Max driver output voltage	$\pm$ 25 V	-7..12 V
Min driver output voltage (with load)	$\pm$ 5 V	$\pm$ 1.5 V

# 10.3. CANBUS vodilo

## CANBUS (ISO-11898-2):

- ▶ Zgodovina
- ▶ Področja uporabe
  - ▶ **Avtomobilska** industrija
  - ▶ Industrijska **avtomatika**, pametne stavbe
- ▶ Pregled protokola, arbitraže, fizičnega nivoja
- ▶ Praktični primer: Pametna hiša, IEX-2 protokol

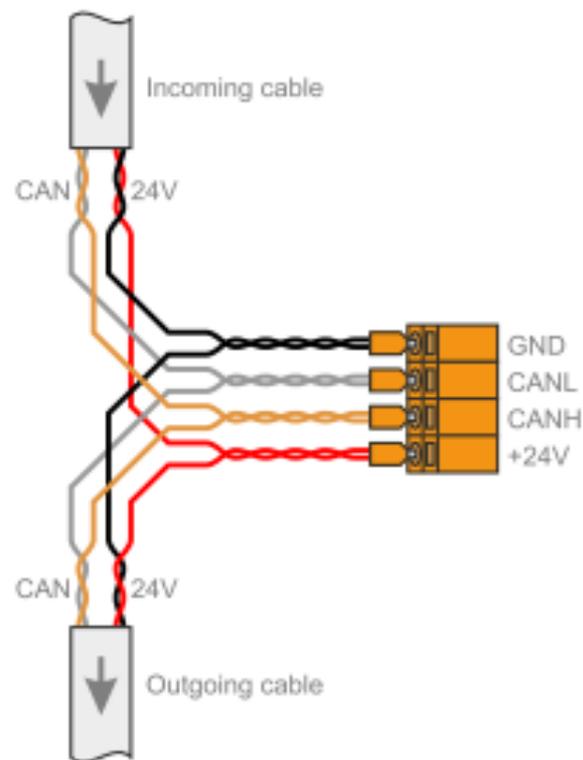


## Lab. vaja LV5:

- ▶ **Gradniki in shema** testnega sistema
- ▶ **Programiranje** sistema
- ▶ **Meritve signalov** na povezavah

# CANbus na kratko

- ▶ **CAN**bus – **C**ontroller **A**rea **N**etwork bus
- ▶ CAN (Controller Area Network) je serijsko vodilo za komunikacijo med vgrajenimi mikrokrmilniki
- ▶ CAN bus na kratko :
  - ▶ serijsko vodilo
  - ▶ dve žici (CAN\_H, CAN\_L) + napajanje,
  - ▶ diferencialni prenos signala
    - ▶ odpornost na šum.
  - ▶ max 1 Mbit/s, 40m,
  - ▶ sporočila do 8 bajtov (latenca)
- ▶ CAN-FD standard, ISO 11898-2:2016
  - ▶ 2Mbps, 5Mbps, do 64 bajtov/sporočilo

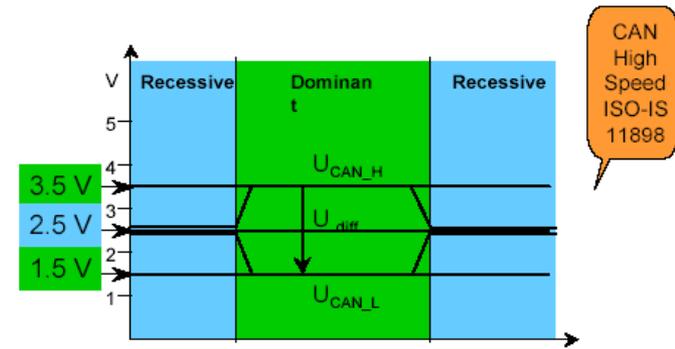


# CANbus na kratko

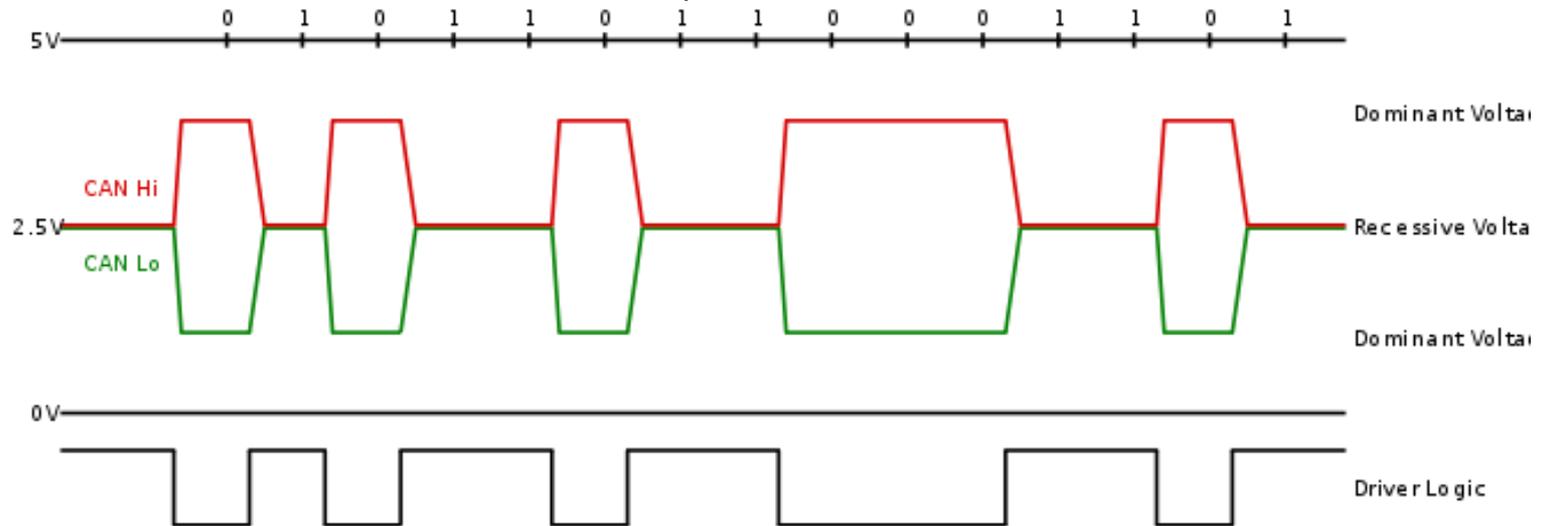
---

- ▶ Komunikacijski protokol:
  - ▶ CSMA/CD
    - ▶ Carrier Sense Multiple Access/Collision Detection (Avoidance)
  - ▶ z NDA
    - ▶ Non-Destructive Arbitration
- ▶ Sporočilno orientiran :
  - ▶ ni naslova, vsako sporočilo ima svojo **ID številko** (prioriteta, vsebina)
  - ▶ potrjevanje (ACK) vsaj enega sprejemalca

# CANbus na kratko

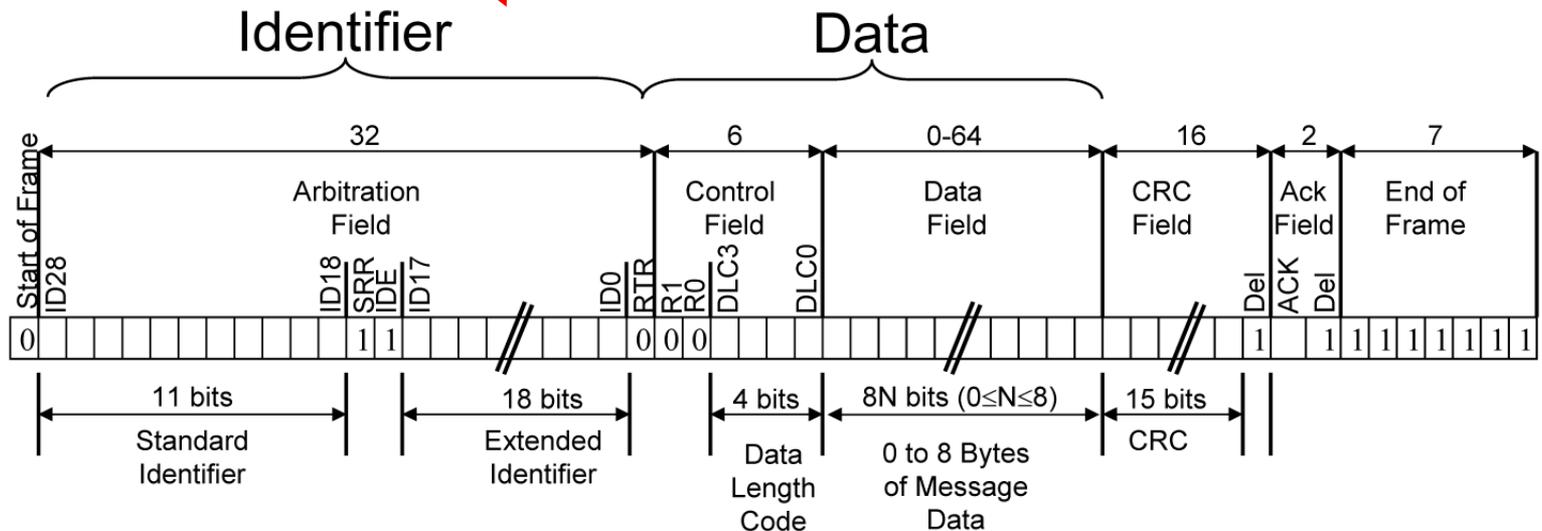


- ▶ **Diferencialni prenos** običajno na parici - Non-Return To Zero (NRZ) in bit-stuffing.
- ▶ Wired – AND povezava: vozlišče z logično 0 prevlada
  - ▶ 0 .. „dominant“, 1 .. „recessive“)



# CANbus na kratko

- ▶ Prenos podatkov
  - ▶ Format okvirja
  - ▶ Protokol – sporočilno naravn
  - ▶ Detekcija napake
    - ▶ Nivo Bitov (branje, „bit stuffing“)
    - ▶ Nivo sporočila (CRC, okvir, ACK napake)



# CANbus na kratko

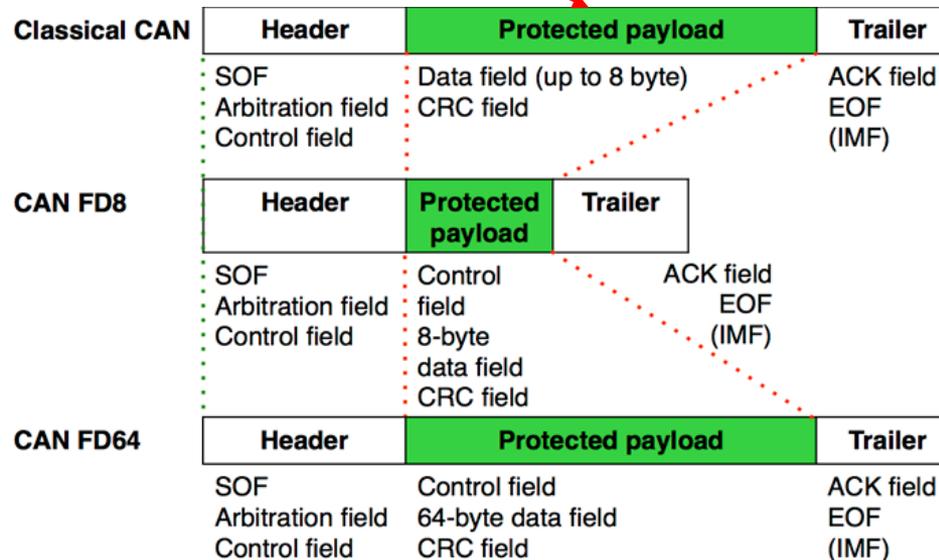
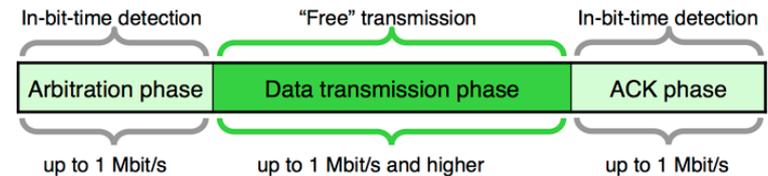
## ▶ CAN FD

▶ Rezultat trenda potrebe po večji pasovni širini

▶ Združljiv s CAN

▶ 2 izboljšavi :

- ▶ Daljše podatkovno polje (do 64 bajtov)
- ▶ Višje bitne hitrosti v podatkovnem bloku



# CANbus zgodovina

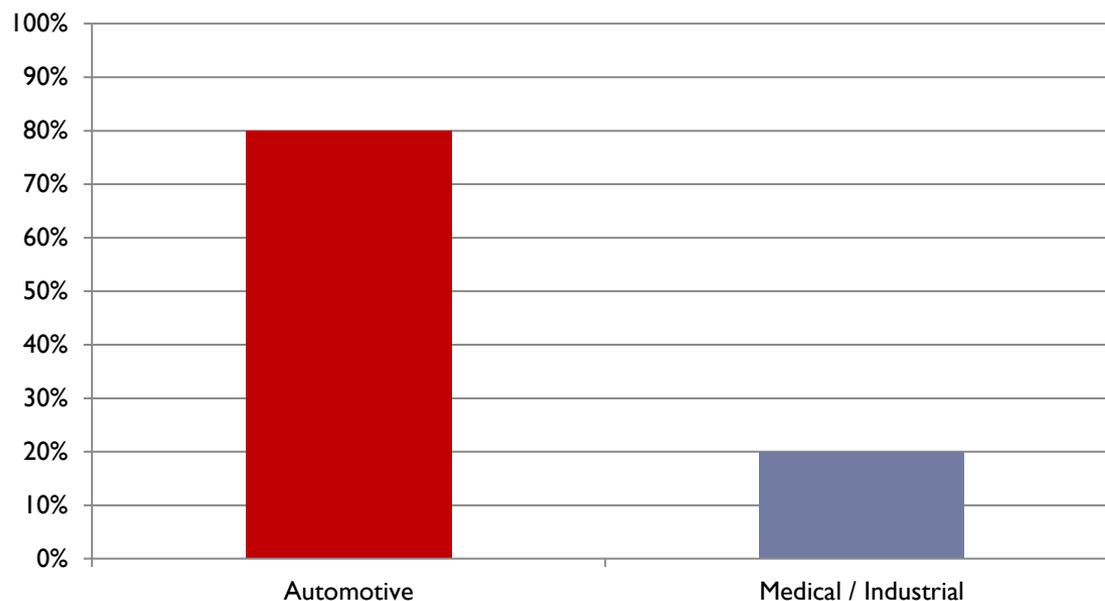
---

- ▶ 1983 : Prvi CANBUS projekt - Bosch
- ▶ 1986 : CAN protokol predstavljen (BMW 850)
- ▶ 1987 : Prvi CAN mikrokontrolerji prodani
- ▶ 1991 : CAN 2.0A specifikacija objavljena
- ▶ 1992 : Mercedes-Benz uporabi CAN omrežje
- ▶ 1993 : ISO 11898 standard
- ▶ 1995 : ISO 11898 amandma
- ▶ Dandanes : večina vozil uporablja CANbus, industrijska avtomatika

# Uporaba CANBUS-a

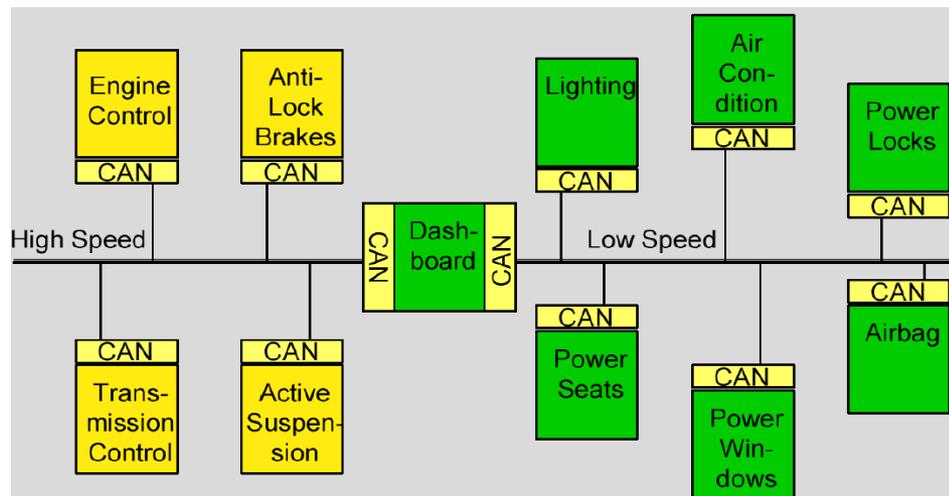
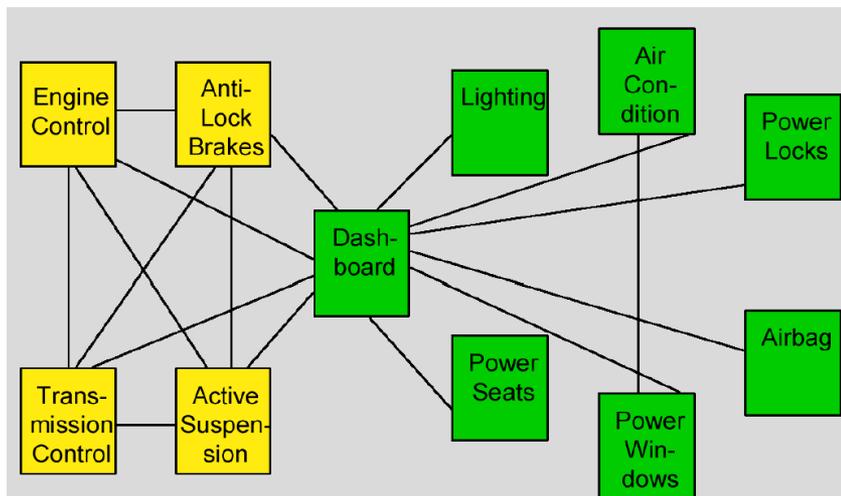
- ▶ Zasnovan za avtomobilsko industrijo
- ▶ Danes: Industrija/medicina

## CANBUS Tržni deleži



# Zakaj vodilo ?

Primeri povezav brez (levo) in z (desno) CANbus v avtomobilu

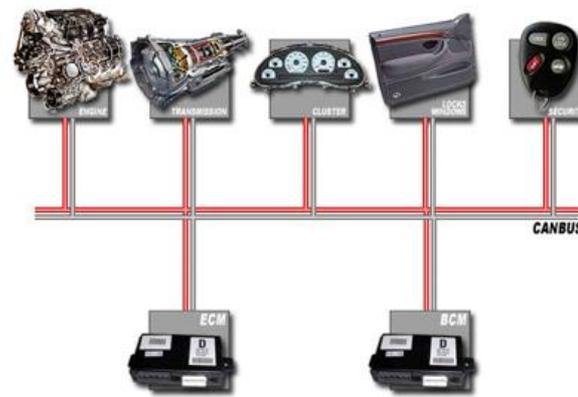


Conventional multi-wire looms

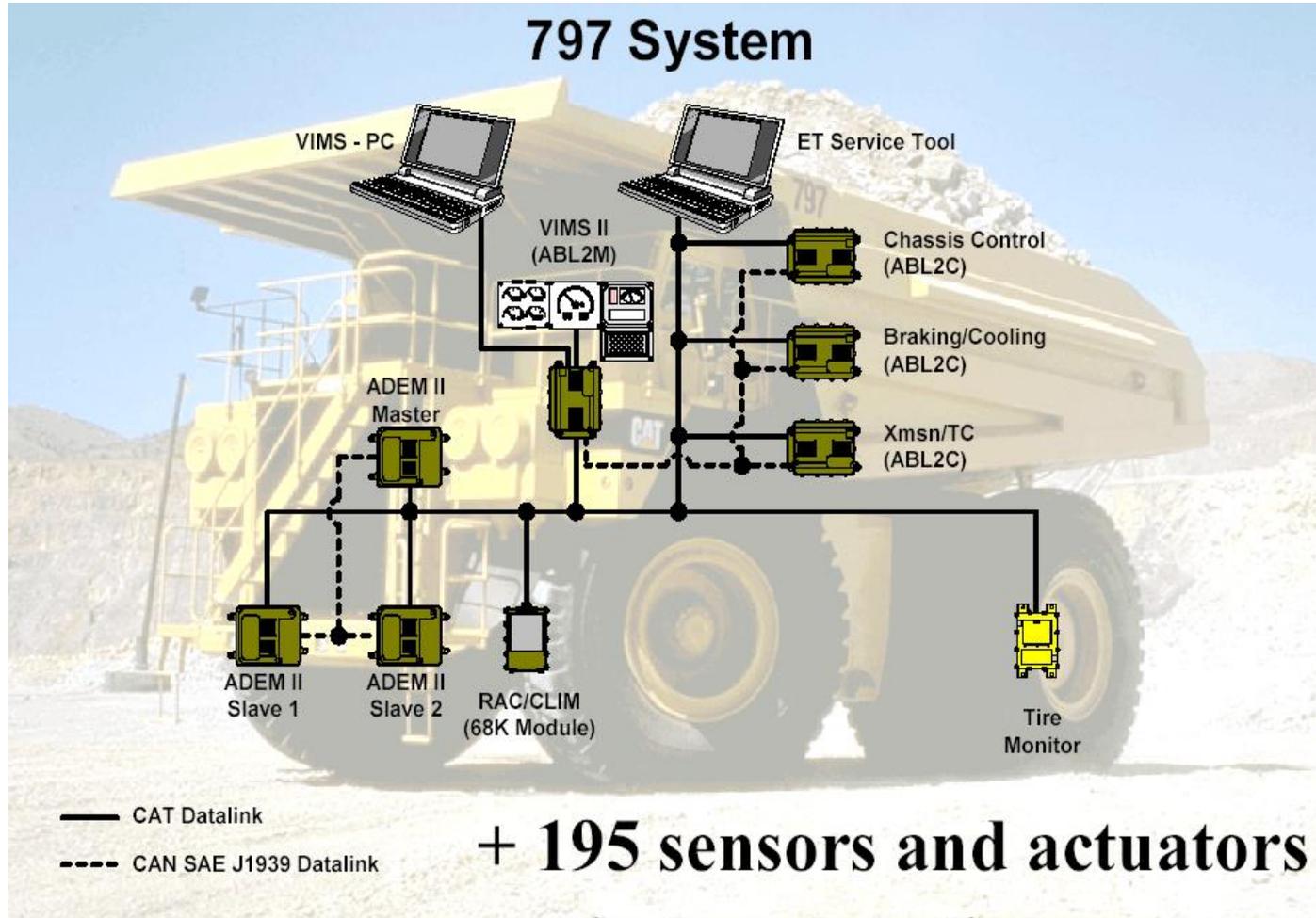


CAN bus network

VS.



# CAN (SAE J1939) Primer: Caterpillar 797

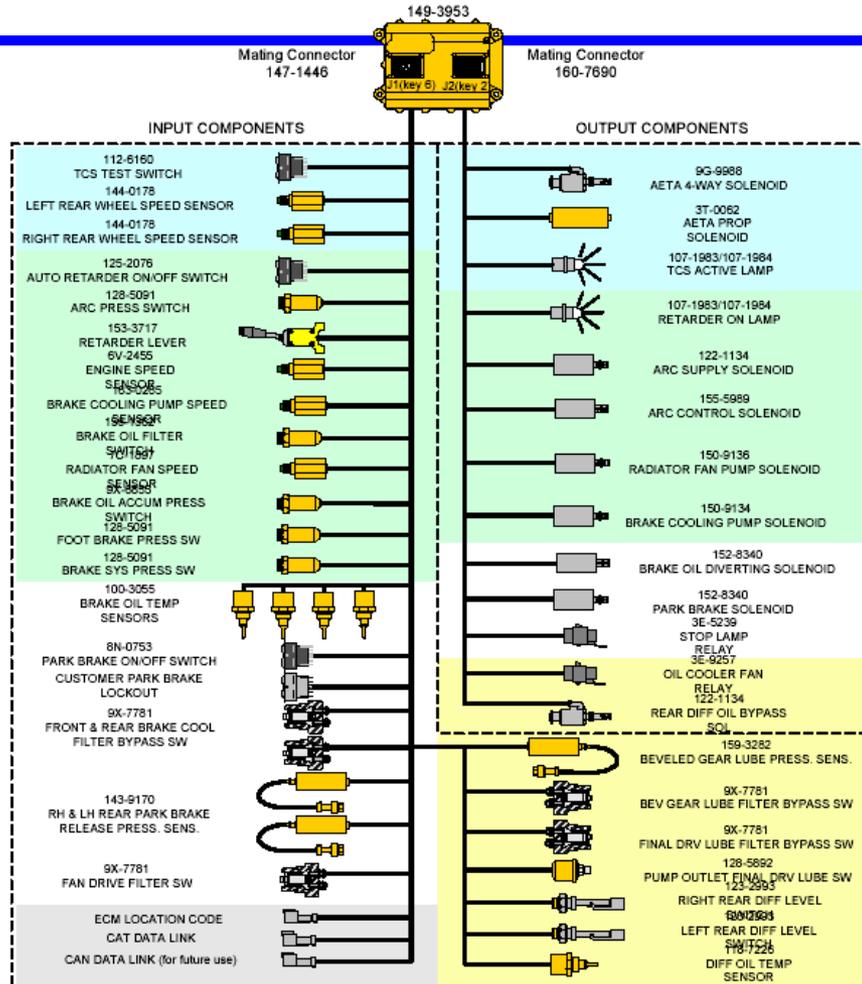


797sys.vsd  
6-18-98  
dab/yrf  
Warning: All paper copies of this document are uncontrolled

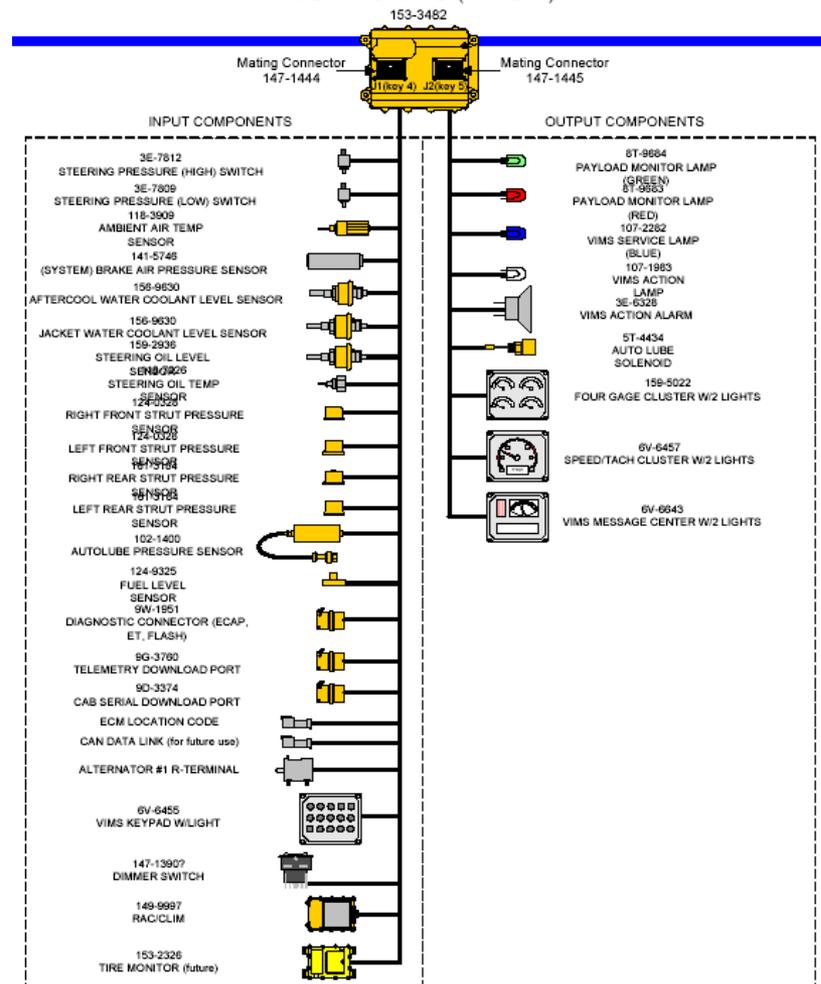
[Caterpillar]

