

## 3rd Slide Set

# Computer Networks

Prof. Dr. Christian Baun

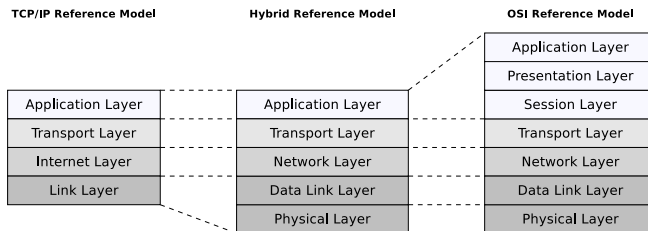
Frankfurt University of Applied Sciences  
(1971–2014: Fachhochschule Frankfurt am Main)  
Faculty of Computer Science and Engineering  
[christianbaun@fb2.fra-uas.de](mailto:christianbaun@fb2.fra-uas.de)

# Learning Objectives of this Slide Set

- Physical layer (part 2)
  - Devices of the Physical Layer
    - Repeaters and Hubs
  - Impact on the collision domain
  - Encoding data with line codes
    - Non-Return-To-Zero (NRZ)
    - Non-Return-To-Zero, Inverted (NRZI)
    - Multilevel Transmission Encoding - 3 Levels (MLT-3)
    - Return-to-zero (RZ)
    - Unipolar RZ encoding
    - Alternate Mark Inversion (AMI code) = Bipolar encoding
    - B8ZS
    - Manchester code
    - Manchester II code
    - Differential Manchester encoding
    - 4B5B
    - 6B6B
    - 8B10B
    - 8B6T

# Physical Layer

- Functions of the Physical Layer
  - Bit transmission on wired or wireless transmission paths
  - Provides network technologies (e.g. Ethernet, WLAN, ...)
  - Transmission Media
  - Frames from the Data Link Layer are encoded with line codes into signals



- Devices: Repeater, Hub (Multiport Repeater)
- Protocols: Ethernet, Token Ring, WLAN, Bluetooth,...

Image Source: StarTech

- 
- The diagram illustrates a bus network topology. It features three main components: Host A, a Repeater/Hub, and Host B. Host A and Host B are represented as vertical stacks of seven nodes each, numbered 1 to 7 from bottom to top. Node 1 is shaded gray, while nodes 2 through 7 are white. The Repeater/Hub is a single gray node labeled '1' positioned between the two host stacks. Below Host A and the Repeater/Hub is a horizontal line labeled 'Medium'. Below the Repeater/Hub and Host B is another horizontal line labeled 'Medium'. This setup shows a shared communication medium where all nodes can potentially receive data.



Image Source: [www.planet.com.tw](http://www.planet.com.tw)

- **Hubs** are Repeaters with  $> 2$  interfaces
- Forwards all incoming signals to all its output ports
- Repeaters and Hubs have no physical or logical network addresses
  - Reason: They just forward the received signals
  - They operate transparent and communicate only on Physical Layer



(Repeater)



(Hub)

## Topology of Hubs

- **Physical topology:** Star network because of the cabling
- **Logical topology:** Bus network, because equal to a long cable, where all network devices are connected with, a Hub forwards incoming signals to all other interfaces
  - For this reason, each terminal device, which is connected to a Hub, can receive and analyze the entire traffic, passing the Hub
- Advantages of Hubs over the physical bus network topology:
  - Better reliability, because the failure of individual cable segments does not result in a complete network failure
  - Adding or removing network devices does not cause network interruptions
- All nodes in the network that are connected to a Hub, are located in the same **collision domain**

Prof. Dr. Christian Baun – 3rd Slide Set Computer Networks – Frankfurt University of Applied Sciences – WS2021 7/41

- The media access protocols are part of the Data Link Layer ( $\Rightarrow$  slide set 6)

---

- Repeaters and Hubs increase the collision domain
  - Reason: These devices can not analyze signals
    - They only forward signals
- **Repeater**
  - In a network with CSMA/CD, all segments connected with Repeaters belong to the same collision domain
- **Hubs**
  - All ports (and thus all computers that are connected to a Hub) belong in a network with CSMA/CD to the same collision domain

Beyond a certain number of network devices, no data transmissions are possible any more, because all transmissions are destroyed by collisions





- 5-4-3 rule**  $\Leftarrow$  applies only for Repeaters and Hubs!

