

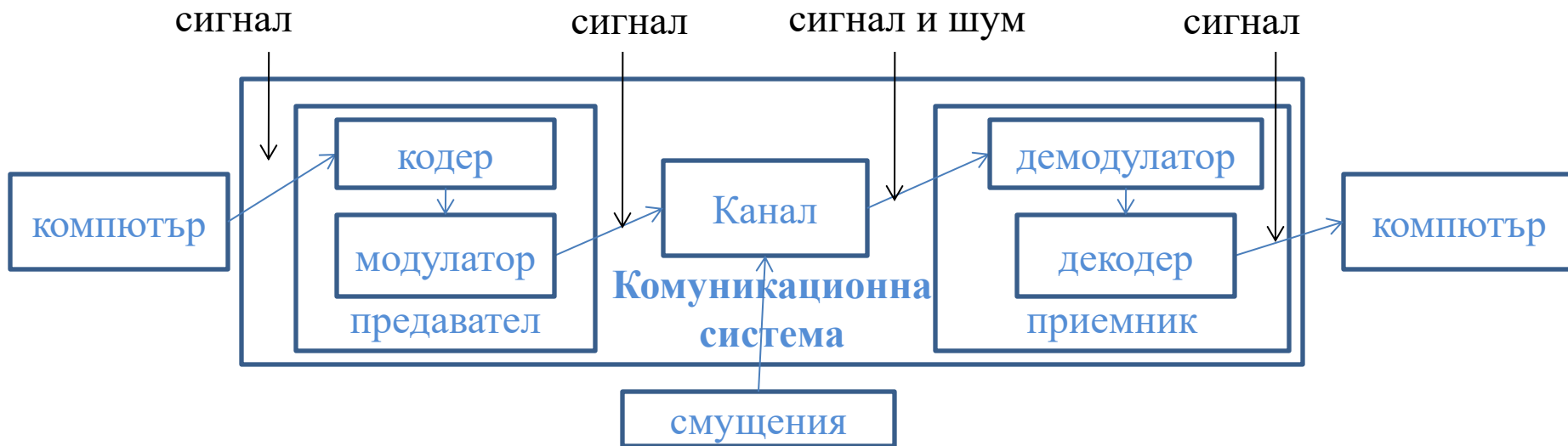
Синхронизация при предаване на данни. Асинхронно и синхронно предаване.

доц. д-р инж. Айдын Хъкъ

ОСНОВНИ МОМЕНТИ

- Синхронизация при предаване на данни.
- Асинхронно предаване.
- Синхронно предаване.

Режими на предаване



- Потоците от данни могат да се състоят от различни елементи:
 - битове – 0 / 1
 - символи – поредици от 8 бита
 - блокове – поредица от различен брой битове или символи
- Режими на предаване според вида на данните:
 - Битово
 - Байтово
 - Символно

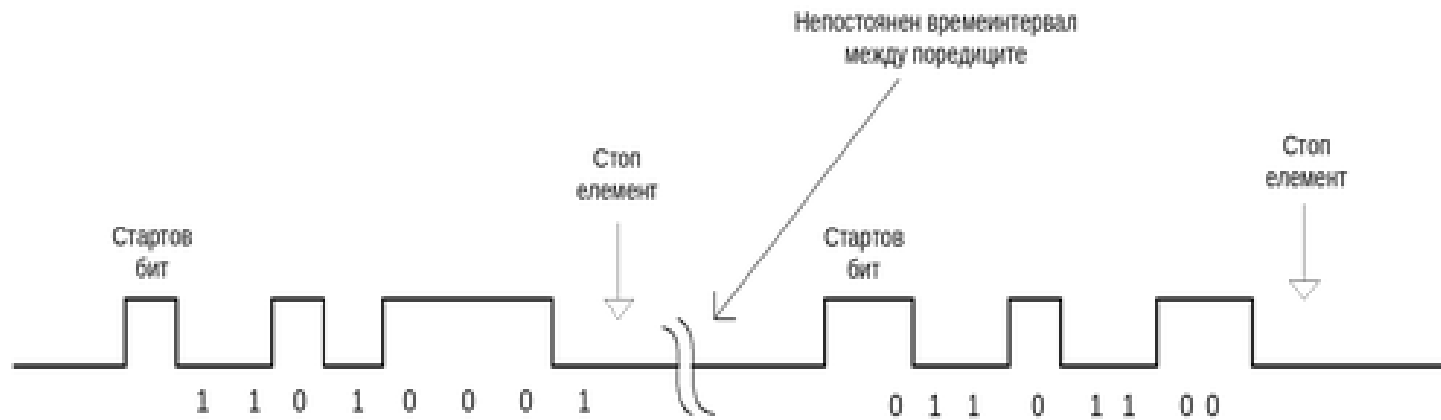
Режими на предаване

- Приемникът трябва да може да разбере, че е започнало предаване на данни и да може да се синхронизира с предавателя.
- Приемникът определя началото и края на всеки елемент на получаваните данни.
- Тази задача е известна като **синхронизация**.
- Режими на предаване, според начина на синхронизация:
 - асинхронен;
 - синхронен.

