Applied Electronics

C6 – Serial Connections

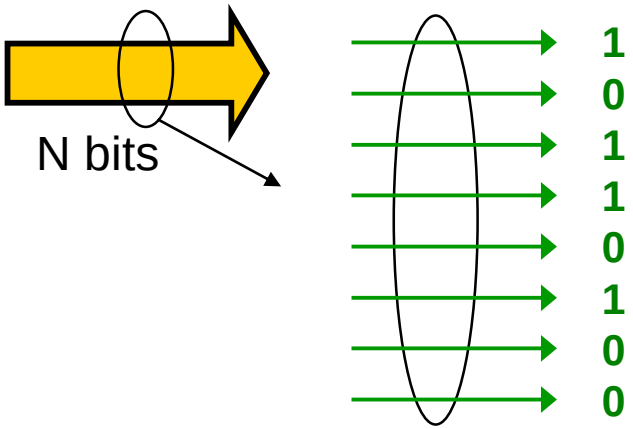
- Limits of parallel buses
- Eye diagram
- Basic asynchronous protocol
- Examples of synchronous protocols

Lecture C6: Serial Connections

- Why serial connections?
- Skew and synchronization
- Crosstalk, Inter Symbol Interference
- Eye diagram
- Asynchronous serial protocols
- Synchronous serial protocols
- References
  - D. Del Corso: Telecommunication Electronics: Ch. 5.5

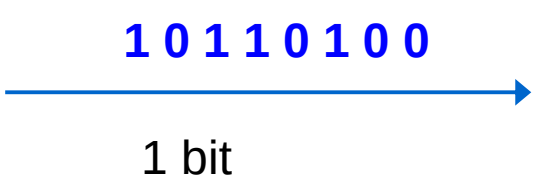
Parallel and serial connections

- Parallel connection



Parallel
→ the bits are “carried” by several conductors at the same time

- Serial connection

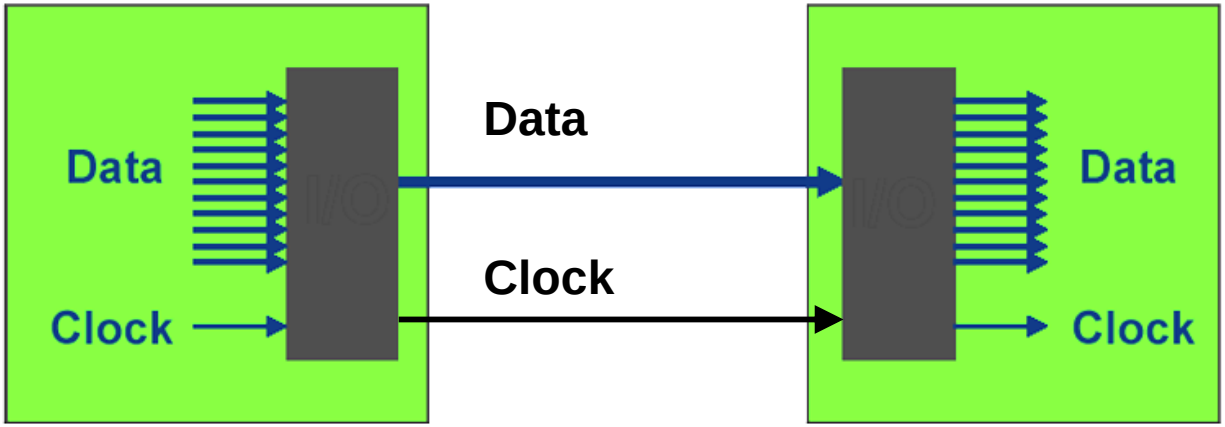


Serial
→ the bits are “carried” by a single conductor at different times

From parallel bus to serial bus

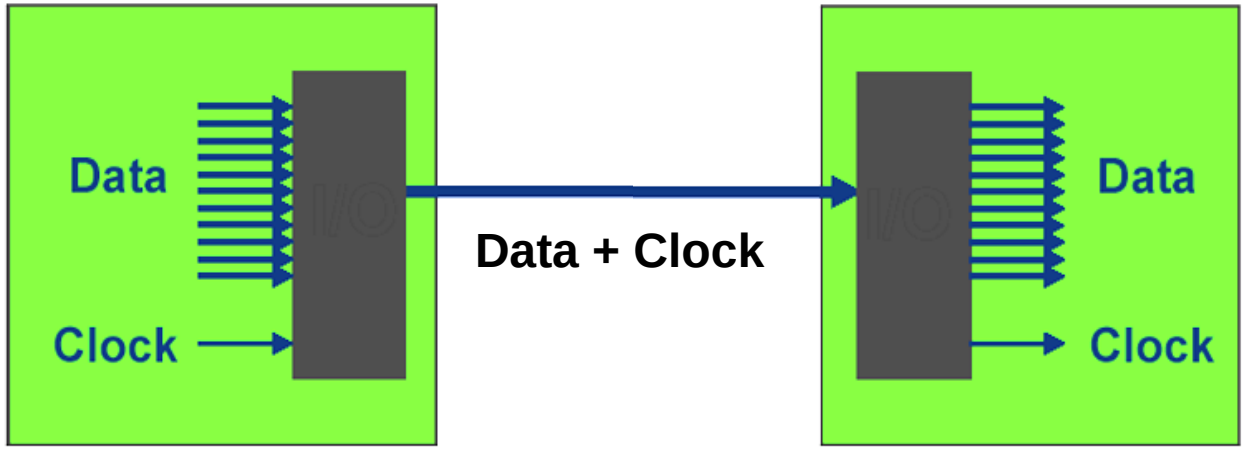
- Limits of parallel connections
  - Speed is limited by propagation (t_{TX}) and skew t_K
  - Source Synchronous transfers eliminate t_{TX}, but not the skew t_K
  - Multipoint structures require terminations → power!
  - To increase throughput → higher parallelism
    - Larger connectors
    - Increased electromagnetic compatibility issues (EMI and EMC)
    - Increased power consumption
- Solution: migration to serial connections
  - ATA → SATA (Serial ATA)
  - PCI → PCI Express
  - Parallel printer port → USB

Serial link



- Transfer of N bits → N cycles: high LATENCY
- N drivers, 1 cycle → 1 driver, N cycles: same energy
- Few signals → lower generated electromagnetic interference (EMI) and received (EMC, compatibility)
- Lower data/clock skew (timing misalignment): 1 data signal instead of N data signals, but still present

Self-synchronizing serial connection



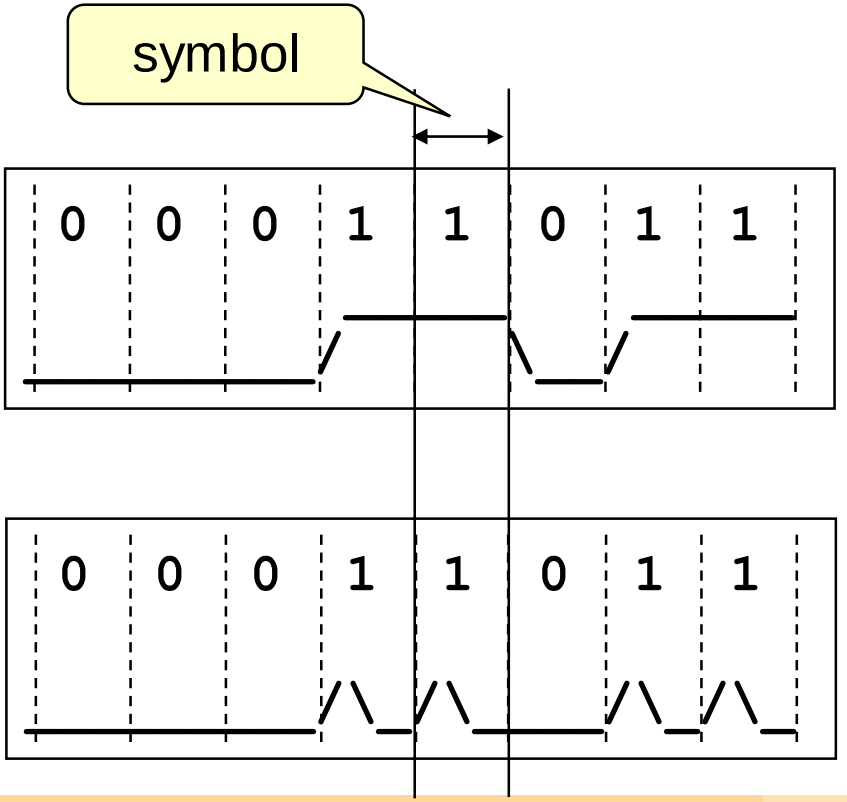
- Data and Clock are carried by the same wire
  - ♦ No misalignment due to skew
- Clock/Data embedding technique
  - ♦ Requires specific protocols

Characteristics of serial connections

- Advantages of serial connections
  - Few conductors (up to only one with appropriate protocols)
  - Simplified routing or cabling
  - Lower power consumption (one driver)
  - Best for long distance and/or high speed
- Problems of serial connections
  - Cycle means transfer of a symbol (1 or more bits)
    - Delay in the transfer of multi-bit packets, latency
  - Synchronization at cycle level
    - Separate clocks → periodic resynchronization (byte, message)
    - "Embedded" clock → self-synchronization (at all levels)
  - Flow control mechanism at transaction level