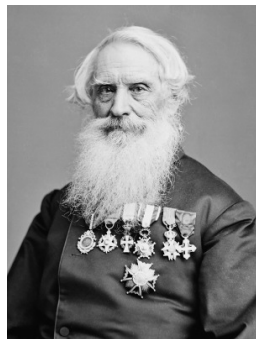
- Prof. Dr. Christian Baun – 3rd Slide Set Computer Networks – Frankfurt University of Applied Sciences – WS2021 10/41

# Encoding Data

Image source: Wikipedia

- Efficient data encoding is important not only since the rise of computer networks
- An example for an efficient encoding is the **Morse Code**, invented by Samuel Morse from 1838

A	. —	M	— —	Y	— . — —
B	— . . .	N	— .	Z	— — . .
C	— . — .	O	— — —	1	. — — — —
D	— . .	P	. — — .	2	. . — — —
E	.	Q	— — . —	3	. . . — —
F	. . — .	R	. — .	4	. . . . —
G	— — .	S	. . .	5	. . . . .
H	. . . .	T	—	6	— . . . .
I	. .	U	. . —	7	— — . . .
J	. — — —	V	. . . —	8	— — — . .
K	— . —	W	. — —	9	— — — — .
L	. — . .	X	— . . —	0	— — — — —



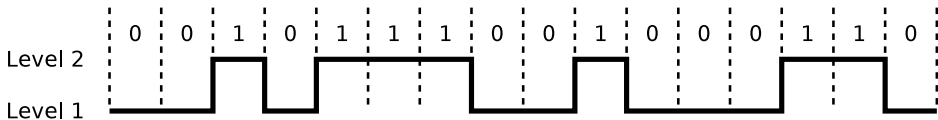
Samuel Morse (1791 – 1872)

# Encoding Data in Computer Networks

- The encoding is called **line code** in this context, and specifies how signals are transmitted on the transmission medium
- Specific signal sequences correspond with bit sequences in the data stream
- Computer networks must implement these operations:
  - ① Conversion of binary data ( $\Rightarrow$  binary numbers) into signals (encoding)
  - ② Transmission of signals from sender to receiver
  - ③ Conversion of the signals back into the binary data (decoding)
- Different ways exist to encode bits into signals
- The most simple way of representing logical 0 and 1 is by using different voltage levels
  - This line code is called **Non-Return-To-Zero (NRZ)**
    - Example: A logical 0 can be encoded by one signal level (e.g. 0 V) and a logical 1 by a different one (e.g. 5 V)

## Non-Return-to-Zero (NRZ)

- This line code encodes...
  - a logical 0 bit with physical signal level 1 (low value)
  - a logical 1 bit with physical signal level 2 (high value)



Implemented by the serial CAN (Controller Area Network) bus system, which was developed by Bosch in the 1980s for connecting control devices in cars

- When transmitting a long series of logical 0 bits or logical 1 bits, the physical signal level does not change
  - This results in 2 problems:
    - 1 **Baseline Wander**
    - 2 **Clock Recovery**

## Non-Return-to-Zero (NRZ) – Baseline Wander

- Problem: Shift of the average signal level (**Baseline Wander**) when using NRZ
- The receiver distinguishes the physical signal levels by using the average signal level of a certain number of received signals
  - Signals below the average signal level, interprets the receiver as logical 0 bit
  - Signals above the average signal level, interprets the receiver as logical 1 bit
- When transmitting long sequences of logical 0 or 1 bits, the average signal level may shift so much, making it difficult to detect a change of the physical signal

### Sources

- Steve Zdancewic. 2004. <http://www.cis.upenn.edu/~cse331/Fall104/Lectures/CSE331-3.pdf>
- Charles Spurgeon, Joann Zimmerman. *Ethernet: The Definitive Guide*. O'Reilly (2014)

### Detailed source, which explains baseline wander from the electrical engineering perspective

- Maxim Integrated (2008). *NRZ Bandwidth – LF Cutoff and Baseline Wander*. <http://pdfserv.maximintegrated.com/en/an/AN1738.pdf>