Асинхронно предаване

- Когато данните за предаване се генерират в случайни интервали от време.
 - въвеждане на текст от клавиатура
 - връзка с принтери, модеми, факсове и др.
- Обикновено всеки символ се кодира със 7 или 8 бита, със схема на кодиране ASCII и EBCDIC.
- Когато данните се предават случайно, може да има дълги интервали от време, в които да няма предаване по линията.
- Приемникът трябва да може да се синхронизира отново, когато предаването започне отново.
- Предаването и приемането се тактуват независимо и не се синхронизират.

Асинхронно предаване на последователност от символи (1)



- Не е необходимо да има времева връзка между последователни знаци (или байтове данни).
- Отделните символи могат да бъдат разделени от произволен период на неактивност.

Асинхронно предаване на последователност от символи (2)

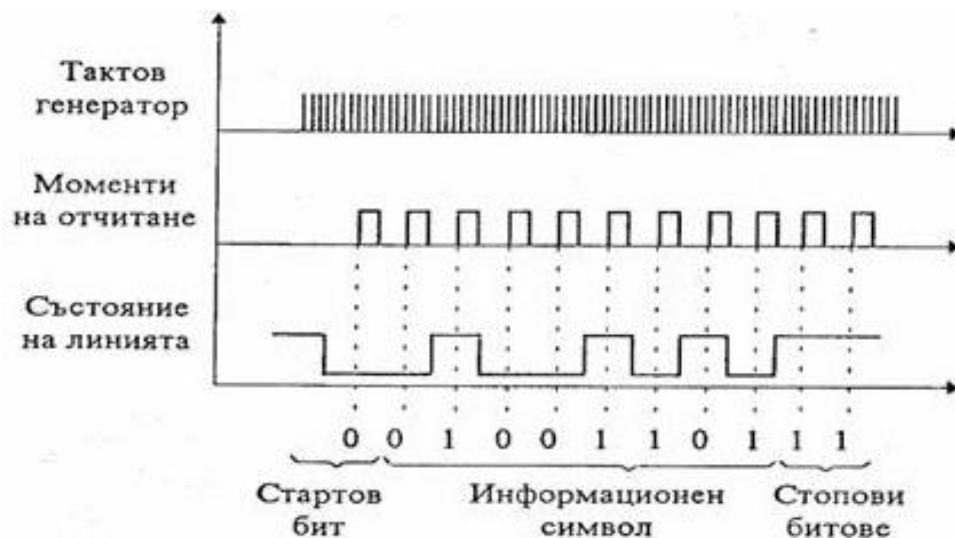
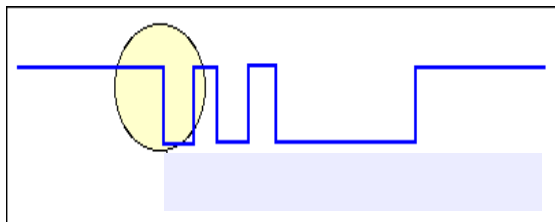


Данни - с фиксиран размер и формат:

- 1 стартов бит
- 5-8 информационни бита - започва от най-низшия бит
- Р бит - контрол по четност - опционно
- 1 или 1,5 или 2 стопови бита

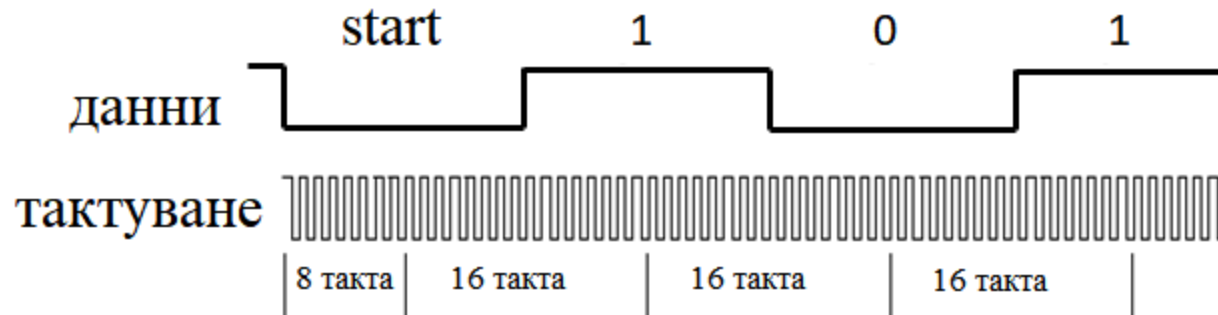
Когато няма предаване между предавател и приемник връзката е в свободно положение двоична 1-ца.