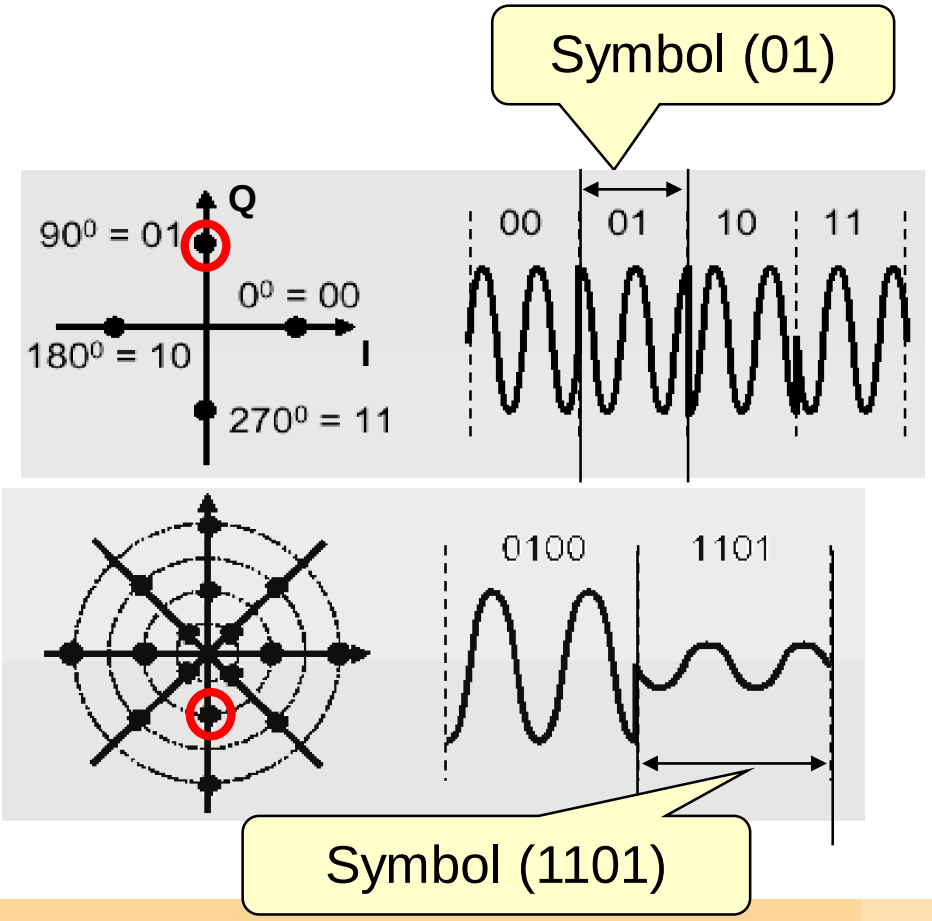
Bits and Symbols

- Transmitted signal is a sequence of SYMBOLS
- A symbol may represent one or more bits
  - Bit rate (BPS, b/s): bit/s
  - Baud Rate: symbols/s
  - Efficiency: bit/symbol
- NRZ: 0→L; 1→H
  - 1 bit/symbol
  - Bit rate = Baud rate
- RZ: 0→L; 1→impulse
  - 1 bit/symbol
  - Bit rate = Baud rate



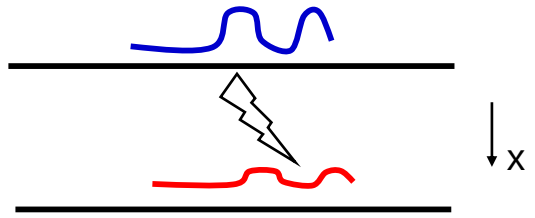
Constellation of signals

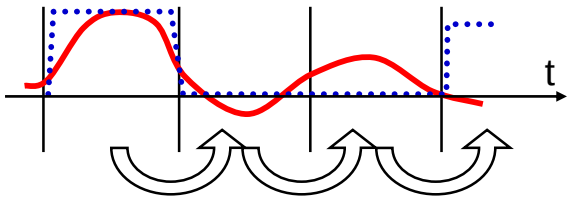
- Signals can be represented by the I/Q components (in-phase/quadrature)
  - Constellation of signals
- Example 1: QPSK
  - 4 phase values
  - 2 bits/symbol
  - Efficiency = 2
- Example 2: QAM
  - 4 phases, 4 amplitudes
  - 4 bits/symbol
  - Efficiency = 4



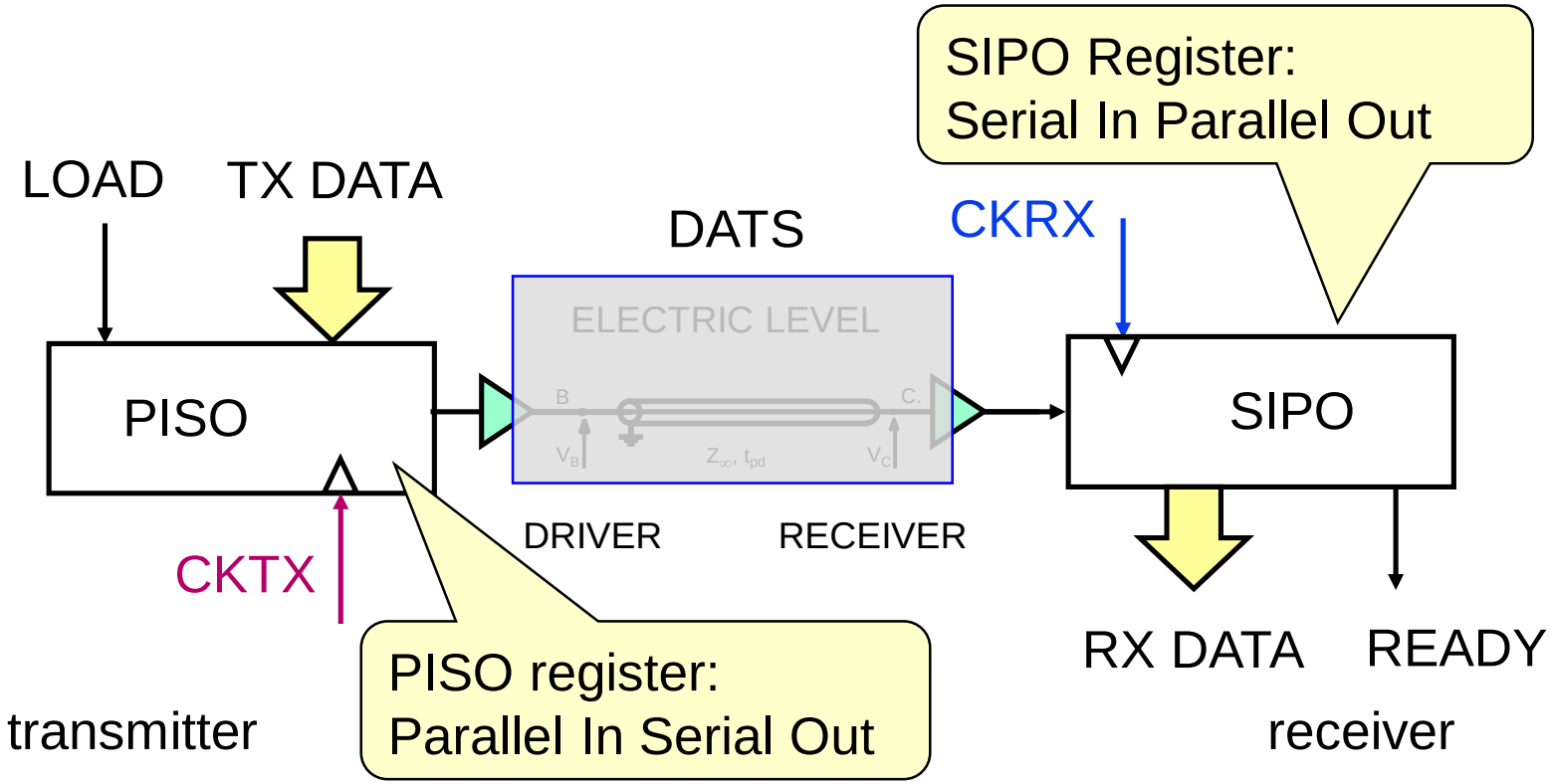
Intersymbol interference (ISI)

- Information can be altered by “close” signals
- Parallel connection → crosstalk
  - Signal transfer from one conductor to the others
  - Same time, different place (the other conductor)
- Serial connections → ISI (Inter Symbolic Interference)
  - Signal transfer from one time interval to another
  - Same place (only one conductor), different time





