UNIWERSYTET
O P O L S K I

INSTYTUT MATEMATYKI I INFORMATYKI

ul. Oleska 48, 45–052 Opole
tel. +48 77 452 72 05
fax +48 77 452 72 07
im@math.uni.opole.pl
www.math.uni.opole.pl

Sieci Komputerowe

Zasady komunikacji w sieciach komputerowych

dr Zbigniew Lipiński
Instytut Matematyki i Informatyki
ul. Oleska 48
50-204 Opole
zlipinski@math.uni.opole.pl

Zagadnienia

Klasyfikacja protokołów komunikacyjnych
Protokoły w modelu OSI
Nazwy pakietów w warstwach modelu OSI
Kapsułkowanie pakietów
Kapsułkowanie pakietów TCP/IP
Protokół Spanning Tree
Typy komunikacji: simplex, duplex
Typy komunikacji: broadcast, multicast, unicast
Komunikacja: synchroniczna, asynchroniczna
Kodowanie sygnałów typu NRZ (Non-Return to Zero), NRZI, Manchester
Zasada Nyquista

Podział protokołów komunikacyjnych

Podział protokołów komunikacyjnych ze względu na rodzaj sieci:

- protokoły sieci lokalnych (sieci LAN)
- protokoły sieci rozległych (sieci WAN).

Podział protokołów ze względu na pełnione funkcje:

- protokoły usług i aplikacji (ftp, telnet, http, POP, IMAP, SMTP, DNS),
- protokoły transportowe (TCP, UDP),
- protokoły służące do wykrywania błędów i informowania o błędach (ICMP),
- protokoły służące do wyznaczania tras przesyłania datagramów IP (protokoły routing'u).

Każdej warstwie modelu OSI można przypisać zbiór protokołów komunikacyjnych które służą do realizacji zadań (usług) danej warstwy.

Do realizacji zadania przez konkretną aplikację lub usługę wykorzystywane są protokoły kilku (lub wszystkich) warstw.

Protokół danej warstwy, po zrealizowaniu zadania w przekazuje dane do warstwy sąsiedniej (górnej lub dolnej) w celu realizowania kolejnych zadań przez odpowiednie protokoły.

Protokoły w modelu OSI

Przykład: Usługa serwera WWW. Przeglądarka internetowa żąda od serwera przesłania strony WWW.

W **warstwie Aplikacji** generowane zapytania i odpowiedzi są przesyłane w **protokole HTTP**.

W **warstwie Transportowej** HTTP wykorzystuje **protokół TCP**.

W **warstwie Sieci** zapytanie są kapsułkowane w **datagramy IP**.

W **warstwie Łącza Danych** datagramy IP kapsułkowane są w **ramki ethernetowe (**dołączany jest nagłówek ramki ethernetowej).

W **warstwie Fizycznej** bity danych kodowane są na bity służące do modulacji i przesyłane są w formie sygnału elektrycznego lub fali elektromagnetycznej . Np. w standardzie Ethernet sygnały **kodowane są w standardzie Manchester**.

W procesie komunikacji wykorzystywane są również protokoły:

- **ICMP,** do obsługi błędów w czasie transmisji,
- **ARP ,** do znalezienia fizycznego adresu serwera WWW (adresu MAC) ,
- **DNS,** do znalezienia adresu IP serwera WWW, jeżeli był użyty URL,
- **RIP**, **OSPF , inne,** do znalezienia i wybrania trasy przesyłania datagramów IP w sieci internetowej,
- **ppp**, **X.25**, **ISDN**, do komunikacji przez łącza telefoniczne, jeżeli był użyty modem,
- **SSL, TLS,** do komunikacji szyfrowanej,
- **Kerberos, Radius, inne,** do uwierzytelnienia klienta lub serwera WWW,
- **RPC**, **LPC,** do komunikacji między komponentami serwera WWW,
- do komunikacji z bazami danych, jeżeli serwer pobiera dane na stronę z bazy danych.

Protokoły warstwy Aplikacji

Aplikacja	Protokół	Opis
telnet	TELNET, Network Terminal Protocol	Zdalne zarządzanie hostami.
ftp	FTP, File Transfer Protocol	Protokół do przesyłania plików.
Klient, serwer usługi WWW	HTTP, HyperText Transport Protocol	Protokół do przesyłania plików tekstowych.
Klient, serwer usługi DHCP	DHCP, Dynamic Host Configuration Protocol	Protokół służący do dynamicznego przydzielania hostom parametrów TCP/IP (adres IP, maska, adres IP domyślnej bramy, adresy IP serwerów DNS, inne).
Klient, serwer usługi DNS	DNS, Domain Name System	Przesyła informacje o adresie IP na podstawie dnsowej nazwy hosta, urla.
Klient, serwer poczty elektronicznej	SMTP, POP, IMAP	Protokoły służące do przesyłania wiadomości elektronicznych.
Monitor, agent SNMP	SNMP, Simple Network Management Protocol	Zarządzanie i monitorowanie urządzeń w sieci.
	NFS, Network File System	Network File System służy do udostępniania zasobów między hostami unixowymi, RFC 1813.
System Active Directory	DAP, LDAP, Lightweight Directory Access Protocol	Umożliwia dostęp do usług katalogowych X.500.

Protokoły warstwy Sesji

W warstwie sieci określane są standardy zarządzania sesją (budowa, rozłącznie sesji, parametry sesji).

Przykłady protokołów warstwy sesji:

- **RPC**, Remote Porcedure Call. Protokół służący do komunikacji między aplikacjami lub komponentami aplikacji znajdującymi na różnych hostach.
- protokół **ASP** (protokół sesji AppleTalk).
- system **X-Windows.**
- protokół obsługujący język zapytań **SQL**.

Protokoły warstwy Transportowej

W warstwie Transportowej wykorzystywane są protokoły:

- **TCP**, (ang.) Transmision Control Protocol
- **UDP**, (ang.) User Datagram Protocol
- **SPX**, (ang.) Sequenced Packet Exchange.

Protokoły warstwy Aplikacji korzystające z TCP:

FTP, TELNET, HTTP, POP, SMTP, NFS v3, DNS, LDAP.

Protokoły warstwy Aplikacji korzystające z UDP:

DHCP, DNS, SNMP, NFS v2, TFTP.

Protokoły warstwy Sieci

- **IPv4**, (ang.) Internet Protocol wersja 4
- **IPv6**, (ang.) Internet Protocol wersja 6
- **IPX**, (ang.) Internetwork Packet Exchange (Novell).

Protokoły IP, IPX służą do transmisji pakietów (datagramów) w sieci Internet.

- **ICMP**, (ang.) Internet Control Message Protocol.

Obsługuje poprawność przesyłanych datagramów przez protokół IP.

- **ARP**, (ang.) Address Resolution Protocol.

Wyznacza adresy MAC kart sieciowych na podstawie adresu IP hostów.

Protokoły routingu dzieli się ze względu obszar działania na:

- protokoły wewnętrzne (Interior Gateway Protocols)
- protokoły zewnętrzne (Exterior Gateway Protocols).

Protokoły routingu dzieli się ze względu sposób budowy tablic routingu na:

- protokoły wektorowe,
- protokoły stanu łącza,
- protokoły hybrydowe.

Protokoły warstwy Sieci

Protokoły routingu dzieli się ze względu obszar działania na:

- protokoły wewnętrzne (Interior Gateway Protocols)
- protokoły zewnętrzne (Exterior Gateway Protocols).

Protokoły wewnętrzne (IGP) wykorzystywane są przez routery znajdujące się wewnątrz dużych podsieci (***autonomicznych podsieci***).