Асинхронно предаване - приемник



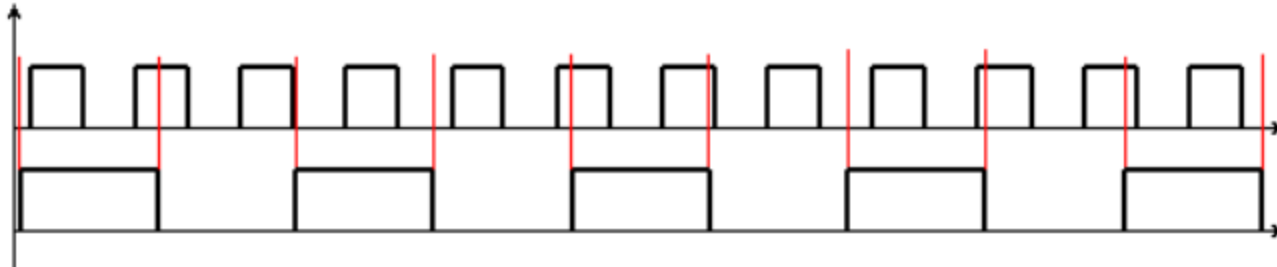
- Приемането започва по първия падащ фронт на сигнала, следващ след:
 - стопов бит,
 - период на мълчание по линията.
- Приемане - стартира се делителят на честота в приемната страна
- Отчети се вземат най-близо до средата на битовия интервал, за да се гарантира, че са преминали преходните процеси и нивото на сигнала е установено.

Асинхронно предаване - тактуване



- Падащият фронт на стартовия бит инициира тактуването.
- Приемникът тактува с N пъти по-висока скорост от скоростта на предаване на данните (обикновено $N=16$), за да е сигурно, че отчитането ще попадне близо до средата на битовия интервал.

Асинхронно предаване - грешки



- За да се отчете правилно пристигналия бит:
 - трябва да се появи точно в необходимия момент,
 - да стои неизменно във времето необходимото време,
 - да изчезне в необходимия момент.
- Изместването на отчитането от средата на битовия интервал при скорост 1Mbps с 1% след 50 бита ще доведе до “излизане” от битовия интервал – грешно отчитане.

Асинхронно предаване – блокова синхронизация



Header

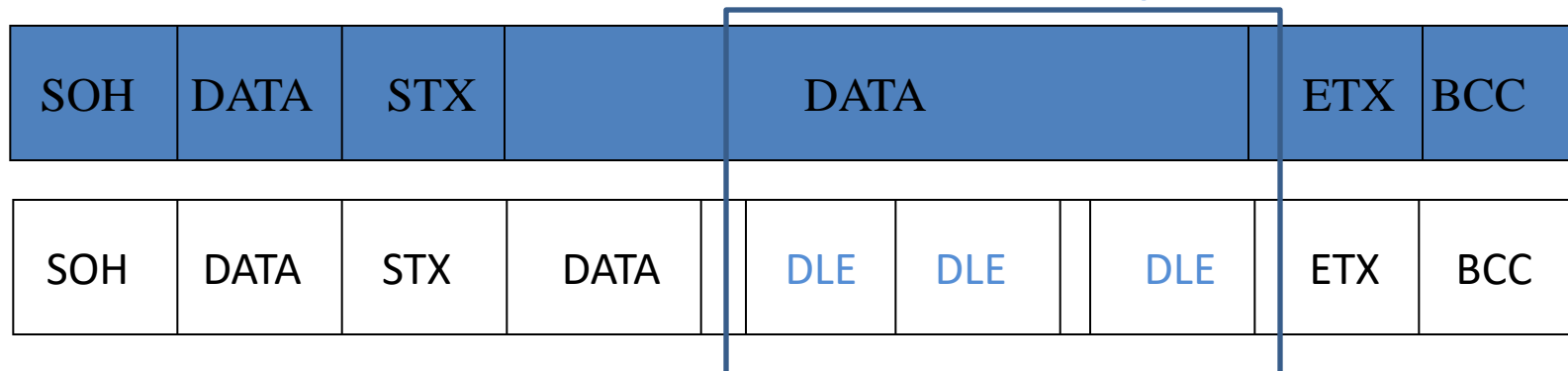
Block Control
Character

Обикновено
MOD256

Формат на блока (кадъра)

- Header - номер на кадъра, тип на кадъра, дължина на кадъра
- SOH – Start Of Heading
- STX – Start of TeXt
- ETX – End of TeXt
- BCC – Block Control Character