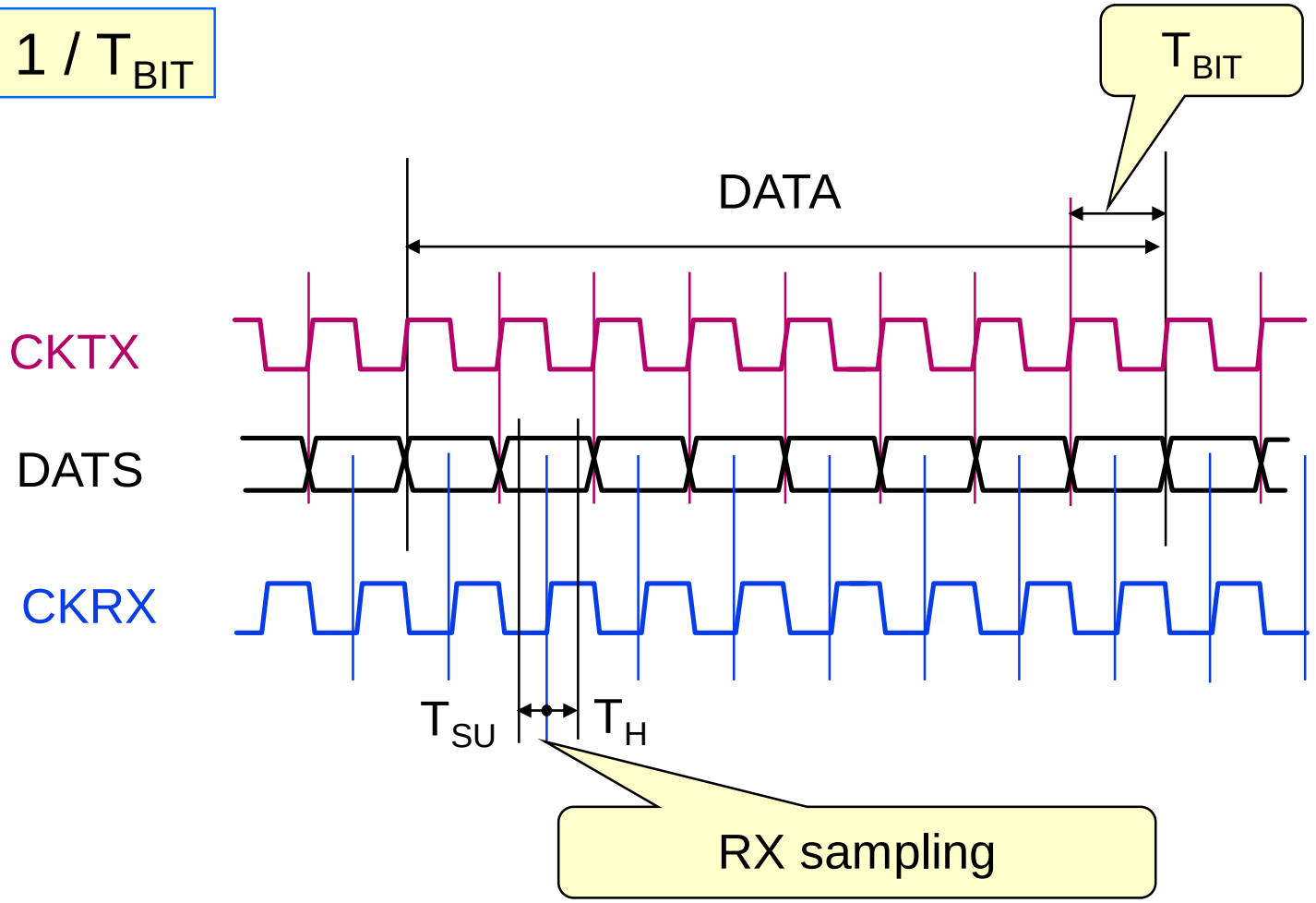
Basic serial connection



TX and RX clocks must be synchronized
(same frequency and known phase relationship)

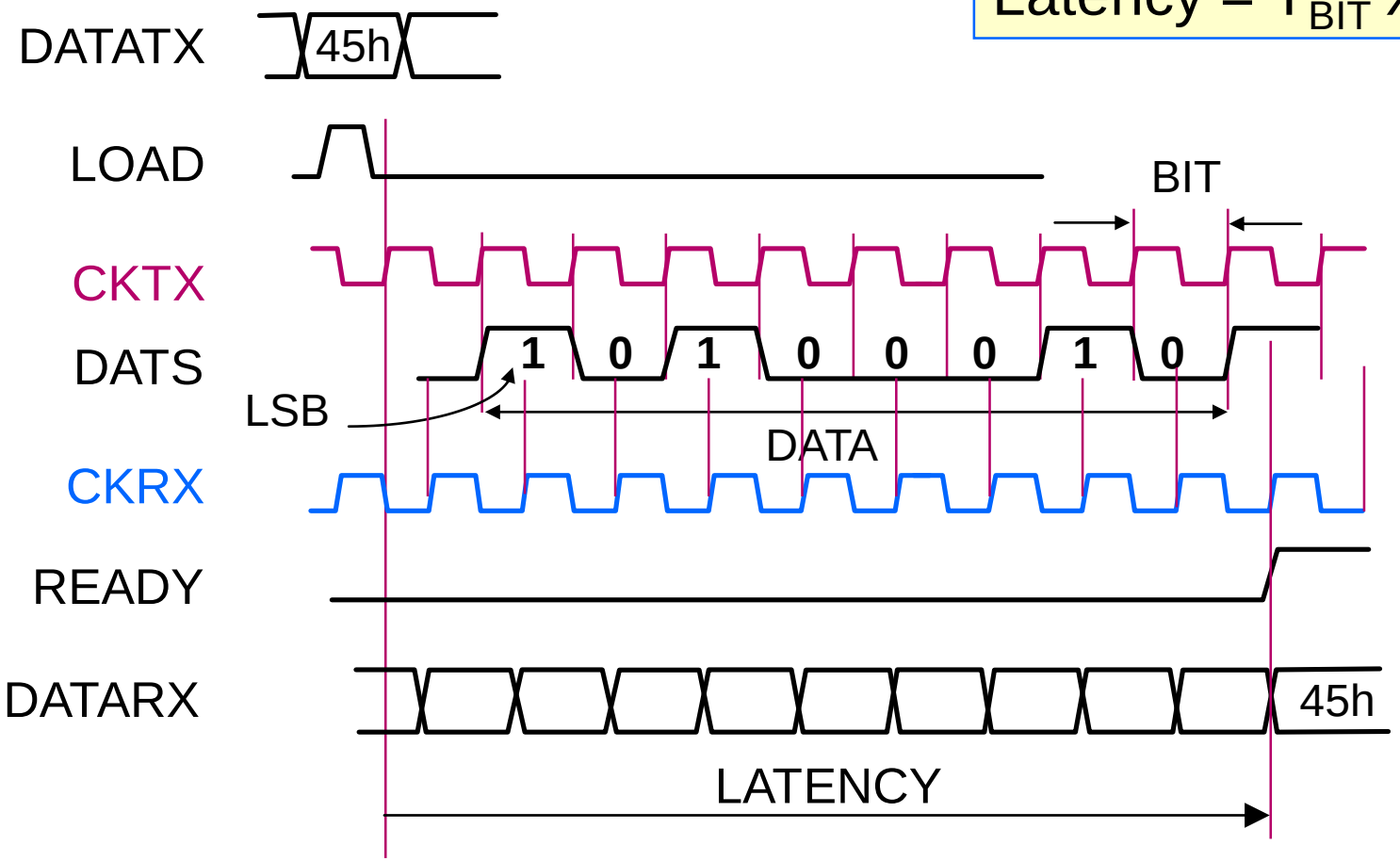
Serial transmission rate

Bit Rate = 1 / T_{BIT}



Example: transmit “01000101” (45H)

Latency = T_{BIT} x num. of bits



Levels of synchronism

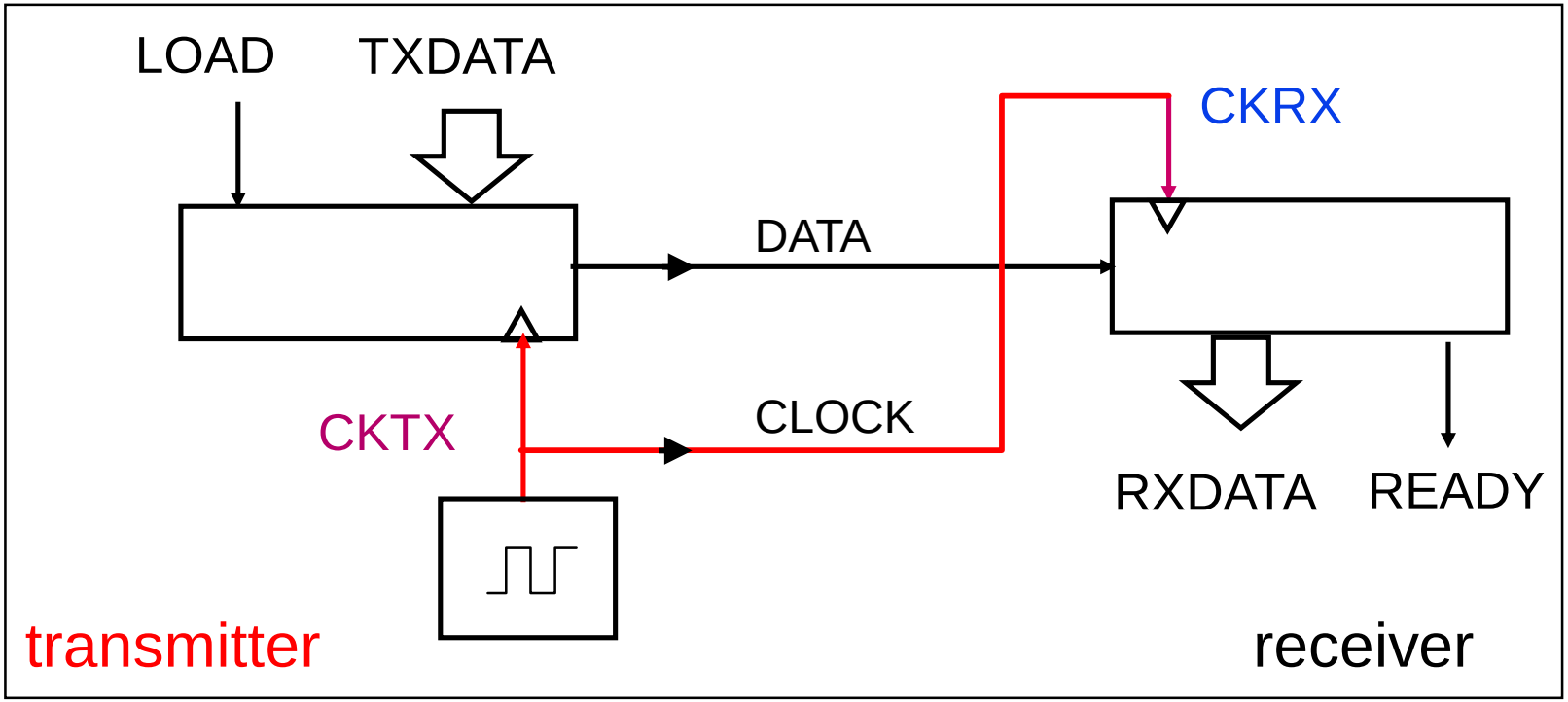
- Bit synchronism
  - Guarantees correct sampling of the single bit (correct loading in the receiver register)
  - Related to the phase between Data and CKRX transitions
- Character synchronism
  - Guarantees correct recognition of MSB and LSB
    - Position of the bits in the receiver register
  - Linked to the activation of the READY signal
- Message Synchronism:
  - Assign meaning to characters
    - Header, length, priority, data, ...