# Caterpillar example

## BRAKING/COOLING CONTROL

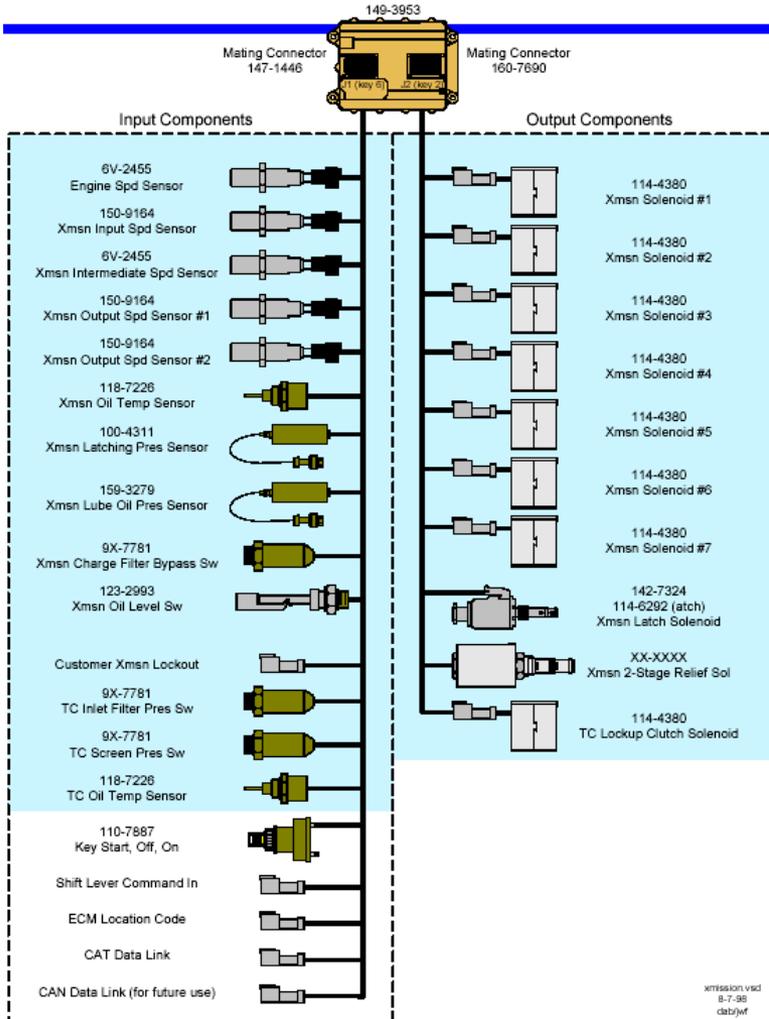


## MONITORING (VIMS-II)

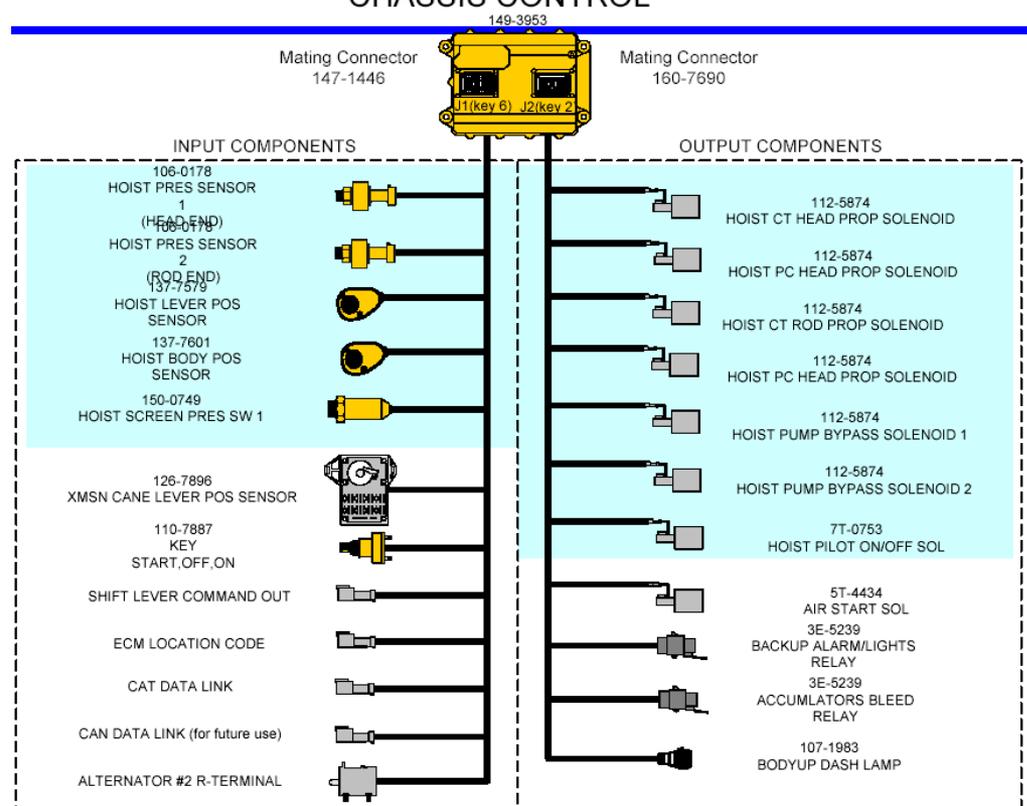


# Caterpillar example

## 797 - TRANSMISSION CONTROL



## CHASSIS CONTROL

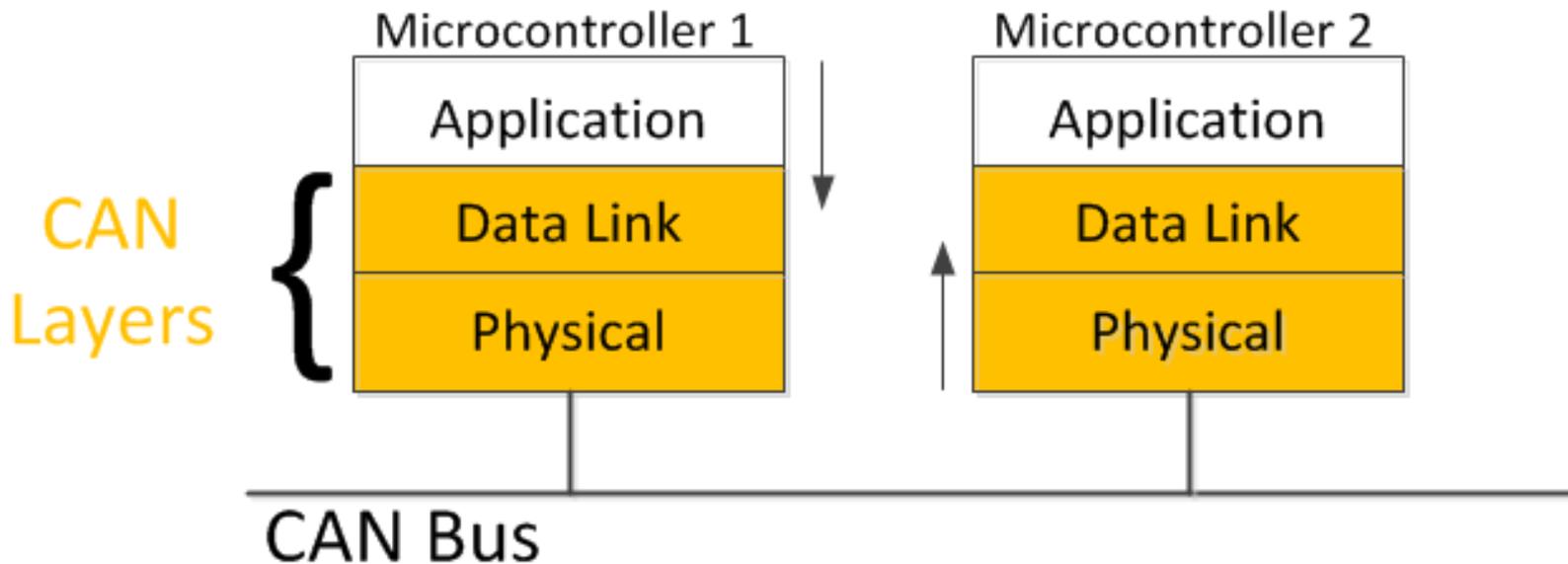


## “SAE Standard Workload” (53 messages) V/C = Vehicle Controller

Signal Number	Signal Description	Size /bits	J /ms	T /ms	Periodic /Sporadic	D /ms	From	To
1	Traction Battery Voltage	8	0.6	100.0	P	100.0	Battery	V/C
2	Traction Battery Current	8	0.7	100.0	P	100.0	Battery	V/C
3	Traction Battery Temp, Average	8	1.0	1000.0	P	1000.0	Battery	V/C
4	Auxiliary Battery Voltage	8	0.8	100.0	P	100.0	Battery	V/C
5	Traction Battery Temp, Max.	8	1.1	1000.0	P	1000.0	Battery	V/C
6	Auxiliary Battery Current	8	0.9	100.0	P	100.0	Battery	V/C
7	Accelerator Position	8	0.1	5.0	P	5.0	Driver	V/C
8	Brake Pressure, Master Cylinder	8	0.1	5.0	P	5.0	Brakes	V/C
9	Brake Pressure, Line	8	0.2	5.0	P	5.0	Brakes	V/C
10	Transaxle Lubrication Pressure	8	0.2	100.0	P	100.0	Trans	V/C
11	Transaction Clutch Line Pressure	8	0.1	5.0	P	5.0	Trans	V/C
12	Vehicle Speed	8	0.4	100.0	P	100.0	Brakes	V/C
13	Traction Battery Ground Fault	1	1.2	1000.0	P	1000.0	Battery	V/C
14	Hi&Lo Contactor Open/Close	4	0.1	50.0	S	5.0	Battery	V/C
15	Key Switch Run	1	0.2	50.0	S	20.0	Driver	V/C
16	Key Switch Start	1	0.3	50.0	S	20.0	Driver	V/C
17	Accelerator Switch	2	0.4	50.0	S	20.0	Driver	V/C
18	Brake Switch	1	0.3	20.0	S	20.0	Brakes	V/C
19	Emergency Brake	1	0.5	50.0	S	20.0	Driver	V/C
20	Shift Lever (PRNDL)	3	0.6	50.0	S	20.0	Driver	V/C
21	Motor/Trans Over Temperature	2	0.3	1000.0	P	1000.0	Trans	V/C
22	Speed Control	3	0.7	50.0	S	20.0	Driver	V/C
23	12V Power Ack Vehicle Control	1	0.2	50.0	S	20.0	Battery	V/C
24	12V Power Ack Inverter	1	0.3	50.0	S	20.0	Battery	V/C
25	12V Power Ack I/M Contr.	1	0.4	50.0	S	20.0	Battery	V/C
26	Brake Mode (Parallel/Split)	1	0.8	50.0	S	20.0	Driver	V/C

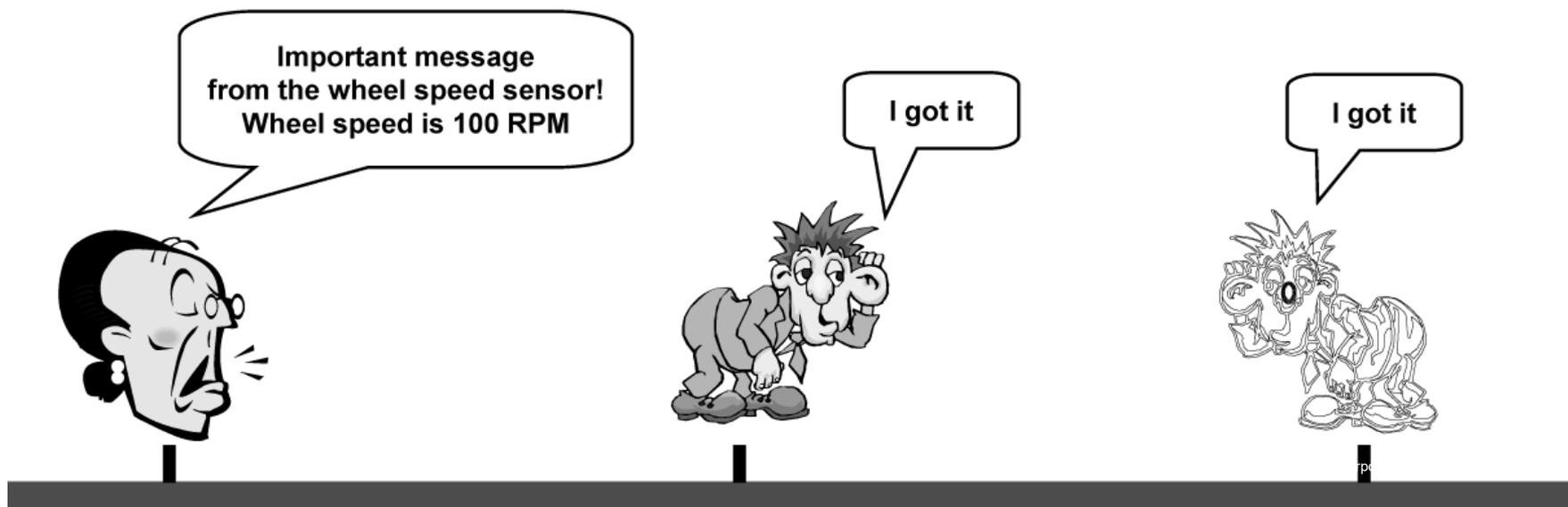
# CANBUS in OSI Model

- ▶ CAN je zaprto omrežje
  - ▶ ni potrebe po varnosti, sejah in prijavah
  - ▶ ni zahtev po uporabniški izkušnji.
- ▶ Fizični in podatkovni nivoji:

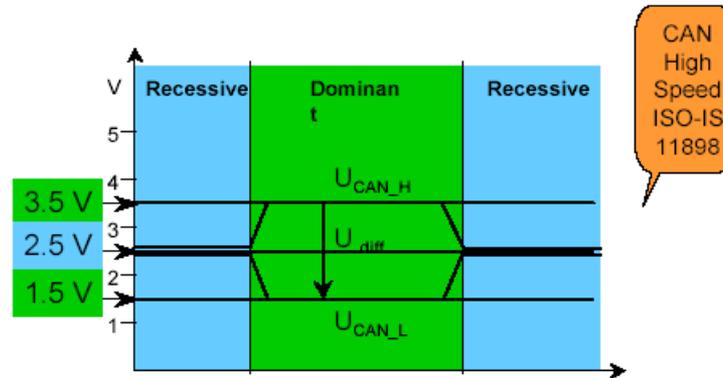


# Sporočilno orientiran protokol

- ▶ Vsako vozlišče – sprejemnik in oddajnik
- ▶ Pošiljatelj pošilja sporočilo **vsem ostalim**
- ▶ **Vsa vozlišča berejo sporočilo** in se odločajo, ali je za njih pomembno.
- ▶ **Vsa vozlišča potrdijo** sprejem brez napak (ne vemo, katero je potrdilo)



# CANbus napetostni nivoji ISO-IS 11898



## •Recesivni bit:

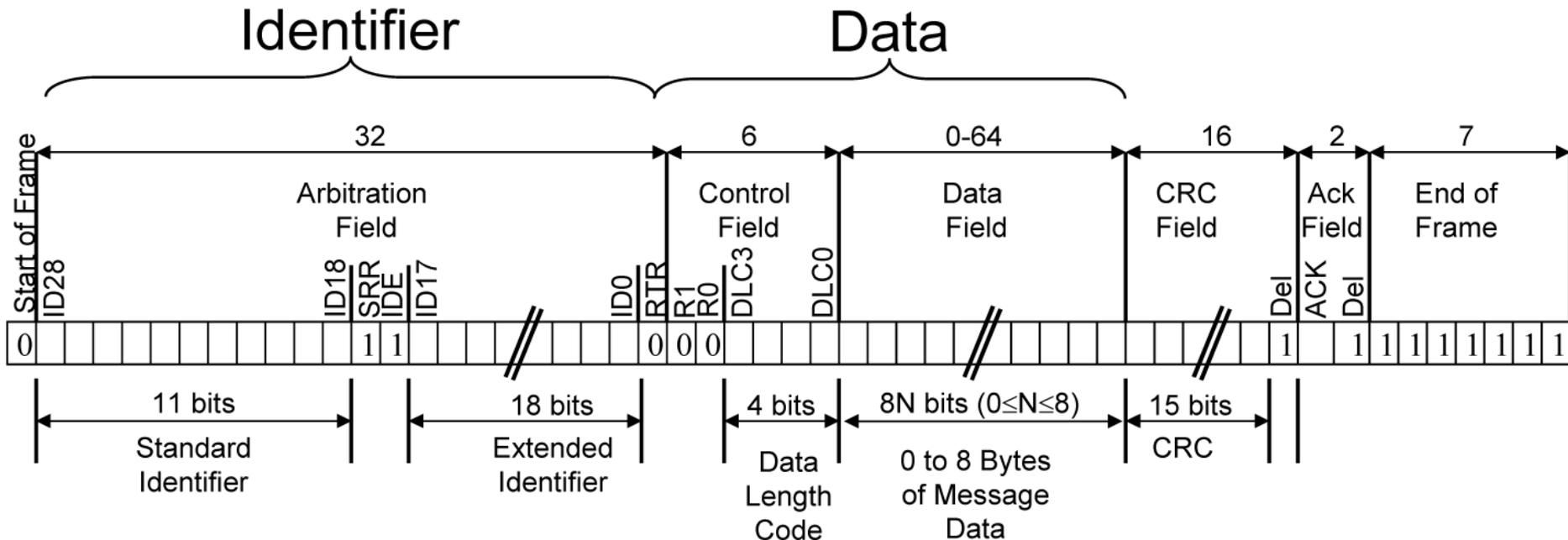
- obe liniji na približno 2.5V
- diferencialna napetost  $CAN\_H$  in  $CAN\_L \approx 0V$

## •Dominantni bit:

- $CAN\_H$  na pribl. 3.5 V in  $CAN\_L$  pribl. 1.5 V
- diferencialna napetost  $CAN\_H$  in  $CAN\_L \approx 2V$

# Format sporočila

- ▶ Vsako sporočilo ima ID, podatke in dodatke
- ▶ ID - 11 ali 29 bitov
- ▶ Data - do 8 bajtov
- ▶ Dodatki - start (SOF), CRC, ACK, end (EOF)



# Format sporočila

## CAN vs. RS-485: Why CAN Is on the Move

By Robert Gee, Executive Business Manager, Core Products Group, Maxim Integrated

Field Name	Bit Length	Description
SOF	1	Start of frame
Identifier (green)	11/29; 12/32	Represents the message priority (11 or 29 bits for standard CAN and extended CAN; 12 or 32 bits for CAN-FD)
RTR (blue)	1	Remote transmission request
IDE	1	Identifier extension bit
r0	1	Reserved bit for future protocol expansion
DLC (yellow)	4/8/9	Code for number of data bytes (4-bit for standard CAN; 8 or 9 bits for CAN-FD)
Data Field (red)	0-64 (0-8 bytes); 0-512 (0-64 bytes)	Data to be transmitted (0-8 bytes for standard CAN; 0-64 bytes for CAN-FD)
CRC	15	Cyclic redundancy check
CRC Delimiter	1	Assigned recessive (1)
ACK slot	1	Dominant bit if error-free message; recessive to discard errant message
ACK Delimiter	1	Acknowledgement delimiter
EOF	7	End of frame

Table 1. CAN Message Data-Frame Format

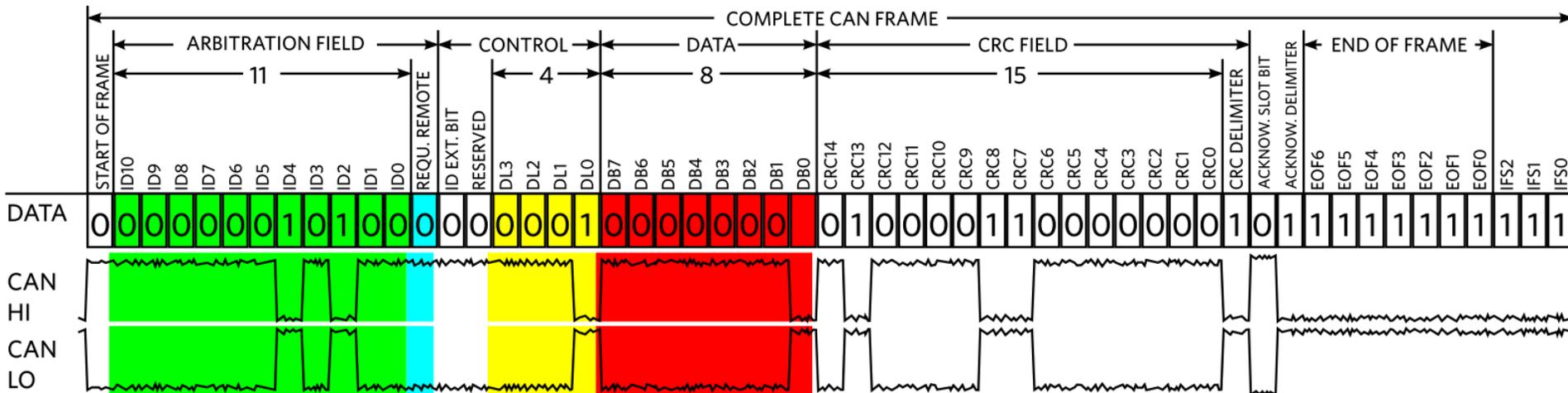
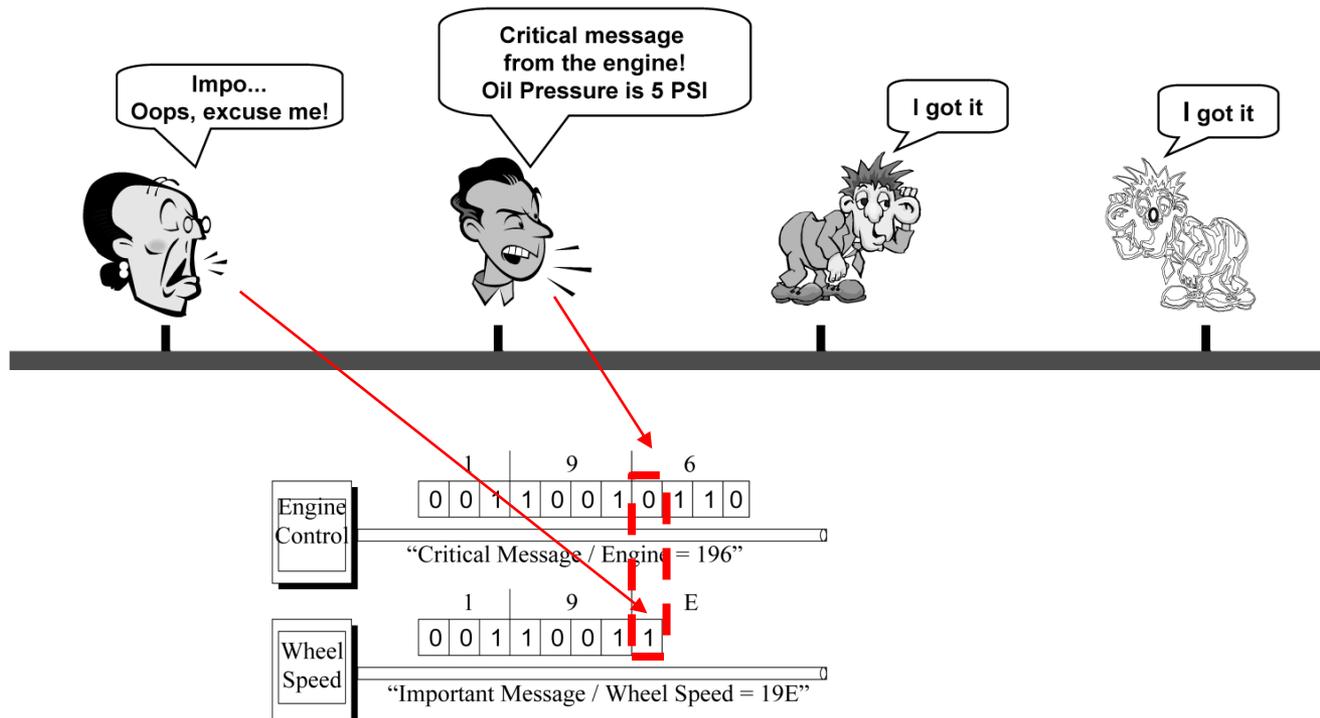


Figure 4. CAN Message Data-Frame Format

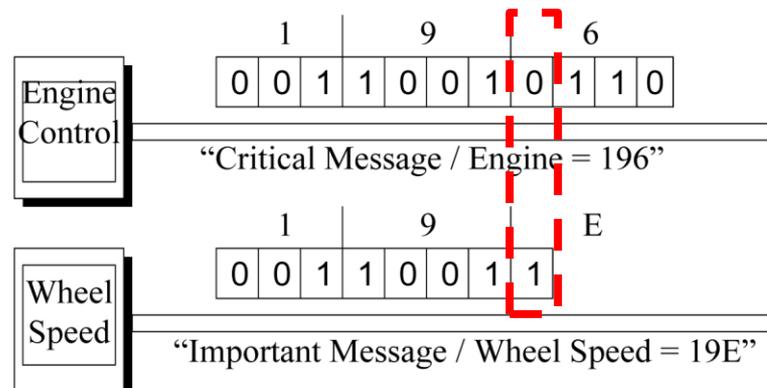
# Arbitraža (NDA- Non-Destructive Arbitration)

- ▶ Arbitraža – potrebna, ko več vozlišč poskusi **oddajati hkrati**
- ▶ Do konca sporočila lahko oddaja **samo eden naenkrat**
- ▶ Vozlišča **berejo vodilo** in **oddajajo samo v trenutkih neaktivnosti**
- ▶ Vozlišča z „višjo“ **ID prioriteto** nato nadaljujejo z oddajo.