Асинхронно предаване –Byte Stuffing

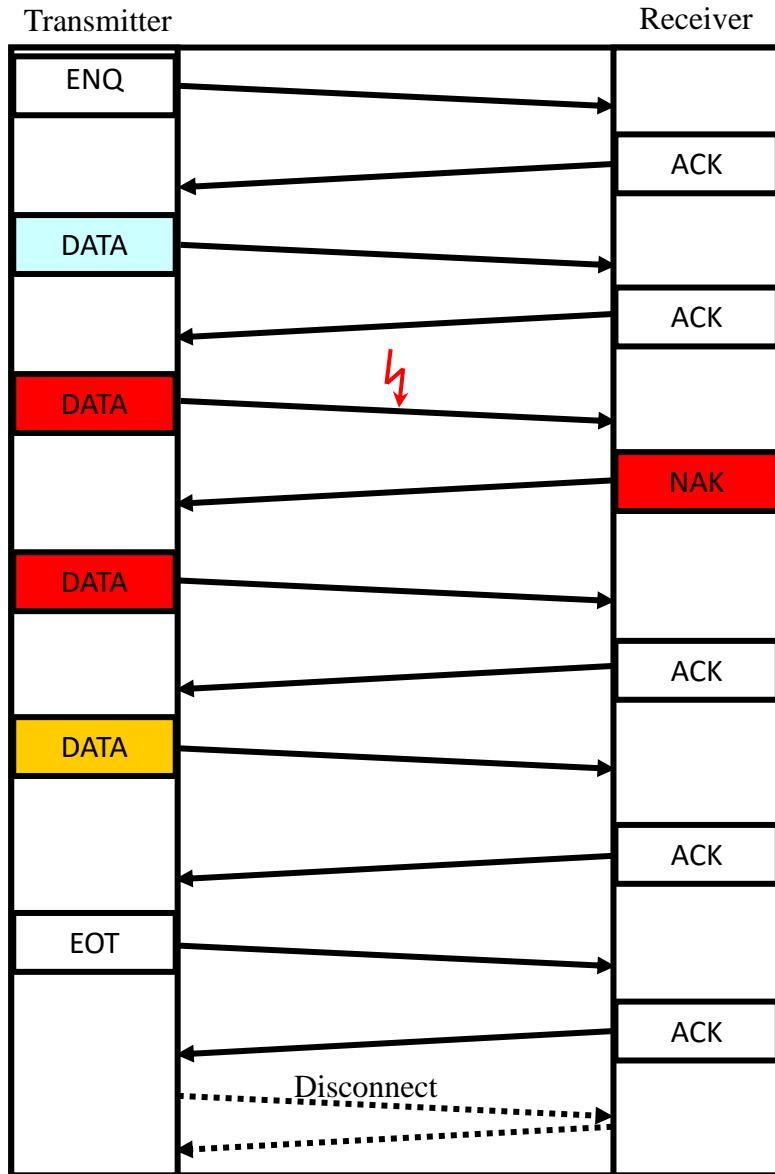


- Предаването става с техниката на вмъкване на символи - Byte Stuffing
 - Вмъкване на DLE – Data Link Escape
- Приемникът е винаги готов за приемане на данни.
- Той не заделя време за начална синхронизация.
- Взаимодействието между предавател и приемник е просто.
- Няма ограничение за броя на последователно изпратените еднакви битове.

Byte stuffing –bit stuffing

- Байт (обикновено символ ESC), се добавя към секцията с данни във фрейма, когато има символ със същия шаблон като флага.
- Всеки път, когато приемникът срещне символа ESC, той се премахва от секцията с данни и третира следващия символ като данни, а не като флаг.
- Но проблемът възниква, когато текстът съдържа един или повече символи ESC, които са част от текста. Тогава те се маркират отново от ESC, т.е. ако символът ESC е част от текста, се добавя допълнителен, който да показва, че вторият е част от текста.

Асинхронно предаване - протокол



- EOB – End Of Block
- EOT – End Of Transmission
- ACK – ACKnowledge
- NAK – Negative AcKnowledge

Недостатъци на асинхронното предаване

- Използването на стартов бит, стопови битове и бит за контрол по четност/нечетност за всеки предаден байт или символ го прави неефективен за предаване на големи съобщения.
- Методът за битова синхронизация при увеличаване на скоростта на предаване става все по-ненадежден.

Синхронно предаване

- Използва се за големи блокове от данни с високи скорости на предаване.
- Фреймовете се предават като последователен поток от битове без закъснение между всеки 8 бита.
- Приемникът се стреми да запази синхронизацията с получавания поток от битове през цялото време на получаването на данните.
- Тактуването при приемника работи в синхрон с получавания сигнал.