Protokoły zewnętrzne (EGP) wykorzystywane są przez routery łączące '***autonomiczne podsieci***'.

Protokoły wewnętrzne

RIP, (ang.) Routing Information Protocol (protokół wektora odległości).

IGRP, (ang.) Interior Gateway Routing Protocol.

EIGRP, (ang.) Enchanced IGRP (protokół wektora odległości).

OSPF, (ang.) Open Shortest Path First (protokół stanu łącza).

IS-SI, (ang.) Intermediate System-to- Intermediate System (protokół stanu łącza).

Protokoły zewnętrzne

EGP, (ang.) Exterior Gateway Protocol

BGP, (ang.) Border Gateway Protocol, RFC 1163, następca EGP (protokół wektora odległości).

Protokoły warstwy Łącza Danych

Sieci LAN:

- Ethernet
- Token Ring
- FDDI.

Sieci WAN:

- SLIP, (ang.) Serial Line Internet Protocol
- PPP, (ang.) Point-to-Point Protocol
- PPTP, (ang.) Point-to-Point Tunneling Protocol
- X.25
- Frame Relay
- HDLC, (ang.) High-level Data Link Control protocol
- ATM, (ang.) Asynchronous Transfer Mode.

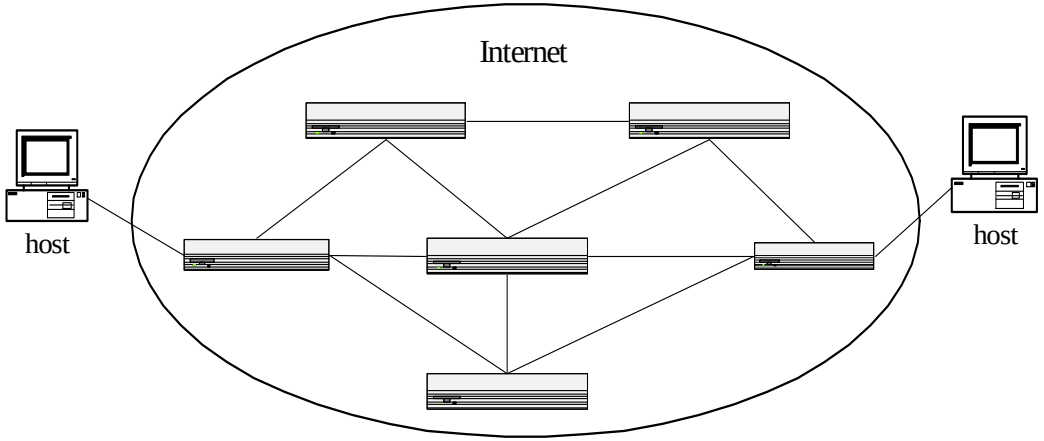
Protokół Spanning Tree

Spanning Tree Protocol, Standard IEEE 8021.d.

Urządzenia takie, jak switche budują tablice tras przesyłania pakietów.

Aby nie dochodziło do powstawania tras z pętlami IEEE opracowało protokół Spanning Tree.

W sieciach Ethernet switche wymieniają ramki Spanning Tree, zwane ramkami BPDU, (ang.) Bridge Protocol Data Units w celu blokowania tras które tworzą pętle.



Protokol Spannig Tree. Trasa z petlami.

Uporządkowanie bajtów

Systemy operacyjne, urządzenia transmisyjne w sieci mogą stosować dwa rodzaje uporządkowania przesyłanych bajtów.

Systemy Unixowe: **Big Endian**.
System Windows: **Little Endian**.

Bajty przesyłane w **sieciach IP** maja uporządkowanie **Big Endian**.

W formacie Big Endian bajty sa uporządkowane od lewej do prawej (**->**).
W formacie Little Endian bajty uporządkowane są od prawej do lewej (**<-**).

Przykład:
Liczba 258 zapisana w układzie dwójkowym ma postać

100000010 => 00000001 00000010

Liczba 258 może być zapisana jest przy pomocy 2 bajtów w formacie:

Big Endian: 00000001 00000010

Little Endian: 00000010 00000001

Nazwy pakietów w warstwach modelu OSI

Warstwa	TCP	UDP
Aplikacji, Prezentacji, Sesji	strumien	wiadomosc
Transportowa	segment	pakiet
Sieci	datagram	
Łącza Danych	ramka	
Fizyczna	bit	

Mówimy:

- strumień HTTP, POP, SMTP,
- wiadomość DHCP, DNS,
- segment TCP,
- pakiet UDP,
- datagram IP, datagram ICMP, datagram ARP, datargram RIP,
- ramka ethernetowa, ramka token ring, ramka fddi.

Kapsułkowanie pakietów

```
# ----- naglowek ethernetowy  -----
Packet 10 arrived at 12:35:32.17
Packet size = 69 bytes
Destination = 8:0:20:85:55:4b, Sun
Source      = 0:e0:f7:22:3f:d9, CISCO Router
Ethertype = 0800 (IP)

# ----- naglowek IP  -----
Version = 4
Header length = 20 bytes
Type of service = 0x00 (normal)
Total length = 55 bytes
Identification = 43775
Flags = 0x0
   .0.. .... = may fragment
   ..0. .... = last fragment
Fragment offset = 0 bytes
Time to live = 252 seconds/hops
Protocol = 6 (TCP)
Header checksum = 3a45
Source address = 152.148.13.32,  server.math.edu.pl
Destination address = 155.181.171.109, client
No options
```

```
# ----- naglowek TCP  -----
Source port = 23
Destination port = 36869
Sequence number = 2486243368
Acknowledgement number = 1913975089
Data offset = 20 bytes
Flags = 0x18
..0. .... = No urgent pointer
...1 .... = Acknowledgement
.... 1... = Push
.... .0.. = No reset
.... ..0. = No Syn
.... ...0 = No Fin
Window = 8760
Checksum = 0xc10c
Urgent pointer = 0
No options

# ----- dane aplikacji TELNET:   -----
  dane
```

Kapsułkowanie pakietów

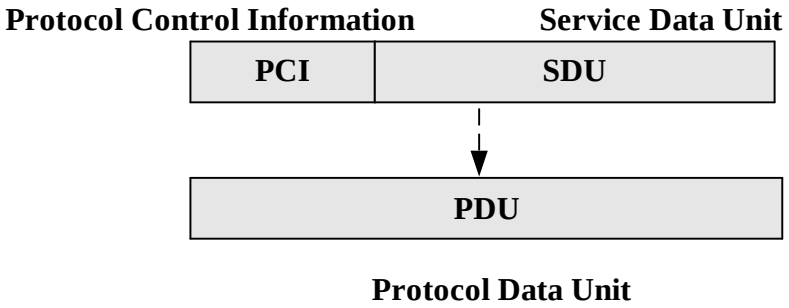
Kapsułkowanie, (ang.) encapsulation

nazywany jest proces tworzenia pakietu PDU z pakietów SDU i PCI.

W procesie przechodzenia z warstwy wyższej do niższej pakietu danych zwanego **Service Data Unit** (SDU), zostaje dołączony nagłówek **PCI** (**Protocol Control Information**).