# Avoid Baseline Wander

- In order to prevent **Baseline Wander**, when using a line code with 2 physical signal levels, the usage of both signal levels must be **distributed equally**
  - Therefore, the data to be transmitted must be encoded in a way, that the signal levels occur equally often
    - The data must be **scrambled**
- If a network technology uses 3 or 5 physical signal levels, the average signal level must match the middle signal level over the time

- You can imagine the local clock as an internal signal, switching from low to high. A low/high pair is a clock cycle



## Avoid the Problem of Clock Recovery

- One option: Using a **separate line, which transmits just the clock**

A network technology with a separate signal line just for the clock is the serial bus system I<sup>2</sup>C (Inter-Integrated Circuit)

But like comparable systems this bus system is only suited for local application and cannot be used to span large distances

- In computer networks, a separate signal line just for the clock is **not practical** because of the cabling effort
  - Instead, it is recommended to **increase the number of signal level changes** to enable the clock recovery from the data stream

The next slides present several line codes, which all...

- (more or less successful) try to solve the challenges of baseline wander and/or clock recovery
- must consider the limitations of the transmission medium used
  - Fiber-optic cables and wireless transmissions via infrared and laser provide just 2 physical signal levels
  - Copper cables and wireless transmissions via radio waves provide  $\geq 2$  physical signal levels

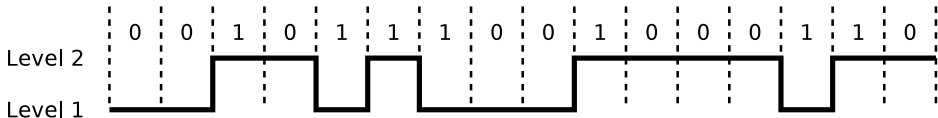
# I will not discuss all Lines Codes during Class!



- It makes no sense to show you all these line codes during class
  - It is not the best way to memorize the most important information about line codes
  - And it is boring
- Best practice is to do the line code related exercises from exercise sheet 2 by using the course material
  - I will assist you during the exercise sessions

## Non-Return-to-Zero, Inverted (NRZI)

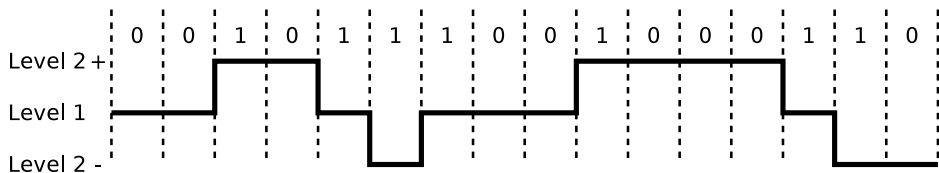
- Transmit a logical 1 bit  $\Rightarrow$  signal level change at the beginning of the clock
- Transmit a logical 0 bit  $\Rightarrow$  signal level remains unchanged for an entire clock
- **Clock recovery is impossible** for series of logical 0 bits
- The usage of the signal levels is not equally distributed
  - Therefore, **baseline wander can occur**



Implemented by Ethernet 100BASE-FX (Multi-mode fiber) and FDDI

## Multilevel Transmission Encoding - 3 Levels (MLT-3)

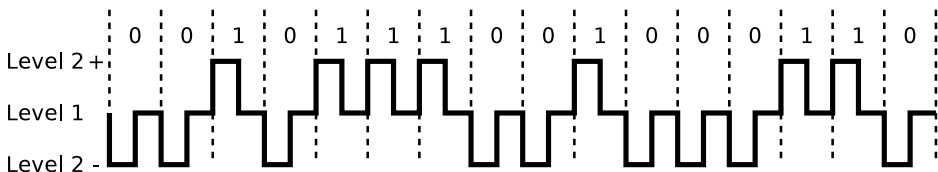
- This line code uses 3 signal levels +, 0 and -
  - If a logical 0 bit is transmitted, no signal level change takes place
  - A logical 1 bit is alternating encoded, according to the sequence [+ , 0 , - , 0]
- Just as for NRZI, the **clock recovery problem exists** with series of logical 0 bits and **baseline wander can occur**