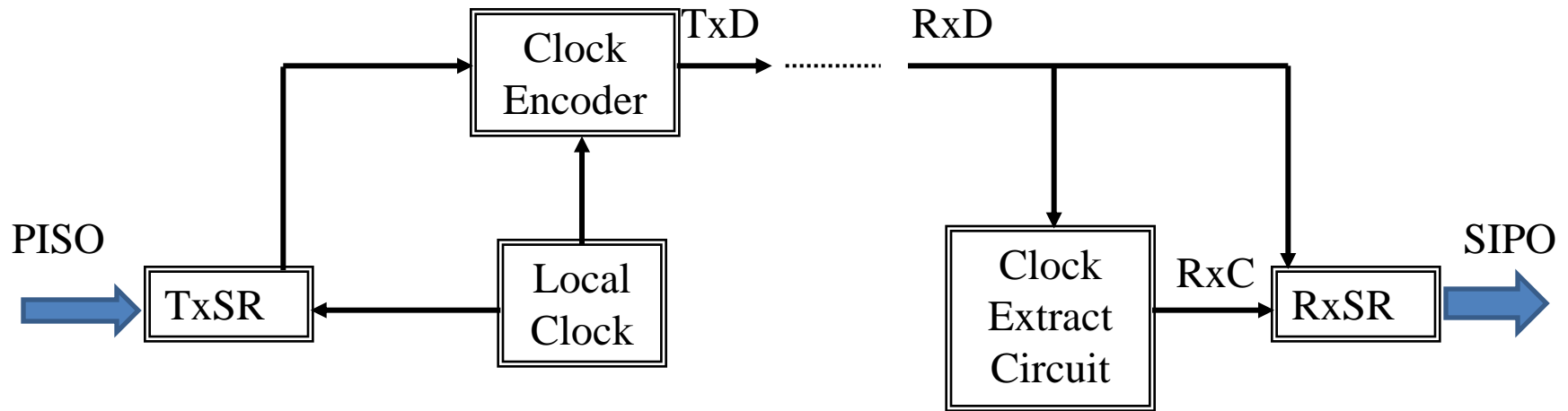
Синхронно предаване

- Предимства:
 - Значително по голяма скорост на обмен.
 - Обемът на предаваните данни значително надхвърля обема на придружаващата ги управляваща информация, което води до по-ефективно използване на канала.
 - Може да се използват по-съвършени методи за модулация.
 - Прилагането на различни типове контрол повишава достоверността на предаването, при което се облекчава възстановяването на сигнала.
- Недостатъци:
 - Увеличаване на апаратурата,
 - Необходимост от буферна памет,
 - Усложнява се обработващата логика, както в приемната, така и в предавателната част.

Синхронно предаване - битова синхронизация (1)

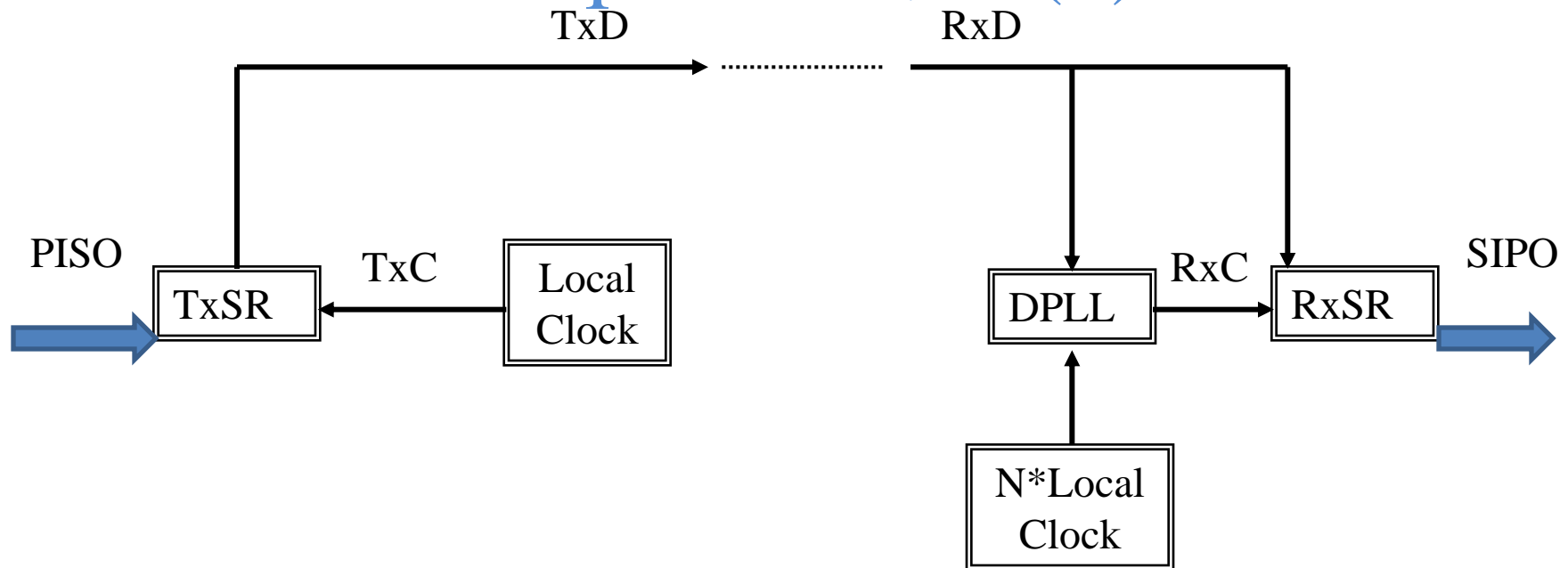
- Синхронизирането на битовете става:
 - по външни сигнали, разпространени по канала на връзка;
 - с помощта на методи за кодиране, при които фазата на информационния бит се съдържа в сигнала:
 - Със схема на кодиране с връщане към нула;
 - С преходно-ориентирана схема на кодиране.
 - синхронно работещи тактови генератори в приемащата и предаващата част. Изисква достатъчно преходи за запазване на синхронизацията:
 - Използва “вмъкване на бит” (bit stuffing);
 - Използва за NRZ схеми на кодиране.
 - с периодично предаване на синхронизиращата последователност за съгласуване на работата на вътрешните тактови генератори.