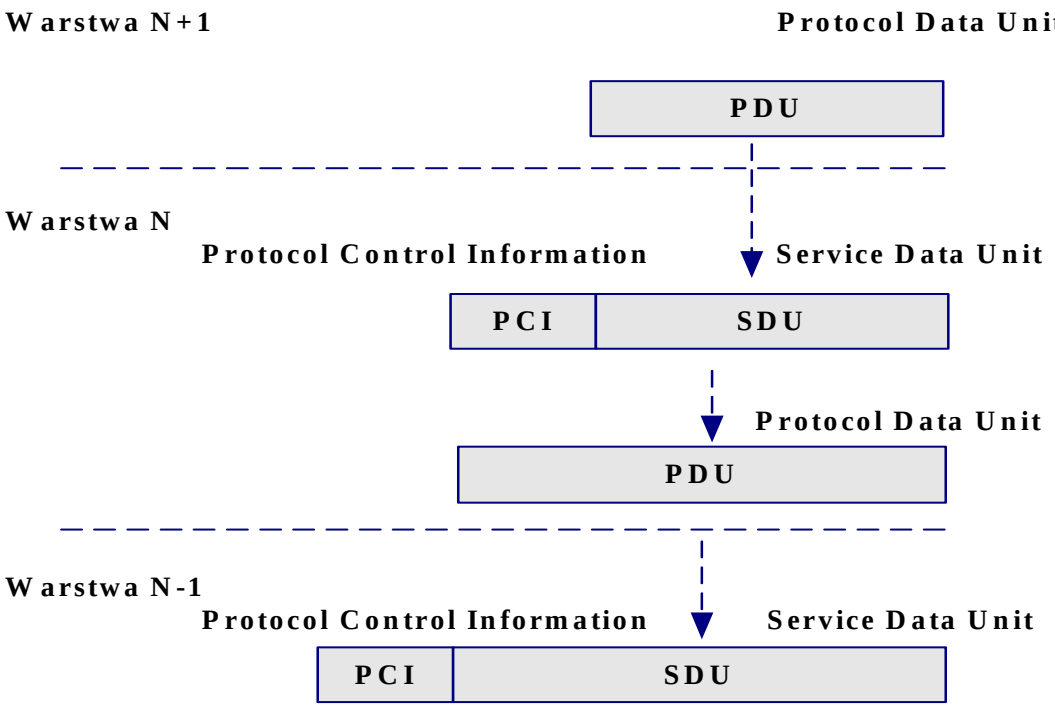
Utworzony pakiet **PCI + SDU** nazywany jest w niższej warstwie pakietem PDU (**Protocol Data Unit)**.

Kapsułkowanie SDU, tzn. dodanie PCI do SDU i utworzenie PDU.



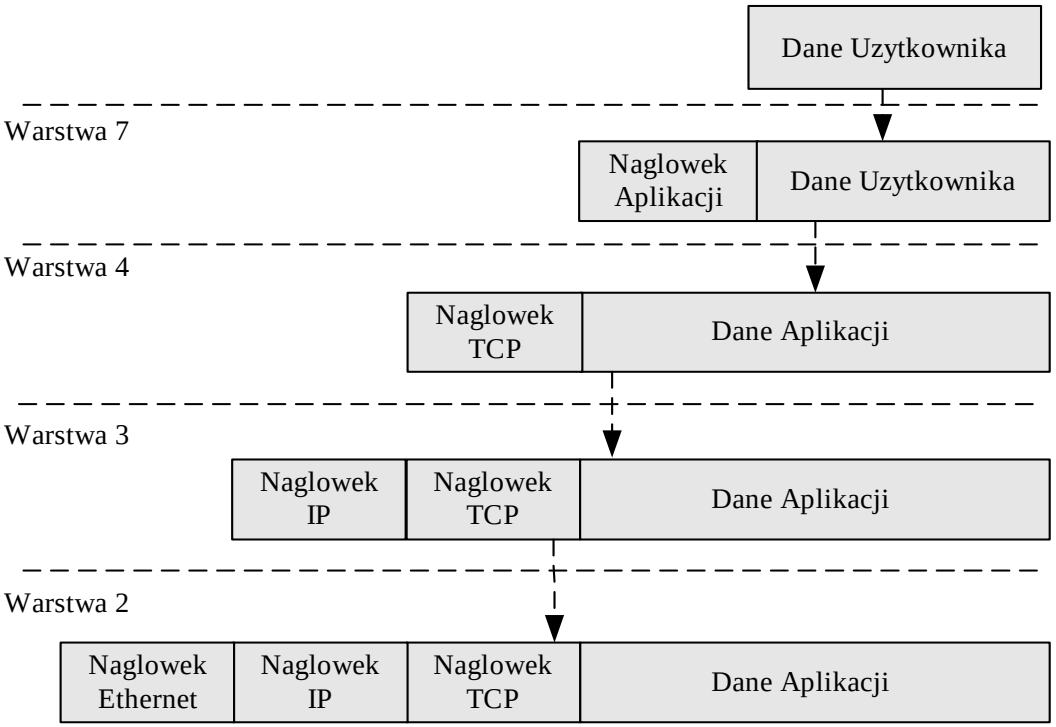
Kapsulkowanie danych w modelu OSI

Kapsułkowanie pakietów: PCI + SDU = PDU



Kapsulkowanie danych w modelu OSI

Kapsułkowanie strumienia danych TCP/IP



Kapsulkowanie danych w modelu OSI

Komunikacja w trybie: simplex, duplex

Komunikacja w **trybie simplex** oznacza, ze dane sa wysyłane tylko w jednym kierunku.

Komunikacja w **trybie duplex** oznacza, ze sa wysyłane w obu kierunkach.

Jeżeli dane sa wysyłane w obu kierunkach niejednocześnie (asynchronicznie) to komunikacja odbywa się w **trybie half duplex**.

Jeżeli dane sa wysyłane w obu kierunkach jednocześnie (synchronicznie) to komunikacja odbywa się w **trybie full duplex**.

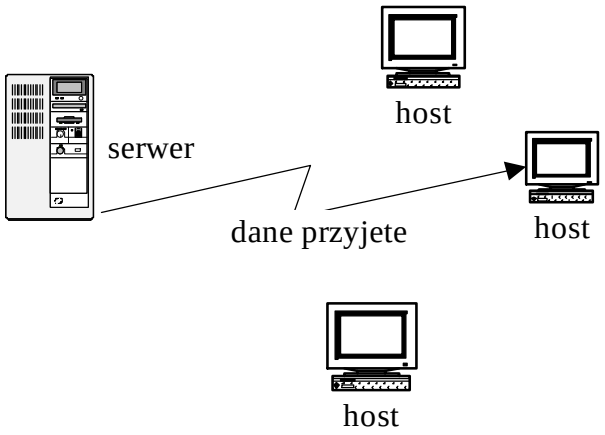
Przykład: rozmowa telefoniczna obywa się w trybie full duplex.

Typy komunikacji: unicast, multicast, broadcast

Transmisja odbywa się w **trybie unicast** jeżeli wysłane dane docierają tylko do jednego odbiorcy (jest tylko jeden nadawca i jeden odbiorca).