Implemented by Ethernet 100BASE-TX

## Return-to-Zero (RZ)

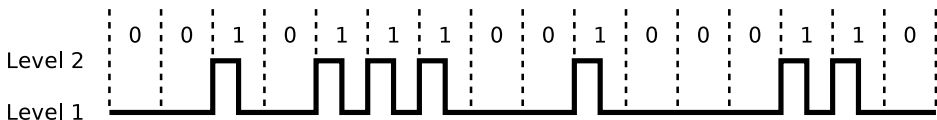
- RZ uses 3 signal levels too
  - Transmit a logical 1 bit  $\Rightarrow$  high signal level is transmitted for a half clock and then the signal level returns to the middle signal level
  - Transmit a logical 0 bit  $\Rightarrow$  low signal level is transmitted for a half clock and then the signal level returns to the middle signal level



- Advantage: Each transmitted bit causes a signal level change
  - Enables the receiver to do the **clock recovery** (synchronization)
- Drawbacks:
  - Requires **double as much bandwidth** compared with NRZ
  - **Baseline wander can occur** for series of logical 0 bits or 1 bits

# Unipolar RZ Encoding

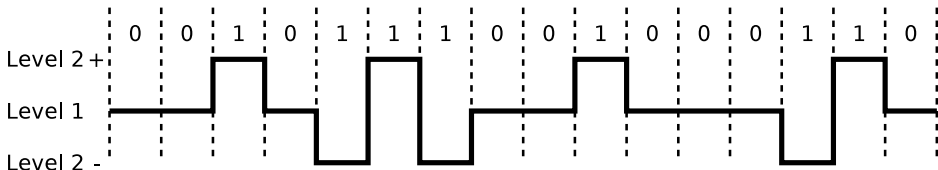
- Special form of return-to-zero (RZ)
  - Uses only 2 signal levels
    - Logical 0 bits are encoded as low signal level
    - Transmit a logical 1 bit  $\implies$  high signal level is transmitted for a half clock and then the signal level returns to the low signal level
- **Clock recovery is impossible** for series of logical 0 bits
- The usage of the different signal level is not equally distributed
  - Therefore **baseline wander can occur**



This line code is used for optical wireless data transmission via IrDA in the transmission mode SIR

## Alternate Mark Inversion (AMI code) = Bipolar Encoding

- Uses 3 signal levels (+, 0 und -)
  - Logical 0 bits are encoded as middle signal level (0)
  - Logical 1 bits are alternating encoded as high (+) or low signal level (-)
- Benefit: **Baseline wander cannot occur**
- Drawback: **Clock recovery is impossible** for series of logical 0 bits
- Error detection is partly possible because the signal sequences ++, --, +0+ and -0- are illegal



# AMI Line Code in Practice and Scramblers

The ISDN  $S_0$  bus uses a modified version of the AMI line code

- With this variant, logical 1 bits are encoded as middle signal level and logical 0 bits are alternating encoded as high signal level or low signal level
- When the AMI line code is used, clock recovery is impossible for the receiver, if series of logical 0 bits are transmitted
  - For this reason, a **scrambler** is often used, after AMI line code encoding
    - A scrambler is a device, which modifies a bit stream according to a simple algorithm in a way, that it is simple to reverse back to the original bit stream
  - In this case, scramblers are used, to interrupt long series of logic 0 bits
    - This makes the clock recovery for the receiver possible

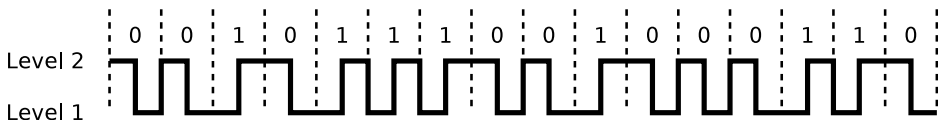


## Bipolar With 8 Zeros Substitution (B8ZS)

- To avoid problems with long series of logic 0 bits, in practice, a slightly modified version of the AMI line code is used  
⇒ **B8ZS**
- B8ZS prevents a loss of synchronization for longer series logical 0 bits by implementing 2 **modification rules** for sequences of 8 logical 0 bits
  - +00000000 is encoded as: +000+-0-+
  - -00000000 is encoded as: -000-+0+-
- In fact, both substitution rules are **code violations**
  - In both substitution rules, 2 positive and negative signal levels occur, one after another
    - This makes the substitutions for the receiver **recognizable**
- In contrast to AMI, **no scramblers are required**, when B8ZS is used
  - Reason: longer series of logical 0 bits are not a problem with B8ZS
- Just as with the AMI line code, **baseline wander cannot occur**

