## Prenos údajov

Rýchlosť

Synchronizácia

Formát a kódovanie správ

Spoľahlivosť

Protokol

Prenosový kanál

## Prenos údajov

Príjemca je pomalý, nestíha

Príjemca je obsadený

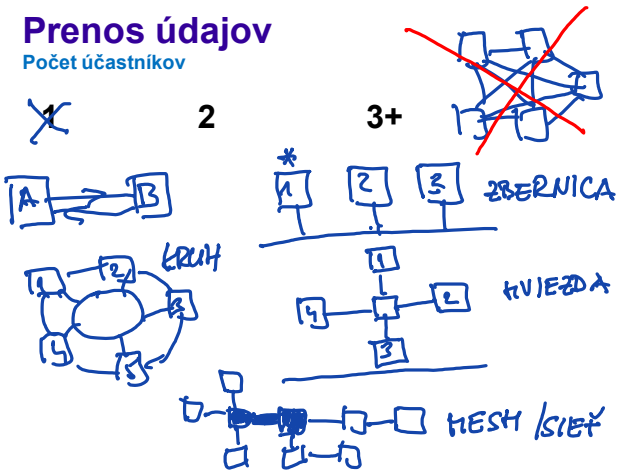
Príjemca nefunguje – nezdržuj a odpoj

Príjemca nestihol začiatok

Príjemcovi nedorazila správna správa (ako to vie?)

## Prenos údajov

Počet účastníkov



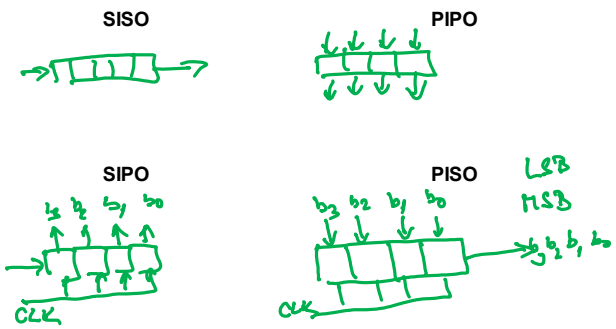

## Prenos údajov

Rýchlosť: tvar prenášaných údajov

Sériové – Informácia sa prenáša bit po bite.

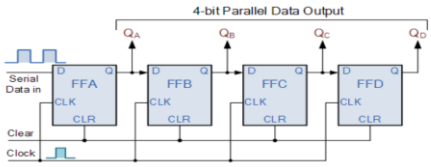
Paralelné – v jednom cykle zbernice sa prenáša viacbitové slovo (8, 16, 32, 64 - bitov).

## Sériovo – paralelné prevodníky



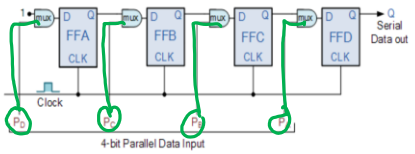
## Serial – Parallell

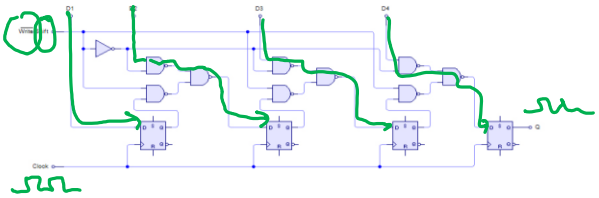
4-bit Serial-in to Parallel-out Shift Register

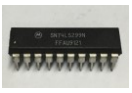



## Parallell – Serial

4-bit Parallel-in to Serial-out Shift Register



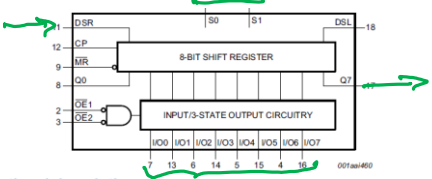


## 74HC299

8-bit universal shift register; 3-state

Rev. 4 — 26 February 2016 Product data sheet

Riadenie




DATA

### 6. Functional description

Table 3. Function table[1]

| Input | | | | Response |
|---|---|---|---|---|
| MR | S1 | S0 | CP | |
| L | X | X | X | asynchronous reset; Q0 to Q7 = LOW |
| H | H | H | ↑ | parallel load; I/On → Qn |
| H | L | H | ↑ | shift right; DSR → Q0, Q0 → Q1, etc. |
| H | H | L | ↑ | shift left; DSL → Q7, Q7 → Q6, etc. |
| H | L | L | X | hold |