How to synchronize at the bit level

- Same Clock for TX and RX, generated by
  - TX: Source Synchronous (like WRITE, constrained by t_K)
  - RX: Asynchronous (like READ, constrained by t_K and t_{TX})
- Independent Clock Generators
  - No synchronization
    - Does not guarantee t_{SU} and t_H → **metastability**
  - Periodic RX Clock synchronization
    - Asynchronous protocol (serial)
- Extracting the RX Clock from the received signal (CDR)
  - Occasional synchronization (from transitions)
    - Requires a time limit on the interval between transitions
  - Continuous synchronization → embedded clock modulations

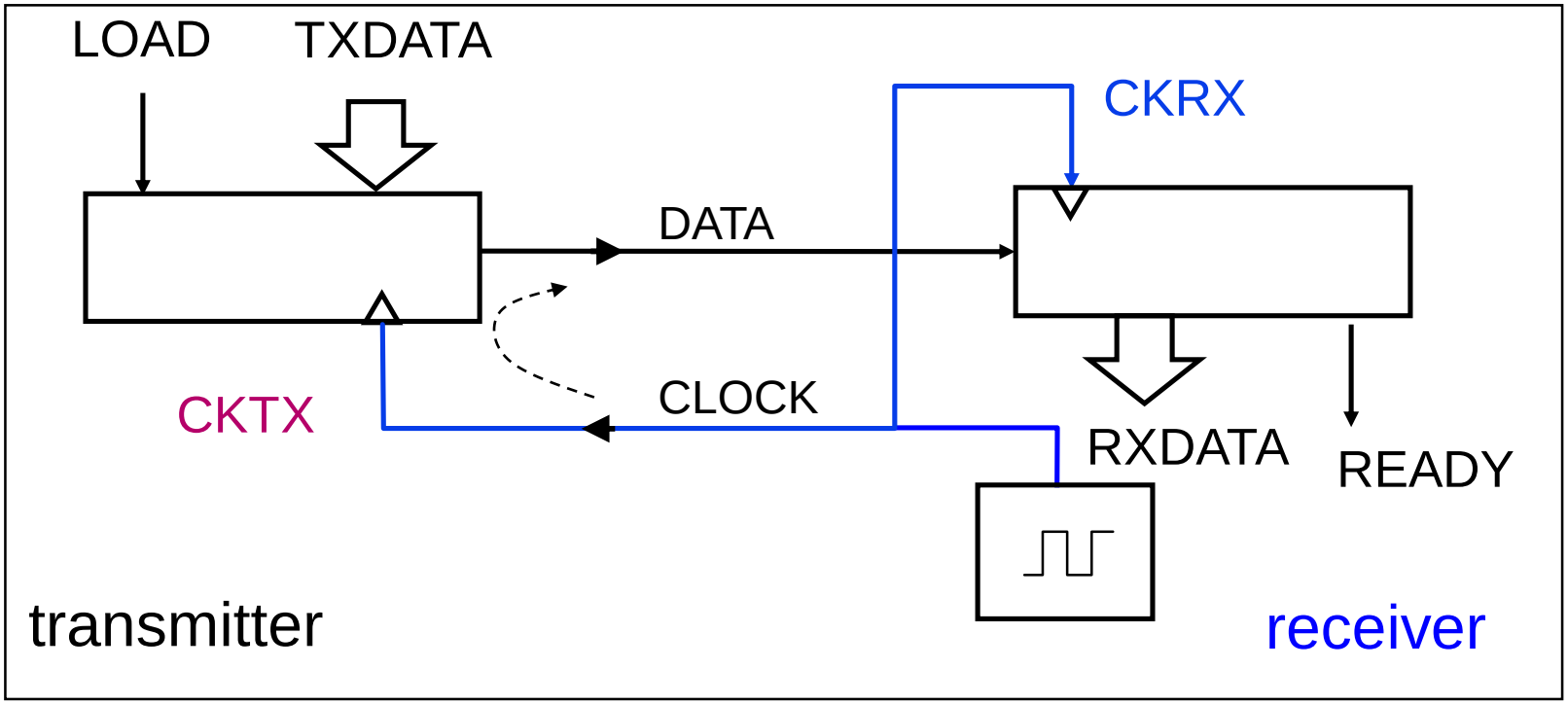
Single clock – generated by TX

- Two wires, fixed speed, limited by t_K (synchronous Write)
- Advantages: maximum speed (Source Synchronous)



Single clock – generated by RX

- Two wires, speed limited by t_K and t_{TX} (asynchronous Read)
- Advantages: RX controls the speed (handshake!)

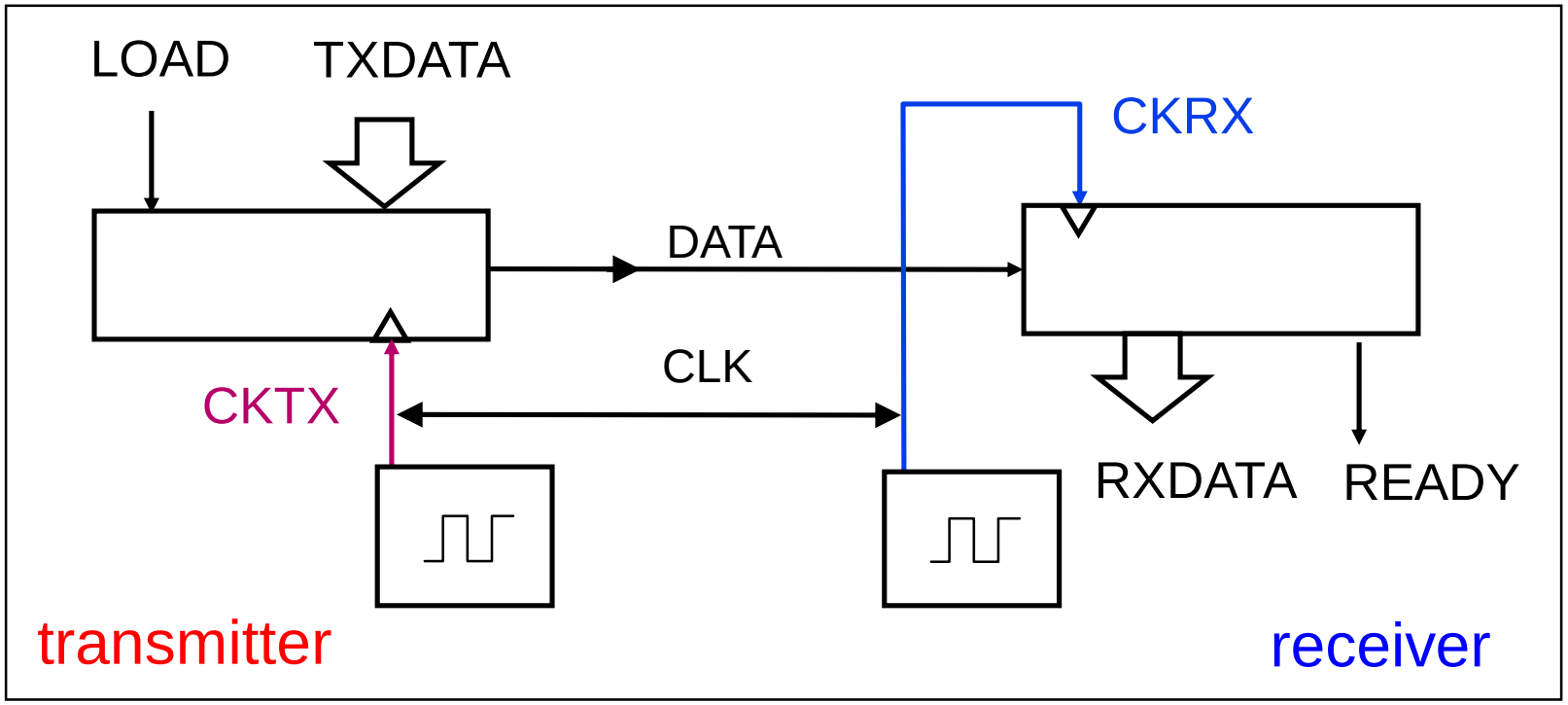


Transfer speed

- Necessary to ensure t_{SU} and t_H for the receiver register
  - The interconnection changes temporal relationships
- Cadence limited by these synchronization constraints
  - Clock generated at the transmitter (TX)
    - Clock and data move in the same direction
    - Timing variations come only from skew
    - $T_{CKmin} = t_{SU} + t_K$ (t_H included in t_{CO})
  - Clock generated at the receiver (RX)
    - Two t_{TX}: from RX to TX for clock, from TX to RX for data
    - Timing variations depend on t_{TX} and t_K
    - $T_{CKmin} = 2\,t_{TX} + 2\,t_K + t_{SU}$ (t_H included in t_{CO})