## Manchester Encoding (1/2)

- Uses 2 signal levels
  - A logical 1 bit is encoded with a rising edge
    - Change from signal level 1 (low value) to signal level 2 (high value)
  - A logical 0 bit is encoded with a falling edge
    - Change from signal level 2 (high value) to signal level 1 (low value)
- If 2 identical bits follow each other, at the end of the bit cell, the signal level changes to the initial level
  - Bit cell = time period, that is reserved for the transmission of a single bit



10 Mbps Ethernet (e.g. 10BASE2 and 10BASE-T) uses this line code

## Manchester Encoding (2/2)

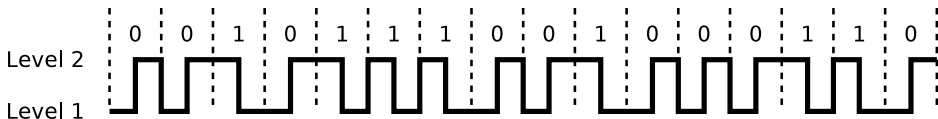
- Advantages:
  - Signal level changes happen all the time to allow clock recovery  
⇒ **Clock recovery is no problem** for the receiver
  - The usage of the signal levels is equally distributed  
⇒ **baseline wander cannot occur**
- Drawback: The transmission of a single bit requires on average 1.5 signal level changes

Because the number of level changes is a limiting factor of the transmission medium, modern network technologies don't use the Manchester encoding as line code

- For this line code, the bit rate is half the baud rate
  - Therefore, the efficiency of the line code is only 50 % compared to NRZ
- **Bitrate:** Transferred payload bits per time unit
- **Baud Rate:** Transferred symbols per second

# Manchester II Encoding

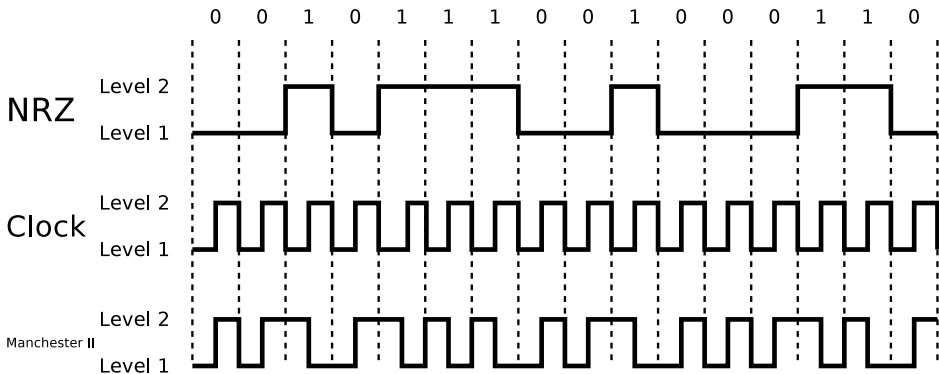
- This line code is the opposite of the Manchester encoding
  - Manchester encoding:
    - Transition from high to low signal corresponds to a logical 0 bit
    - Transition from low to high signal corresponds to a logical 1 bit
  - Manchester II encoding:
    - Transition from low to high signal corresponds to a logical 0 bit
    - Transition from high to low signal corresponds to a logical 1 bit
- Just as for the Manchester encoding, **clock recovery is possible** for the receiver and **baseline wander cannot occur** because the usage of the signal levels is distributed equally



# Manchester II Code

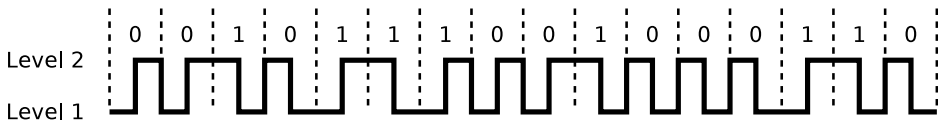
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