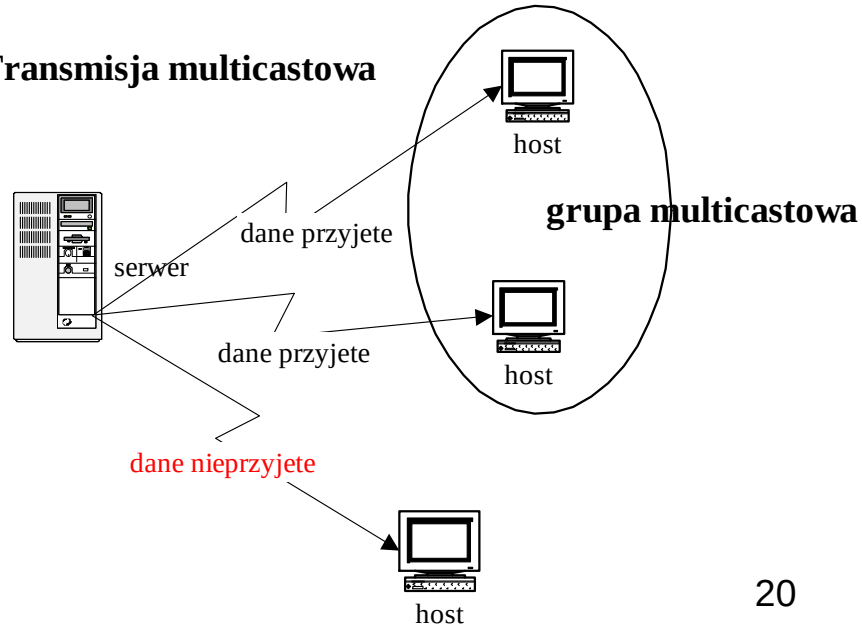
Transmisje w trybie unicast stosują np. protokoły http, smtp, ftp, telnet.

Transmisja unicastowa



Transmisja odbywa się w **trybie multicast'owym** jeśli dane wysyłane z jednego lub kilku węzłów są odbierane przez określoną grupę odbiorców danej sieci (jest jeden lub więcej nadawców i określona grupa odbiorców -**grupa multicastowa**).

Transmisja multicastowa



Typy komunikacji: unicast, multicast, broadcast

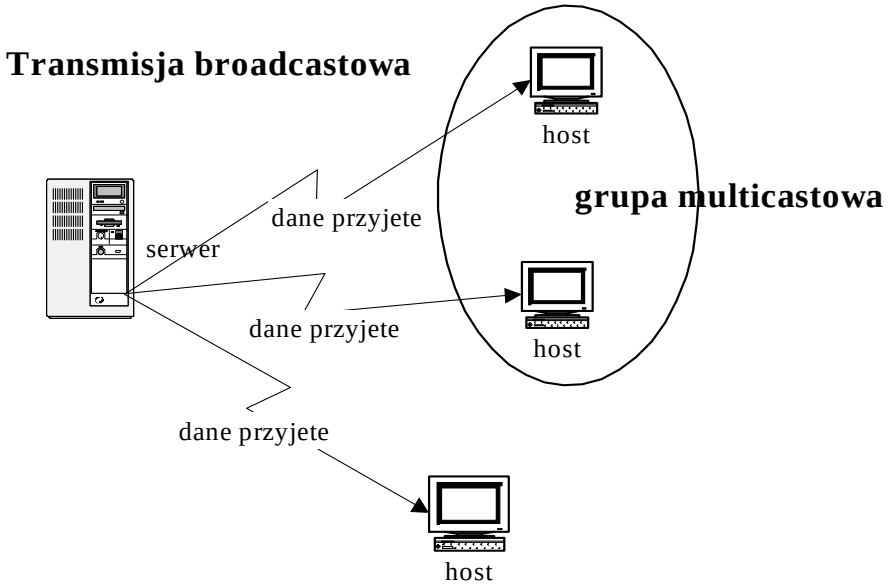
Transmisja odbywa się w **trybie broadcast'owym** jeśli dane wysłane z jednego węzła sieci docierają do wszystkich węzłów danej sieci (jest jeden nadawca i wszyscy odbiorcy).

Transmisje w trybie broadcast'owym stosują np. protokoły ARP, IP.

W sieciach TCP/IP adres broadcastowy jest to ostatni adres z zakresu adresów IP.

Adres IP 0.0.0.0 oznacza wszystkie węzły w sieci.

W warstwie łącza danych adresem broadcastowym jest adres MAC FF:FF:FF:FF:FF:FF



Transmisja asynchroniczna

Transmisja asynchroniczna to sposób wymiany danych w którym nadawca i odbiorca **nie synchronizują** zegarów.

W komunikacja asynchronicznej zegary odbiorcy i nadawcy są niezależne tzn. kolejne wysyłane bajty mogą być wysyłane z dowolnym opóźnieniem.

Nadawca i odbiorca ustalają:

- czas transmisji jednego bitu,
- wielkość transmitowanego znaku (ilość bitów w znaku).

W komunikacji asynchronicznej dane (ciąg bitów) wysyłane są w określonym formacie i o określonej wielkości.

Każdy znak poprzedzany jest bitem początkowym 0 (ang. start bit) i zakończony 1-2 bitami końca (ang. stop bits).

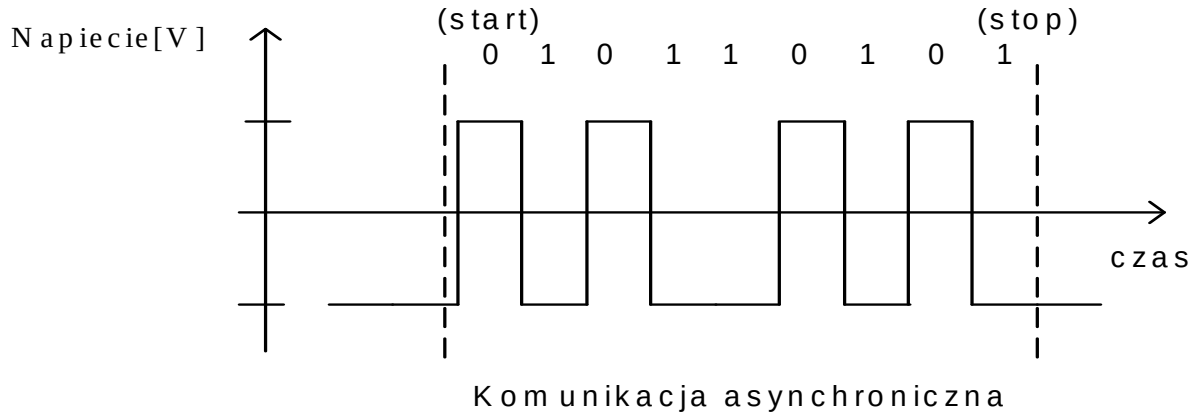
Dodawany jest bit parzystości w celu zabezpieczenia integralności przesyłanego znaku.

Przykładem standardu wykorzystującego transmisję asynchroniczną jest standard **RS-232C.**

Transmisja asynchroniczna

Struktura danych wysyłanych w komunikacji asynchronicznej:

- 1 bit początkowy (start bit), Wartość: 0.
- 8 bitów danych, w tym bit parzystości (jeśli jest).
- 1 lub 2 bity końca (1-2 stop bity), Wartość: 1.



Komunikacja asynchroniczna

Transmisja synchroniczna

Transmisja **synchroniczna** to sposób wymiany danych w którym nadawca i odbiorca uzgadniają parametry transmisji, synchronizują zegary.

Transmisja synchroniczna nazywana jest transmisją z taktowaniem.

Przykłady standardów wykorzystujących transmisję synchroniczną:

- USA: Synchronous Optical Network (SONET)

  Europa: Synchronous Digital Hierarchy (SDH)
- X.25 (komunikacja synchroniczna, full-duplex, do 2Mb/s).
- DDS (digital data service), szybkość 56 kbps.

W transmisji synchronicznej nadawca i odbiorca synchronizują zegary, transmisja zachodzi ze stałą szybkością a odbiorca zlicza przesyłane bity na podstawie czasu transmisji.

Przykład:

'**Preambuła**' - pierwsze 32 bity w ramce ethernetowej, zawiera ciąg bitów 1010…10, służących do synchronizacji zegara odbiorcy z zegarem nadawcy.

Transmisja synchroniczna

Sposób synchronizacji zegarów zależy od metody kodowania bitów.