Синхронно предаване - битова синхронизация (2)



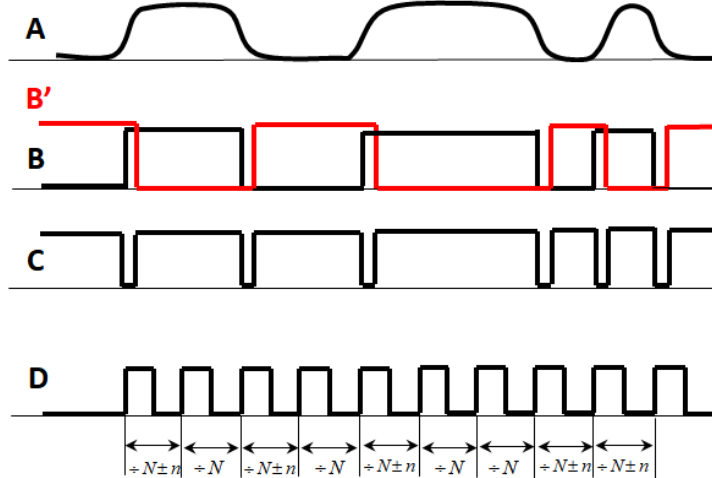
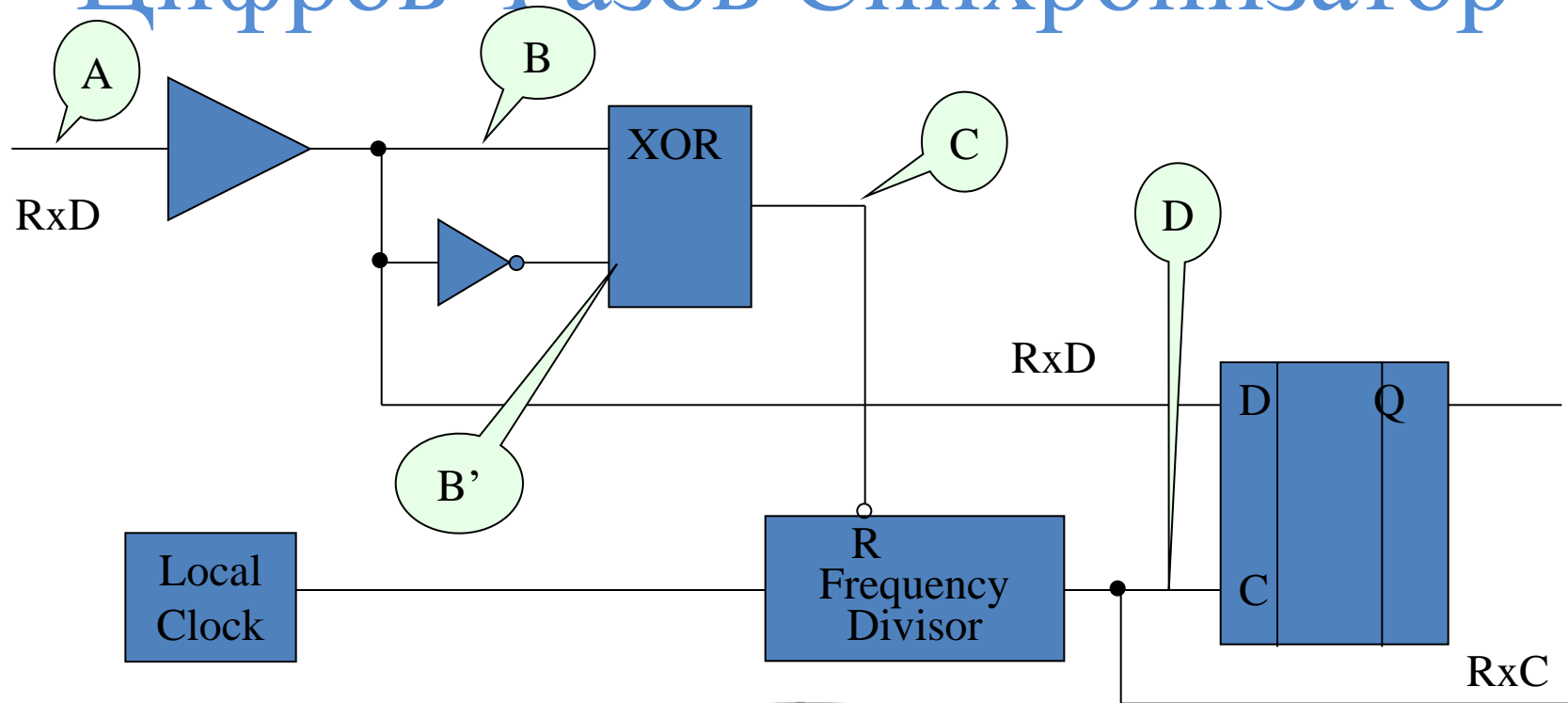
- Тактовата информация се за кодира в предавания сигнал и се извлича в приемната страна:
 - Изисква код с връщане в нула,
 - Преходно ориентирана схема на кодиране.
- Пример за самосинхронизиращи се кодове - Manchester код.