# Arbitraža (Non-Destructive Arbitration)

- ▶ Pomembnost sporočila je določena z IDjem  
nižja vrednost = višja pomembnost
- ▶ Vodilo:
  - ▶ “0” (dominantni bit) na vodilu prevlada nad “1” (recesivni bit) na vodilu
- ▶ Če naprava:
  - ▶ odda “1” in bere “0” → izguba arbitraže
  - ▶ odda in bere enako → nadaljuje z oddajo

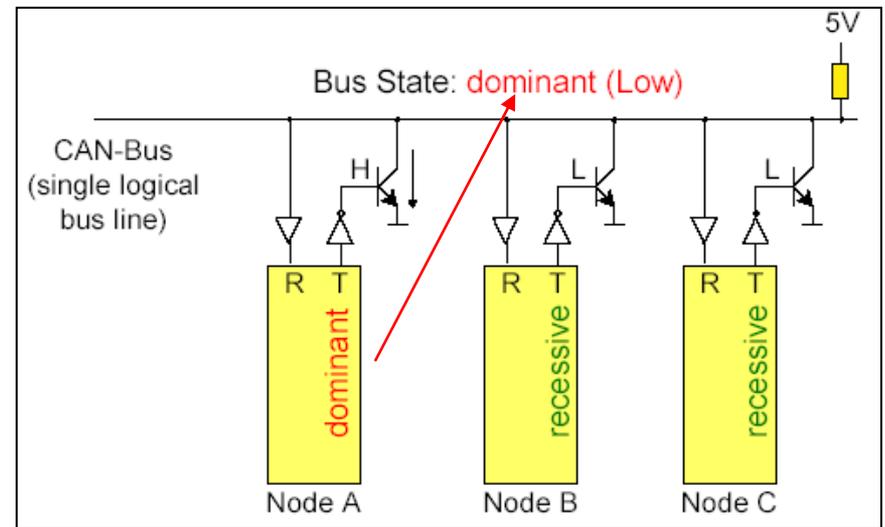
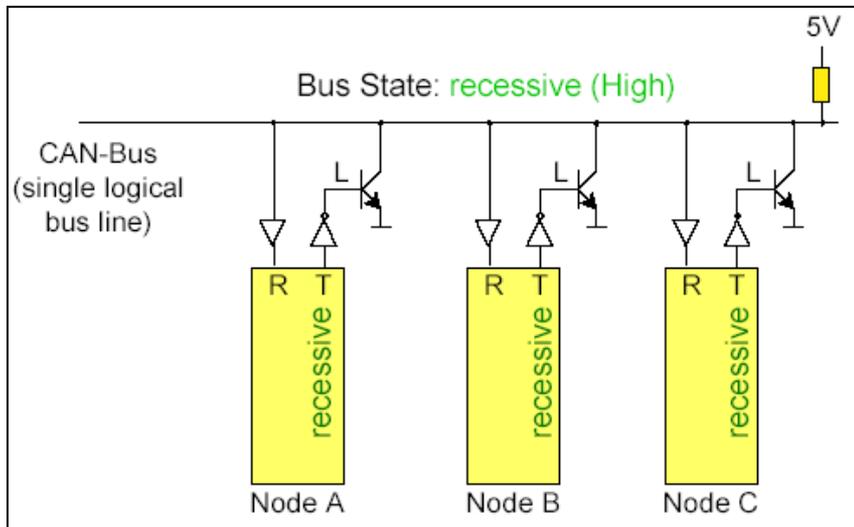


# Wired AND vezava

Stanje "0" (dominantno stanje) na vodilu

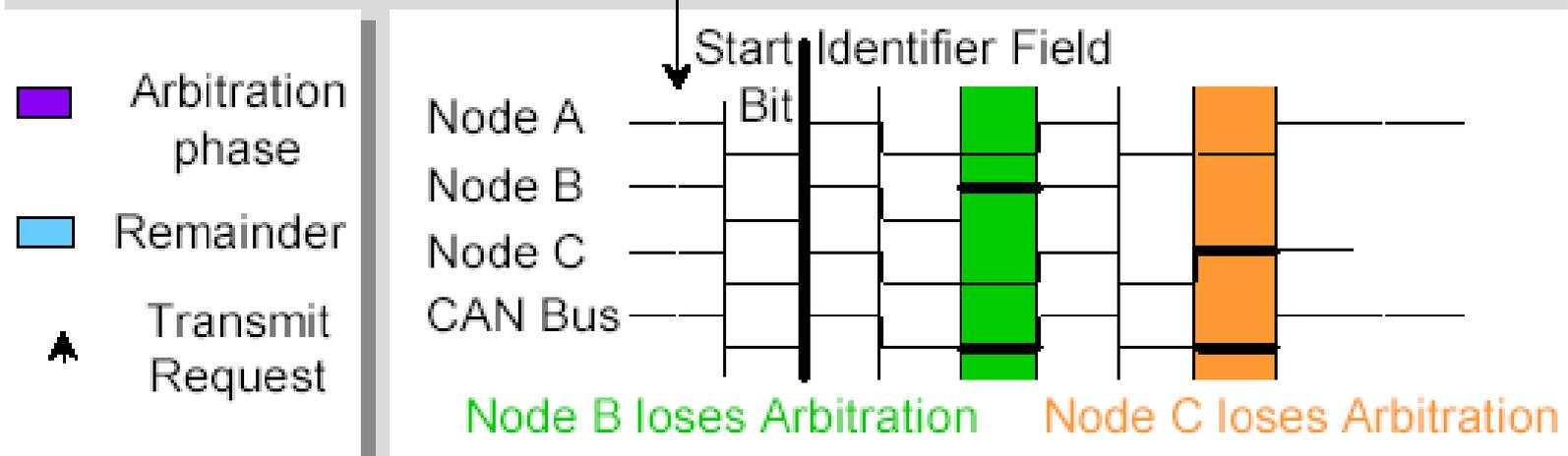
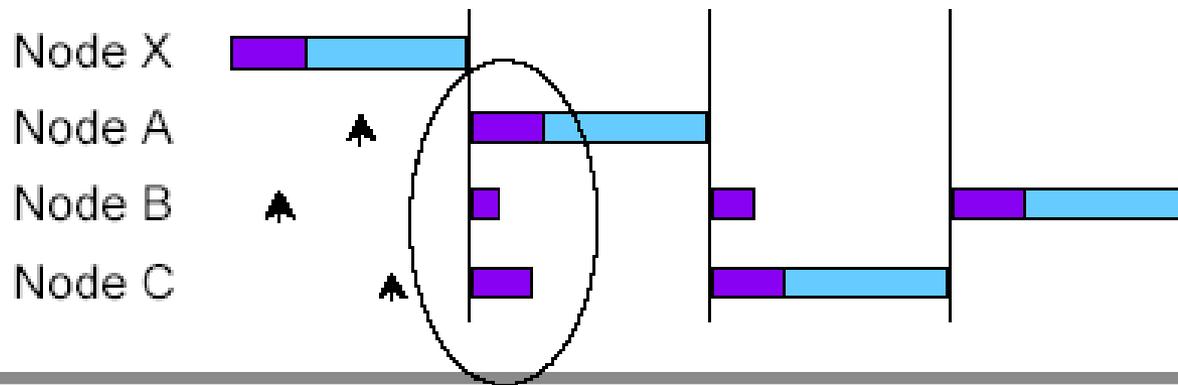
prevlada

ostala stanja "1" (recesivno stanje) na vodilu.



# Primer arbitraže – CSMA/CA NDA

## CSMA/CD NDA – **C**arrier **S**ense **M**ultiple **A**ccess/**C**ollision Avoidance by **N**on **D**estructive **A**rbitration



# Oscilloskop: primer CANbus komunikacije

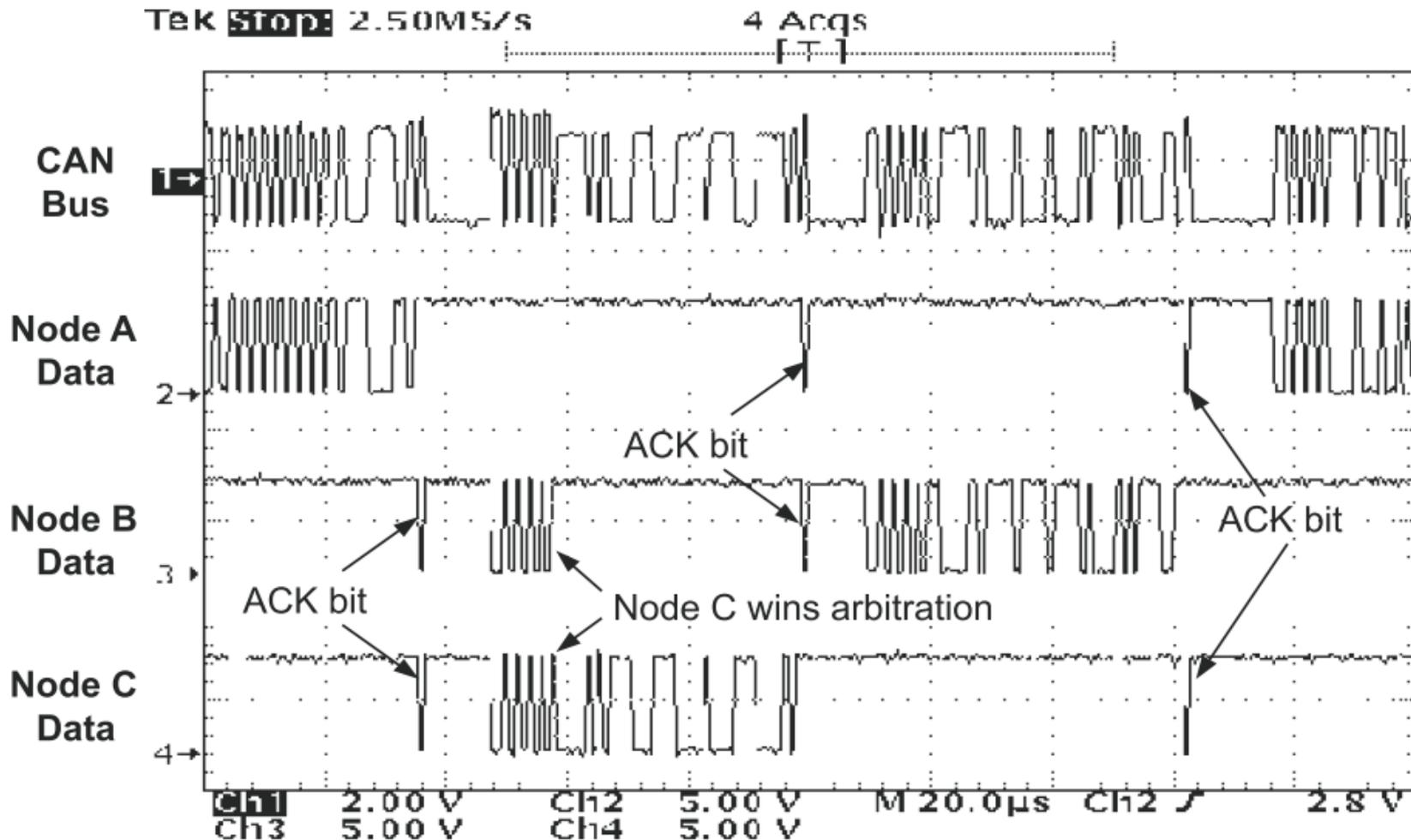
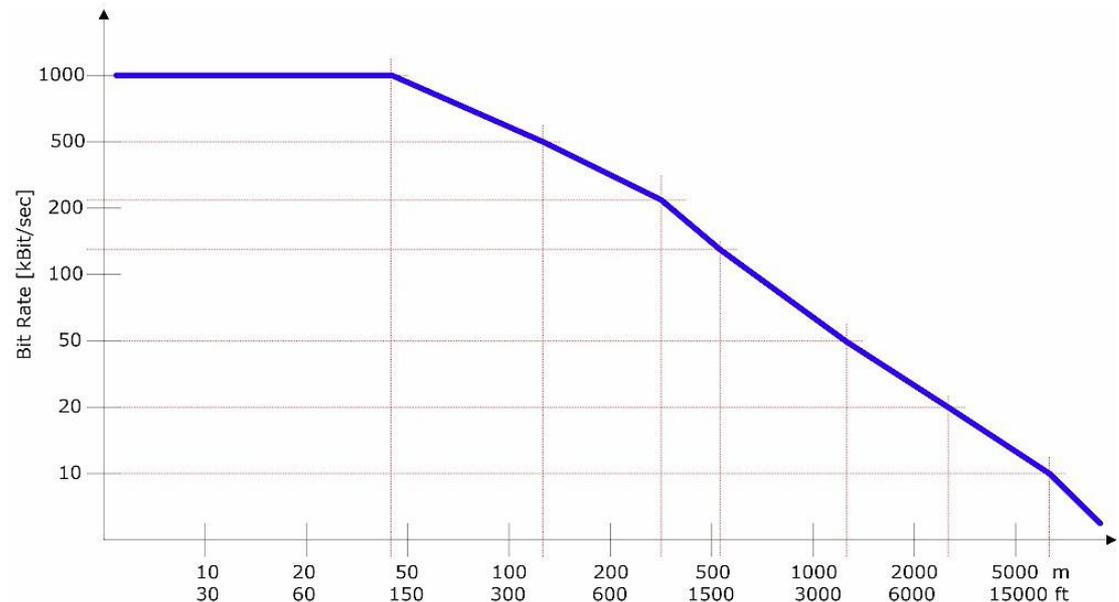


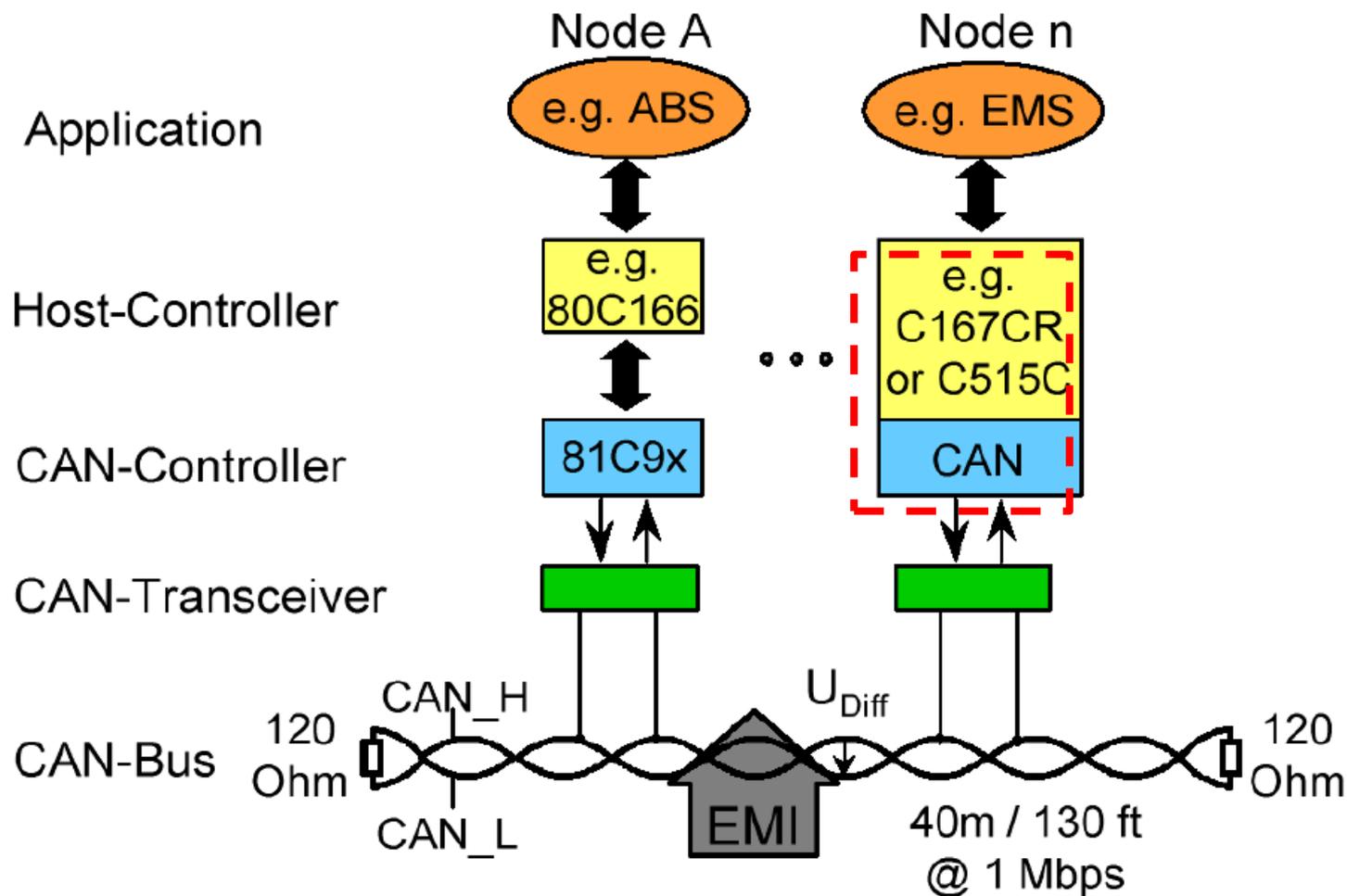
Figure 8. CAN Bus Traffic

# CANBUS - Hitrost komunikacije in razdalja

- ▶ Do 1 Mbit/sec.
- ▶ Standardne hitrosti: 1 MHz, 500 KHz and 125 KHz
- ▶ Max length: do 5000m, odvisno od:
  - ▶ hitrosti
  - ▶ lastnosti povezav
    - ▶ zaključitve, vrsta kabla, topologije, motenj, ...



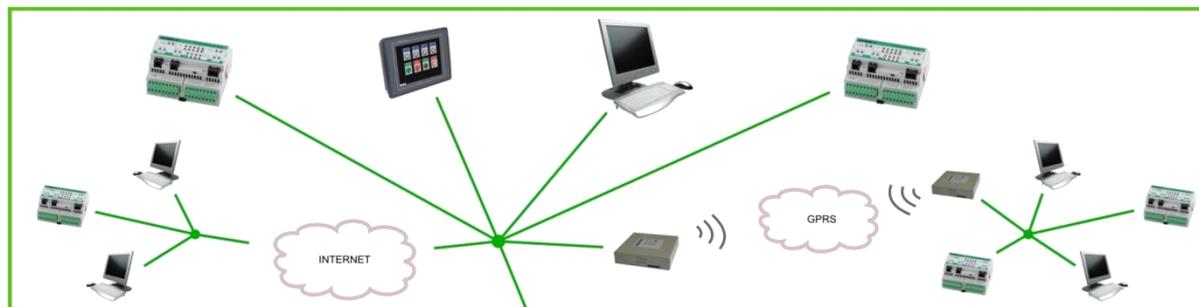
# Možnosti realizacije



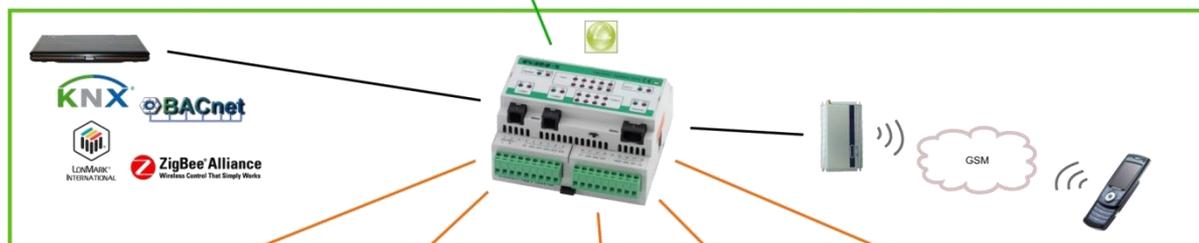
# CANbus v praksi

## INTEGRA BM SYSTEM Industrial & Building Automation

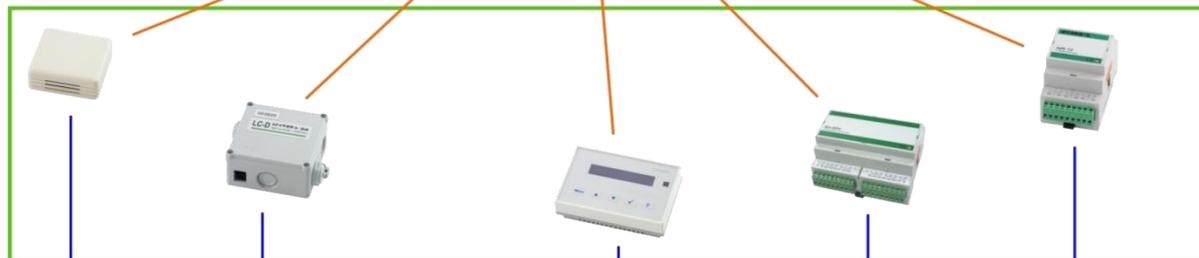
High level network  
(Ethernet, A-Bus,  
Modbus)



CyBro controller



Low level network  
(Canbus)



Dodatki (tipala, daljinci,  
...)



## INTEGRA BM SYSTEM Bus length

## Dolžina, hitrost in topologije

Regarding bus length, two points must be considered:

### 1. Voltage drop

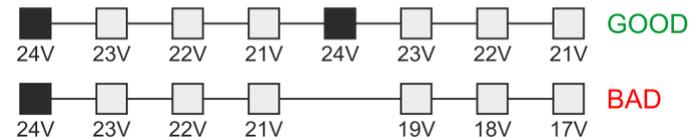
Wire resistance cause voltage drop, which depends of cable length, wire diameter and power consumption. **Cable must be selected** to ensure each module have at least the minimum specified voltage.

### 2. Signal delay

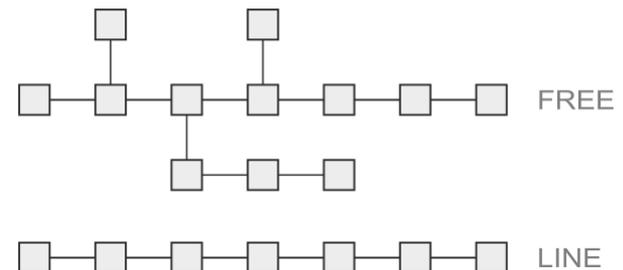
Communication speed is limited with propagation time and bus topology. With **default 100kbps baudrate, 100m is safe without restrictions**. For a longer distance, cable must be connected in **a line (without trunks) and properly terminated**.

Speed\Topology	FREE	LINE
100kbps	100m	300m
50kbps	200m	500m
20kbps	500m	1000m

Secondary power supply



Network topology



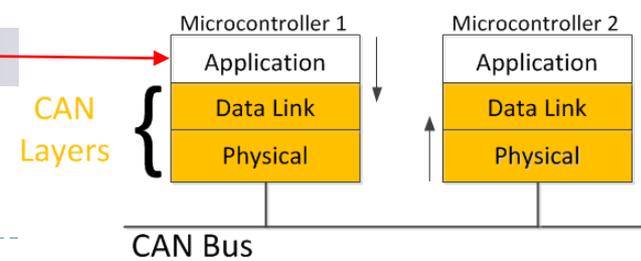
## INTEGRA BM SYSTEM

### IEX protocol

#### (nadgradnja CANBUS)

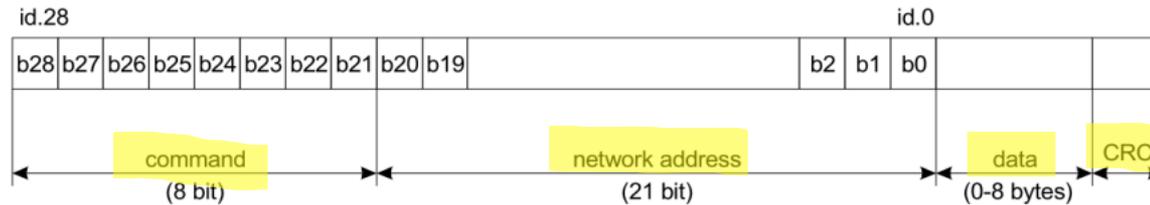
#### IEX PROTOCOL v2.8

#### POVZETEK

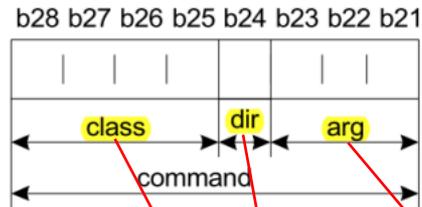


### General

IEX-2 is based on CAN 2.0B. Message format is defined as follows:



### Command summary



NAD – unikatni naslov IEX modula

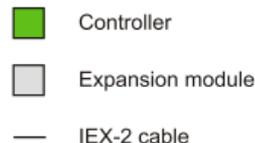
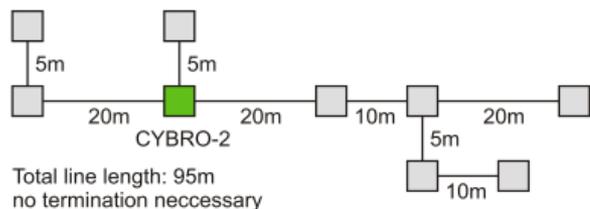
command	class	dir	command	arg	data bytes	description	PCAN view
	0000						
	0001						
	0010						
IX_DATA	0011	1		xxx	data(1..4)	binary inputs	070-07Exxxxxh
QX_DATA		0		xxx	data(1..4)	binary outputs	060-06Exxxxxh
	0100						
	0101						
	0110						
IW_DATA	0111	1		xxx	data(2..8)	analog inputs	0F0-0FExxxxxh
QW_DATA		0		xxx	data(2..8)	analog outputs	0E0-0EExxxxxh
BAUDSYNC	1111	1		111	-	autobaud sync msg	1FFFFFFFh

# INTEGRA BM SYSTEM

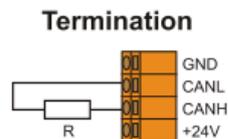
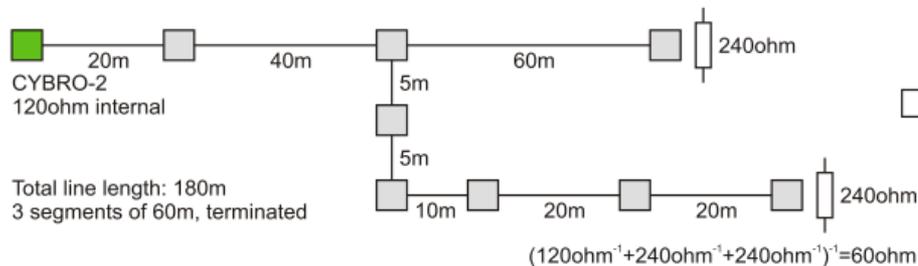
## Cabling topology & Termination

## Primer priporočil: topologija in zaključevanje

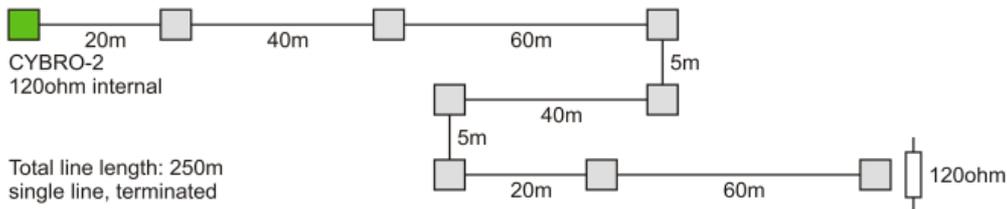
### 1) Total IEX-2 bus length <100m



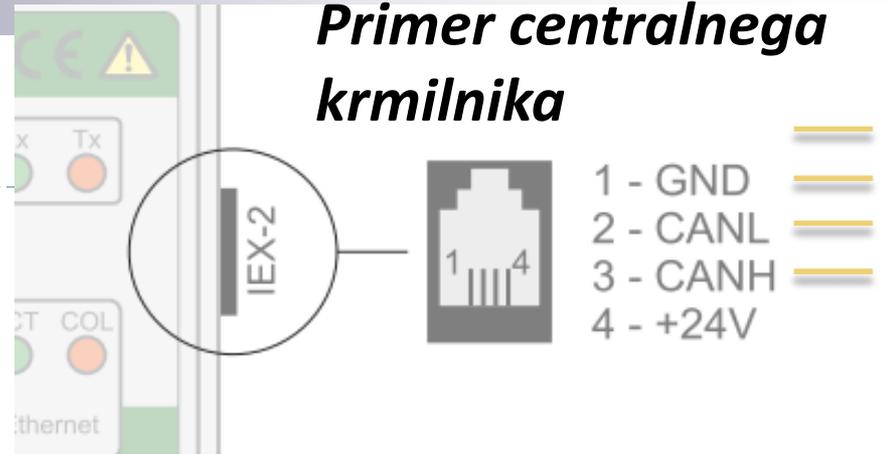
### 2) 100m < Total IEX-2 bus length <200m