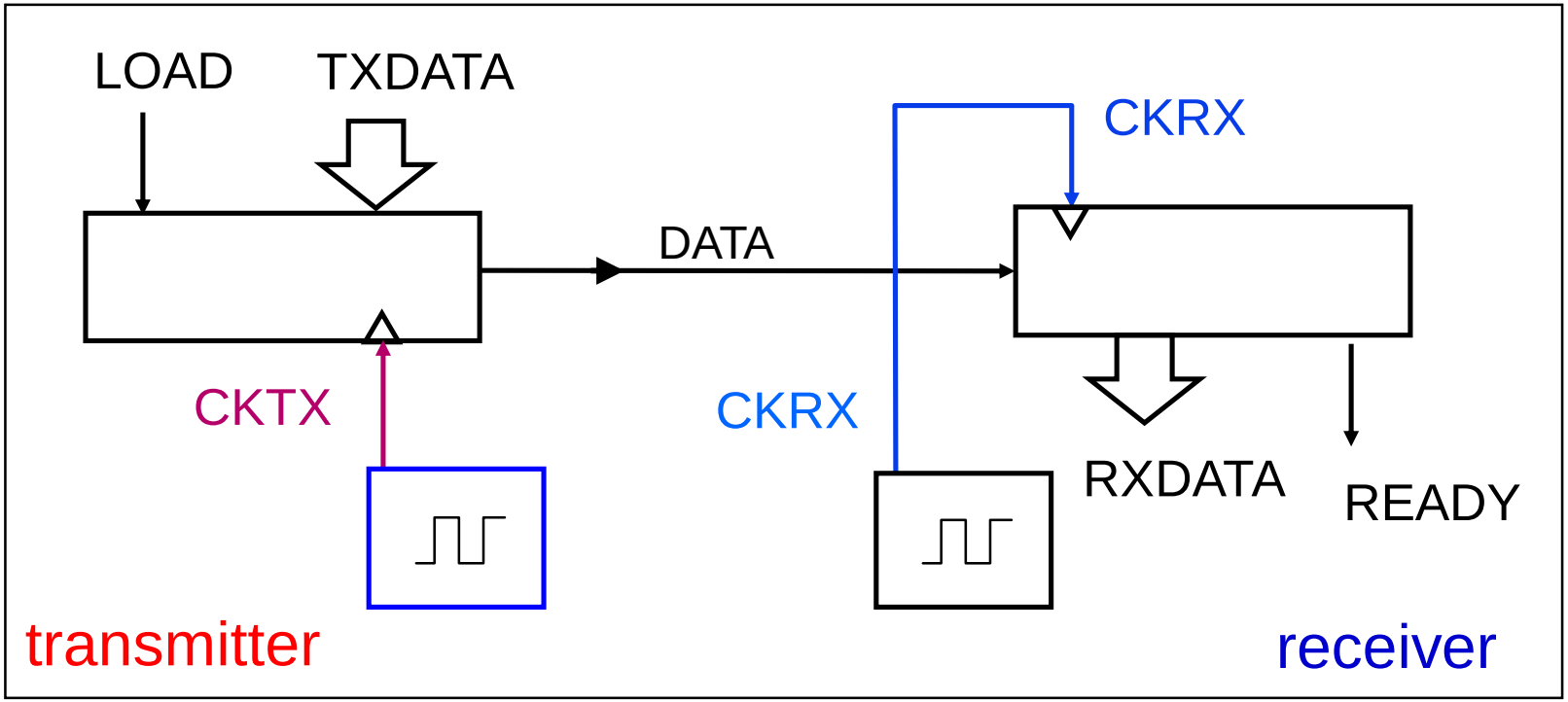
Cooperative Clock (I^2C bus)

- Allows synchronization between multiple units
- Handshake: STB/ACK encoded on the same CLK signal (driven in Open Drain)



Independent isofrequency clocks

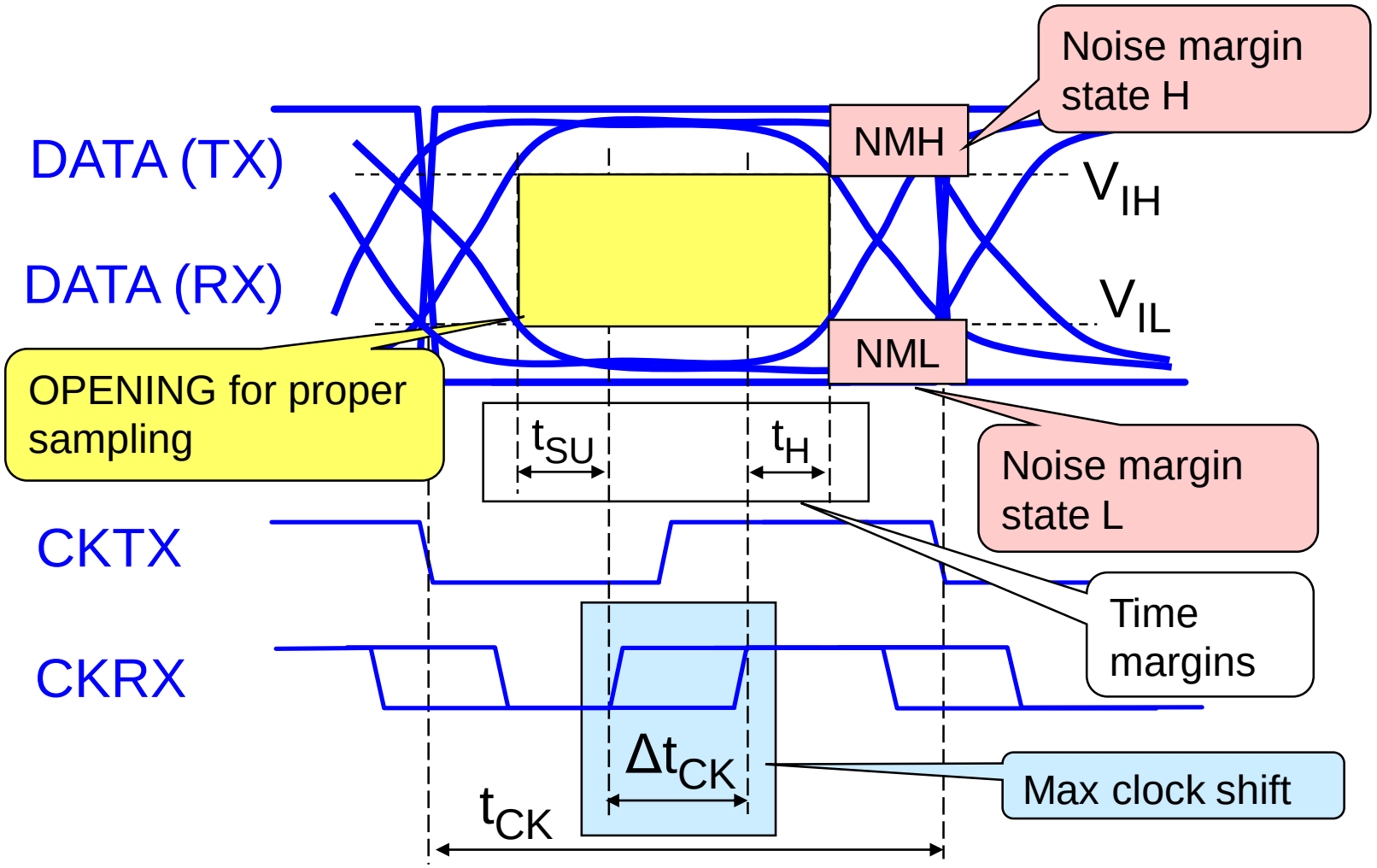
- Requires a single conductor (Data)
- Errors caused by clock offsets



Limits of independent clocks

- Ideal: synchronized clock generators, no noise
- Real case (f_{CK} “nominally equal” on TX and RX sides)
  - The frequency of the clock generators varies
  - The signal is modified by inter-symbolic interference, noise, ...
    - Shift of the sampling time
- After synchronization, correct sampling depends on phase shifts and noise
  - Periodic resynchronization (every S bit)
  - Resynchronization on each bit: embedded clock
- To identify the time and amplitude margins
  - Eye diagram

Eye diagram

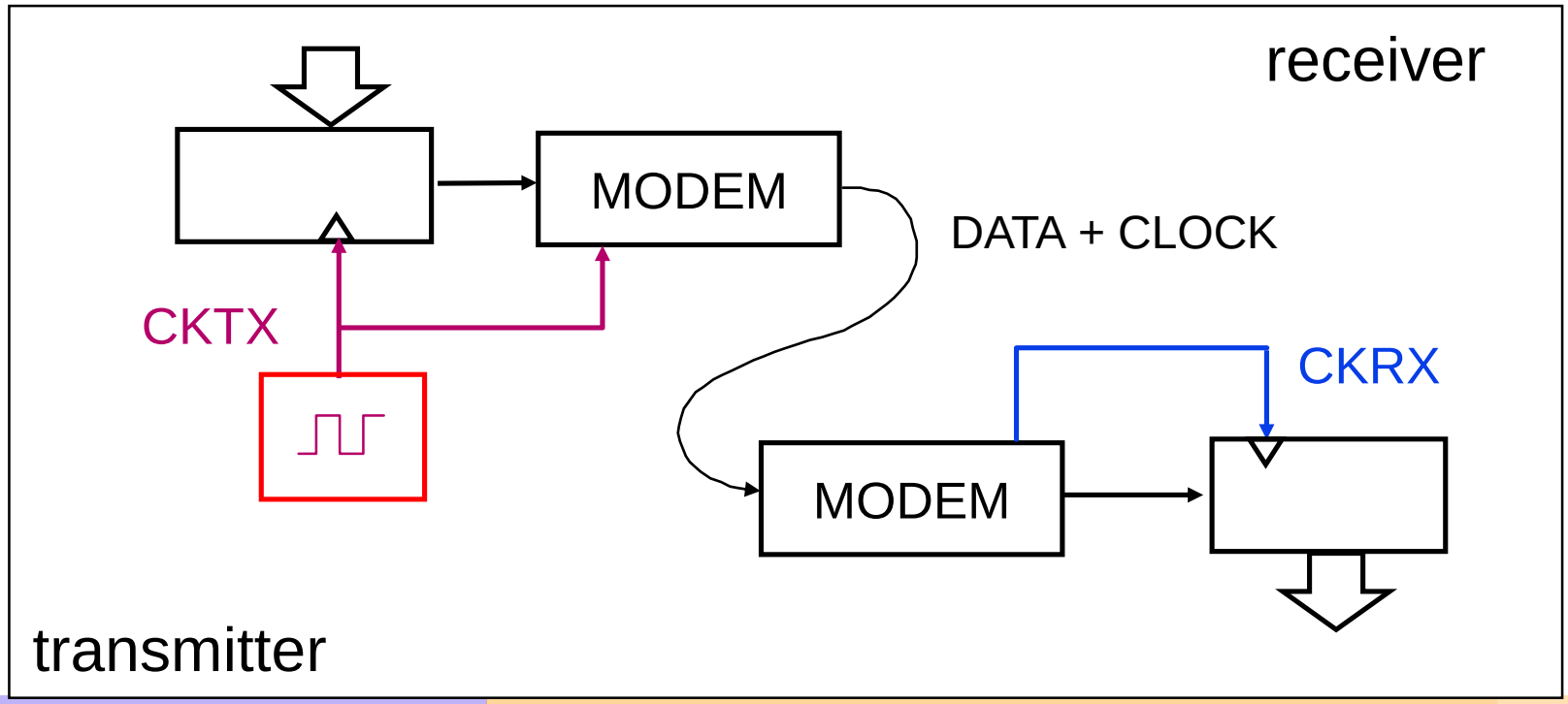

Noise margin state H
NMH
DATA (TX)
V_{IH}
DATA (RX)
V_{IL}
NML
OPENING for proper sampling
t_{SU}
t_H
Noise margin state L
CKTX
Time margins
CKRX
Δt_{CK}
Max clock shift
t_{CK}