[1] H = HIGH voltage level;

TI Home > Semiconductors > Logic > Flip-flop/latch/register > Shift register >

SN74ALS323 (ACTIVE) In English Alert me

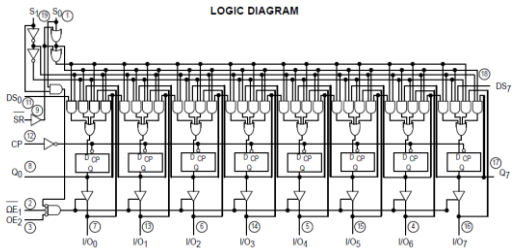
8-Bit Universal Shift/Storage Registers With Synchronous Clear And 3-State Outputs

DATASHEET

8-Bit Universal Shift/Storage Registers With Synchronous Clear And 3-St Outputs datasheet (Rev. A)

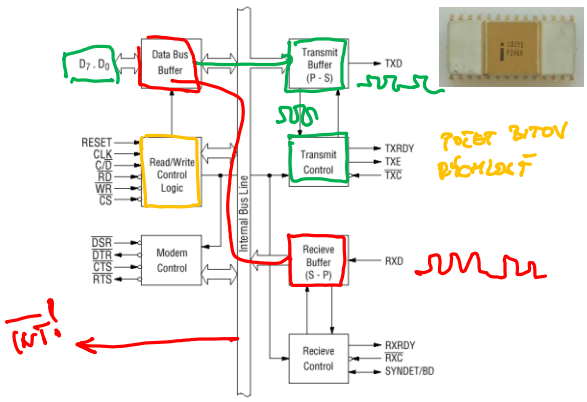
Download

LOGIC DIAGRAM




## I8521 – USART

**Universal Synchronous / Asynchronous Receiver / Transmitter**



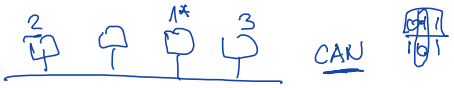



## Zbernice

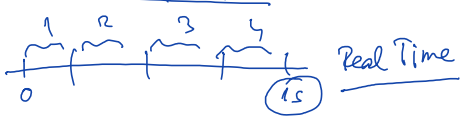
**Arbitrácia**

**Token**

**Priority**




**Time slots**




**Random access**

Eth CSMA/CD

# Zbernice

## Synchronizácia prenosu

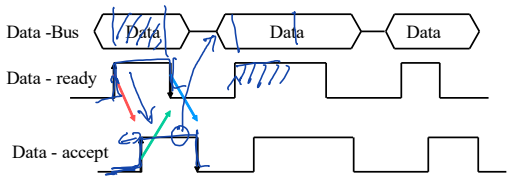
**Synchrónne:**
- prenos je synchronizovaný **hodinovým signálom**
- **rýchlejšie** ako asynchronné.
  Vhodné pre zariadenia pracujúce na **rovnakej frekvencii**

**Asynchrónne:**
- prenos je **riadený jednostranne**, napr. vysielačom.
- **handshaking** („výzva“ – „odpoveď“).
  **Pomalšie**, ale umožňujú spojiť **zariadenia pracujúce na rôznych frekvenciách**.

# Zbernice

## Synchronizácia prenosu

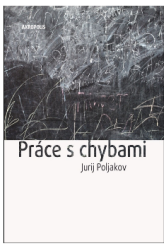
**Handshaking**




Asynchronné časovanie zbernice - handshaking

# Práca s chybami

## a: detekcia




Ako viem, že nastala chyba?