### 3) 200m < Total IEX-2 bus length <300m



# Centralni krmilnik Cybro-2



**Primer centralnega  
krmilnika**

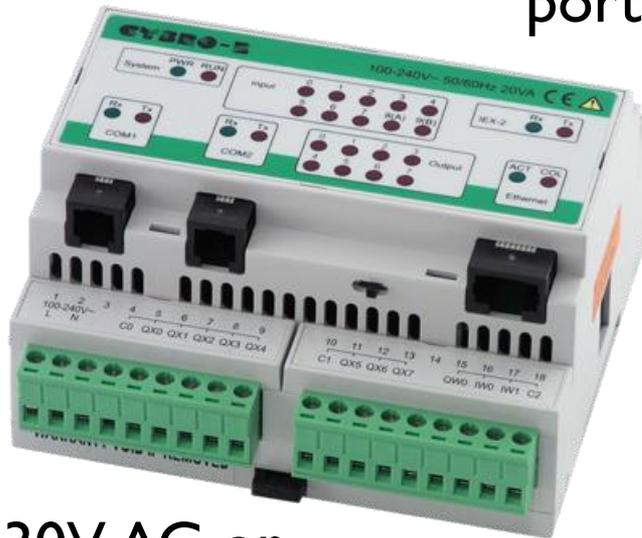
- 1 - GND
- 2 - CANL
- 3 - CANH
- 4 - +24V

Ethernet  
port

2 x RS-232  
port

CAN  
interface

Core  
Modular  
Block



Digital and analog  
I/O  
Communication  
and status LED  
signalization

230V AC or  
24V DC

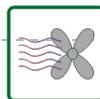
Removable  
connectors

Retentive and  
permanent EEPROM  
memory

# Primeri modulov

## IEX MODULE

FC



### FC

fan coil module

#### SPECIFICATIONS:

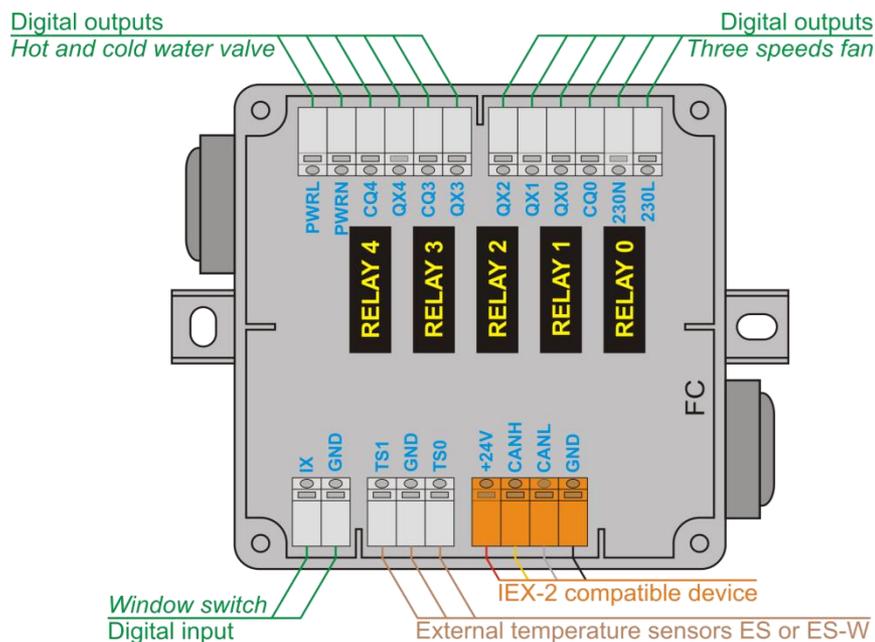
- 1 x digital input
- 5 x relay output
- 2 x input temperature measurement
- 24V DC power supply consumption: 110mA

#### MECHANIC:

field mountable

#### TYPE:

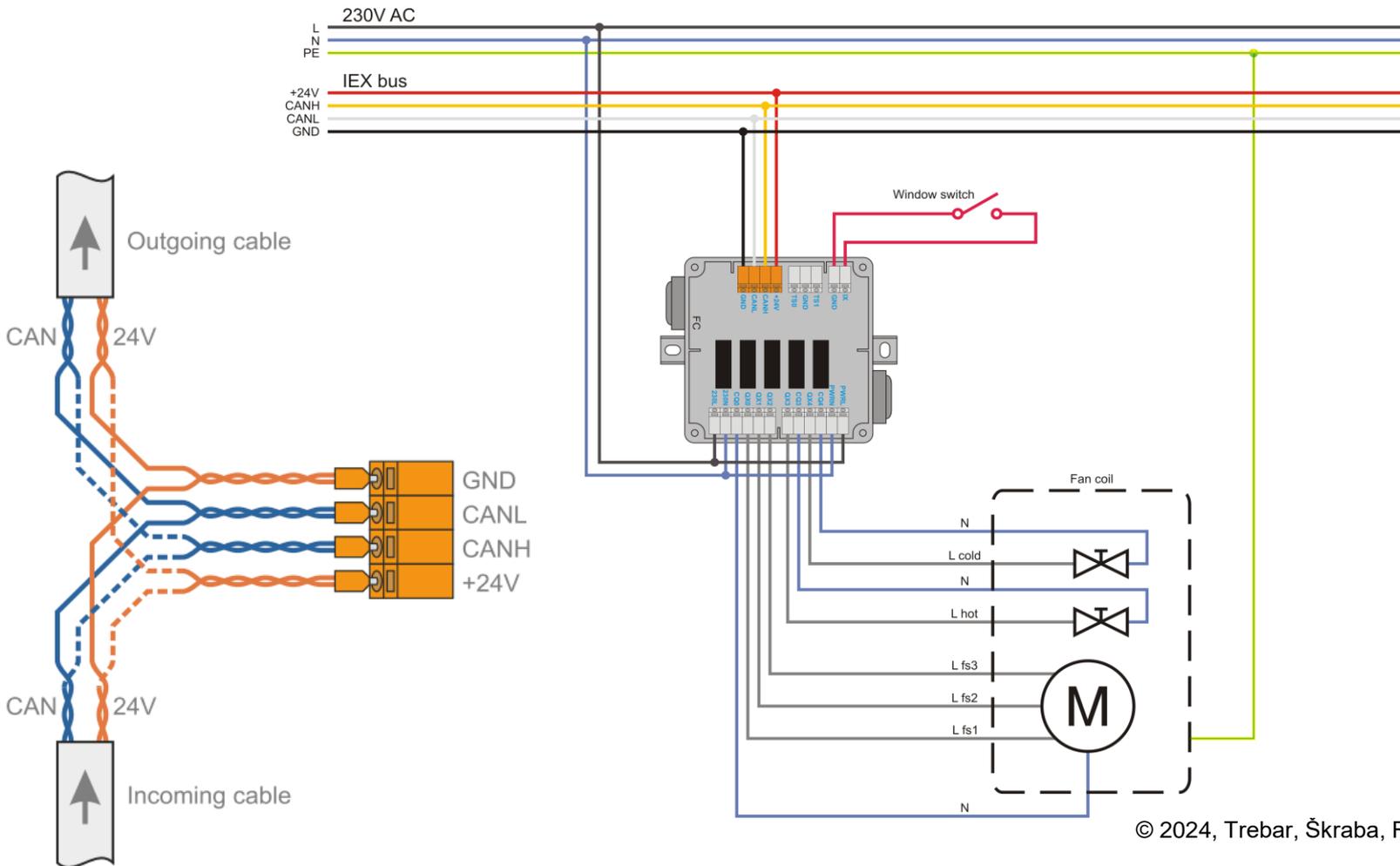
FC-FB



# Primeri modulov

## IEX MODULE FC

FC

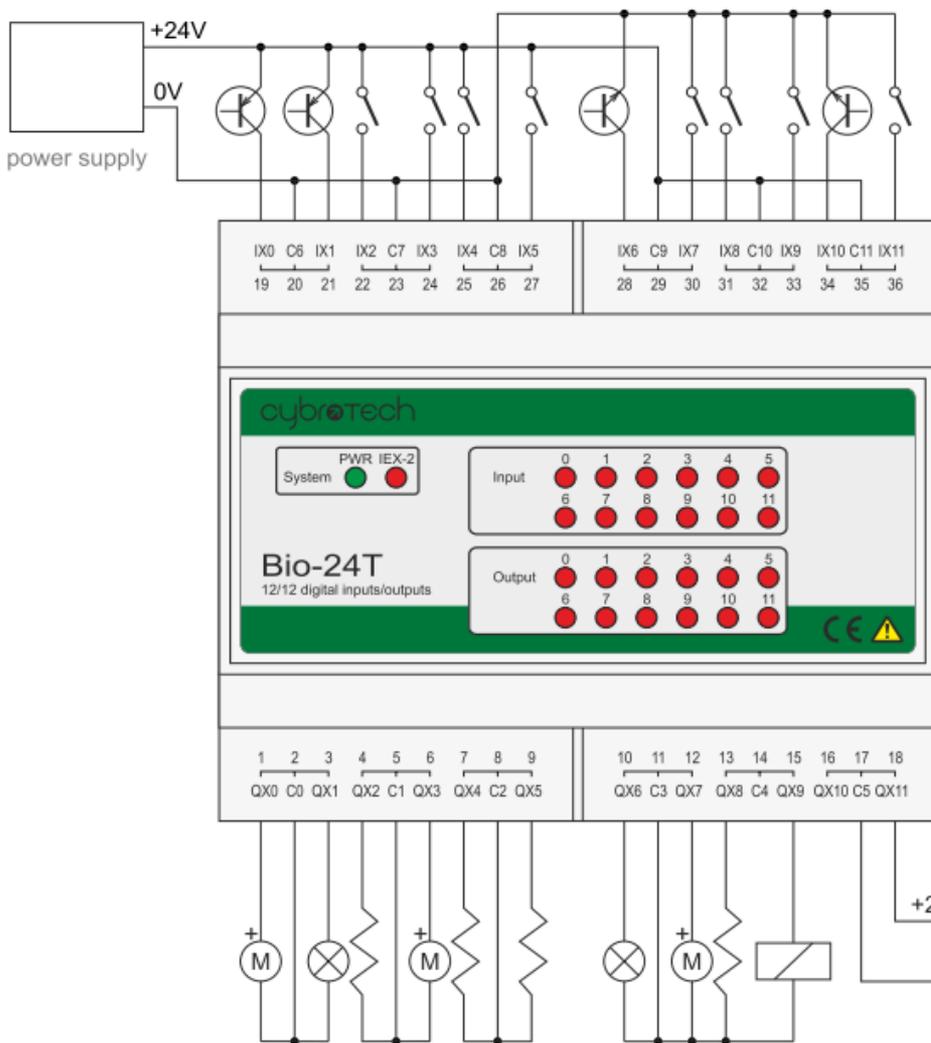


# Primeri modulov

## IEX MODULE Bio-

## Bio-24T

Wiring diagram



IEX-2 module  
12 opto-isolated PNP transistor  
outputs 1A  
12 opto-coupler inputs 24V

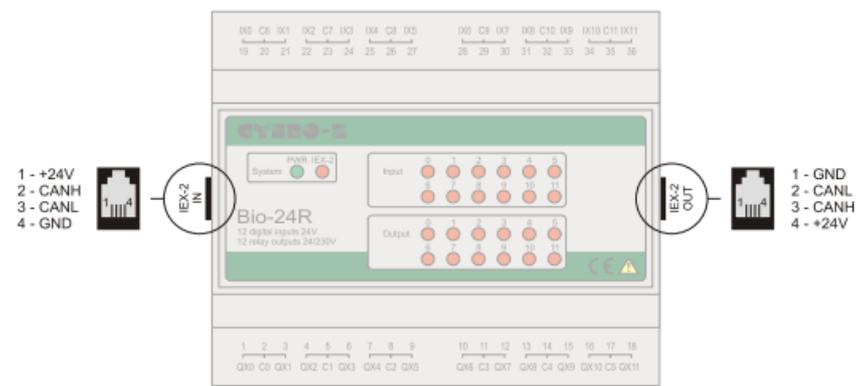
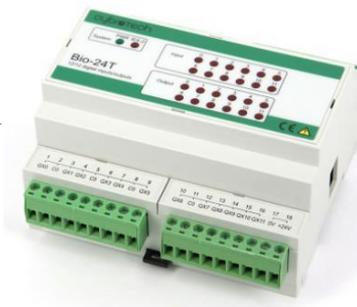
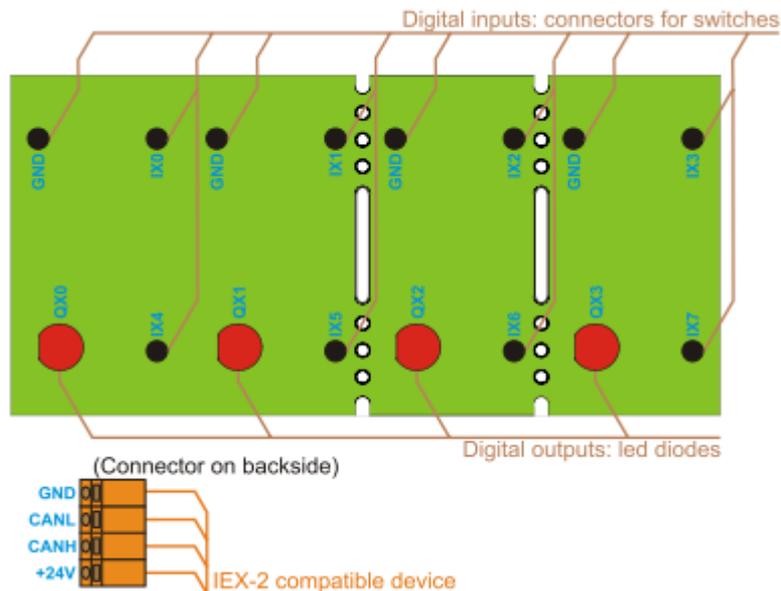


Figure 3: IEX-2 input and output ports.

# Primeri modulov

## IEX MODULE SW-

## SW-L

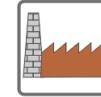


IEX-2 module  
4 switches  
4 LED illuminations  
Designed for Legrand, Bticino and TEM switches



### Technical specifications

IX (8 digital inputs)	for connecting 4 switches
Current	2.5mA/12V
QX (4 digital outputs)	
Led illumination	3mm red led-diodes
Power supply	24V DC (18..26V DC), over IEX-2 bus
Power consumption	40mA
Mounting	2 x switch: flush box (diameter 60mm, depth 55mm), in wall 3 x switch: flush box (size 95x58mm, depth 49mm), in wall 4 x switch: flush box (size 120x58mm, depth 49mm), in wall
Dimensions	89x44x38mm



CyPro v2.7.6 - C:\Users\R\Documents\Sluzba\Vaje\VIN\_Vh\_Izh\_naprave\VIN\_2016\_17\Vaje\13 Labvaja LV5\_Canbus\VIN\_vaje.cyp

File Edit View Project Program Tools Window Help

New Open Save Print Cut Copy Paste Environment Configuration Hardware Allocation Masks Sockets Send Monitor Start Stop

Project Tree

- Project: VIN\_vaje.cyp
  - Program: New Program
    - Hardware
    - Masks
    - Sockets
    - ST: function main:void;
    - Description

Local Allocation

Name	Type	Attributes	Description
main			

```
if fp(clock_10s) then
  bio00_qx00 := !bio00_qx00 ;
end_if ;

bio00_qx01 := !bio00_qx00 ;

if fp(bio00_ix00) then
  bio00_qx02 := !bio00_qx02 ;
end_if ;

if fp(sw00_ix01) then
  bio00_qx00 := !bio00_qx00 ;
end_if ;

if fp(clock_10ms) then
  bio00_qx02 := !bio00_qx02 ;
end_if ;
```

Online Monitor

Monitor01

History	Variable name	Type	Value	Base
	clock_10s	bit		0 Dec
	bio00_ix00	bit		0 Dec
	bio00_ix01	bit		0 Dec
	bio00_qx00	bit		0 Dec
	bio00_qx01	bit		0 Dec
	bio00_qx02	bit		0 Dec
	sw00_ix01	bit		0 Dec

Speed: 50ms (16s total)

Close



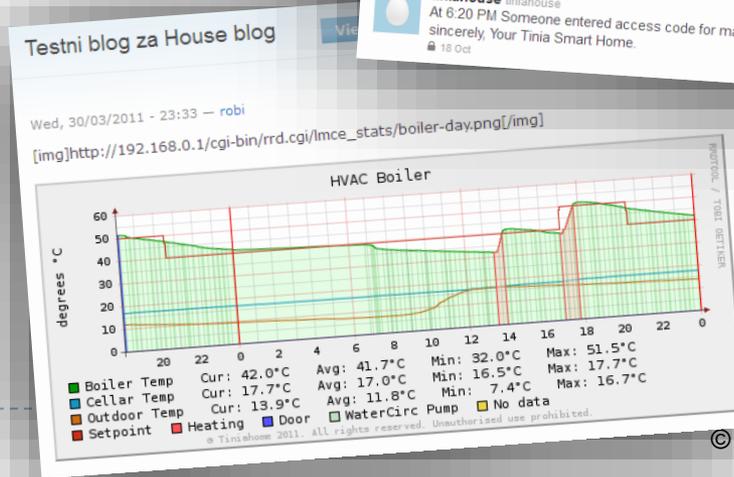
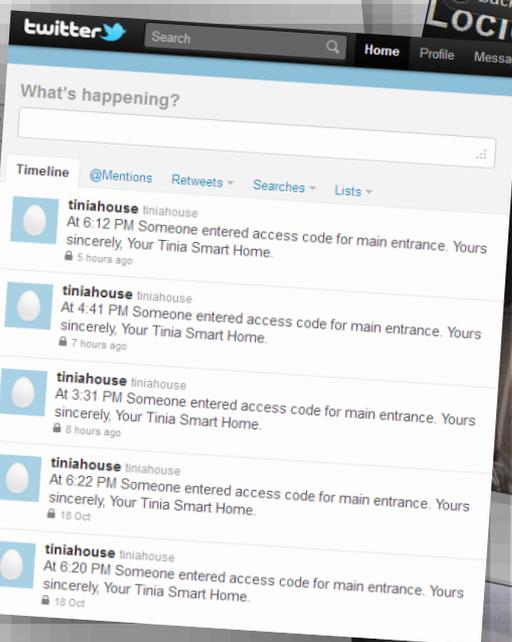
## Tinia – prijazen dom

### Kratek opis

#### TBS – „Tinìa Building Server“:

*Nadzor, upravljanje in vizualizacija delovanja prijaznega doma.*

- majhen, varčen, tih (5W)
- povezuje zgradbo in pametno mesto
- informiranje, povratna inf.
  - pametni telefoni, tablice
  - splet, soc.omrežja
- programiranje s pravili, vtičniki
- povezava s soc.omrežji
  - Twitter, FaceBook

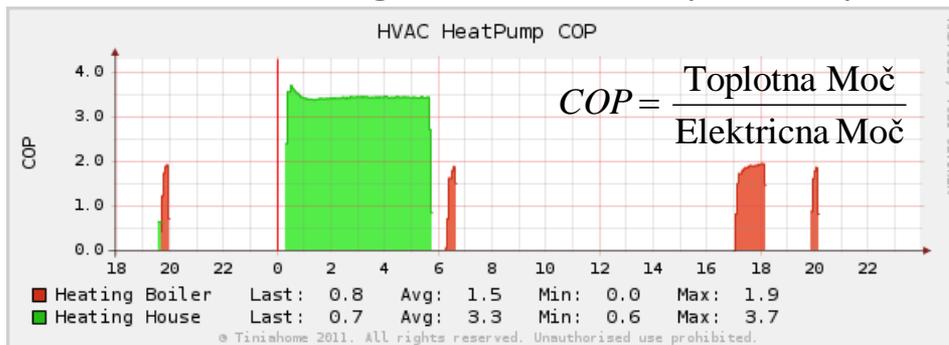


# Ogrevanje, prezračevanje



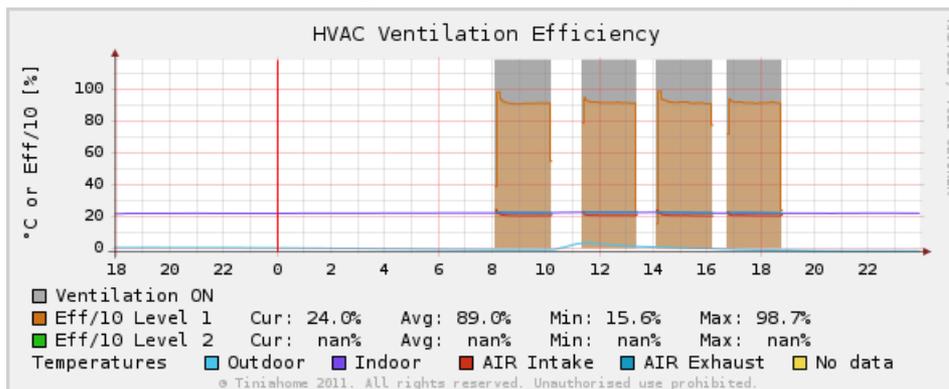
Primer zimskega dneva – COP toplotne črpalke :

## Toplotna črpalka zemlja-voda



Primer zimskega dneva – učinkovitost rekuperacije

## Prezračevanje s povratkom toplote (rekuperacija)



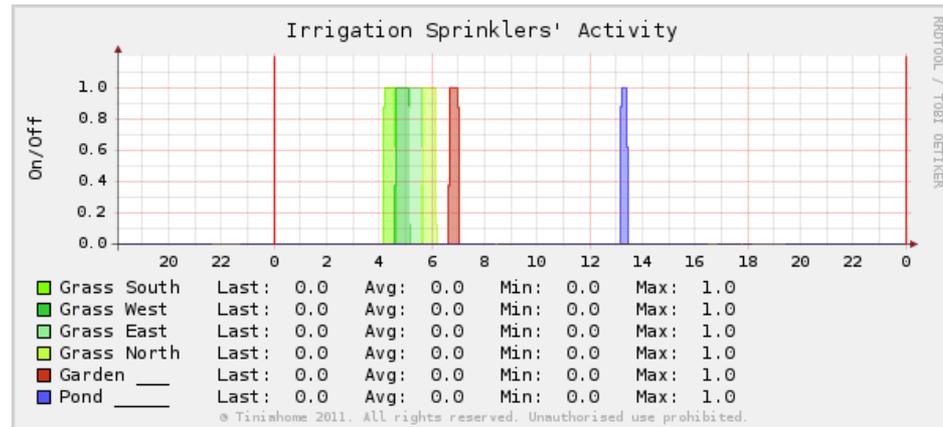
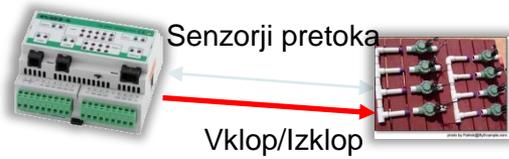
$$EFF \approx \frac{\text{Svež zrak temp.} - \text{Zunanji zrak temp.}}{\text{Izčrpažrak temp.} - \text{Zunanji zrak temp.}} [\%]$$

# Zalivanje (vrt, zelenica, ribnik)



Primer poletnega dne - Zalivanje

Zalivanje  
(vrt, zelenica, ribnik)



# Tipala

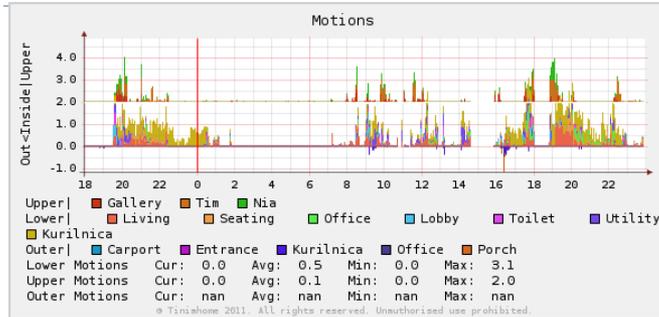
## Zaznave gibanja v hiši - primer



### Detekcija gibanja/prisotnosti

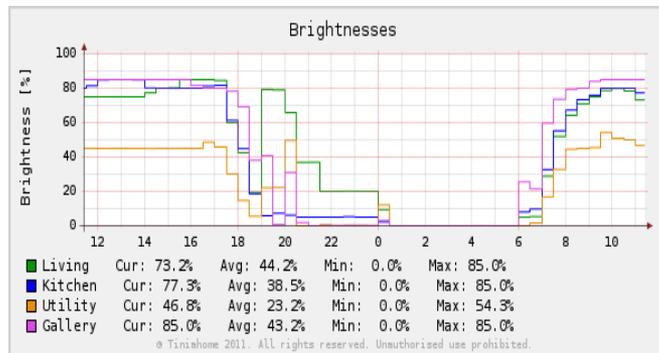


Motion Detector

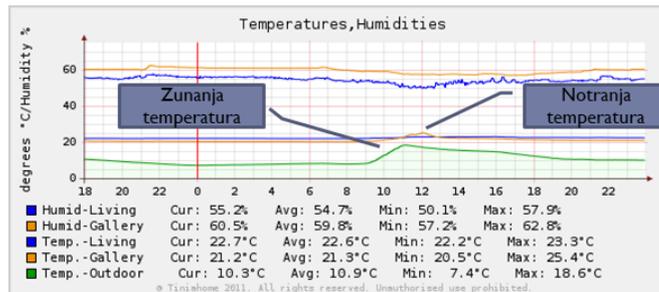


### Osvetljenost prostorov

### Meritve zveznih vrednosti



### Temperatura, vlaga

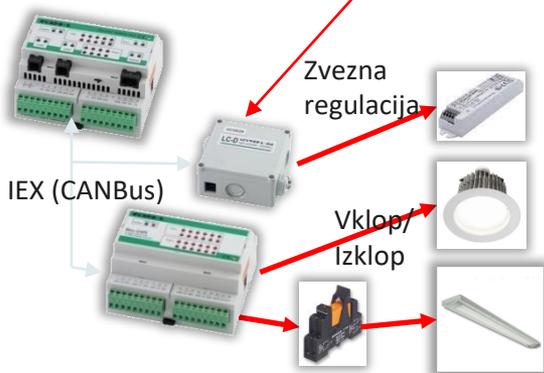




# Praktični primer nadzora in vizualizacije: Osvetlitev (aktuatorji)

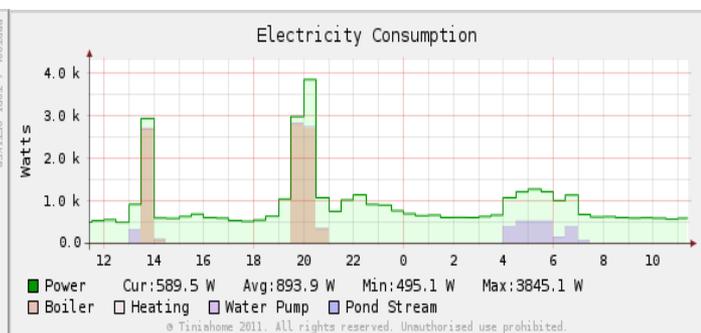
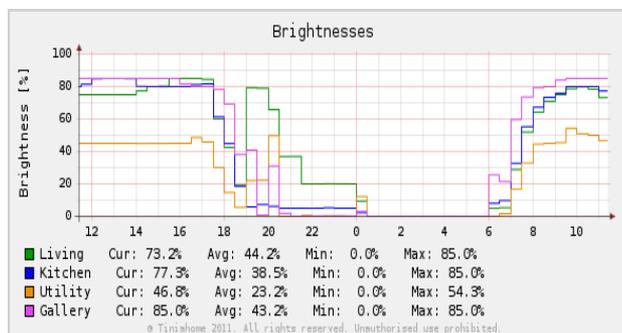
## Vklop/Izklop in zvezna regulacija razsvetljave

- Vklop/Izklop BIO-24R in BIO-24T
- Zvezna regulacija s pomočjo DALI balastov - Cybrotech modul LC-D



- Luči se upravljajo v skupinah
- Običajno krmiljena s pomočjo scen in zaznavanje osvetljenosti:
  - Statične scene** – npr. : Prehrana, Obisk, Romantika, TV, Branje, Relaksacija, ...
  - Dogodkovne scene**: Ko se vklopi TV, nastavi bližnjo luč na 20%.
- Zmanjševanje porabe :
  - Časovne luči** (izklopi po določenem času odsotnosti)
  - Vklopi luč samo, ko je to **res potrebno** (trenutna osvetljenost)
  - Nastavi zvezne luči samo na **potrebno stopnjo** (glede na osvetljenost)

## Primer meritev osvetljenosti in nadzora porabe el. energije (glavni porabnik el. energije so posebej izpostavljeni)



# Pasivno ogrevanje/hlajenje...



Rolete, žaluzije, Okna

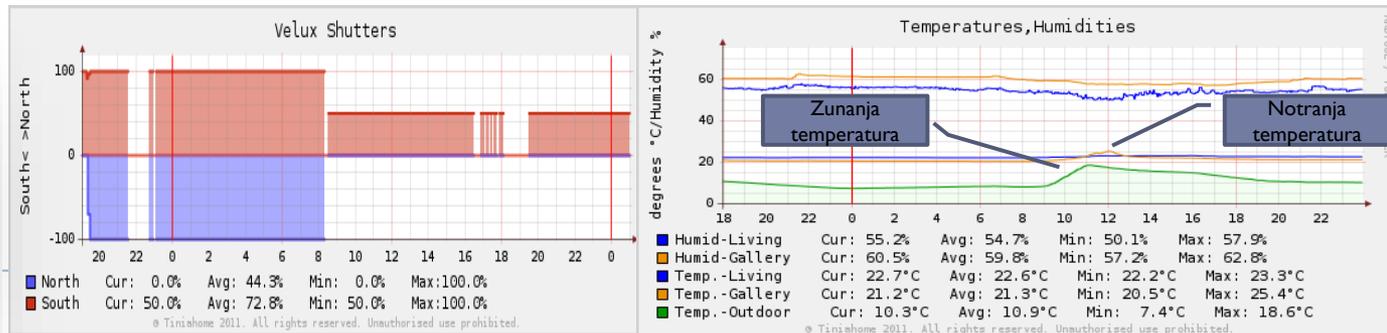
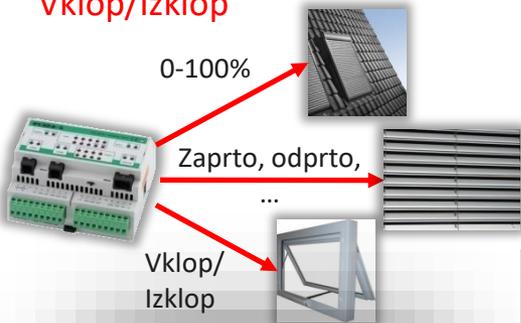
•Rolete: med 0% - 100%  
(0% odprte, 100% zaprte)

•Žaluzije imajo stanja :  
Zaprto(100%), Senčeno(75%),  
Odprto(50%), Solarno pasivno  
(25%), Dvignjeno(0%).