Eye diagram parameters

- Eye opening is the correct sampling zone (t, A)
  - V: from V_{IL} to V_{IH}, t at least for $t_{SU} + t_H$
- Time axis: errors Δt_{CK} for phase shifts of CK
  - Tolerances and frequency drifts (medium term)
  - Shifts in each clock period → skew and jitter
- Amplitude axis: noise, distortion, ISI
  - V_{OH}, V_{OL}, V_{IH}, V_{IL} (driver and receiver)
  - Slope of edges (channel band)
  - Noise and intersymbol interference (ISI)
- Resynchronization interval → linked to $t_{CK}/\Delta t_{CK}$

Synchronous modem

- Single channel (single conductor)
- TX: merges data and clock (clock embedding)
- RX: splits data from clock → continuous synchronization



Transfer cadence

- Single conductor → no skew
  - Can operate with very high bit rates (10 Gb/s)
  - Must merge/separate data and synchronism
  - Synchronism adds information → higher bandwidth (special symbols or modulations)
  - Synchronism overhead depends on oscillator precision
- From bits to symbols
  - Coding: operation on *digital* information flow
    - Adds synchronism and/or error checking
  - Modulation: operation on *analog* signals
    - Adapt the signal to the channel characteristics (band, noise)

Performance of serial links

- Speed (bit rate)
  - Limited by the sampling jitter
  - Can be improved by reducing timing “errors”
- Link length
  - Limited by attenuation, distortion, noise
  - Improved by increasing the voltage range
- Both can be improved with differential signals
  - High noise immunity with low range excursion
- Two optimization “families”
  - Short links, high speed
  - Longer links, low speed

Asynchronous and synchronous links

- Asynchronous serial link
  - Bits grouped in characters
  - Discontinuous transmission
  - Clock synchronization at the beginning of the character

- Synchronous serial link
  - Bits grouped in packets (packet/frame, various sizes)
  - Continuous transmission
  - Clock synchronization on each bit (encoding/modulation)

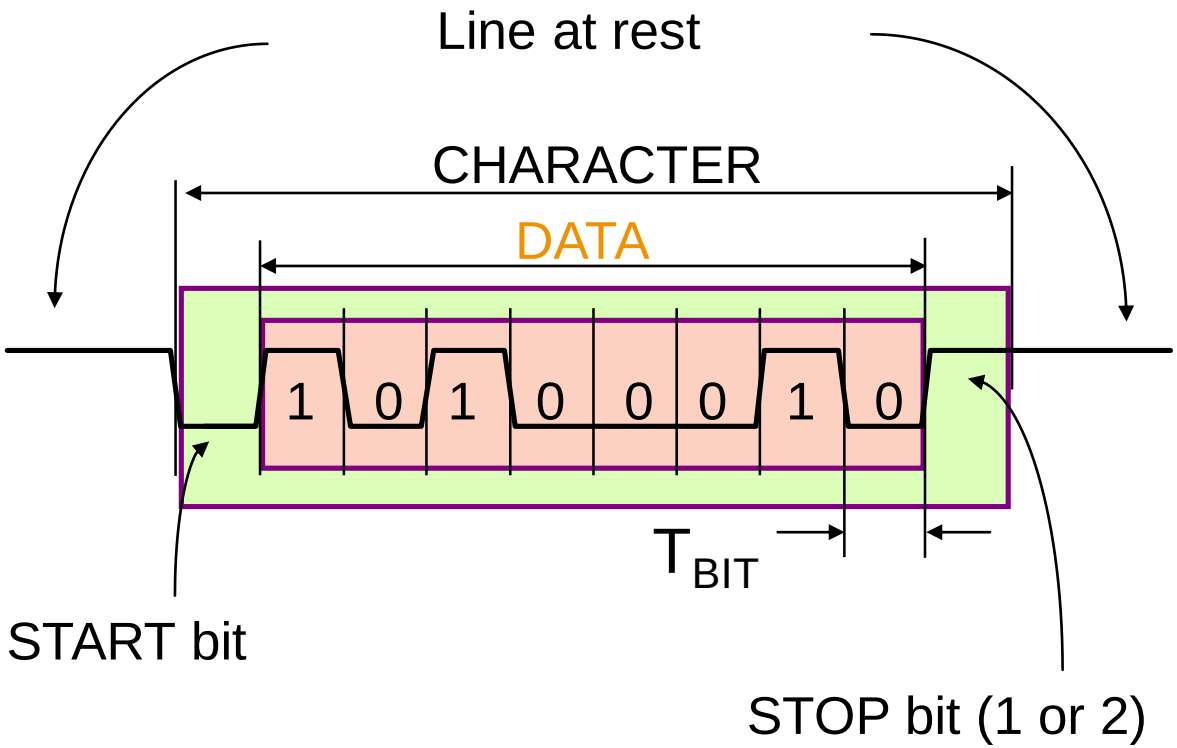
Lecture C6: Serial Connections

- Why serial connections?
- Skew and synchronization
- Crosstalk, Inter Symbolic Interference
- Eye diagram
- Asynchronous serial protocols
- Synchronous serial protocols

Asynchronous serial links

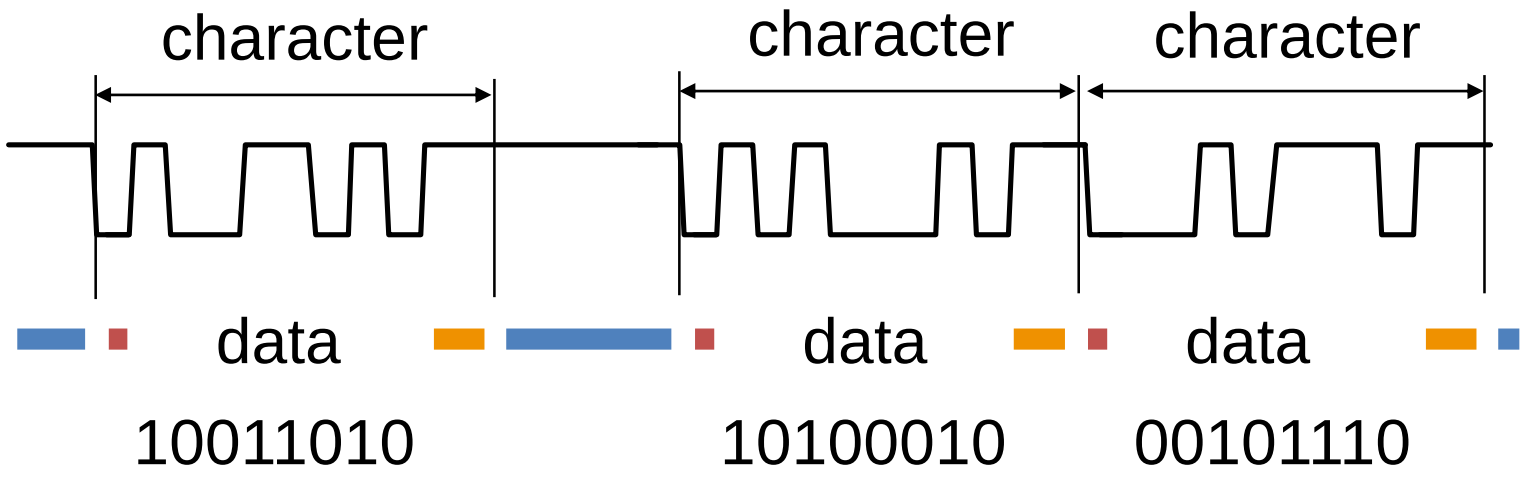
- The line at rest has a defined state (high)
  - The transmission of a character can begin at any time
  - The beginning of the character is indicated by a START symbol
- RX side clock is generated at the receiver and is synchronized by the falling edge of the Start bit
  - Oscillators have errors and drifts
  - Synchronization is maintained for a limited time
  - Must resynchronize periodically the RX clock
- To ensure that the Start symbol (bit L) is detected
  - Insert at least one Stop symbol (bit H) after each character

Asynchronous serial character



45H transmission (LSB first)

Sequence of characters



UART

- UART: Universal Asynchronous Receiver / Transmitter
- TX side
  - Conversion Parallel → Serial with PISO register
  - Insertion of Start and Stop symbols
  - Possible insertion of bits for error detection (parity)
- RX side
  - Synchronism of Bit and Character (from the Start Bit)
  - Conversion Serial → Parallel with SIPO register
  - Check of the format of each character (Stop Bit)
  - Possible error check (parity)

Synchronization in the UART

- On transition H → L and keep L for $T_{BIT}/2$
  - Recognition of a Start Bit
  - Sampling time synchronization (clock from bits)
  - Start the bit/character counter
- At the end of the character
  - Parity check (if required)
  - Check the Stop bit
  - Activation of the “RX data ready” flag (Ready)
- Wait for new H → L transition (start of another character)
- Limits
  - Number of bits / character is limited by Δf_{CK} between TX and RX

Standard RS 232 connections

- Created to connect terminals or computers to Modems
- A standard for the physical layer and defines
  - Signals on connector (25 pins)
  - Command signals for Modem
  - Electrical levels
  - It does not define a character format
- Often used to connect peripherals
  - High noise immunity, low speed
  - Simplified versions (example: 9 pin)
- Upper layers defined by V24, V92,….