Синхронно предаване-битова синхронизация (3)

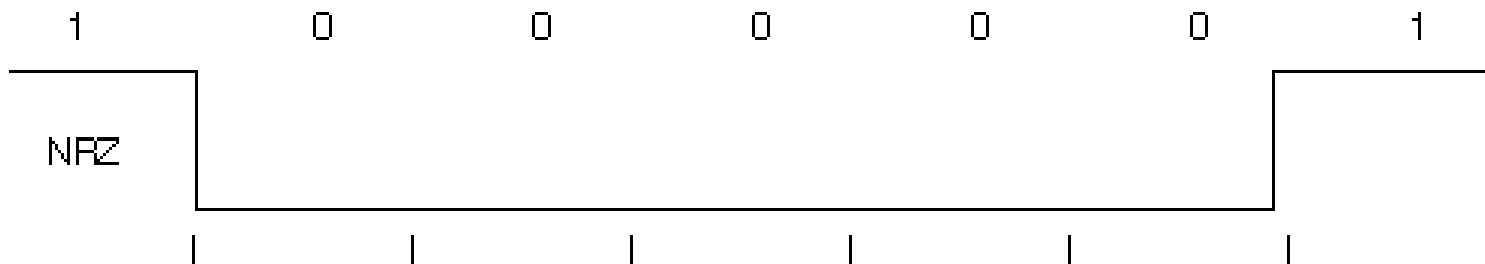


- Приемникът поддържа локален тактов генератор, който пази синхронизацията с получения сигнал, използвайки Digital Phase Lock Loop (DPLL)
 - Използва “вмъкване на битове” (bit stuffing)
- Пример
 - Може да се използва за NRZ схеми на кодиране

Цифров Фазов Синхронизатор



Non-Return to Zero (NRZ) Encoding



- При предаване на дълга поредица от символи с еднаква стойност при NRZ кодирането няма предаване по канала.
- Липсата на предаване не дава възможност на приемника с неговия DPLL надеждно да възстанови фазата, тъй като става невъзможно откриването на границите на получаваните битове от приемника.
- Затова в Ethernet LANs се използва Manchester кодиране, а не NRZ кодиране.

Техника “вмъкване на бит” (Bit Stuffing)

Оригинални данни

(a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 0 0 1 0

Предавани данни без
стартови и стопови битове

(b) 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0

Stuffed bits

Получени данни след
премахване на вмъкването

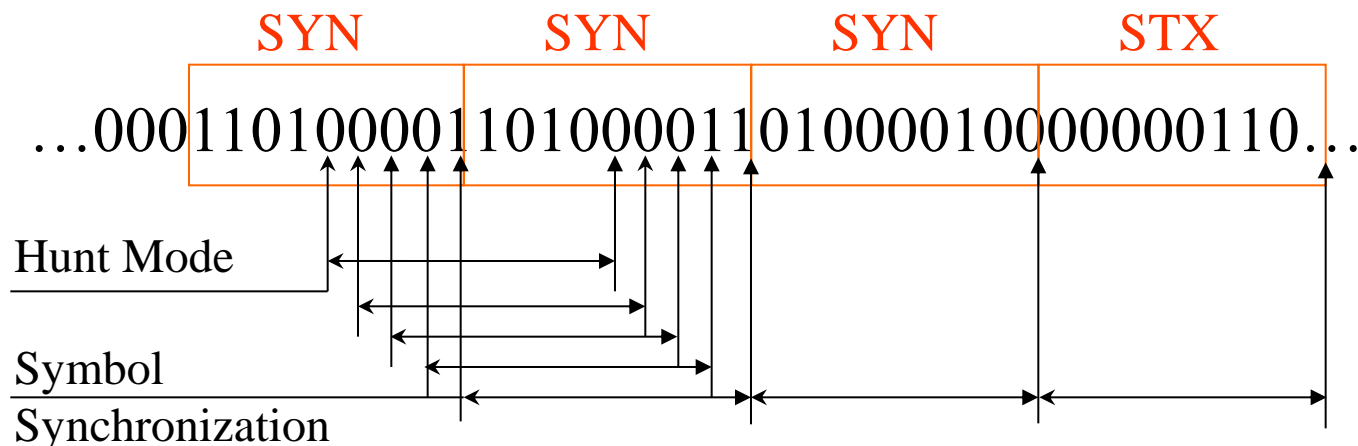
(c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