•Motorizirana okna:  
Vklop/Izklop

- Strešna okna z roletami :
  - severna, običajno:
    - Odprta v toplem vremenu za boljšo osvetlitev (poletje)
    - Zaprta v hladnem vremenu za ohranjanje toplote (zima)
  - južna, običajno:
    - Odprta v hladnem, sončnem vremenu za pasivno ogrevanje (zima, pomlad)
    - Zaprta v vročem vremenu proti pregrevanju (poletje)
- Žaluzije:
  - Senčene ali zaprte ob izrazitem sončnem vremenu poleti
  - Odprte v "solarni" poziciji ob sončnih dnevih pozimi
- Motorizirana okna (s komarniki) :
  - Odprta v poletnih nočeh za pasivno ohlajanje

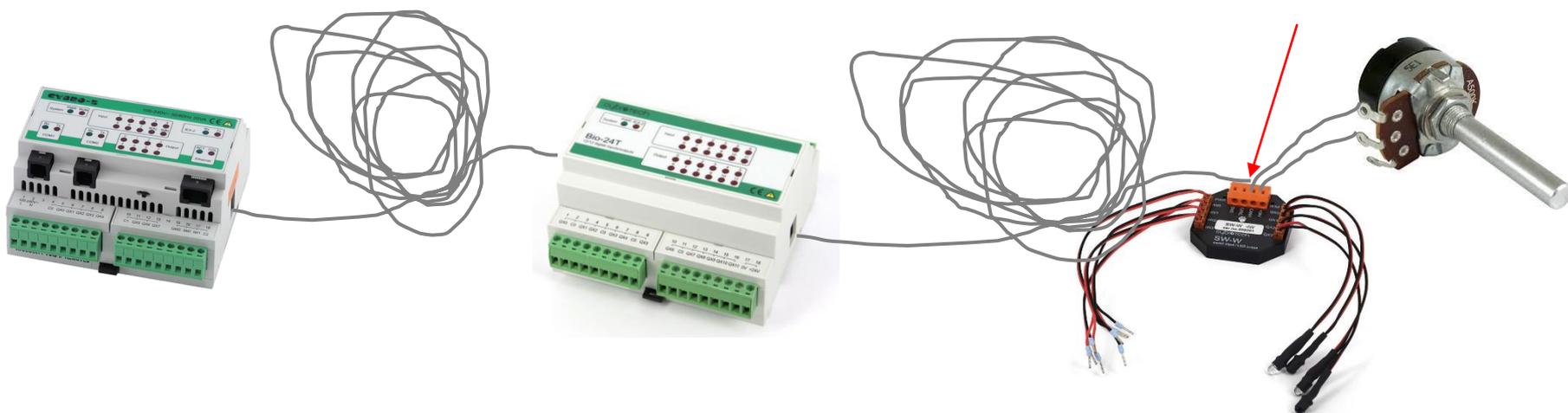
Primer stanj rolet in temperatur v sončnem zimskem dnevu:



# Lab. vaja – meritve CANBUS

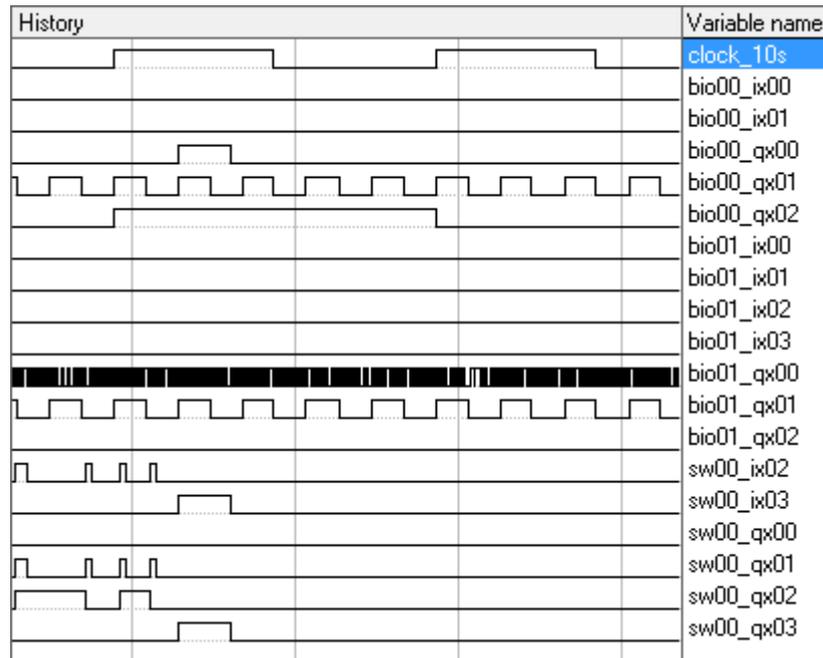
Izmerite stanje na vodilu pri :

- Različnih zaključitvah na koncu vodila
  - Odprte sponke, 500ohm, zaključitev (107ohm)
- Dveh različnih bitnih hitrostih (500kb/s, 100kb/s)



# Lab. vaja – kontrolni program

## Monitor



## Program

```
// Periodic tasks
if fp(clock_10s) then
    bio00_qx02 := !bio00_qx02 ; // Red LED every 10 secs
end_if ;

if fp(clock_1s) then
    bio00_qx01 := !bio00_qx01 ; // Red LED every 1 sec
    bio01_qx01 := !bio01_qx01 ; // Red LED every 1 sec
end_if ;

if fp(clock_10ms) then
    bio01_qx00 := !bio01_qx00 ; // Red LED every 10 msec
end_if ;

if fp(bio00_ix00) then
    bio00_qx02 := !bio00_qx02 ; // Red LED on keypress
end_if ;

// SW Switch -> LED indicator & ventilator
sw00_qx03 := sw00_ix03;
bio00_qx00 := sw00_ix03;

sw00_qx01 := sw00_ix02; // SW Key -> LED indicator

if fp(sw00_ix02) then
    sw00_qx02 := !sw00_qx02 ; // SW Key -> change LED indicator
end_if ;
```

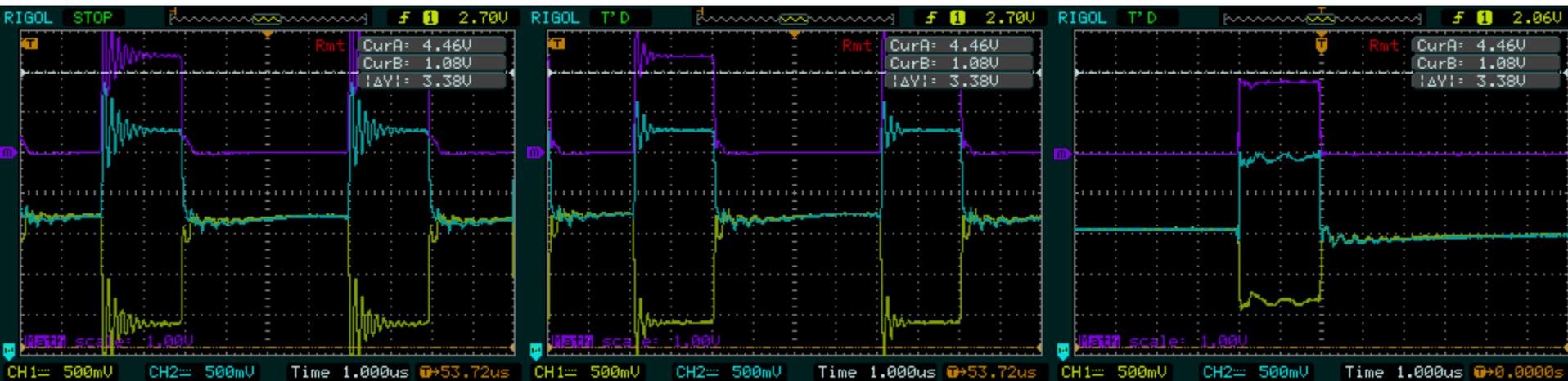
# Lab. vaja – meritve

500kb/s:

Odprte sponke

500ohm

107ohm



3 zavitki UTP kabla s spojnikami – cca 40m...



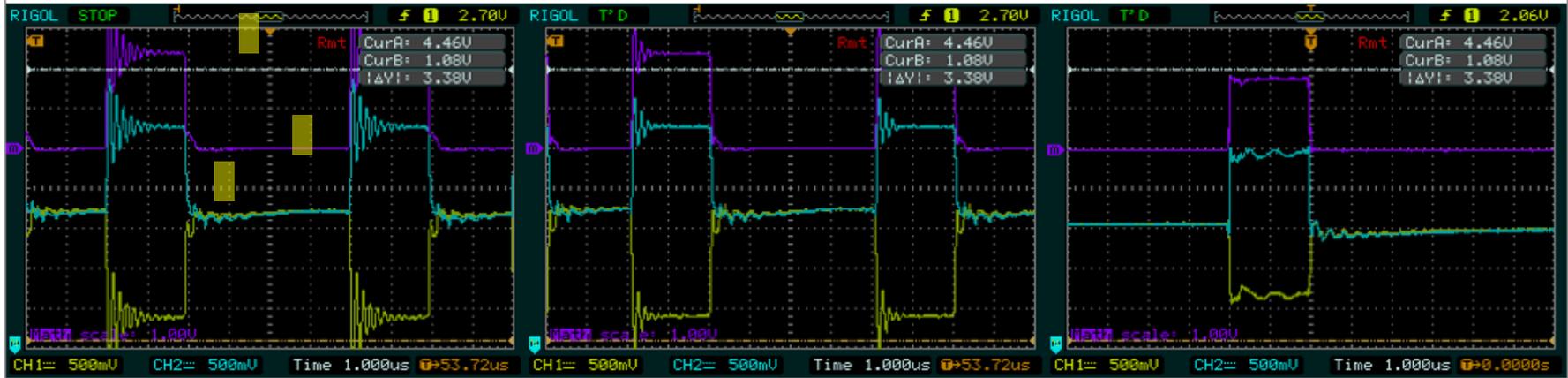
# Lab. vaja – primerjava hitrosti

500kb/s:

Odprte sponke

500ohm

107ohm

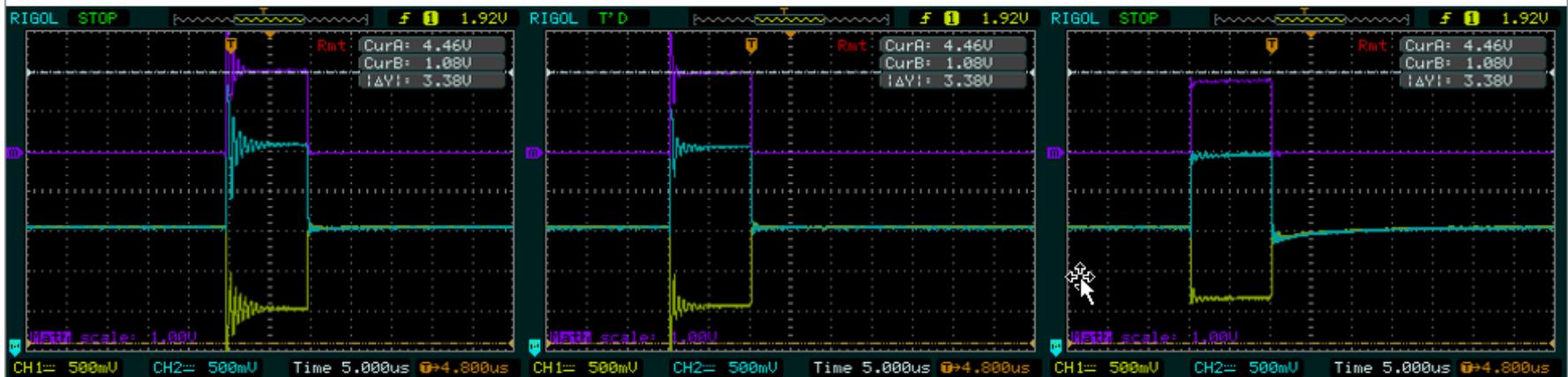


100kb/s:

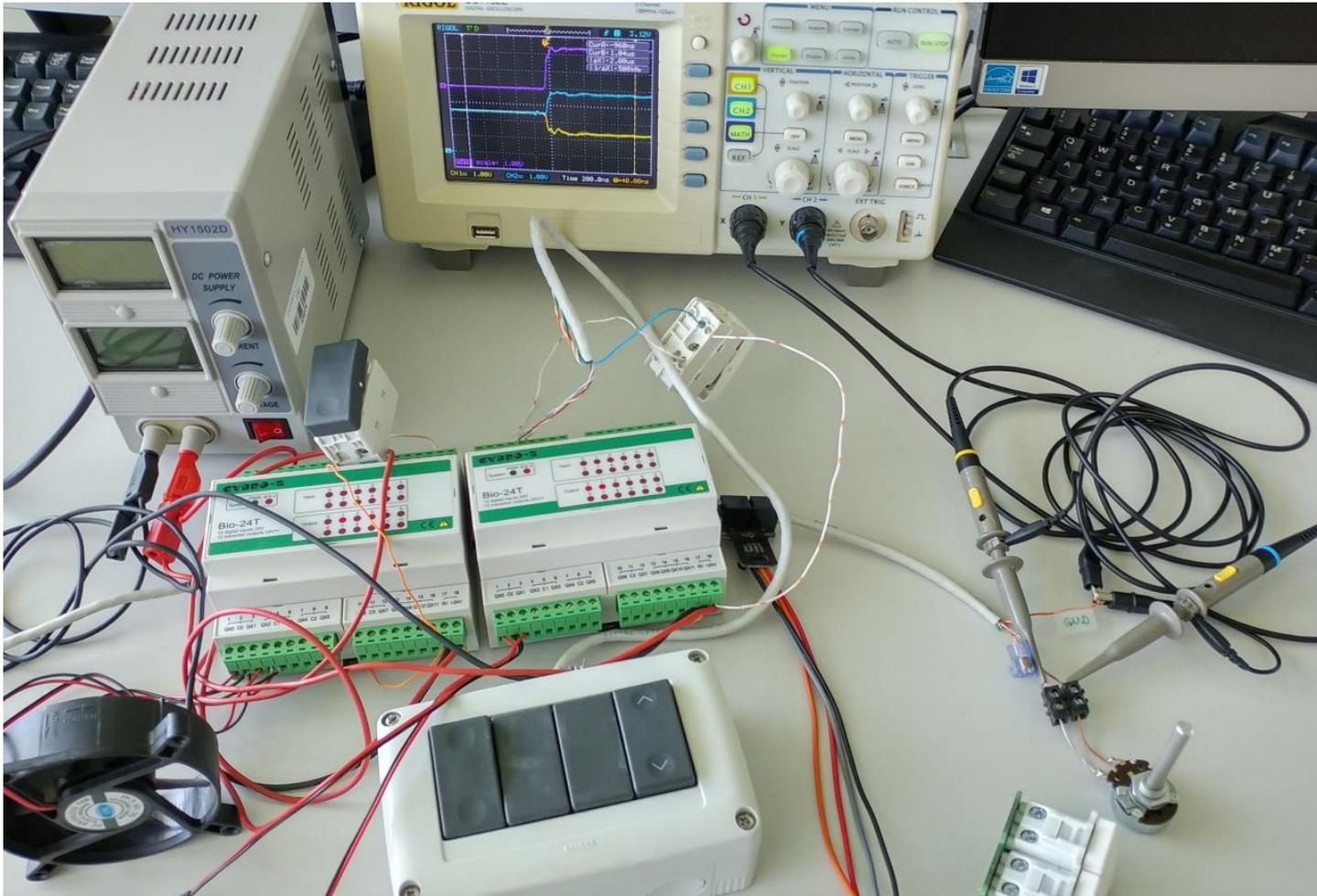
Odprte sponke

500ohm

107ohm

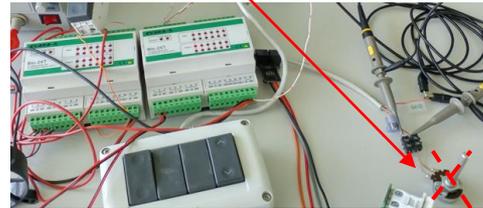


# Lab. vaja - meritve

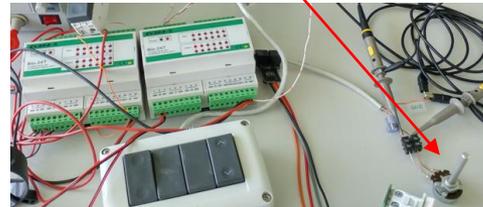


# Lab. vaja - meritve

Nezaključena linija

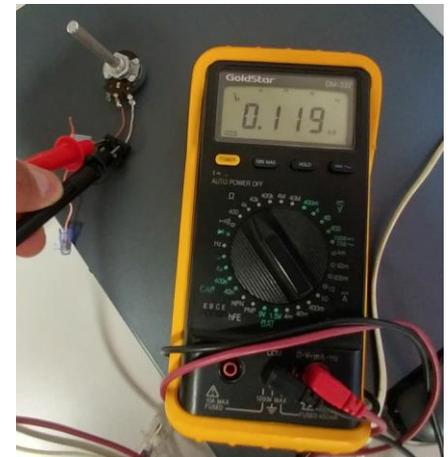


Zaključena linija



Ročna meritev

$R_0$



# Lab. vaja – STM32 kot IEX modul

## STM32F4 Discovery + shield MikroE



New Program - ST: function main: void;

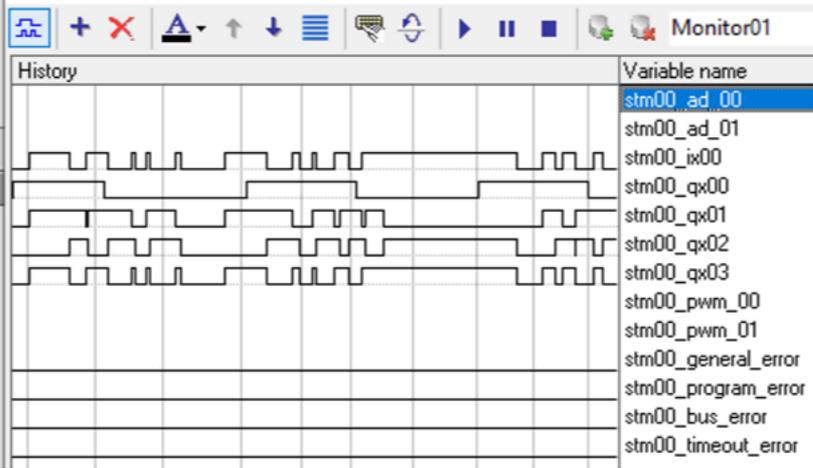
```
if fp(clock_10s) then  
  stm00_qx00:=!stm00_qx00;  
end_if;
```

```
if fp(stm00_ix00) then  
  stm00_qx01:=!stm00_qx01;  
end_if;
```

```
if fn(stm00_ix00) then  
  stm00_qx02:=!stm00_qx02;  
end_if;
```

```
stm00_qx03:=stm00_ix00;
```

Online Monitor



Hardware Setup

Slot	Name	Description	NAD	Prefix
CPU Unit	CyBro-2	10 binary inputs, 8 binary outputs, 4 a...	7332	
Slot 1	STM32F4	STM32F4 Multi Sensor 1 user key inp...	750	stm00
Slot 2				

# RS-485 vs CANBUS

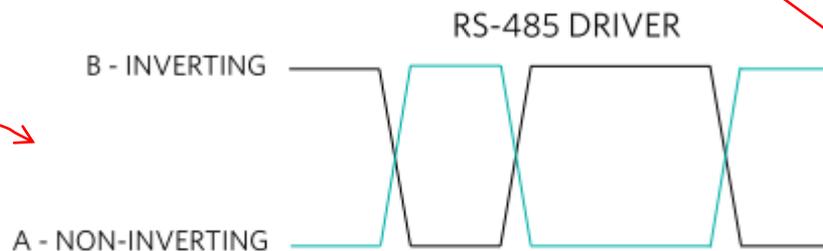
*Kako razrešiti ?*

## Podobno/enako:

- ▶ Diferencialni prenos
- ▶ Multi-master
- ▶ Zaključitev 120Ω

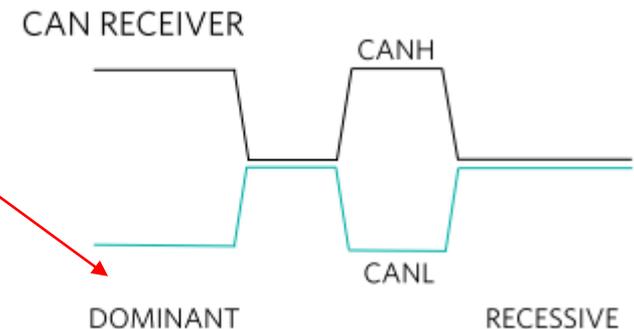
## Prednosti RS485 :

- ▶ Višja hitrost – do 35Mbit/s
- ▶ Obe stanji sta aktivno vodeni
- ▶ CANBUS (Wired AND) ima recisivno in dominantno stanje



## Prednosti CANBUS :

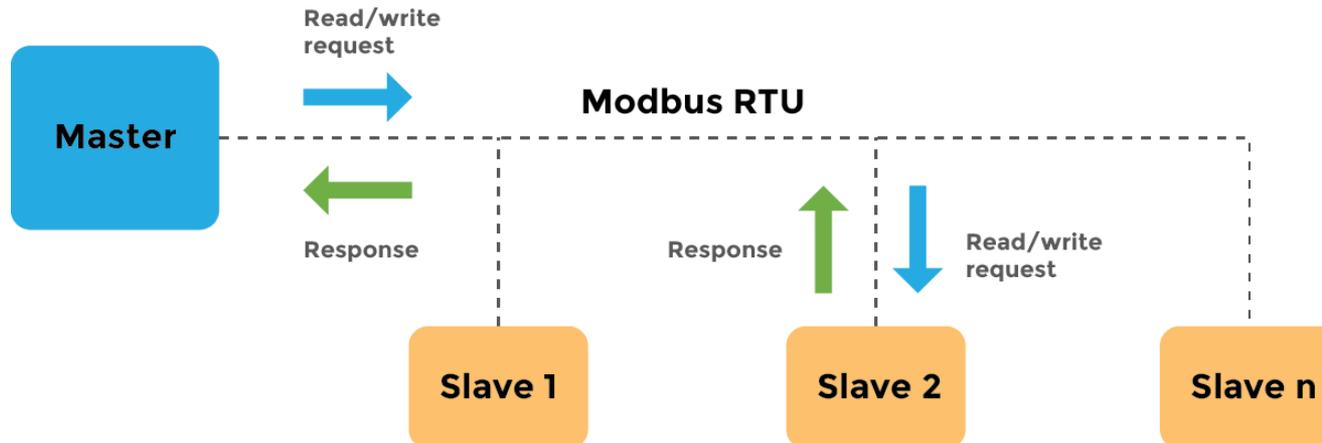
- ▶ Multi-master oddajanje
  - ▶ CANBUS arbitraža
  - ▶ RS485 –konflikt, poraba toka, segrevanje
- ▶ Dodatna preverjanja (nivo sporočila)
  - ▶ CRC, format sporočila
- ▶ Dodatna preverjanja(bitni nivo)
  - ▶ Spremljanje stanja linije (poslano/sprejeto)
  - ▶ **Potrditev (Acknowledge)**
  - ▶ **Bit-stuff (6. bit)**