Lecture C6: Serial Connections

- Why serial connections?
- Skew and synchronization
- Crosstalk, Inter Symbolic Interference
- Eye diagram
- Asynchronous serial protocols
- Synchronous serial protocols

Synchronous protocols

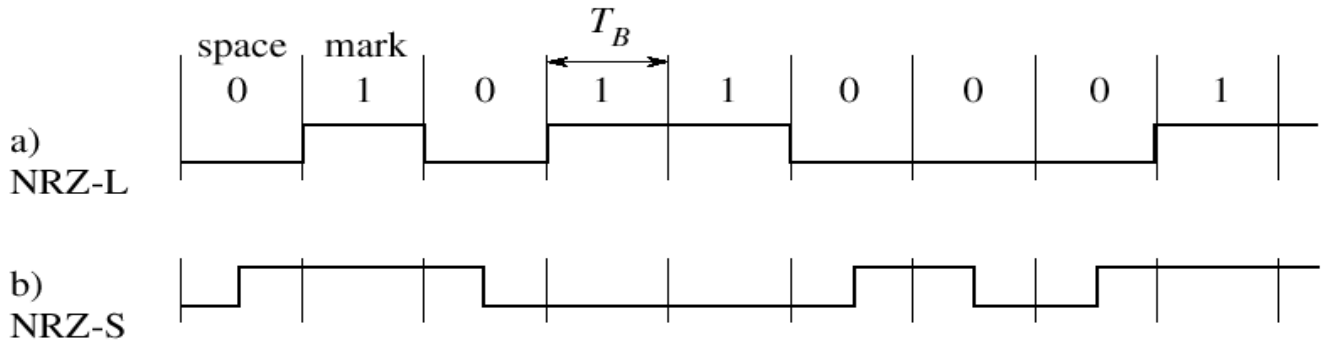
- Asynchronous protocols → periodic synchronization
- Synchronous protocols → continuous synchronization
  - How to maintain bit synchronization?
  - Include time references, i.e., transitions
- Clock extraction from data (CDR: Clock-Data Recovery)
  - Transitions in known positions within each T_{BIT}
  - Embedded Clock, synchronous modulations, ...
- Clock synchronization from data
  - Ensure transitions earlier than the maximum allowed interval
  - Bit stuffing: insertion of additional bits to ensure transitions
  - BxBy: x bits replaced with y bits ($y > x$) to guarantee transitions

Codes for serial connections

- NRZ: Not Return to Zero
  - NRZ-L, NRZ-M, NRZ-S, ...
- RZ: Return to Zero (unipolar)
  - RZ-M, RZ-S
- Codes BxBy, ...
- Multilevel encodings: MLT-3 (MultiLevel Trans.): +, 0, –
  - Ternary code, require higher SNR
- Manchester encoding and derivatives
- …

NRZ coding

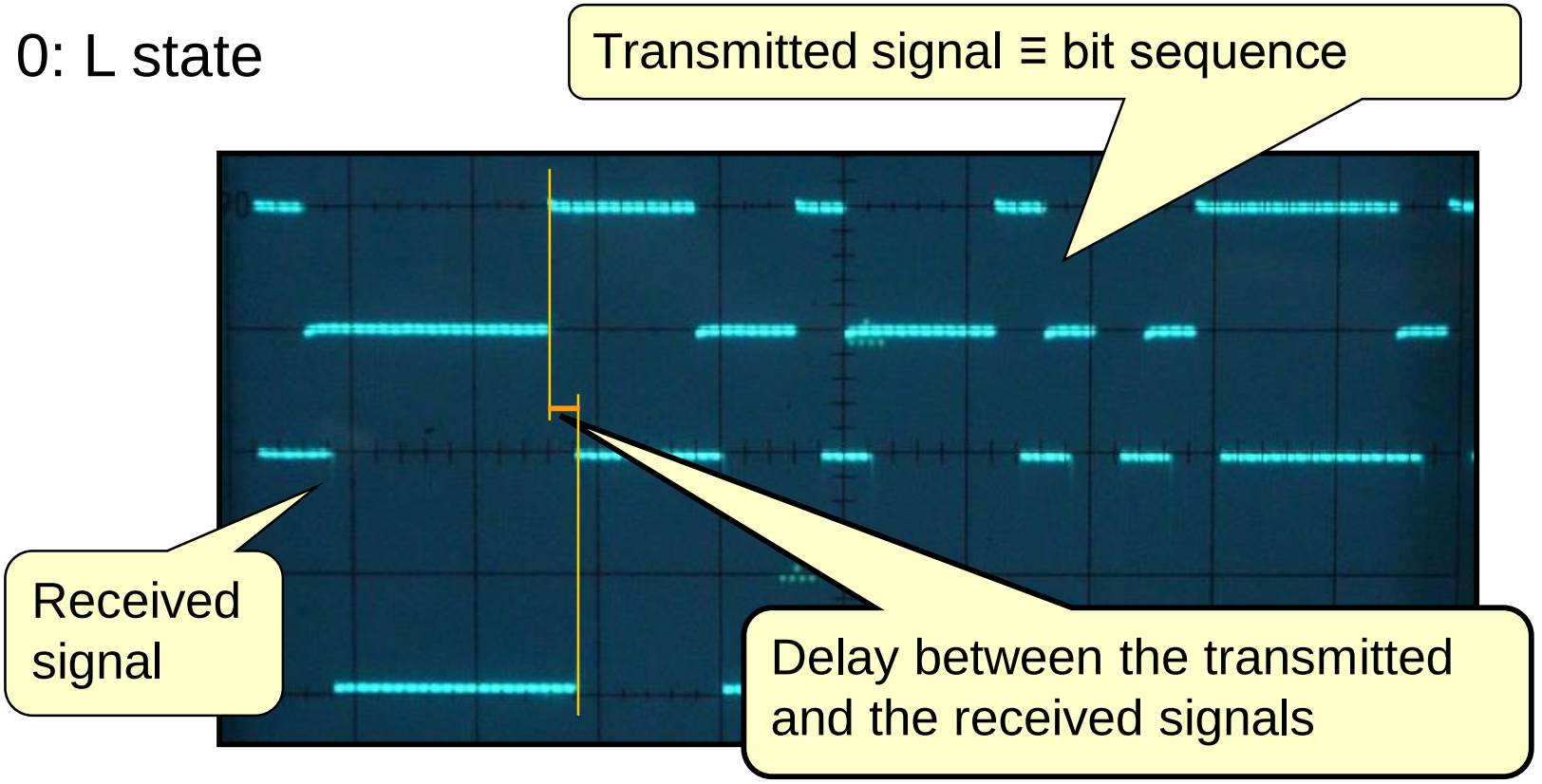
- NRZ: Not Return to Zero
  - NRZ-L: 1 represented by H state, 0 by L state
  - NRZ-M: 1 is encoded in a transition
  - NRZ-S: 0 is encoded in a transition
  - Each symbol can end up either H or L



- Unreliable CK recovery in data seq. without transitions
- Bandwidth: 1 transition/bit → f_{MAX} = BitRate / 2

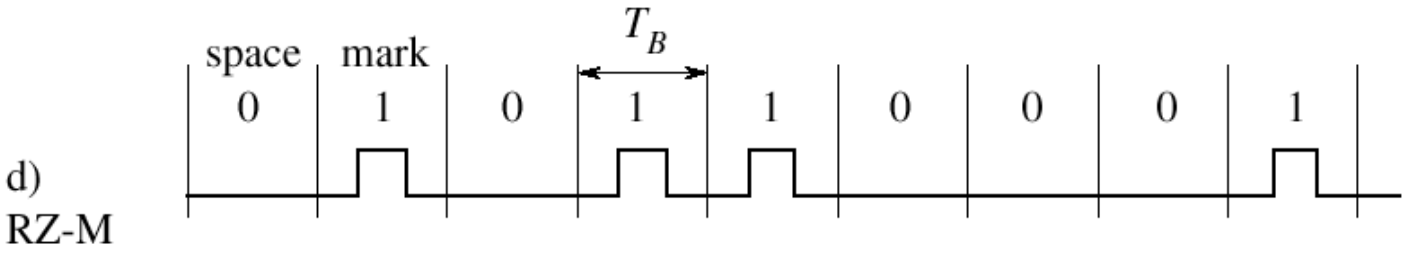
NRZ-L encoding

- NRZ-L → plain binary data, no modifications
  - Without return to 0 (Not Return to Zero – level: NRZ-L)
  - 1: H state
  - 0: L state



RZ coding

- RZ: Return to Zero (unipolar)
  - All symbols start and end with 0 (or another fixed state)
  - RZ-M: 1 is represented by an H (Mark) pulse
  - RZ-S: 0 is represented by an H (Mark) pulse



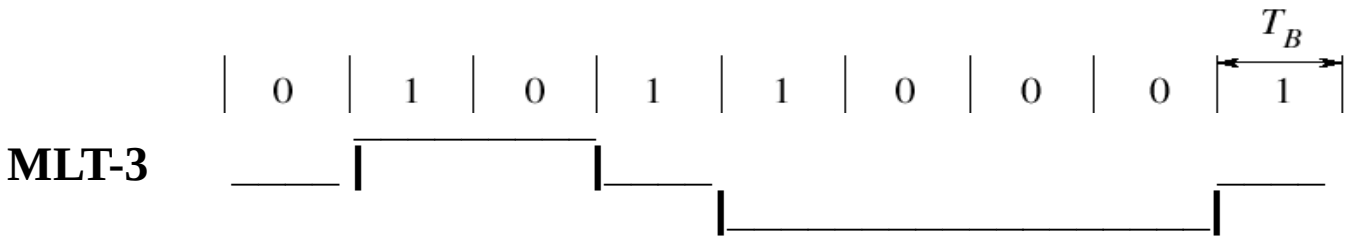
- No transition for sequences (e.g., of 0s for RZ-M)
- Bandwidth: 2 transitions/bit → f_{MAX} = BitRate

Ensure the transitions!