Všetky princípy vyžadujú istú **redundanciu**, t.j. **nadbytočnosť**

1. Timeout
2. Repetition codes
3. Parity bit
3. Checksum
4. Cyclic redundancy checks (CRCs)
5. Error-correcting codes

```
7 bitov   1     8 = 7 + parita
 data          even/pár   odd/nepár
0000000   0    00000000   10000000
1010001   3    11010001   01010001
1101001   4    01101001   11101001
1111111   7    11111111   01111111
```

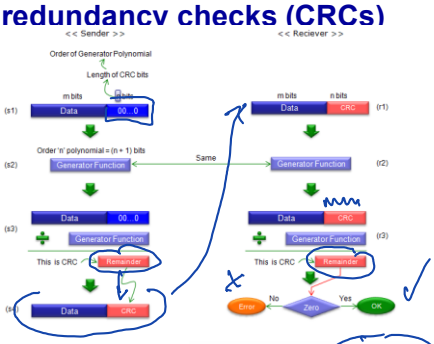
# Práca s chybami

## Checksum

| ASCII | Dec | Hex | Bin |
|---|---|---|---|
| A | 65 | 41 | 0100 0001 |
| H | 72 | 48 | 0100 1000 |
| O | 79 | 4F | 0100 1111 |
| J | 74 | 4A | 0100 1010 |
| | 290 | DE | 1101 1110 |

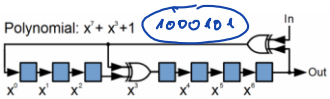
| Bin | Dec | Hex | ASCII |
|---|---|---|---|
| 0100 0001 | 65 | 41 | A |
| 0100 1000 | 72 | 48 | H |
| 0100 1110 | 78 | 4E | N |
| 0100 1010 | 74 | 4A | J |
| | | | |
| 1101 1111 | 289 | DF | |

## Cyclic redundancy checks (CRCs)



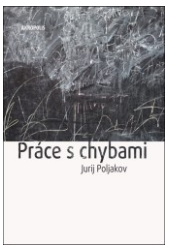

CRC-1 = parity bit
simple to implement in hardware,
easy to analyze mathematically,
good at detecting common errors
check value has a fixed length

Polynomial: $x^7 + x^3 + 1$ 10000101




# Práca s chybami

## b: korekcia




1. Zopakovanie celého prenosu

   + automaticky (nepríde ACK)
   + na vyžiadanie (príde RTS)

2. Zopakovanie **časti** prenosu
3. Samoopravné kódy

# Práca s chybami

## Error-correcting codes - Hamming

| $D_7$ | $D_6$ | $D_5$ | $P_4$ | $D_3$ | $P_2$ | $P_1$ | How to calculate parity bits |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 1 | 0 | This represents the full codeword |
| | | | | | | | |
| 1 | 1 | 0 | 0 | | | | P4 – Even parity of 1 1 0 = 0 |
| | | | | | | | |
| 1 | 1 | | | 1 | 1 | | P2 – Even parity of 1 1 1 = 1 |
| | | | | | | | |
| 1 | | 0 | | 1 | | 0 | P1 – Even parity of 1 0 1 = 0 |

**Detection** of 2 bit errors (assuming no correction is attempted);
**Correction** of single bit errors;
**Cost** of 3 bits added to a 4-bit message.

# Práca s chybami

## Error-correcting codes - Hamming

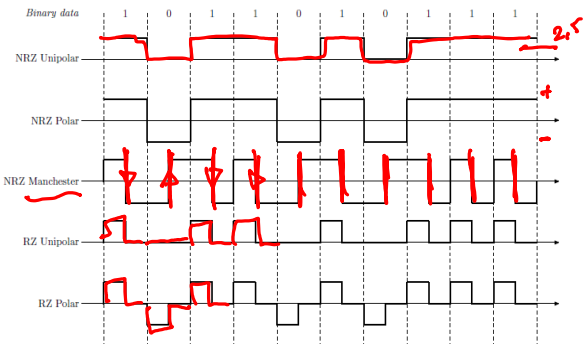
110 0110 111 0110

| $D_7$ | $D_6$ | $D_5$ | $P_4$ | $D_3$ | $P_2$ | $P_1$ | How to calculate parity bits |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | This represents the full codeword |
| | | | | | | | |
| 1 | 1 | 1 | 1 | | | | P4 – Even parity of 1 1 0 = 1 |
| | | | | | | | |
| 1 | 1 | | | 1 | 1 | | P2 – Even parity of 1 1 1 = 1 |
| | | | | | | | |
| 1 | | 1 | | 1 | | 1 | P1 – Even parity of 1 0 1 = 1 |

$A = P_4 + D_7 + D_6 + D_5$ 1
$B = P_2 + D_7 + D_6 + D_3$ 0
$C = P_1 + D_7 + D_5 + D_3$ 1

ABC is 101 in binary, which is 5 in decimal

# Kódovanie




21