# 10.4. MODBUS protokol

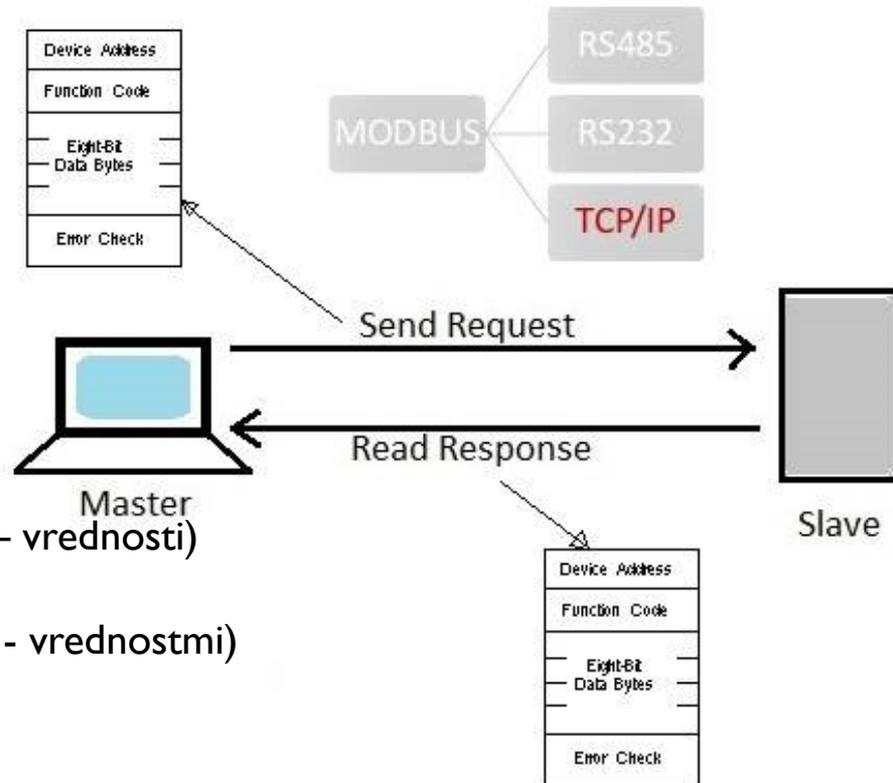
## MODBUS:

- ▶ Opis protokola in fizičnih nivojev
- ▶ Primeri Modbus komunikacij:
  - ❑ STM32F4 (MB master) <-> PC (MB slave)
  - ❑ Cybro kot MB Slave (višje nivojsko omrežje)
  - ❑ Smarteh Lir-2 termo tipalo I6xI2



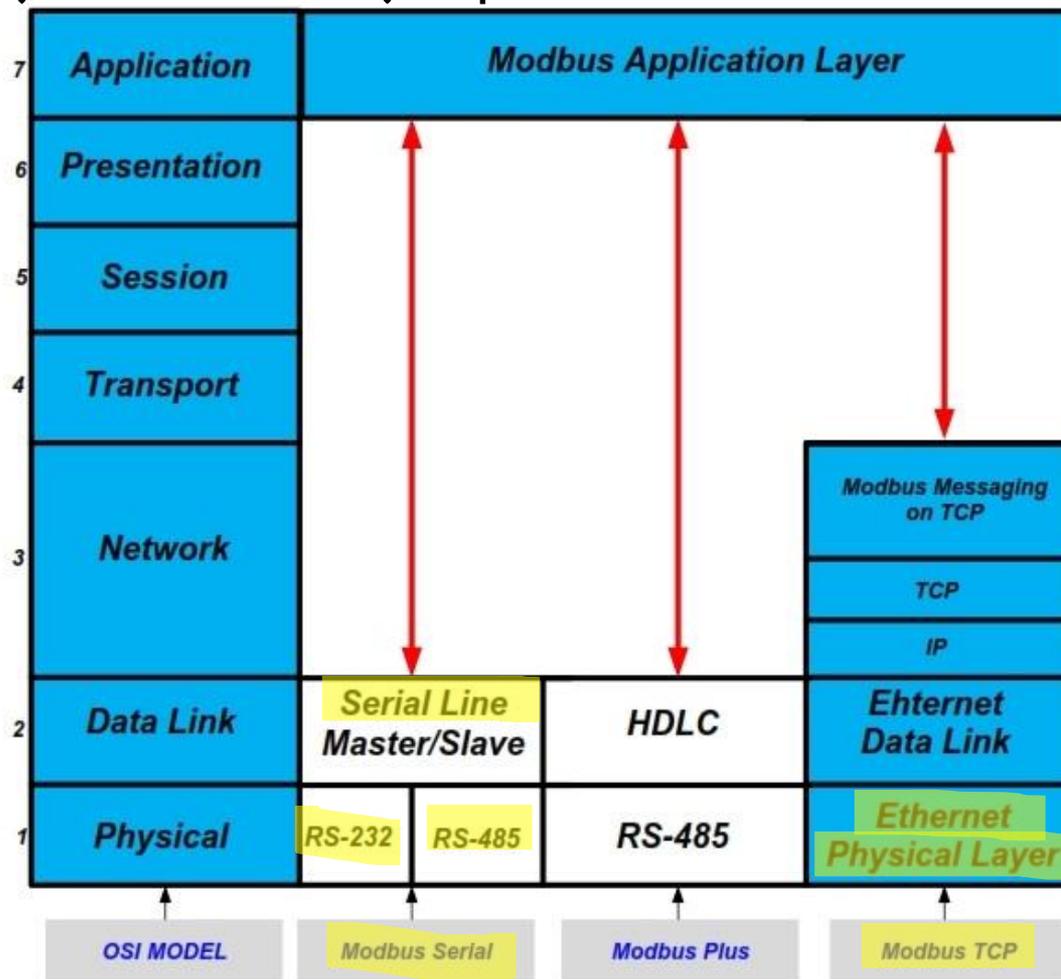
# MODBUS na kratko

- ▶ Višje nivojski komunikacijski protokol:
  - ▶ Različni „fizični“ nivoji:
    - ▶ RS-232
    - ▶ RS-485
    - ▶ TCP
- ▶ Sporočilno orientiran :
  - ▶ Serijska komunikacija:
    - ▶ ASCII (v obliki ASCII znakov)
    - ▶ RTU (v obliki posameznih bajtov – vrednosti)
  - ▶ Ethernet komunikacija:
    - ▶ TCP (v obliki TCP paketov z bajti - vrednostmi)
  - ▶ dvosmerna komunikacija
    - ▶ zahteva (MB Master) →
    - ▶ ← odziv (MB Slave)
  - ▶ V/I prostor: digitalni V/I, analogni V/I
  - ▶ MB ukazi (Function Codes – FC)



# MODBUS na kratko

- ▶ Višje nivojski komunikacijski protokol:



# MODBUS na kratko

## ▶ Sporočilno orientiran :

### ▶ Serijska komunikacija:

#### ▶ ASCII (v obliki ASCII znakov)

START	ADDRESS	FUNCTION	DATA	LRC CHECK	END
1 CHAR :	2 CHARS	2 CHARS	<i>n</i> CHARS	2 CHARS	2 CHARS CRLF

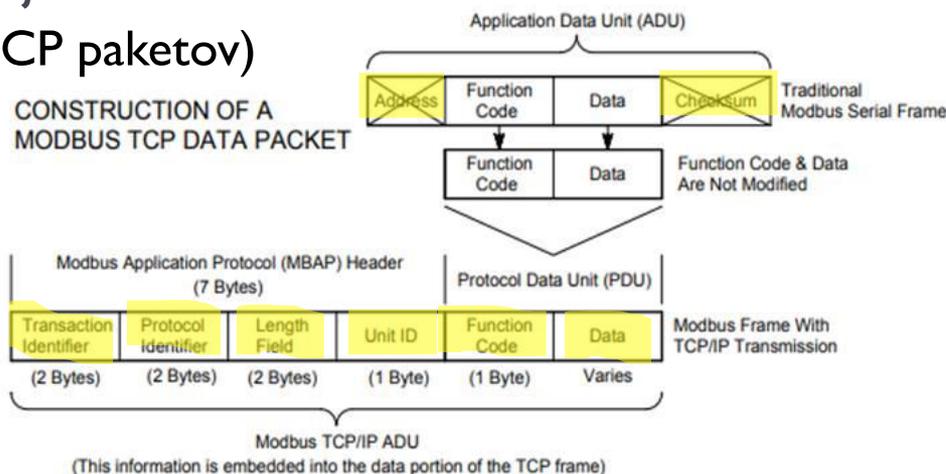
#### ▶ RTU (v obliki posameznih bajtov – vrednosti)

START	ADDRESS	FUNCTION	DATA	CRC CHECK	END
T1–T2–T3–T4*	8 BITS	8 BITS	<i>n</i> x 8 BITS	16 BITS	T1–T2–T3–T4*

\*For T1–T2–T3–T4, 3.5 character times at no communication.

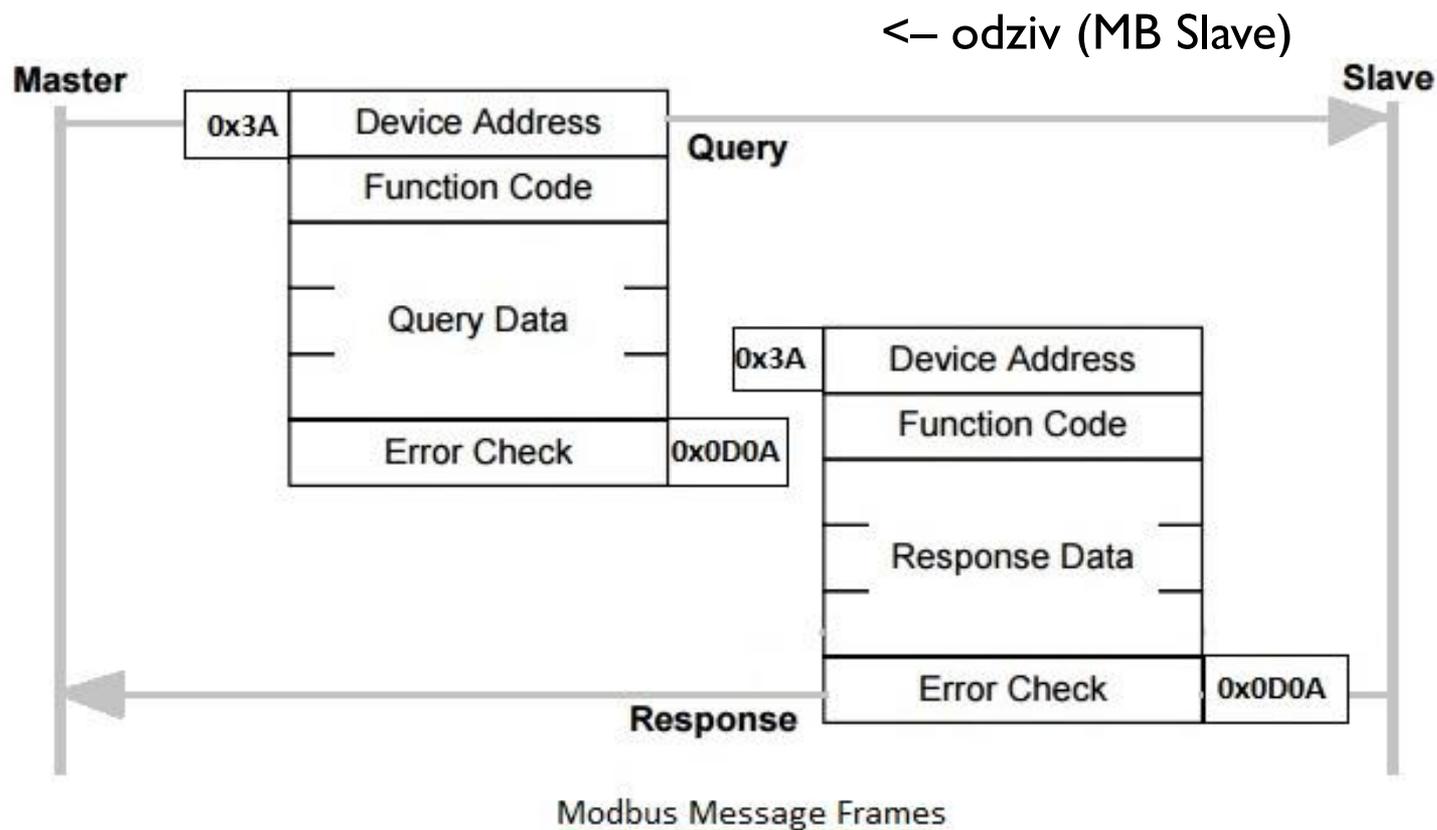
## ▶ Ethernet komunikacija:

### ▶ TCP (v obliki TCP paketov)



# MODBUS na kratko

- ▶ dvosmerna komunikacija
  - ▶ zahteva (MB Master) →



# MODBUS na kratko

- ▶ V/I prostor: digitalni V/I, analogni V/I

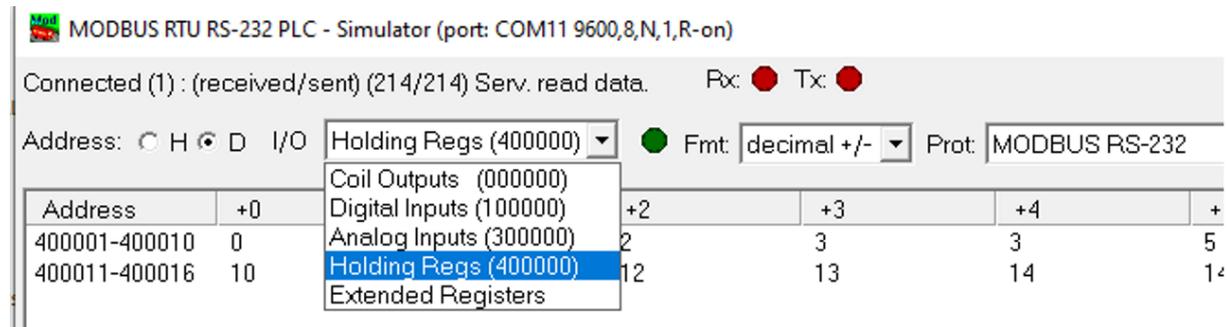
Types of Modbus Data	Data format	Address
Coils (Dig. Outputs)	Bits, Binary values	00001
Digital Inputs	Binary values	10001
Analog Inputs	Analog values	30001
Holding registers (outputs)	Analog values	40001

- ▶ MB ukazi (Function Codes – FC)

Function Code	Action	Table Name	Function Code	Action	Table Name
01 (01 hex)	Read	Discrete Output Coils	04 (04 hex)	Read	Analog Input Registers
05 (05 hex)	Write single	Discrete Output Coil	03 (03 hex)	Read	Analog Output Holding Registers
15 (0F hex)	Write multiple	Discrete Output Coils	06 (06 hex)	Write single	Analog Output Holding Register
02 (02 hex)	Read	Discrete Input Contacts	16 (10 hex)	Write multiple	Analog Output Holding Registers

# MODBUS – Primer 1: STM32 <-> PC (USB VCOMPort)

## ▶ PC – MB slave simulator:



## ▶ STM32F4 – MB master SW :

```
// --- Slave ID=1, Write state 511 of 1 analog output register on addresses 0-15
usModbusUserData[0]= usModbusUserData[1]= usModbusUserData[2]= usModbusUserData[3]=MBCnt;
errorCode = eMBMasterReqWriteMultipleHoldingRegister(1,MBCnt,2,usModbusUserData,1);
MBCnt= (MBCnt + 1) % 16;
```

[https://github.com/LAPSyLAB/STM32F4\\_Discovery\\_VIN\\_Projects/tree/main/STM32\\_USB\\_Modbus\\_Demo](https://github.com/LAPSyLAB/STM32F4_Discovery_VIN_Projects/tree/main/STM32_USB_Modbus_Demo)

# MODBUS – Primer 1: STM32 <-> PC (USB VCOMPort)

## ► Preveritev MODBUS ukazov:

```
// --- Slave ID=1, Read state of 8 coils address 0
errorCode = eMBMasterReqReadCoils( 1, 0, 8 ,1 ); //RXbuffer[3];

// --- Slave ID=1, Write ON state to one coil on addresses 0-7
errorCode = eMBMasterReqWriteCoil(1,MBCnt,0xFF00,1);

// --- Slave ID=1, Write ON state to all coils on addresses 0-7
errorCode = eMBMasterReqWriteMultipleCoils(1,0,8, &MBCnt,1);

// --- Slave ID=1, Read from 8 discrete inputs on addresses 0-7
errorCode = eMBMasterReqReadDiscreteInputs(1,0,8,1); //result in RXbuffer[3];

// --- Slave ID=1, Read from 2 analog input registers from addresses 0-1
errorCode = eMBMasterReqReadInputRegister(1,0,2,1); //results in RXbuffer[3-6];

// --- Slave ID=1, Read state of 2 analog output registers from addresses 0-1
errorCode = eMBMasterReqReadHoldingRegister(1,0,2,1); //results in RXbuffer[3-6];

// --- Slave ID=1, Write state 511 of 1 analog output register on addresses 0-15
errorCode = eMBMasterReqWriteHoldingRegister(1,MBCnt,511,1);

// --- Slave ID=1, Write state 511 of 1 analog output register on addresses 0-15
errorCode = eMBMasterReqWriteMultipleHoldingRegister(1,MBCnt,2,usModbusUserData,1);
```

Rešitev temelji na dveh projektih :

- [FreeModBUS](#) knjižnica
- v Middlewares/Modbus
- in dodatek za Master funkcionalnost
- temelji na viru:  
[https://github.com/armink/FreeModbus\\_Slave-Master-RTT-STM32](https://github.com/armink/FreeModbus_Slave-Master-RTT-STM32)
- predelano za USB VCOMPort komunikacijo in lastne podatkovne strukture
- Modbus Slave (server) **simuliramo na PC** strani z aplikacijo :
- <https://sourceforge.net/projects/modrssi2/>
- izberemo ustrezen COM port (VCOMPort od STM32F4)

[https://github.com/LAPSyLAB/STM32F4\\_Discovery\\_VIN\\_Projects/tree/main/STM32\\_USB\\_Modbus\\_Demo](https://github.com/LAPSyLAB/STM32F4_Discovery_VIN_Projects/tree/main/STM32_USB_Modbus_Demo)

# MODBUS – Primer 2: Cybro kot MB slave

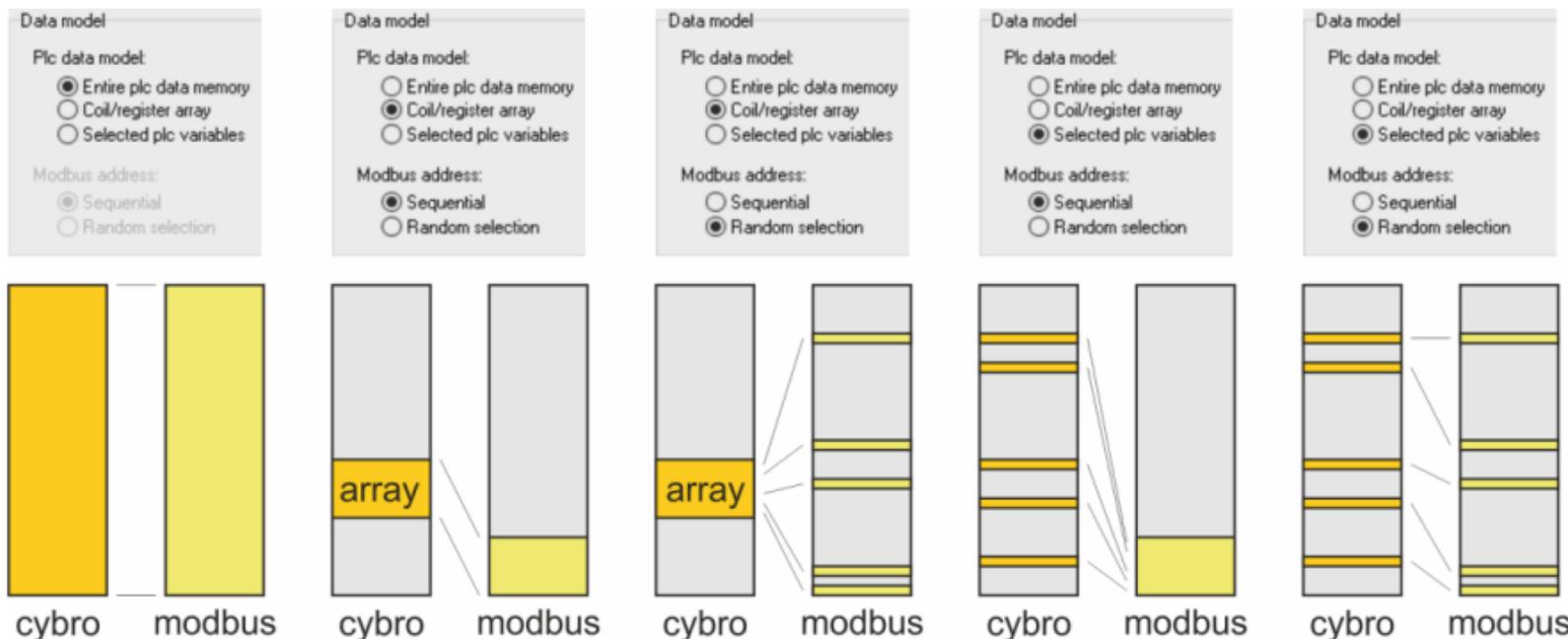
Cybro supports:

- Modbus **RTU slave** (RS232/RS485)
- Modbus **TCP slave** (Ethernet)

Function codes:

code	hex	command
1	01h	READ_COILS
3	03h	READ_HOLDING_REGISTERS
5	05h	WRITE_SINGLE_COIL
6	06h	WRITE_SINGLE_REGISTER
15	0Fh	WRITE_MULTIPLE_COILS
16	10h	WRITE_MULTIPLE_REGISTERS

Modbus data model describes how modbus coils and registers are translated to Cybro memory.



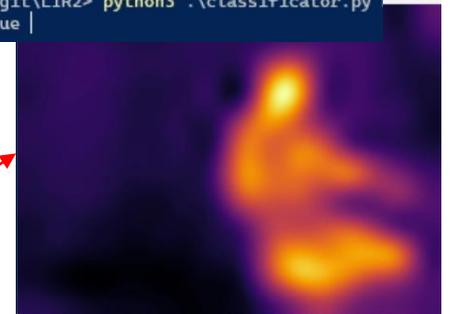
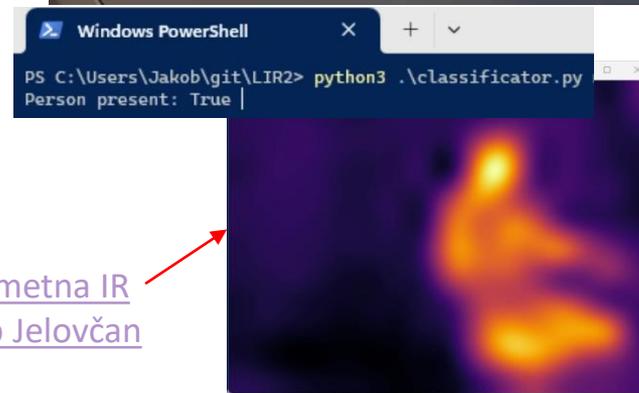
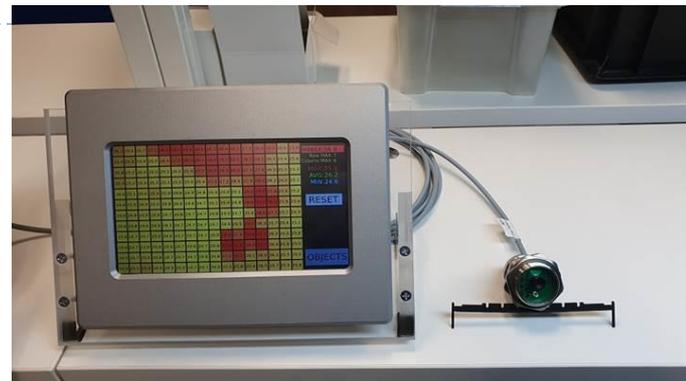
# MODBUS – Primer 3: Smarteh Lir-2 termo tipalo 16x12

Prilagam kratek dokument z navodili za priklop in zagotovitev pravilnega delovanja komunikacije med LIR-2 senzorjem ter osebnim računalnikom.

Poleg tega prilagam še link: <https://github.com/kolan51/ModbusLIR2example>, do GitHub repozitorija, kjer se nahaja osnovna python skripta, ki jo lahko uporabite, kot osnovo za izdelavo vaših lastnih projektov.

V navodilih je na koncu tudi kratek opis uporabe dodatnega projekta za barvni oziroma toplotni izris zajetih meritev, ki pa rabi še en minimalen popravek in ga bom poslal naknadno. Poleg naštetega prilagam še user manual za LIR-2, kjer je podrobnejši opis vseh registrov in samega delovanja.

Pošiljam še dodatni projekt kolega Domna, kot sem obljubil včeraj: [https://github.com/kolan51/LIR2\\_ColorTemp](https://github.com/kolan51/LIR2_ColorTemp). Lahko ga poganjate, kar preko .exe datoteke, navodila za uporabo projekta pa so ravno tako opisana v dokumentu, ki sem ga poslal včeraj.

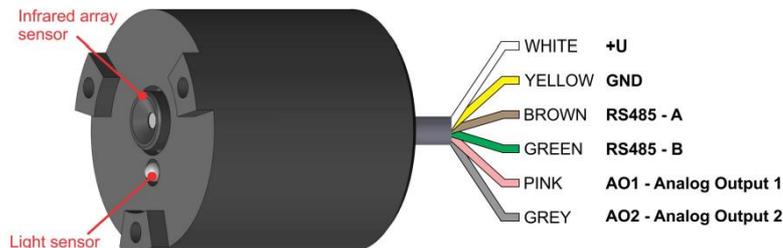


VIN PROJ 2023 Pametna IR kamera 8x8 Jakob Jelovčan

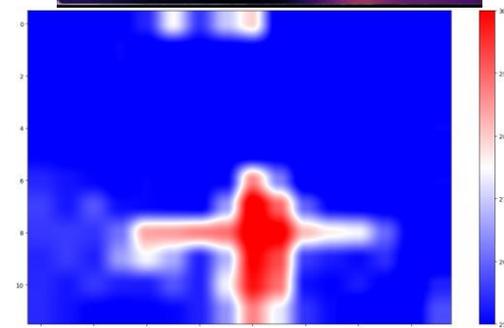
## Smarteh LIR2 Tipalo

Table 1: Features

Infrared array sensor 12 x 16 Pixel
Light indication
Modbus RTU communication protocol
2 analog outputs 0 .. 10 V
IP65



VIN PROJ 2023 Zaznavanje človeka z matrično termo kamero Jakob jelovčan.mp4



## 10.5 Universal Serial Bus (USB)

- Povezovalni standard USB je bil objavljen 1996. Določa:
  - specifikacije za **kable, priključke in protokole za povezavo,**
  - **komunikacijo in napajanje** med osebnimi računalniki in njihovimi perifernimi napravami.
- Za razvoj in vzdrževanje skrbi USB Implementers Forum (USB-IF), ki vključuje podjetja, kot so: Hewlett Packard, Intel, LSI Corporation, Renesas, Microsoft

- Cilji, ki so jih želeli doseči, pri razvoju USB standarda:
  - Nastavljanje stikal ali mostičev na kartici ali napravi ne bo potrebno.
  - Pri instalaciji novih V/I naprav ne bo treba odpirati ohišja.
  - **Samo ena vrsta kabla** za vse vrste naprav.
  - V/I naprave naj dobijo **napajanje po USB kablu.**
  - Možnost priključitve do 127 V/I naprav na krmilnik.
  - Podpora napravam, ki **delujejo v realnem času.**
  - Naprave naj bo možno **nameščati med delovanjem** (ang. hot plug).
  - Po instalaciji nove naprave naj ne bi bilo potreben ponovni zagon (ang. reset) računalnika.
  - Povezava naj bi bila **poceni.**

# USB – na kratko

- ❑ Električno je to **dvosmerna simetrična povezava z enim parom podatkovnih linij (ang. half-duplex).**

- Nizke hitrosti prenosa: LS (Low-speed) = 1,5 Mb/s
- Visoke hitrosti prenosa: HS: High-speed 480 Mb/s
- Maksimalna dolžina:  $l_{max} = 5$  m

- ❑ Za preprečevanje odbojev se uporablja serijska zaključitev pri oddajniku in paralelna zaključitev pri sprejemniku ( $Z_0 = 90 \Omega$ ).