Termin **kodowanie** oznacza sposób zamiany strumienia bitów na sygnały elektryczne.

Przykładowe rodzaje kodowań:

- kodowanie NRZ (Non-Return to Zero),
- kodowanie typu Manchester,
- kodowanie NRZI (Non-Return to Zero Inverted),
- 4B/5B,
- 8B/10B,
- MLT-3, Multi-Level Transition 3,
- 4D-PAM5, four-Dimensional and 5 level Pulse Amplitude Modulation,
- 8B/6T.

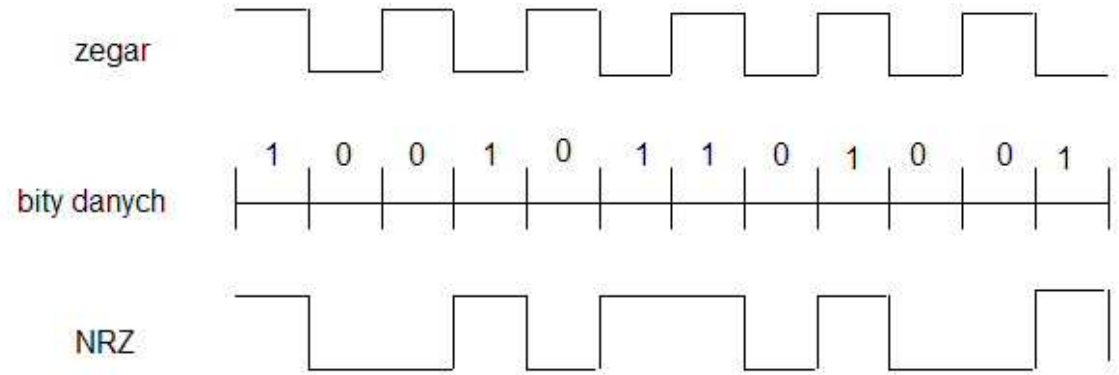
Kodowanie NRZ (Non-Return to Zero)

Kodowanie NRZ, (ang.) Non-return to Zero, kodowanie bez powrotu do zera.

W **kodowaniu NRZ**

- bit 1 reprezentowany jest niskimi napięciem
- bit 0 wysokim napięciem.

Odbiorca rejestruje zmianę napięcia sygnału elektrycznego tylko wtedy, gdy nadawca prześle ciąg bitów **10** lub **01**.
Metoda kodowania NZR stosowana jest w transmisji synchronicznej i asynchronicznej.



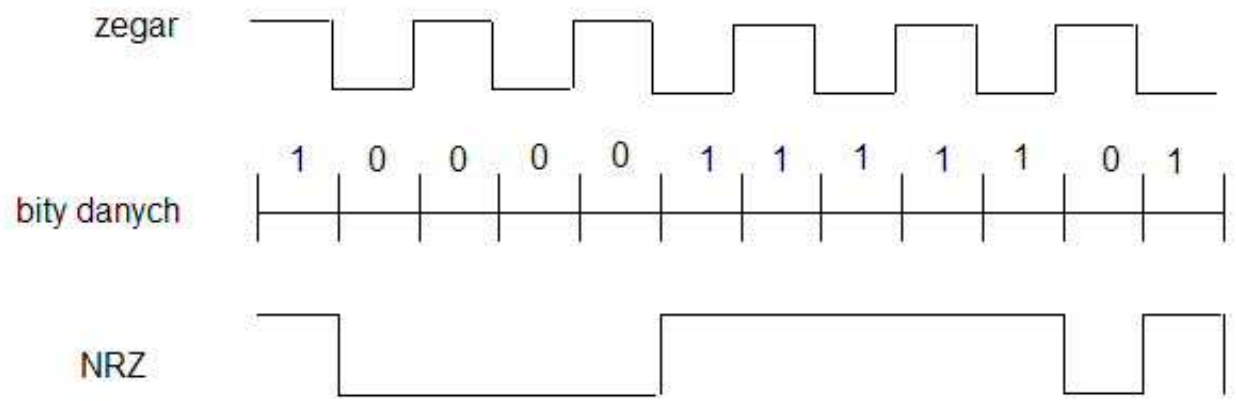
Kodowanie NRZ (Non-Return to Zero)

Jeżeli w transmisji synchronicznej stosowane jest kodowanie NRZ, to pojawia się problem **rozsynchronizowania zegarów**.

Rozsynchronizowanie zegarów może zajść gdy w transmitowanym strumieniu bitów jest ciąg samych 1 lub 0.

Jeżeli przez dłuższy czas nadawany jest ciąg samych 0 lub 1 to odbiorca nie rejestruje zmian napięcia.

Jeżeli dojśdzie do rozsynchronizowania zegarów odbiorca nie może określić ilości przesłanych bitów.

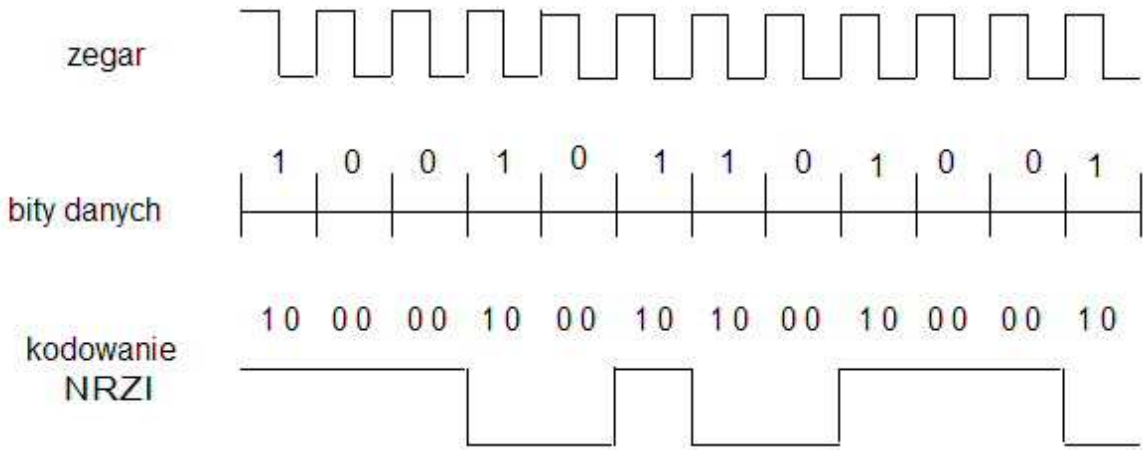


Kodowanie NRZI (Non-Return to Zero Inverted)

Kodowanie NRZI, (ang.) **Nor-Return to Zero Inverted** jest mieszaną metodą kodowania.

Zasada kodowania ciągu bitów o wartości 1 jest taka, jak w kodowaniu Manchester, czyli 1 -> 10.
Zasada kodowania ciągu bitów o wartości 0 jest taka, jak w NRZ, czyli 0 -> 00

1 - oznacza zmianę napięcia, **0 -** brak zmiany. Zmiany napięcia następują przy przejściach **0->1**.



Kodowanie typu Manchester

W kodowaniu **typu Manchester** przesyłany bit (0 lub 1) zamieniany jest na parę bitów 01 lub 10.

Taki sposób kodowania gwarantuje, ze w przesyłanym strumieniu bitów brak jest ciągu samych 1 lub 0 (max. może wystąpić ciąg 11 lub 00).