- The Manchester II encoding is calculated via exclusive or (XOR) of the NRZ encoded data and the clock



# Differential Manchester Encoding

- Also called **Conditional DePhase encoding (CDP)**
  - Transmit a logical 1 bit  $\implies$  only in the middle of the bit cell changes the signal level
  - Transmit a logical 0 bit  $\implies$  a change of the signal level will take place at the beginning and in the middle of the bit cell
- In this variant of the Manchester encoding too, ...
  - is **clock recovery possible** for the receiver and
  - **baseline wander cannot occur**
- Depending on the initial signal level, **2 signal sequences, inverse to each other, are possible**



Token Ring (IEEE 802.5) uses this line code

# Summary

- All line codes presented so far have drawbacks

## ① **Baseline wander**

- Problem with series of logical 0 bits and 1 bits when NRZ is used
- Problem with series of logical 0 bits when NRZI, MLT-3, Unipolar RZ and AMI are used

## ② **Clock recovery**

- Not guaranteed when NRZ, NRZI, MLT-3, Unipolar RZ and AMI are used

## ③ **Lack of efficiency**

- With the variants of the Manchester encoding

## Possible Solution: Line Codes that encode Groups of Bits

- Modern network technologies encode the bit stream first with a line code that...
  - works **efficient**,
  - **ensures clock recovery** and
  - **avoids baseline wander**
- These encodings **improve** the bit stream in a way, that an encoding afterwards with the line codes NRZ, NRZI and MLT-3 does not result in any problems
- Examples of line codes, which improve the bit stream first, are 4B5B, 5B6B and 8B10B
  - These line codes encode fixed-size input blocks into fixed-size output blocks
- The objective is to achieve the positive characteristics of the Manchester encoding and a high efficiency at the same time



## 4B5B Encoding

- Maps groups of 4 payload bits onto groups of 5 code bits
    - With 5 bits, 32 different encodings are possible
      - Only 16 encodings are used for data (0–9 and A–F)
      - Some of the remaining 16 encodings are used for connection control
    - Because of the additional bit, added to each group of 4 bits payload, the output is increased by factor 5/4
      - Efficiency of the 4B5B encoding: 80%
    - Each 5-bit encoding has a maximum of a single leading 0 bit and in the output data stream, a maximum of three 0 bits in a row
      - Therefore, **clock recovery** for the receiver is **possible**
  - After the encoding with 4B5B, **another encoding** e.g. with NRZI or MLT-3 takes place
    - If 4B5B is combined with NRZI (for 2 signal levels) or with MLT-3 (for 3 signal levels), **baseline Wander cannot occur**
- Ethernet 100BASE-TX: After 4B5B, a further encoding with MLT-3 takes place
  - FDDI and Ethernet 100BASE-FX: After 4B5B, a further encoding with NRZI takes place

- The missing 5-bit combinations are invalid because they contain more than a single leading 0 bits or more than two 0 bits in a row

If Fast Ethernet 100BASE-TX is used, frames begin with JK and end with TR

## 5B6B Encoding (1/2)

- Maps groups of 5 payload bits onto groups of 6 code bits
  - Of the 32 possible 5-bit words, 20 are mapped to 6-bit words that contain an equal number of 1 bits and 0 bits  
⇒ **neutral inequality** (*balanced*)
  - For the remaining twelve 5-bit words, a variant with two 1 bits and four 0 bits and a variant with four 1 bits and two 0 bits exist  
⇒ **positive or negative inequality** (*unbalanced*)
- As soon as the first 5-bit word without neutral inequality need to be encoded, the variant with the positive inequality is used
  - For encoding the next 5-bit word without neutral inequality, the variant with the negative inequality is used
    - The variants with positive or negative inequality alternate

## 5B6B Encoding (2/2)

- After the encoding with 5B6B, another encoding with NRZ takes place
  - This is possible, because if 5B6B is used, **clock recovery is possible** for the receiver and **baseline wander cannot occur**
- Advantage compared to the Manchester encoding: higher baud rate
  - Efficiency:  $5/6 = 83.\bar{3}\%$