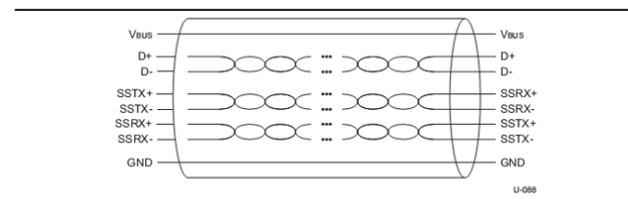
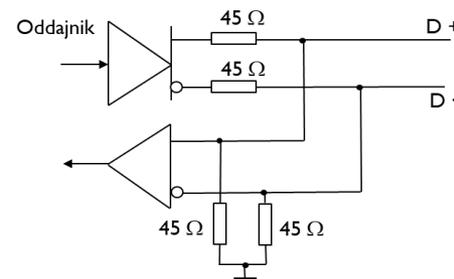
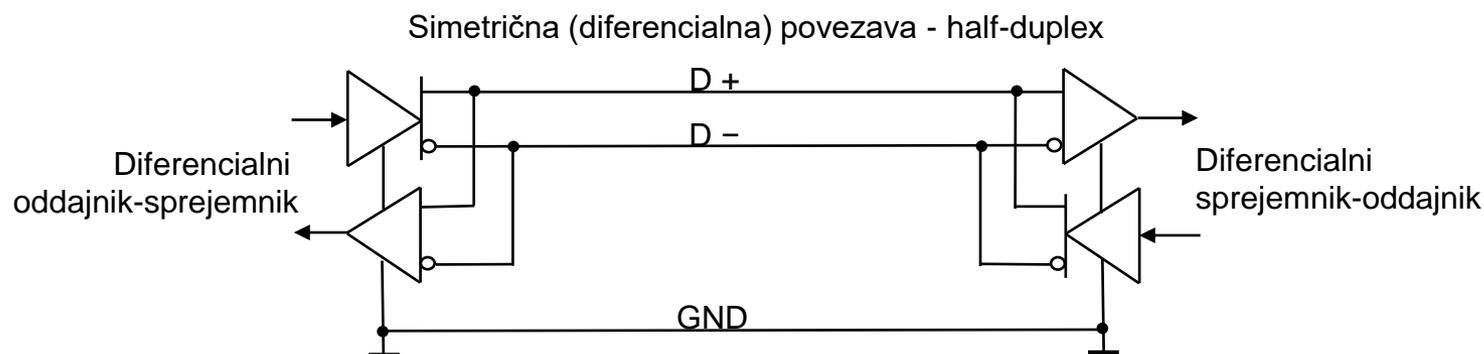


Figure 3-2. USB 3.0 Cable



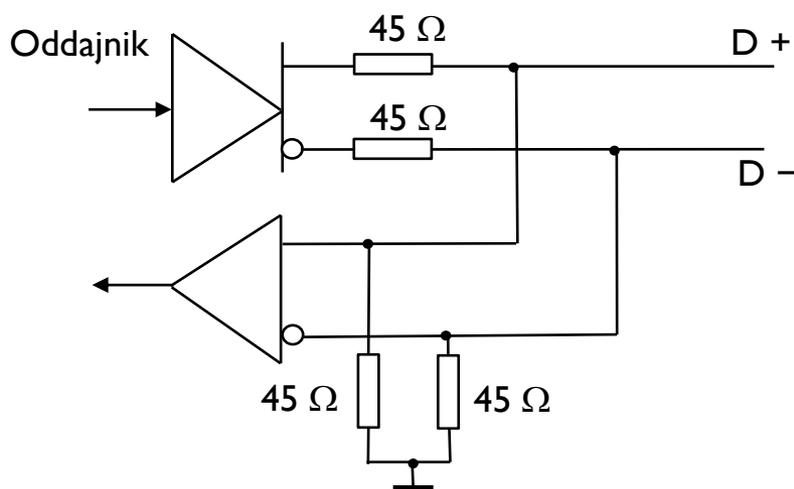
- ❑ Podatki se prenašajo med dvema napravama v majhnih količinah, **kot paketi**. Z vsakim paketom se prenese določeno število bajtov (enota digitalnih informacij).
- ❑ Logično je USB sistem videti **kot množica povezav** (cevi) med USB napravami in korenskim vozliščem (gostiteljskim krmilnikom).
- ❑ Vsi prenosi potekajo med korenskim vozliščem in USB napravami. **Prenos med napravami ni možen.**

- ❑ Električno je to dvosmerna simetrična povezava z enim parom podatkovnih linij (ang. half-duplex).

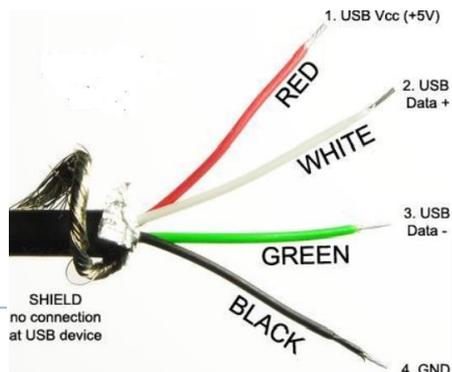


- Nizke hitrosti prenosa: LS (Low-speed) = 1,5 Mb/s  
 Minimalna izhodna napetost pri visokem nivoju (1)      Maksimalna izhodna napetost pri nizkem nivoju(0)  
 $V_{OH} > 2,8V$        $V_{OL} < 0,3V$
  - Visoke hitrosti prenosa: HS: High-speed 480 Mb/s  
 Minimalna izhodna napetost pri visokem nivoju (1)      Maksimalna izhodna napetost pri nizkem nivoju(0)  
 $V_{OH} = 400 \text{ mV} \pm 10\%$        $V_{OL} = 0V \pm 10 \text{ mV}$
- Maksimalna dolžina:  $l_{max} = 5 \text{ m}$

- Za preprečevanje odbojev se uporablja serijska zaključitev pri oddajniku in paralelna zaključitev pri sprejemniku ( $Z_0 = 90 \Omega$ ).

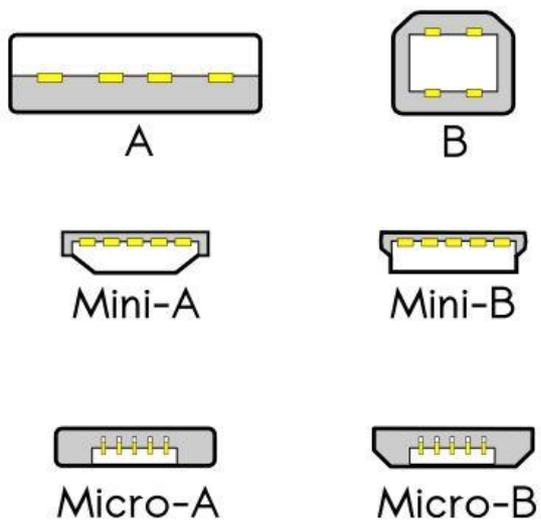


- USB kabel

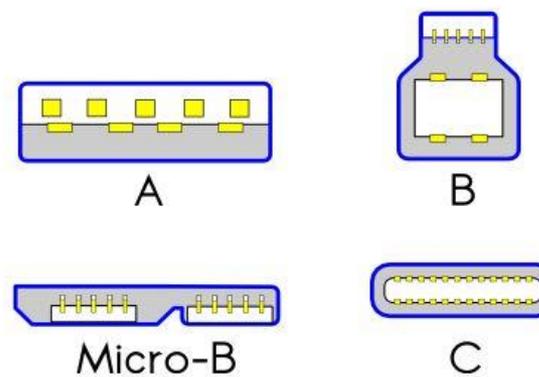


## □ Strojna oprema - Konektorji USB

### USB 1.0 - 2.0



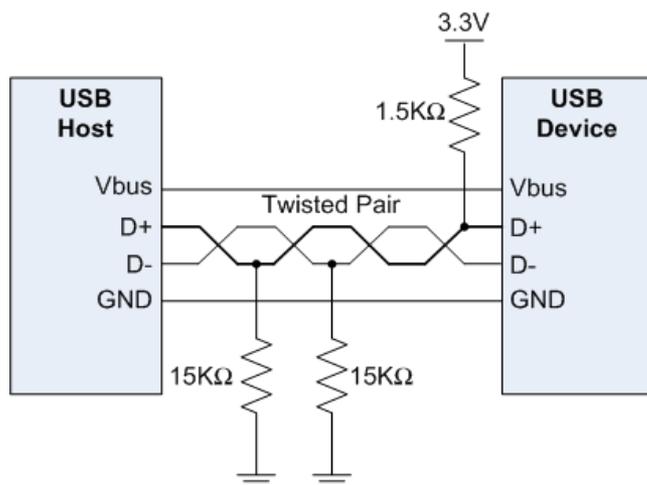
### USB 3.0 - 3.1



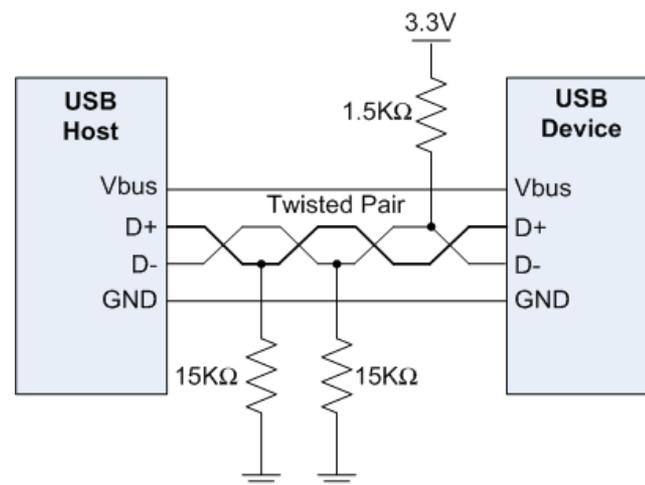
[https://en.wikipedia.org/wiki/USB\\_hardware](https://en.wikipedia.org/wiki/USB_hardware)

## □ Identifikacija hitrosti :

- ▶ low speed (1.5 Mbit/s = 187 KB/s)
- ▶ full speed (12 Mbit/s = 1.5 MB/s)
- ▶ high speed (480 Mbit/s = 60 MB/s)
- ▶ super speed (4.8 Gbit/s = 600 MB/s)



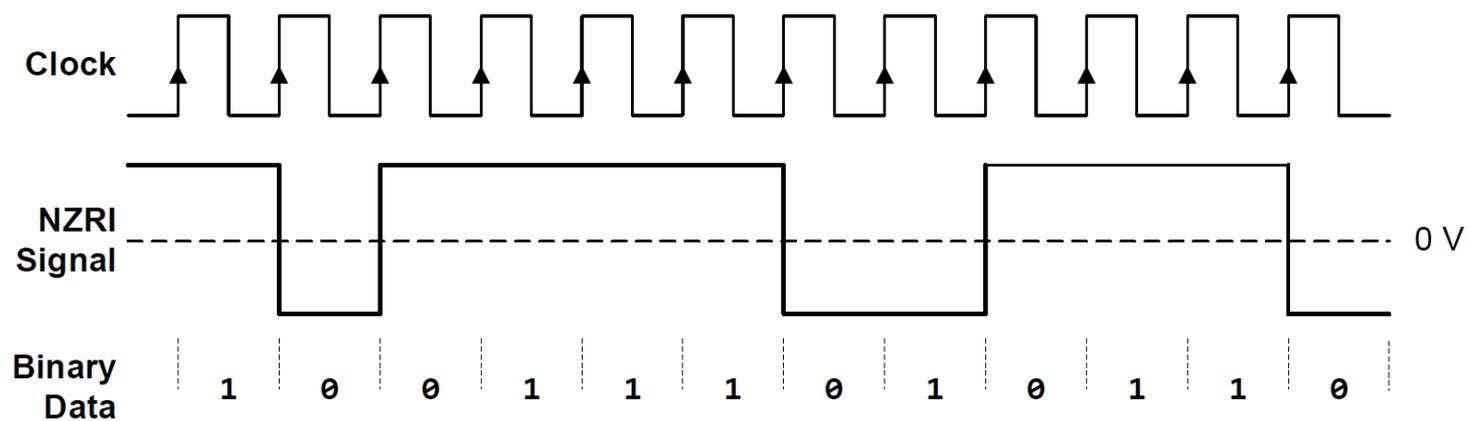
Full-speed mode (12 Mbits/s)  
1.5KΩ pull-up na D+



Low-speed mode (1.5Mbits/s)  
1.5KΩ pull-up na D-

□ Kodiranje NRZI :

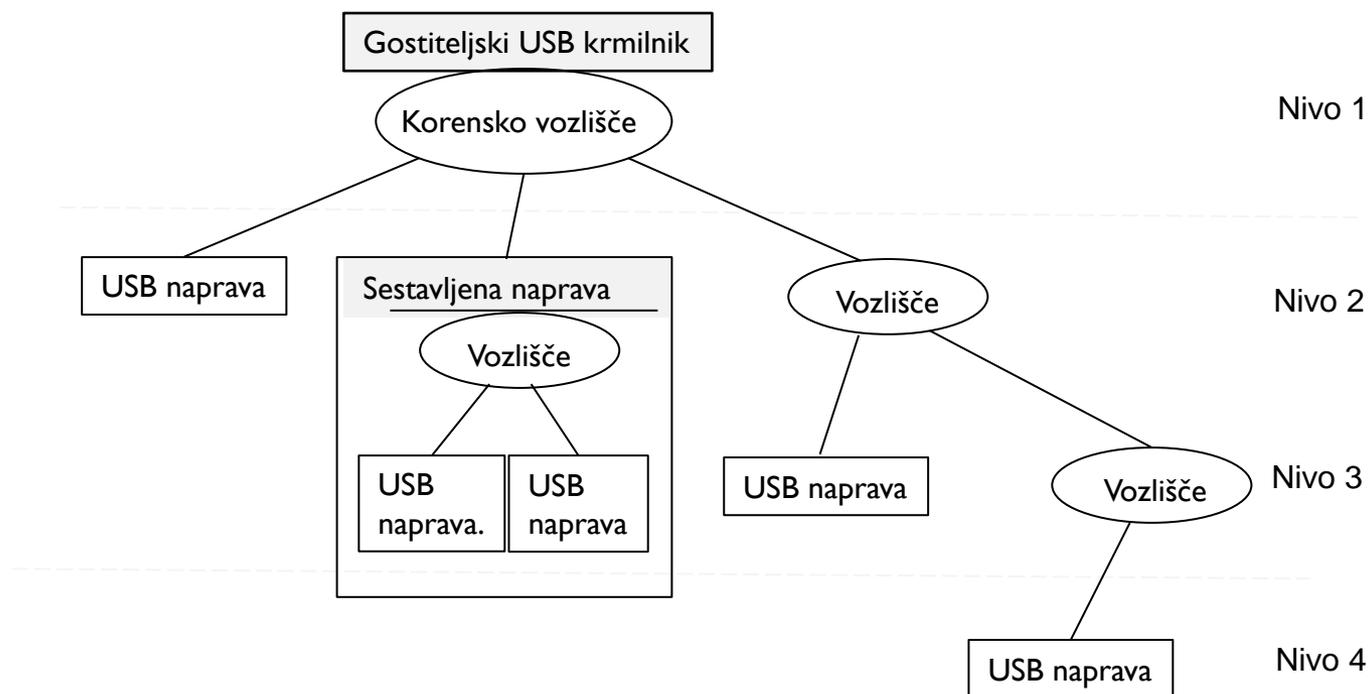
- ▶ 0: sprememba napetosti
- ▶ 1: ni spremembe



## 10.5.1 Topologija USB

### □ Splošno:

- Naprave USB so fizično povezane v vozlišče.
- Za delovanje je potreben gostiteljski USB krmilnik (ang. root host),
- Vsaka USB naprava ima tudi svoj USB krmilnik.

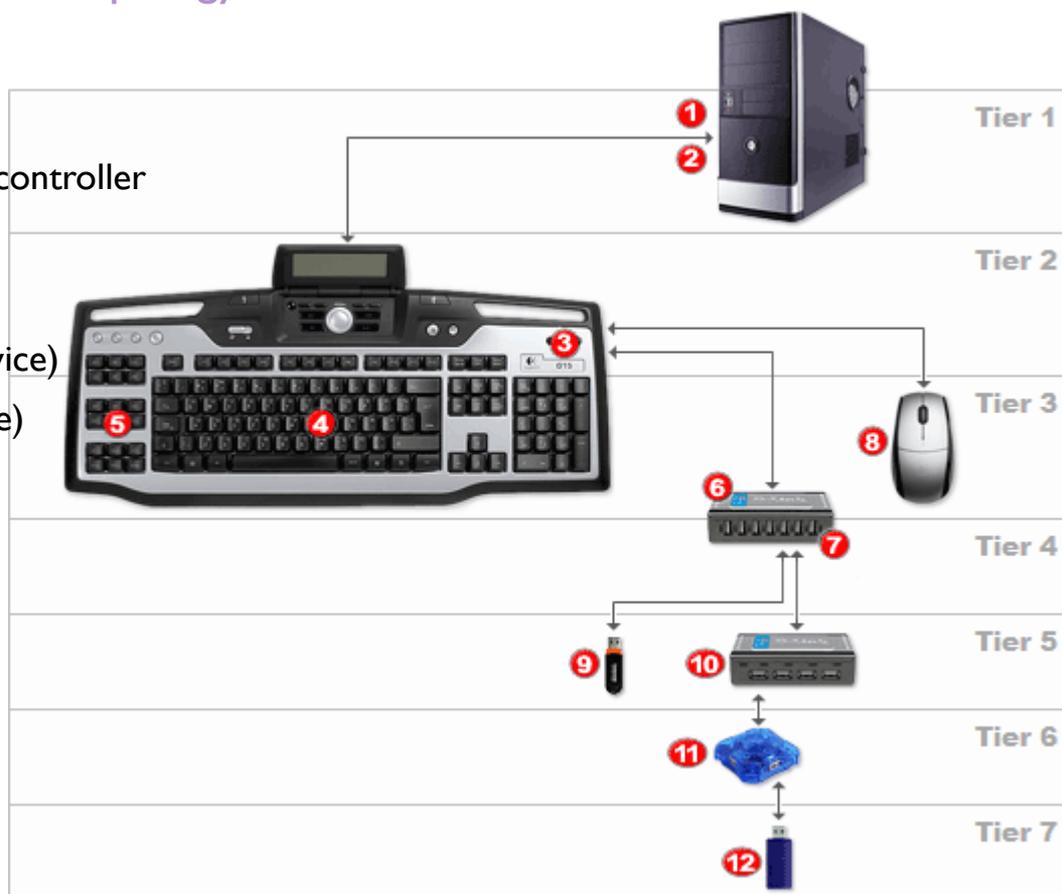


- ❑ Maksimalno število nivojev vključno s korenskim vozliščem je 7.
- ❑ Vsaka naprava ali vozlišče predstavlja en nivo. Sestavljena naprava npr. tipkovnica in sledilna plošča predstavljata dva nivoja.
- ❑ Logično USB gostiteljski krmilnik z vsako napravo komunicira kot bi bil z njo direktno povezan (točka-v-točko).
- ❑ Logične povezave  $\Rightarrow$  cevi (bit pipe).
- ❑ Urin signal se po povezavi ne pošilja, na sprejemni strani se urin signal restavrira iz podatkovnega zaporedja.
- ❑ Uporablja se NRZI kodiranje (bomo spoznali v nadaljevanju)
- ❑ Ko priključimo novo USB napravo, gostiteljski krmilnik to zazna in pošlje prekinitveno zahtevo v CPE.
- ❑ Operacijski sistem (OS) poizve kakšna naprava je to in kakšno hitrost zahteva, oziroma koliko pasovne širine bo zasedla.
- ❑ OS dodeli novi napravi enoveljaven naslov (1 – 127) ter naslov in ostale parametre povezave naloži v konfiguracijske registre krmilnika USB naprave.
- ❑ Neinicializirane naprave začnejo z naslovom 0, da se lahko v nadaljevanju uporabijo.
  
- ❑ Namestitev poteka tako brez konfiguracijskih posegov uporabnika (ang. on the fly).

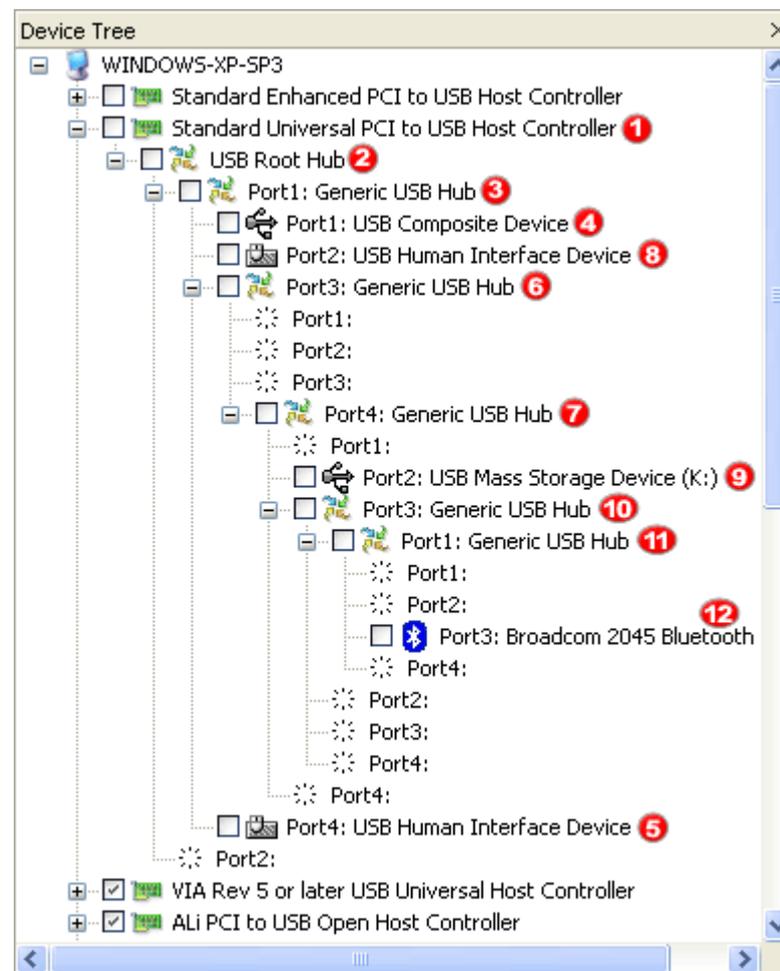
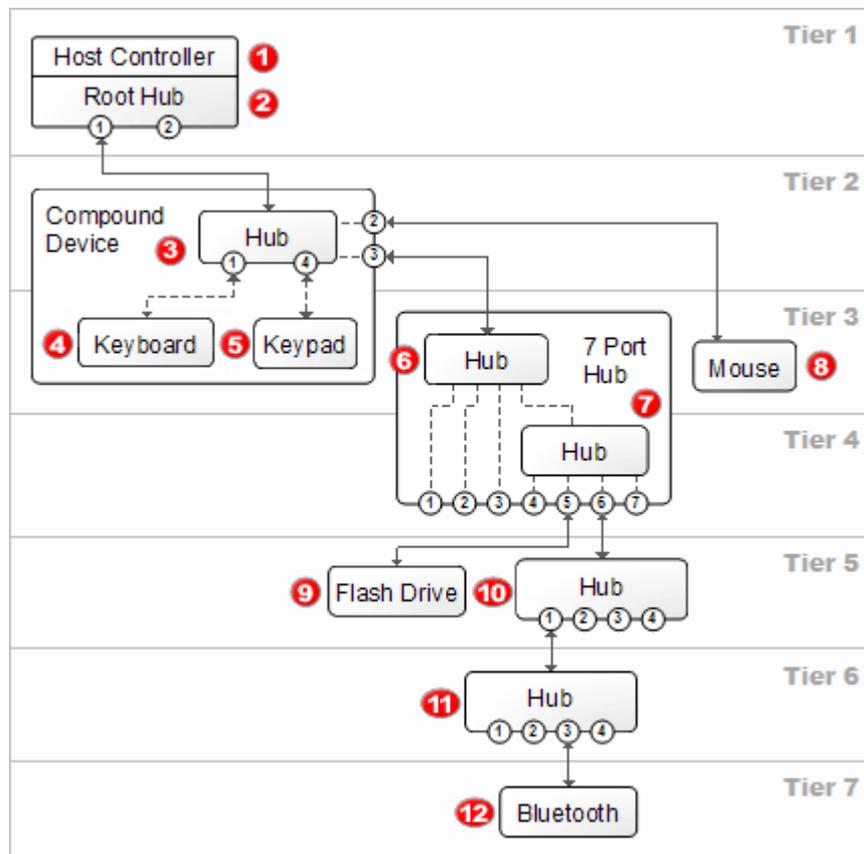
□ Fizična topologija vodila – Uporabniški pogled

<http://www.usblyzer.com/usb-topology.htm>

1. USB host with host controller
2. 2-port root hub integrated into the host controller
3. 4-port hub integrated into the keyboard  
(part of the compound device)
4. USB keyboard (part of the compound device)
5. USB keypad (part of the compound device)
6. 4-port hub (part of the 7-port hub)
7. 4-port hub (part of the 7-port hub)
8. USB mouse
9. USB flash drive
10. 4-port hub
11. 4-port hub
12. USB bluetooth adapter

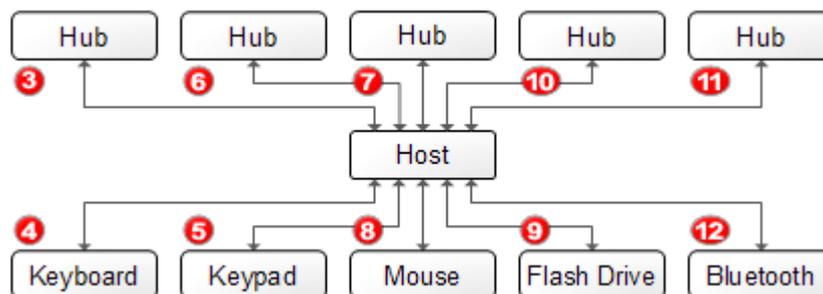


## ❑ Fizična topologija vodila USB – podrobnejši pogled

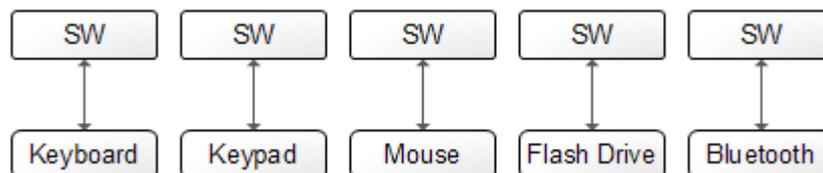


□ Logična topologija vodila USB:

- Fizično so naprave povezane nivojsko, v zvezdni topologiji.
- Gostitelj komunicira z V/I napravami, kot da bi bile direktno povezane na 'Root hub'.



□ Pogled programskega odjemalca – program odjemalca komunicira z napravo, kot da bi bila direktno povezana na gostitelja.