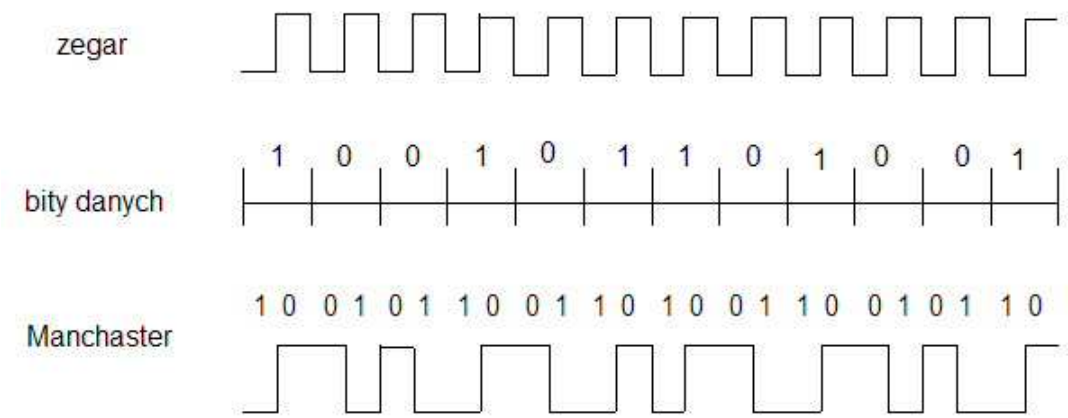
Zasada kodowania typu Manchester.

Wartość bitu	**Wartość wysłana**
logiczne 0	0 1
logiczne 1	1 0

Metoda kodowania Manchester stosowana jest w standardzie Ethernet, RS-232 (transmisja asynchroniczna).

Przykład: Transmisja strumienia bitów 01111001.

Ciąg bitów	kodowany jest jako
0 1 1 1 1 0 0 1	01 10 10 10 10 01 01 10



Kodowanie 4B/5B

Kodowanie 4B/5B + kodowanie MLT-3 stosowane w standardzie **100BaseTX**.
Kodowanie 4B/5B + NRZI stosowane w standardzie **100BaseFX**.

Zamienia ciąg 4 bitów na ciąg 5-bitowy.
Istnieją 32 kody 5-bitowe.
16 kodów służy do reprezentowania danych:

4B/5B Encoding

4-bit	5-bit	4-bit	5-bit
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Pozostałe kody służą do sterownia transmisją.
11111 – idle,
11000 – start of stream delimiter 1 (pierwsze 5 bitów znacznika początku),
10001 – start of stream delimiter 2 (drugie 5 bitów znacznika początku),
01101 – end of stream delimiter 1 (pierwsze 5 bitów znacznika końca),
00111 – end of stream delimiter 2 (drugie 5 znacznika końca),
00100 – błąd transmisji.

Pozostałe kody nie są oznaczone.

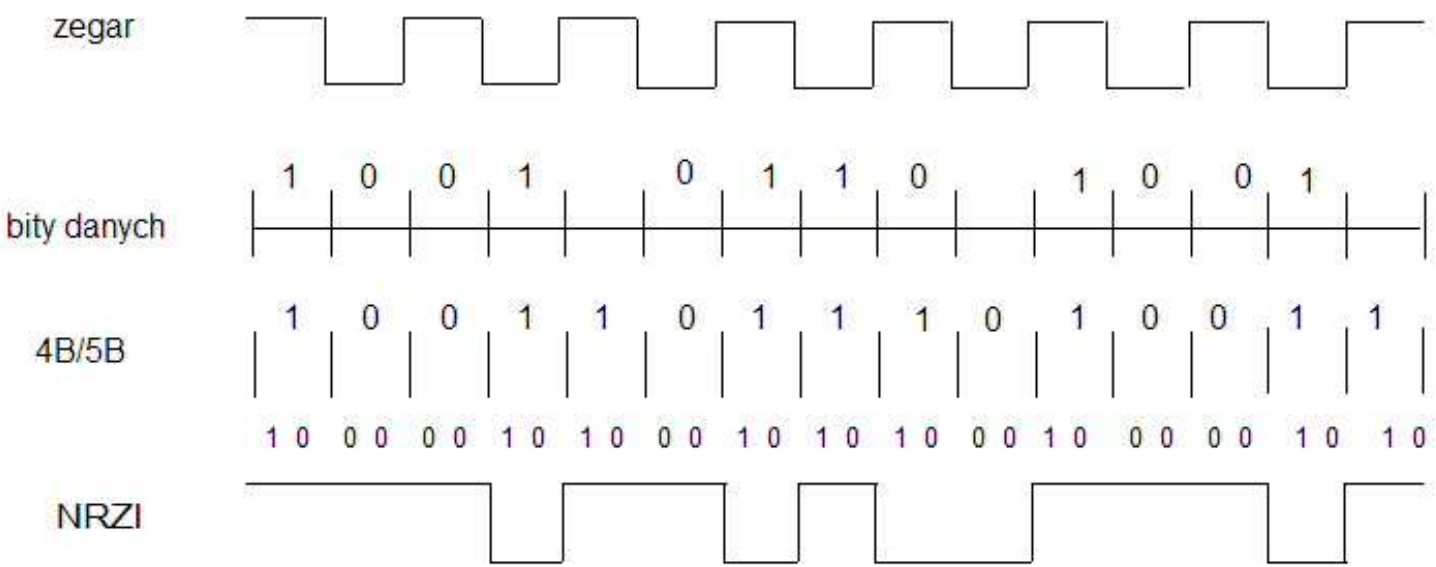
Kodowanie 4B/5B

4 bitowe słowa

1001 0110 1001

są kodowane na 5 bitowe słowa

10011 01110 10011.



Kodowanie 8B/10B

Kodowanie 8B/10B + 4D-PAM5 stosowane w standardzie **1000Base-T**.

Kodowanie 8B/10B + NRZ stosowane w standardach **1000Base-SX,1000Base-LX, 1000Base-CX**.

Zamienia 8 bitów na 10-bitowy ciąg.

Struktura kodu:

• każdy 10 bitowy ciąg musi zawierać 5 jedynek, 5 zer; 4 jedynki, 6 zer albo 6 jedynek, 4 zera,

• 3 najbardziej znaczące bity (y) są zamieniane na 4 bity (kodowanie 3B/4B),

• 5 pozostałych bitów (x) jest zamienianych na 6 bitów (kodowanie 5B/6B),

• każda grupa danych jest oznaczana D.x.y, x=0-31, y=0-7,

np. D.0.0 = 000 00000, D.6.2 = 010 00110, D.30.6 = 110 11110,

• istnieje 12 oktetów kodów sterujących, oznaczane K.x.y,

• w celu zapewnienia równowagi napięciowej stosuje się tzw. *running disparity (RD)*, czyli bieżącą zmianę parzystości.

Tabele konwersji zawierają przypisania grup kodowych 8B/10B o parzystości dodatniej (więcej „1”, „0”), ujemnej i neutralnej.

W trakcie transmisji, po uzyskaniu synchronizacji kolejno przesyłane grupy kodowe powinny mieć parzystość wyznaczoną przez poprzednią grupę.

Początek transmisji ma parzystość -1 (RD = -1).

Kodowanie 8B/10B

12 grup kodów specjalnych K.x.y.