- The time reference is given by the transitions
  - NRZ and RZ: long sequences without transitions are possible
  - Void of time references to synchronize the RX clock
- Avoid long sequences without transitions
  - Encodings with Embedded clock (immersed clock)
    - Ternary codes (MLT), Manchester codes, ...
  - Bit stuffing (inserting bits)
    - Insertion of 0/1 in sequences of 5 consecutive 1/0
  - 4-bit groups without transitions → 5 bit with transitions
    - BxBy codes (4/5, 8/10, 32/36, ...)
- The additional transitions increase the bandwidth

MLT-3 encoding

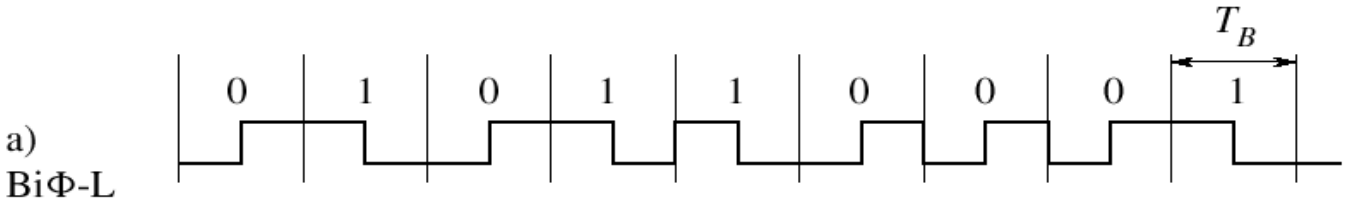
- Multilevel encodings: MLT-3 (MultiLev Trans): +, 0, –
  - 0: No change; 1: change
    - If previous state is + or –, it changes to 0
    - If previous state is 0, change to + or –, opposite to the last one
  - Ternary code, requires higher SNR



- Bandwidth (sequence of 1): f_{MAX} = BitRate / 4
- Transitions not guaranteed → requires other techniques
- Used for 100Base-T Ethernet

Manchester Codes

- Phase modulation (variations of π; Bi-Phase L)
  - 0: transition L → H
  - 1: transition H → L



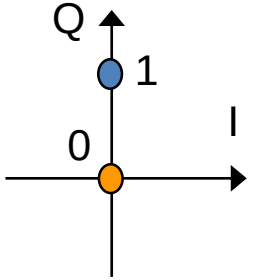
- One transition/bit (at least) → self-synchronizing
- Bandwidth: 2 transitions/bit → f_{MAX} = BitRate
- Variants
  - MFM, M2FM: Manchester :2

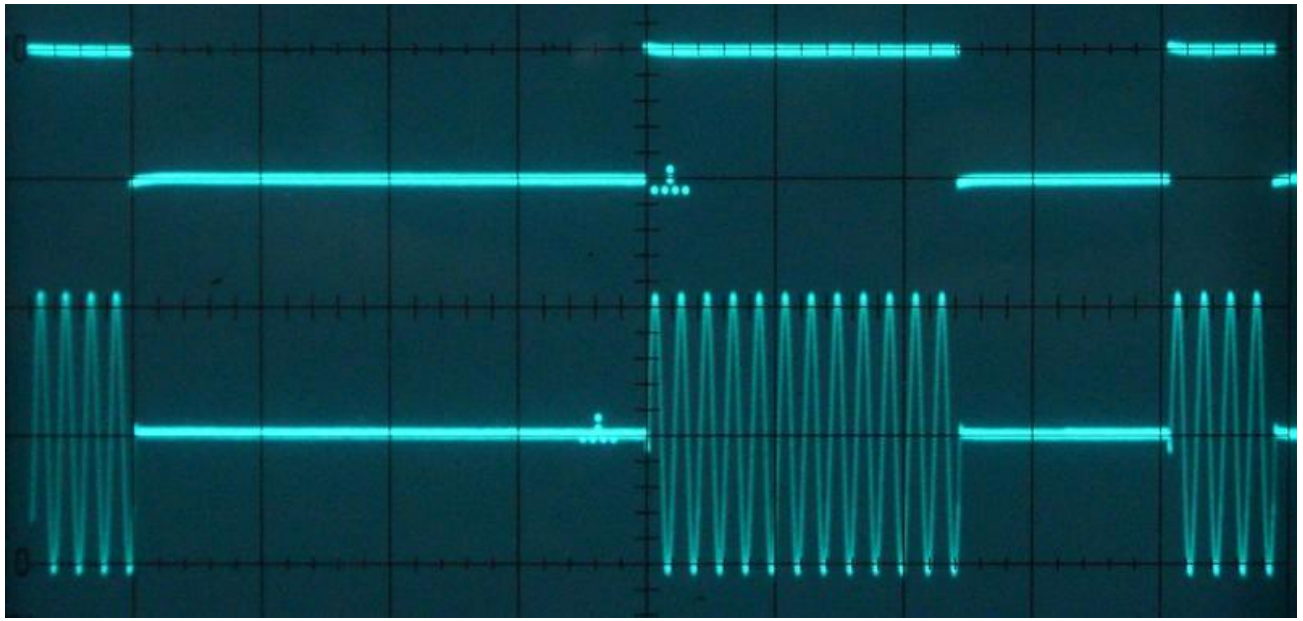
Modulations

- Conventional AM/FM modulations (analog)
  - No constraint between carrier and data rate (bit rate)
  - Non-integer carrier/bit rate ratios are possible
  - It does not allow clock recovery
- Modulations synchronous (mostly digital)
  - Integer carrier/bit rate ratios
  - Carrier phase set for each bit range
  - Clock extraction from the data is possible
    - Clock Data Recovery: CDR
    - Requires continuous presence of a carrier (even at low levels)

ASK modulation example

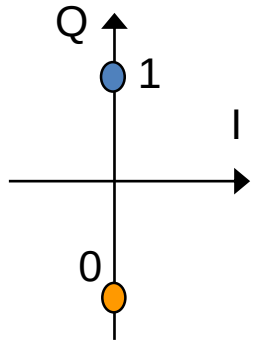
- ASK (PAM): Amplitude Shift Keyed
  - 1: carrier
  - 0: no carrier (no CDR)
    - Or reduced carrier to ensure CDR

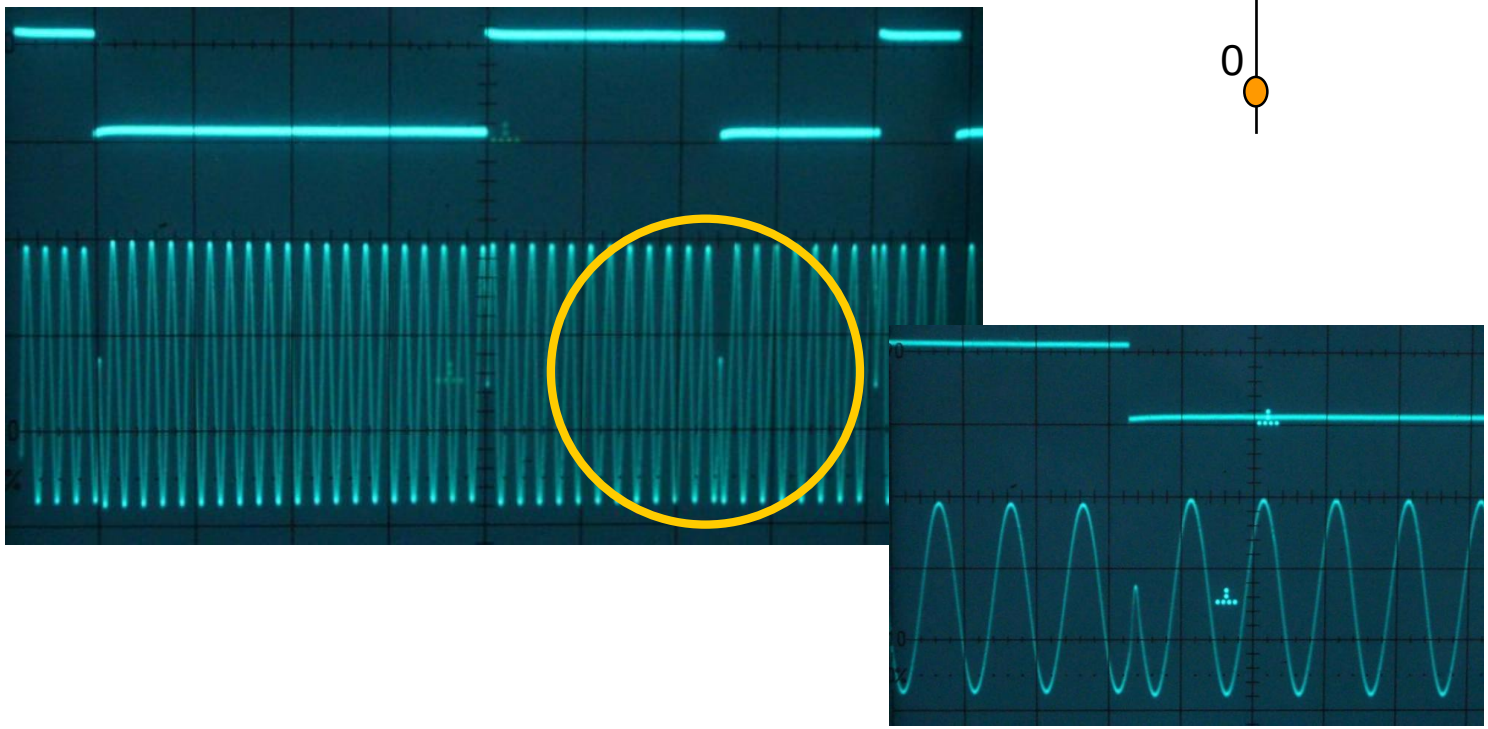




PSK modulation example

- PSK: Phase Shift Keyed
  - Carrier phase controlled by modulation
  - Carrier always present – CDR possible!





Lecture C6 - final test

- What are the advantages of a serial link?
- List the parameters that describe a serial link.
- Differences between synch. and asynch. serial links.
- What is an eye diagram?
- Params influencing the max speed of a serial link?
- Under what conditions can transmitter and receiver use independent clock generators?
- Define the maximum resynchronization interval?
- Indicate an example of encoding with embedded clock.