## 10.5.2 Prenos podatkov

- ❑ Podatki se prenašajo med dvema napravama v majhnih količinah, **kot paketi**. Z vsakim paketom se prenese določeno število bajtov (enota digitalnih informacij).
- ❑ Pošljejo se tudi druge informacije, kot so:
  - izvor podatkov in ponor podatkov,
  - dolžina podatkov,
  - podrobnosti o odkritih napakah
- ❑ Tipi prenosov podatkov:
  - ‘Interrupt transfer ‘ – naprave zahteva prekinitev, majhne količine podatkov, manj pogosti prenosi (tipkovnica, miška)
  - ‘Bulk transfer ‘ - večje količine podatkov, nizka prioriteta (tiskalniki, skenerji)
  - ‘Isochronous transfer ‘ – lahko pride do napak, prenos paketov se ne prekine in se ne pošilja ponovno (audio, video)
  - ‘Control transfer ‘ : konfiguracija in kontrola naprave USB.
  - <https://www.cmd-ltd.com/advice-centre/usb-chargers-and-power-modules/usb-and-power-module-product-help/usb-data-transfer-guide/>

- ❑ Logično je USB sistem videti **kot množica povezav** (cevi) med USB napravami in korenskimi vozliščami (gostiteljskim krmilnikom).
- ❑ Vsi prenosi potekajo med korenskimi vozliščami in USB napravami. Prenos med napravami ni možen.
- ❑ Prenos med dvema USB napravama je možen le, če ima krmilnik ene od USB naprav tudi gostiteljske sposobnosti (ang. dual role device, OTG).
- ❑ USB gostiteljski krmilnik pošlje vsako milisekundo ( $1,00 \text{ ms} \pm 0,05 \text{ ms}$ ) nov okvir in tako vzdržuje sinhronizacijo z napravami.
- ❑ Okvir, ki ga sestavljajo paketi, je povezan z določeno povezavo USB.
- ❑ **Prvi paket** v okvirju (SOF) je **vedno v smeri od korenkega vozlišča proti napravi**. Ostali paketi v okvirju pa so lahko v eni ali drugi smeri.

## Primer komunikacije med USB krmilnikom in USB napravami

SOF: Start of Frame;

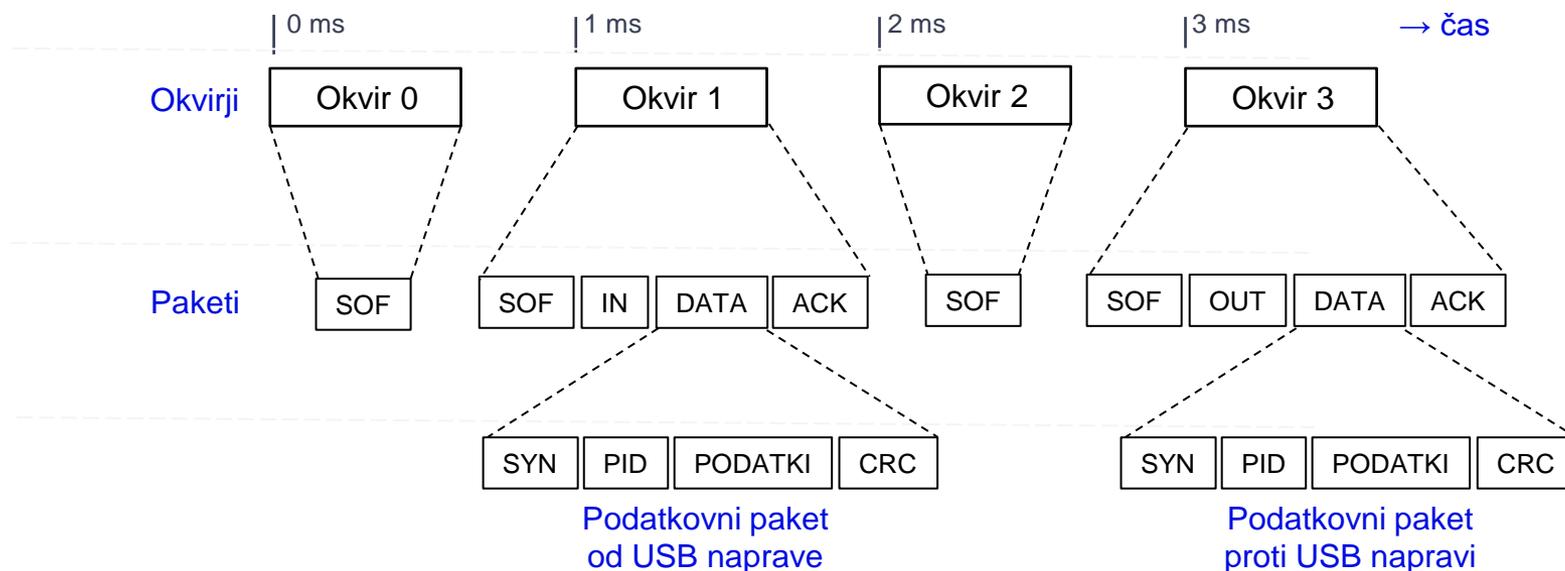
IN: prenos od naprave;

OUT: prenos proti napravi (vsi vedno od krmilnika proti napravi)

SYN: 8-bitno sinhronizacijsko polje

PID: 8-bitna oznaka vrste paketa

CRC: 16-bitno polje za detekcijo napak



## 10.5.3 Razvoj standarda USB

---

- ❑ USB 1.0 (1996): 1,5 Mb/s Low-speed – LS
- ❑ USB 1.1 (1998): Bolj stabilno delovanje, 12 Mb/s Full-speed – FS
- ❑ USB 2.0 (2000): 480 Mb/s High-speed (HS), Kodiranje NRZI, Dolžina povezave do 5m, S petimi razdelilniki (hubi) do 30m, Obremenitev vodila pri +5V do 500mA (5 x 100 mA), Možno hkratno delovanje LS, FS in HS naprav, Potreben je gostiteljski krmilnik (Host controller)
- ❑ USB OTG (USB On-the-Go; leto 2001): Dodatek k standardu USB 2.0, Periferna naprava lahko prevzame vlogo gostiteljske naprave (krmilnika, ki vodi prenos) – dual role device, Novi konektorji (mini A, mini B), manjša poraba
- ❑ USB 3.0 (leto 2008): Maksimalna teoretična hitrost 5 Gb/s (realno 3,2 Gb/s) - Super-speed – SS, Kodiranje 8b/10b, Dodatni 4 vodniki (2 x diferencialni signal v eno in drugo smer), Praktična dolžina povezave do 3m (standard dolžine direktno ne omejuje), Hkratni dvosmerni prenos (full-duplex), Novi konektorji (kompatibilni z USB 2.0), Obremenitev vodila do 900 mA (6 x 150 mA), Kompatibilnost z USB 2.0
- ❑ USB 3.1 (leto 2013): Maksimalna teoretična hitrost 10 Gb/s SuperSpeed+ , Kodiranje 128b/132b ( $\Rightarrow$  hitrost  $10 \cdot 10^9 \text{ b/s} \cdot (128/130) / 8 \text{ b/B} = 1,23 \text{ GB/s}$ ), Konektor USB C, ki pa ni del specifikacij USB 3.1
- ❑ USB 3.2 (leto 2017) in USB 4 (leto 2019)

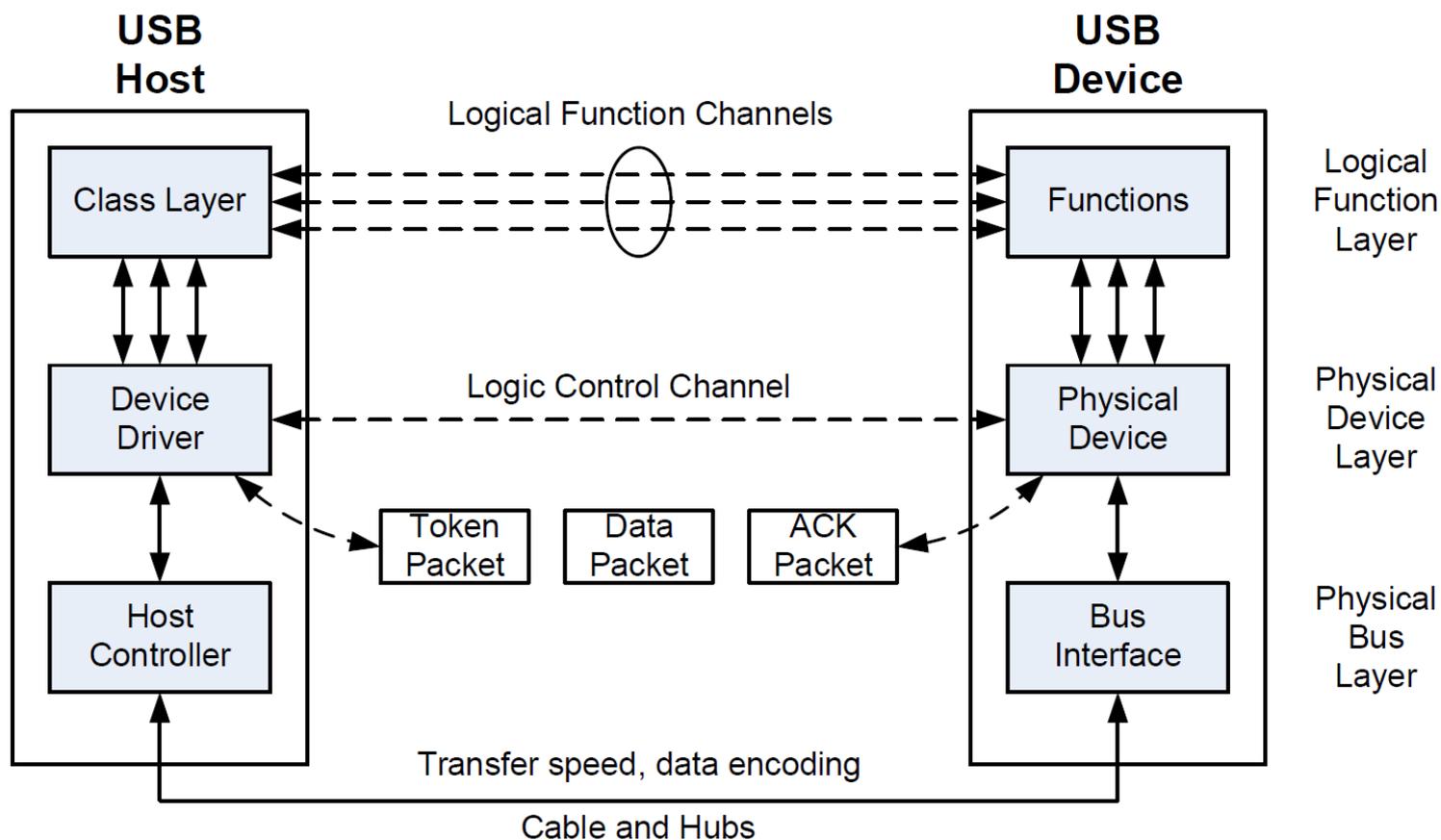
## 10.5.4 Napajanje in polnjenje preko kabla USB

---

- ❑ Napajanje USB naprav po USB kablu:
  - Napetost na korenskem vozlišču je  $5\text{ V} \pm 5\%$ , USB naprave morajo delovati do 4,4 V.
  - Enota obremenitve (ang. unit load) je definirana s tokom:  
USB 2.0  $\Rightarrow$  100 mA (= 0,5 W pri napetosti 5 V)  
USB 3.0  $\Rightarrow$  150 mA (= 0,75W)
  - Ko se USB naprava priklopi na vodilo, dobi 100 mA toka, med delovanjem lahko zahteva še dodatne obremenilne enote, do največ 5 (6 pri USB 3.0).
  - USB naprave se delijo v:
    - Varčne naprave (ang. Low power: tok  $\leq$  100 mA)
    - Požrešne naprave (ang. High power: tok  $\leq$  500 mA)
    - Naprave z zunanjim napajanjem (ang. Self powered devices)
- ❑ USB polnjenje akumulatorjev:
  - CDP (Charging Downstream Port) - napajalni in podatkovni vodniki.
    - Možno polnjenje in hkraten prenos podatkov.
    - Maksimalni tok do 5 A, Konektor C 1,5 A.
    - Pri velikih polnilnih tokovih se pri večjih hitrostih lahko pojavijo motnje na podatkovnih povezavah.
  - DCP (Dedicated Charging Port) - samo napajalna vodnika brez podatkovnih (avtomobilski polnilci, ...)

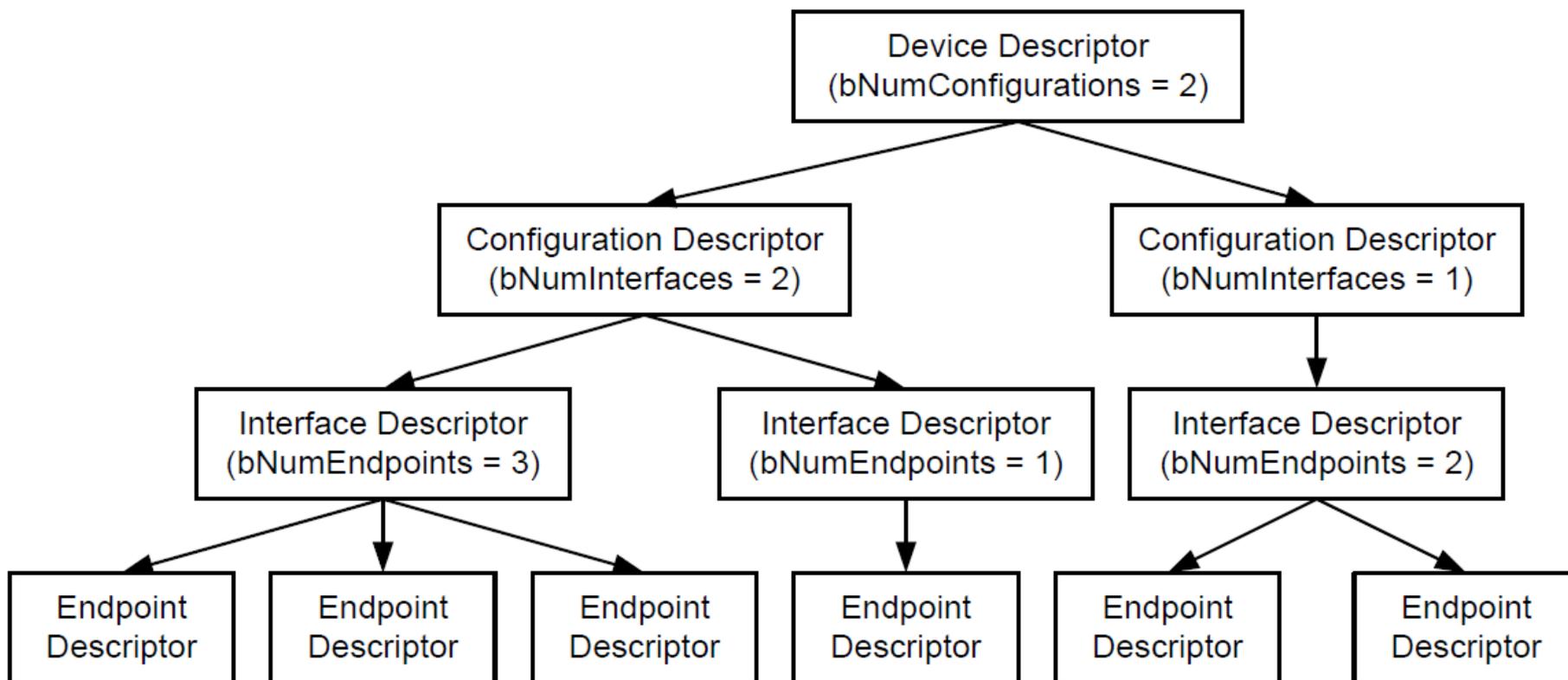
## 10.5.5 USB – programerski vidik

### □ Komunikacijske plasti



## 10.5.5 USB – programerski vidik

### □ Deskriptorji



## 10.5.5 USB – programerski vidik

### □ Deskriptorji

Field Name	Size	Offset
bLength	1	0
bDescriptorType	1	1
bcdUSB	2	2
bDeviceClass	1	4
bDeviceSubClass	1	5
bDeviceProtocol	1	6
bMaxPacketSize	1	7
idVendor	2	8
idProduct	2	10
bcdDevice	2	12
iManufacturer	1	14
iProduct	1	15
iSerialNumber	1	16
bNumConfigurations	1	17

#### Device descriptor

Field Name	Size	Offset
bLength	1	0
bDescriptorType	1	1
bInterfaceNumber	1	2
bAlternateSetting	1	3
bNumEndpoints	1	4
bInterfaceClass	1	5
bInterfaceSubClass	1	6
bInterfaceProtocol	1	7
iInterface	1	8

#### Interface descriptor

Field Name	Size	Offset
bLength	1	0
bDescriptorType	1	1
wTotalLength	2	2
bNumInterfaces	1	4
bConfigurationValue	1	5
iConfiguration	1	6
bmAttributes	1	7
bMaxPower	1	8

#### Configuration descriptor

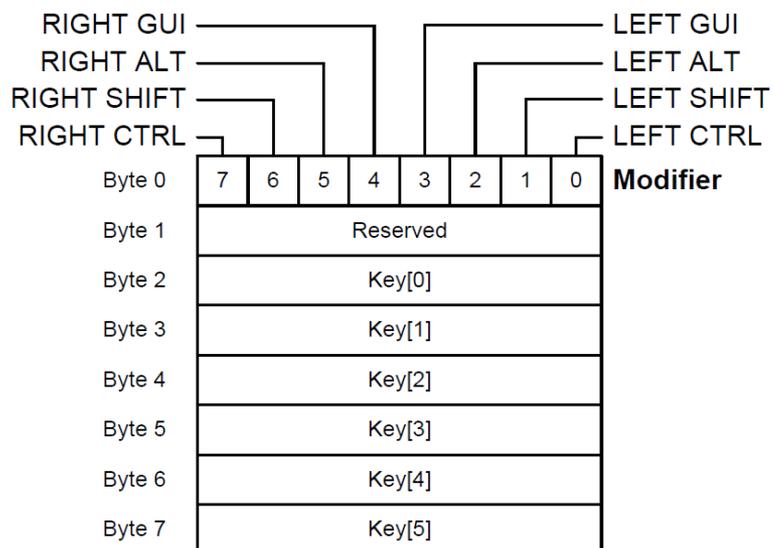
Field Name	Size	Offset
bLength	1	0
bDescriptorType	1	1
bEndpointAddress	1	2
bmAttributes	1	3
wMaxPacketSize	2	4
bInterval	1	6

#### Endpoint descriptor

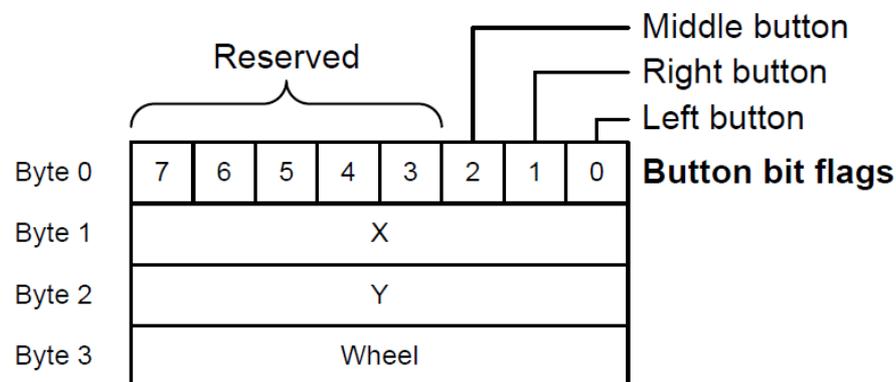
## 10.5.5 USB – programerski vidik

- Primeri podatkovne vsebine („Report format“) – HID profil

### Tipkovnica



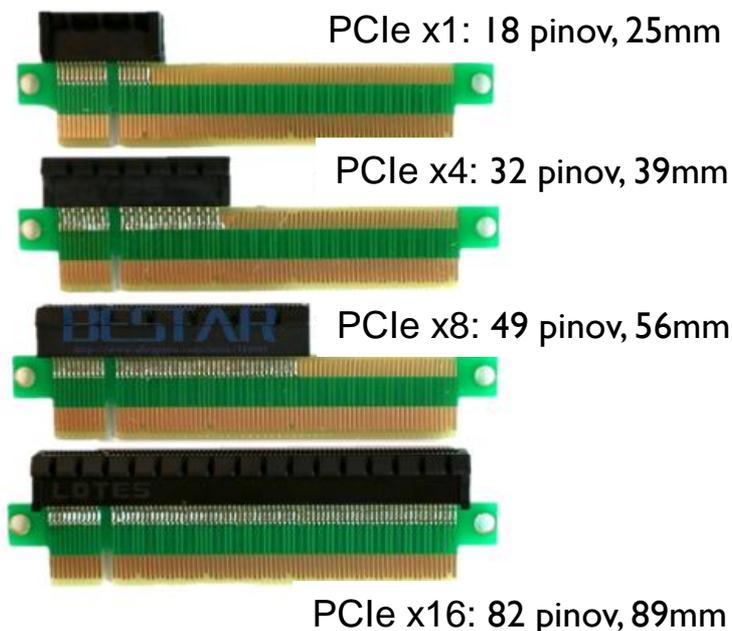
### Miška



## 10.6 PCI Express

### □ PCI Express ≡ PCIe ali PC-E (Peripheral Component Interconnect Express)

- Standarden tip povezav za notranje naprave računalnika.
- Številne matične plošče v računalniku imajo danes reže ali vtična mesta PCIe
- Fizične velikosti rež PCIe z različnim številom pinov so: PCIe x16, PCIe x8, PCIe x4, PCIe x1
- Fizične velikosti PCIe:

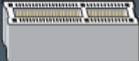
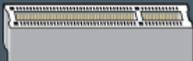
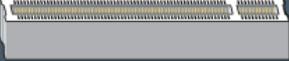


### Verzije PCIe:

Version	Introduced	Line code	Transfer rate per lane [Gbit/s]	×1
1.0	2003	8b/10b	2.5 GT/s	0.250 GB/s
2.0	2007	8b/10b	5.0 GT/s	0.500 GB/s
3.0	2010	128b/130b	8.0 GT/s	0.985 GB/s
4.0	2017	128b/130b	16.0 GT/s	1.969 GB/s
5.0	2019	128b/130b	32.0 GT/s	3.938 GB/s
6.0	2022	242B/256B	64.0 GT/s 32.0 GBd	7.563 GB/s

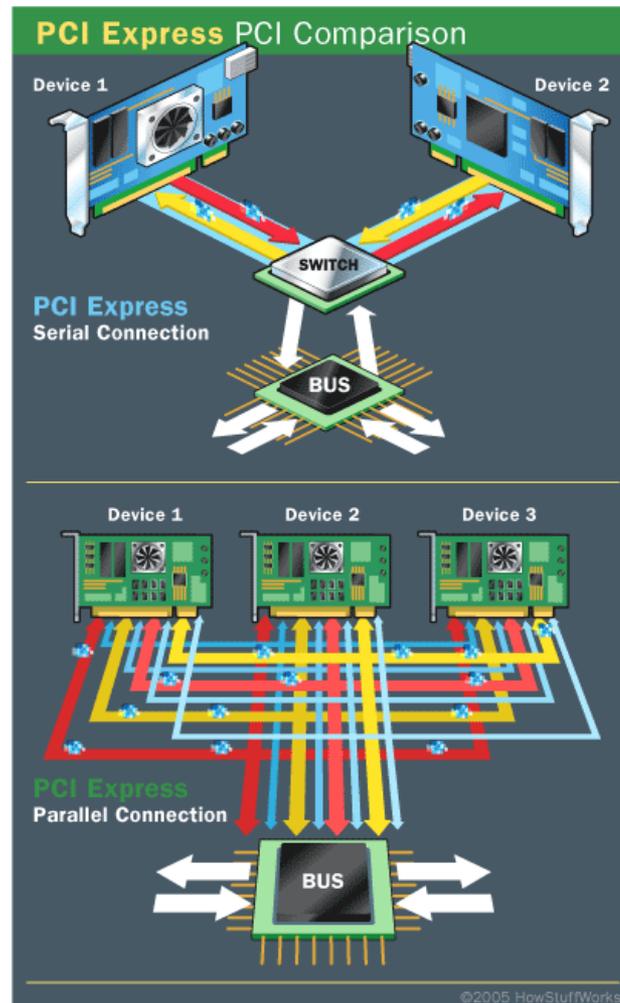
- ❑ Primer povezave naprav na PCIe, <https://computer.howstuffworks.com/pci-express-pictures.htm>



PCI Express Example Connectors	
<b>x1</b>	<p><b>BANDWIDTH</b>            Single direction: 2.5 Gbps/200 MBps            Dual Directions: 5 Gbps/400 MBps</p> 
<b>x4</b>	<p><b>BANDWIDTH</b>            Single direction: 10 Gbps/800 MBps            Dual Directions: 20 Gbps/1.6 GBps</p> 
<b>x8</b>	<p><b>BANDWIDTH</b>            Single direction: 20 Gbps/1.6 GBps            Dual Directions: 40 Gbps/3.2 GBps</p> 
<b>x16</b>	<p><b>BANDWIDTH</b>            Single direction: 40 Gbps/3.2 GBps            Dual Directions: 80 Gbps/6.4 GBps</p> 

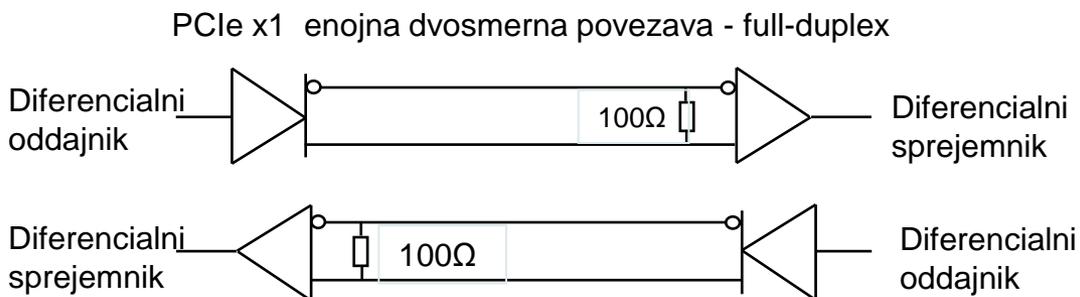
Source: IBM ©2005 HowStuffWorks

Serijska povezava naprav



Paralelna povezava naprav

- ❑ PCI Express je hitra serijska dvosmerna (ang. full-duplex) povezava točka-v-točko.
- ❑ Sestavljata jo dve simetrični (ang. differential) povezavi z LVDS (Low Voltage Differential Signaling), vsaka v eno smer (ang. full-duplex) z zaključitvenim uporom na sprejemni strani.



- ❑ Hitrost prenosa na enojni povezavi (PCIe x1) v eno smer je pri frekvenci ure 2,5GHz enaka 2,5Gb/s.

Zaradi kodiranja 8b/10b dobimo hitrost prenosa

$$2,5\text{Gb/s} * 8/10 = 2\text{Gb/s} = 250\text{MB/s}$$

- ❑ Dolžina povezave je lahko največ 50 cm.
- ❑ <http://www.interfacebus.com/PCI-Express-Bus-PCIe-Description.html>

□ PCIe protokol: Podatki se prenašajo v obliki okvirjev, ki jih sestavljajo:

- Start bajt (Start of Frame - SOF)
- Zaporedna številka (Sequence Number - SEQ #)
- Glava (Header)
- Podatki (Data)
- CRC - biti za detekcijo in korekcijo napak
- Konec okvirja (End of Frame - EOF)



- Dejanska hitrost prenosa na enojni povezavi (PCIe x1), če upoštevamo zgradbo okvirja z najdaljšim podatkovnim poljem 4096 B, je enaka:

$$2,5\text{Gb/s} * 8/10 * 4096 / 4124 = 1,9864\text{Gb/s} = 248,3\text{MB/s}$$

## □ PCI Express protokolski sklad