28 - 11100, 23 – 10111, 27 – 11011, 29 – 11101, 30 - 11110

7 – 111, 6 – 110, 5 – 101, 4 – 100, 3 – 11, 2 - 10

Nazwa	**A B C D E F G H**	**a b c d e i f g h j**
K.28.0	0 0 0 1 1 1 0 0	0 0 1 1 1 1 0 1 0 0
K.28.1	0 0 1 1 1 1 0 0	0 0 1 1 1 1 1 0 0 1
K.28.2	0 1 0 1 1 1 0 0	0 0 1 1 1 1 0 1 0 1
K.28.3	0 1 1 1 1 1 0 0	0 0 1 1 1 1 0 0 1 1
K.28.4	1 0 0 1 1 0 0 0	0 0 1 1 1 1 0 0 1 0
K.28.5	1 0 1 1 1 1 0 0	0 0 1 1 1 1 1 0 1 0
K.28.6	1 1 0 1 1 1 00	0 0 1 1 1 1 0 1 1 0
K.28.7	1 1 1 1 1 1 0 0	0 0 1 1 1 1 1 0 0 0
K.23.7	1 1 1 1 0 1 1 1	1 1 1 0 1 0 1 0 0 0
K.27.7	1 1 1 1 1 0 1 1	1 1 0 1 1 0 1 0 0 0
K.29.7	1 1 1 1 1 1 0 1	1 0 1 1 1 0 1 0 0 0
K.30.7	1 1 1 1 1 1 1 0	0 1 1 1 1 0 1 0 0 0

Kodowanie 4B/5B + MLT-3 (Multi-Level Transition 3)

Trzy etapy kodowania danych w standardzie 100BaseTX.

1. 4B/5B-NRZI

Tablica konwersji kodów 4B/5B-NRZI

dane	kod	dane	kod
0000	10100	1000	11100
0001	01110	1001	11101
0010	11000	1010	11011
0011	11001	1011	11010
0100	01100	1100	10011
0101	01101	1101	10010
0110	01011	1110	10111
0111	01010	1111	10110

2. Zakodowane bity są XOR-owane z wygenerowanym pseudolosowo ciągiem bitów.

3. Kodowanie MLT-3.

MLT-3 są określone 3 poziomy napięcia –V, 0V, +V.

Bit 0 w ciągu bitów nie oznacza zmiany napięcia.

Bit 1 w ciągu bitów oznacza zmianę napięcia jeżeli:

Poprzednie przejście było 0V -> -V, następuje przejście -V -> 0V

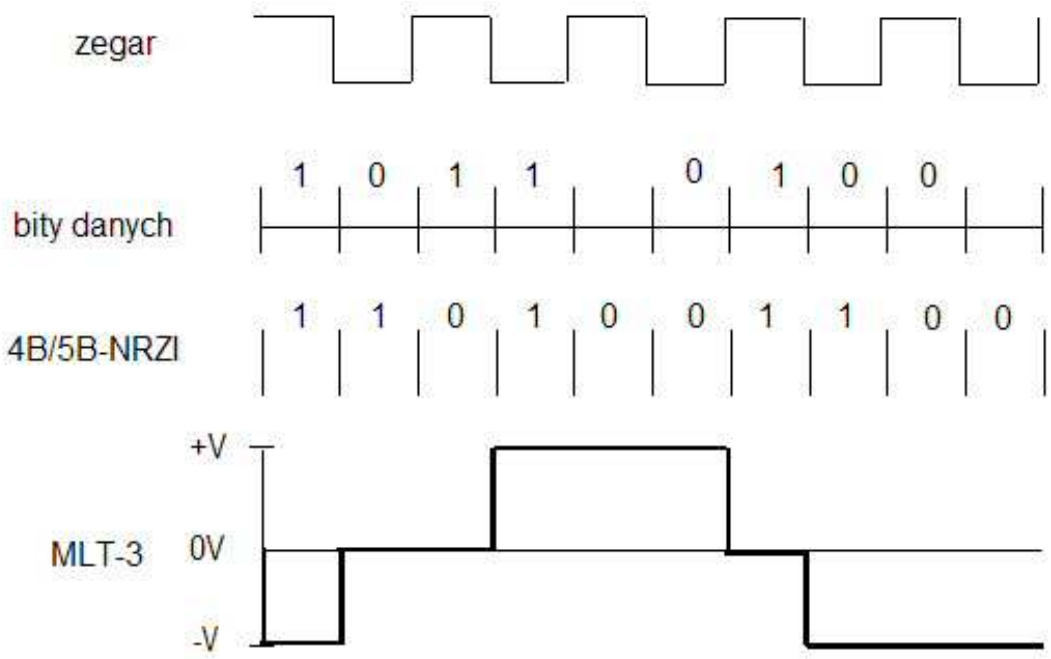
Poprzednie przejście było -V -> 0V, następuje przejście 0V -> +V

Poprzednie przejście było 0V -> +V, następuje przejście +V -> 0V

Poprzednie przejście było +V -> 0V, następuje przejście 0V -> -V

Kodowanie MLT-3 (Multi-Level Transition 3)

Przykład: Kodowanie MLT-3 bez operacji XOR.
1011 0100 -> 11010 01100.



Kodowanie 4D-PAM5

Kodowanie 8B/10B + kodowanie 4D-PAM5 zastosowane w standardzie 1000Base-T.

Kodowanie 4D-PAM5, (ang) four-Dimensional and 5 level Pulse Amplitude Modulation.

- Dane są transmitowane w 4 parach przewodów w skrętce.
- Do modulowania sygnału stosowanych jest 5 wartości napięcia:-1V, -½ V, 0 V, +½ V, +1 V.
- Algorytm zawiera 625 możliwych kodów.
- Stosowane są tylko kody o dużej wzajemnej odległości w przestrzeni stanów w celu zmniejszenia ilości błędów.

Kodowanie 8B/6T

Kodowanie 8B/6T stosowane w standardzie **100BaseT4**.

Algorytm kodowania zamienia ciąg 8 bitów (znaków 2-stanowych) na ciąg 6 kodów 3-stanowych.
Istnieje 729 możliwych kodów, 256 reprezentuje dane, 9 oznacza kody startu, stopu i kody sterujące.

Przykładowe kody:

Tablica konwersji kodów 8B6T		
dane (Hex)	**Binarnie**	**kod 8B6T**
00	00000000	+-00+-
01	00000001	0+-+-0
...	...	...
0E	00001110	-+0-0+
...	...	...
FE	11111110	-+0+00
FF	11111111	+0-+00

Szybkość transmisji. Szybkość modulacji.

Szybkości transmisji, (ang.) bit rate

oznacza szybkość przesyłanych bitów w jednostce czasu.

Szybkości transmisji mierzona jest w bitach/sekundę, [b/s], (ang.) bps.

Szybkości modulacji, (ang.) baud rate

oznacza szybkość zmian sygnału elektrycznego w jednostce czasu.

Szybkości modulacji mierzona jest w **bod'ach.**

Jeżeli każda zmiana sygnału elektrycznego wiąże się przesłaniem jednego bitu to ilość bodów jest równa ilości przesłanych bitów, tzn. **1 b/s = 1 baud.**

W kodowaniu **typu Manchester** szybkości modulacji jest **2 razy większa od szybkości transmisji**.

Transmisja cyfrowa - transmisja analogowa

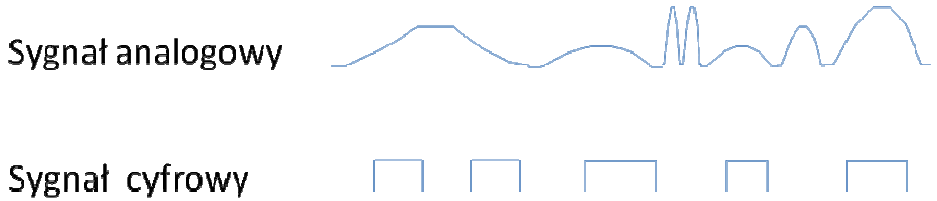
Analogowe/cyfrowe: dane, sygnał, transmisja.