5B6B is used by Fast Ethernet 100Base-VG

---

Prof. Dr. Christian Baun – 3rd Slide Set Computer Networks – Frankfurt University of Applied Sciences – WS2021 37/41

- Used by Gigabit-Ethernet 1000Base-CX, -SX, -LX, FibreChannel, InfiniBand, DisplayPort, FireWire 800 (IEEE 1394b) and USB 3.0

## 8B6T Encoding

- 8B6T = Binary 6 Ternary
  - Useful for network technologies, that use **> 2 signal levels**
- This line code encodes 8-bit blocks as groups of 6 symbols, where each one can represent the state -, 0 or +
  - The symbols of the states represent electrical signal levels
- The encoding is carried out by using a table, which contains all  $2^8 = 256$  possible 8-bit combinations
  - The table shows, that the output of 8B6T makes **baseline wander impossible**, and the frequent signal level changes make **clock recovery possible** for the receiver
- In contrast to 4B5B, 5B6B and 8B10B, which only *improve* the payload and require an encoding with NRZ(I) or MLT-3 afterwards, 8B6T encoded data **can be used directly for transmission**

Fast-Ethernet 100BASE-T4 uses this line code

# 8B6T Encoding (Table)

8-bit sequence	8B6T code	8-bit sequence	8B6T code	8-bit sequence	8B6T code
00	+ - 00 + -	10	+ 0 + - - 0	20	00 - + + -
01	0 + - + - 0	11	+ + 0 - 0 -	21	- - + 00 +
02	+ - 0 + - 0	12	+ 0 + - 0 -	22	+ + - 0 + -
03	- 0 + + - 0	13	0 + + - 0 -	23	+ + - 0 - +
04	- 0 + 0 + -	14	0 + + - - 0	24	00 + 0 - +
05	0 + - - 0 +	15	+ + 00 - -	25	00 + 0 + -
06	+ - 0 - 0 +	16	+ 0 + 0 - -	26	00 - 00 +
07	- 0 + - 0 +	17	0 + + 0 - -	27	- - + + + -
08	- + 00 + -	18	0 + - 0 + -	28	- 0 - + + 0
09	0 - + + - 0	19	0 + - 0 - +	29	- - 0 + 0 +
0A	- + 0 + - 0	1A	0 + - + + -	2A	- 0 - + 0 +
0B	+ 0 - + - 0	1B	0 + - 00 +	2B	0 - - + 0 +
0C	+ 0 - 0 + -	1C	0 - + 00 +	2C	0 - - + + 0
0D	0 - + - 0 +	1D	0 - + + + -	2D	- - 00 + +
0E	- + 0 - 0 +	1E	0 - + 0 - +	2E	- 0 - 0 + +
0F	+ 0 - - 0 +	1F	0 - + 0 + -	2F	0 - - 0 + +

etc.



# Summary

Line code	Signal levels	Baseline wander possible	Signal level change	Self-synchronizing <sup>1</sup>	Efficiency <sup>2</sup>	Directly transferable	Additional encoding
NRZ	2	yes	at changes	no	100%	no	—
NRZI	2	yes	for 1-bits	no	75%	no	—
MLT-3	3	yes	for 1-bits	no	100%	no	—
RZ	3	yes	always	yes	50%	no	—
Unip. RZ	2	yes	for 1-bits	no	75%	no	—
AMI	3	no	for 1-bits	no	100%	no	Scrambler
B8ZS	3	no	for 1-bits	yes	100%	yes	—
Manchester	2	no	always	yes	50%	yes	—
Manchester II	2	no	always	yes	50%	yes	—
Diff. Manch.	2	yes	always	yes	50%	yes	—
4B5B	2	yes	—	yes	80%	no	NRZI or MLT-3
5B6B	2	no	—	yes	83.3%	no	NRZ
8B10B	2	no	—	yes	80%	no	NRZ
8B6T	3	no	—	yes	100%	yes	—

<sup>1</sup> Specifies if the clock recovery is possible with this line code.

<sup>2</sup> Ratio of bit rate (payload in bits per time) and baud rate (signal changes per second).