Za pomocą sygnału cyfrowego można przesyłać dane cyfrowe (ethernetowa karta sieciowa) lub dane analogowe (zapis danych na płycie audio cd).

Za pomocą sygnału analogowego można transmitować dane cyfrowe (np. modem, fale radiowe).

Sygnał analogowy – to sygnał o ciągłych parametrach.

Sygnał cyfrowy – ciąg bitów (0 lub 1) przesyłany jako sygnał o dyskretnych wartościach.



W **transmisji cyfrowej** strumień bitów zamieniany jest ciąg impulsów elektrycznych (np. 1 - niskie napięcie, 0 wysokie napięcie).

W **transmisji analogowej** dane przesyłane są za pomocą elektromagnetycznej fali nośnej o określonej częstotliwości.

Dane są kodowane za pomocą odpowiedniej modulacji fali nośnej.

$$F(t) = A \sin(\omega t + \varphi)$$

A - amplituda fali

ω - częstotliwość fali

φ - faza fali.

Amplituda, częstotliwość, faza fali **A(t), ω(t), φ(t)** mogą zależeć nieliniowo od czasu.

Modulator - zamienia sygnał cyfrowy na analogowy.

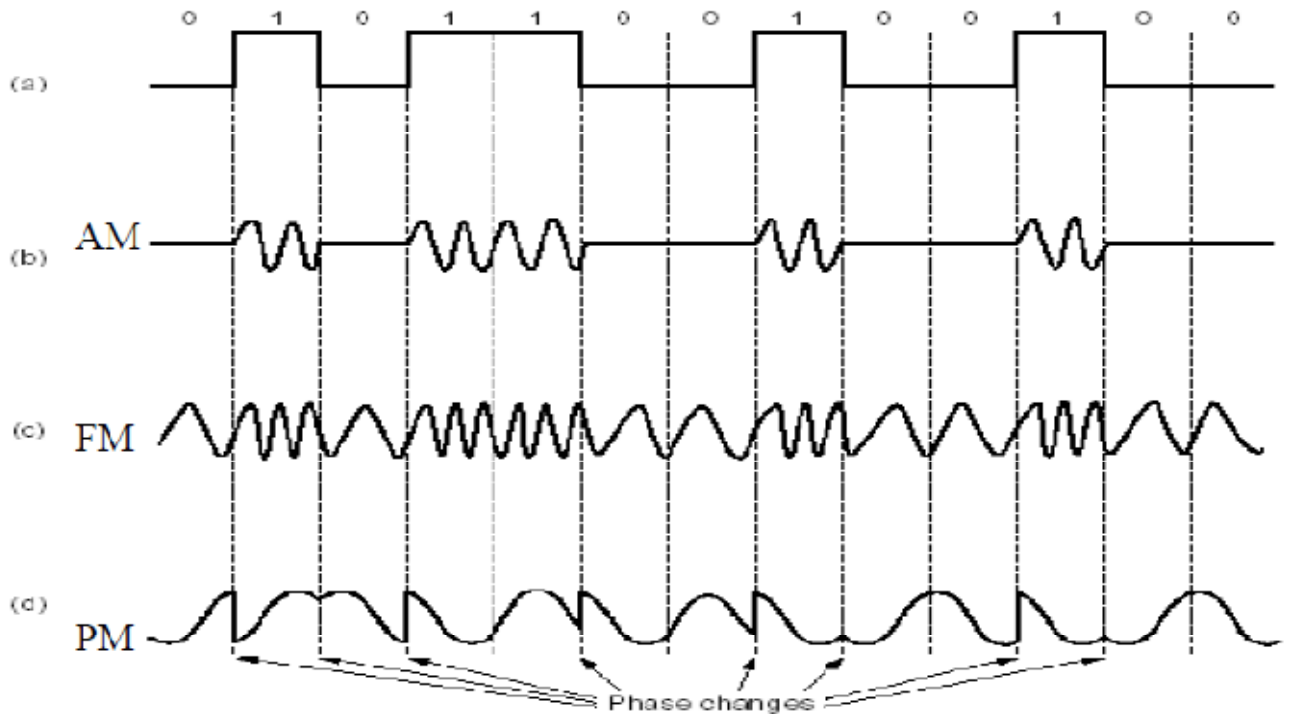
Demodulator - zamienia analogowy na sygnał cyfrowy.

Transmisja cyfrowa - transmisja analogowa

Modulacja – zamiana sygnału cyfrowego na analogowy.

Typy modulacji:

- modulacja amplitudy (AM, ang, amplitude modulation), np. 1 wysoka amplituda, 0 - niska amplituda / brak fali.
- modulacja częstotliwości (FM, ang. frequency modulation), np. 1 wysoka częstotliwość, 0 - niska częstotliwość.
- modulacja fazy (ang. phase shift modulation), np. 1 - zwykła faza , 0 - faza przesunięta.



Szerokość pasma, przepustowość

W transmisji analogowej **szerokość pasma** określa zakres częstotliwości stosowanej do transmisji danych, jednostka herz [Hz].

W transmisji cyfrowej **szerokość pasma** jest to ilość danych, które można przesłać przez sieć w określonym czasie.

Jednostka bity/sekundę [b/s].

Przepustowość w transmisji cyfrowej oznacza ilość bitów przesyłanych w ciągu jednostki czasu.

Przepustowość oznacza rzeczywistą szerokość pasma (def. Cisco).

Przykład. Sieć Ethernet ma przepustowość 10Mb/s.

Zasada Nyquista

Szerokość pasma B

maks. częstotliwość z jaką urządzenie może zmieniać sygnał.

Jednostka - herc [Hz].

Zasada Nyquista (1924 r.)

określa zależność między **szerokością pasma** urządzenia transmisyjnego a maksymalną **liczbą bitów którą to urządzenie może przesłać w ciągu 1s**.

Zasada określa teoretyczne ograniczenie na maksymalną szybkość przesyłania danych.

Jeżeli urządzenie stosuje:

K - możliwych wartości napięcia

B - jest szerokością pasma

To maksymalna szybkość przesyłania danych D jest równa

$$D = 2\,B \log_2 K$$ **[bity/sekunda]**

Przykład:

Standard RS-232 stosuje dwie wartości napięcia +15V, -15V.

K=2, log_2 (2) =1 => D=2*B

Zasada Shannona

W 1948 **Shannon** uwzględnił wpływ szumów na szybkość przesyłania danych

$$C = B \log_2(1 + \frac{S}{N})$$ **[bity/sekunda]**

C- efektywne ograniczenie pojemności kanału [bit/s]

S - średnia moc kanału

N - średnia moc szumów

B - szerokość pasma urządzenia.

Jednostka stosowana to 10*log_10(S/N) [decybel] = [dB].

Oznacznie: **S//N.**

Przykład:

S/N =1000, 10*log_10(1000) = 10*3 = 30 [dB]

S//N = 30.

Przyklad: Maks., teoretyczna szybkość transmisji przez łącza telefoniczne.

Dla łączy telefonicznych

S//N = 10*log_10(S/N) = 30dB;

B = 3000 Hz;

=> C = 3000 * log_2(1 + 1000) = 30 kb/s (mniej niż 56kb/s).

Wydajność sieci

Wydajność sieci charakteryzuje się:

- opóźnieniem
- przepływnością.

Rodzaje opóźnień:

- opóźnienie propagacji
- opóźnienie przełączania
- opóźnienie dostępu (np. CSMA/CD)
- opóźnienie kolejkowania.

Przepływność urządzeń transmisyjnych to **szerokość pasma**.