Синхронно предаване - блоково

SYN	SYN	SOH	DATA	STX	DATA	ETX	BCC
-----	-----	-----	------	-----	------	-----	-----

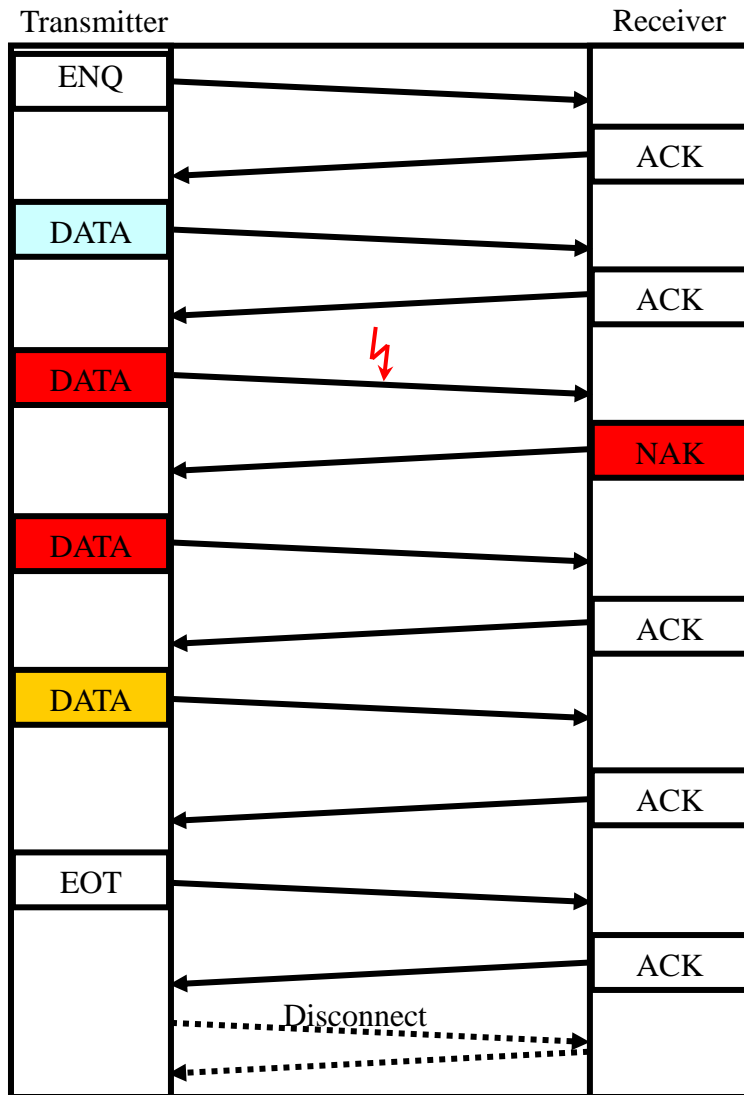
- Синхронизирането по блокове се постига чрез изпращането на специални синхронизиращи символи в началото на блока от данни.
- Контролните символи (SYN) винаги предхождат блока с данни и са два или три.
- Блокът започва със знак за започване на текста (STX).
- В края на текста (ETX) знакът прекратява блока.

Синхронно предаване - СИМВОЛНО



- Необходими са 2 до 7 SYN символа за постигане на символна синхронизация
- Техниката “вмъкване на символ” (Symbol Stuffing) се използва при предаване на двоични данни, аналогично както при асинхронното предаване.
- Пример
 - full и half-duplex мрежите използват byte-ориентирани протоколи

Байтово (символно) ориентиран протокол



- Комуникацията започва с установяване на синхронизация.
- “Договарят” се параметрите на комуникацията, които дефинират инструкциите за предаването.
- Предават се самите данни.
- Предават се АСК, валидиращи предаването.

Битово ориентирано синхронно предаване

- То е кодово независимо.
- Не е необходима символна/байтова синхронизация.
- Данните се предават като дълги последователности от битове, чийто ред не предава специфични инструкции на получателя.
- Пример:
 - Synchronous Data Link Control (SDLC)
 - High-Level Data Link Control (HDLC).

Битово ориентирано синхронно предаване – SDLC/HDLC

Flag	Address	Control	Data	FCS	Flag
------	---------	---------	------	-----	------

- **Flag** – 01111110
- **Address** – единичен, групов или broadcast
- **Control** – един или два байта
- **FCS** (Frame Check Sequence)– CRC 16 бита или 32 бита

Режими на предаване:

SDLC използва само един режим на предаване

- Normal Response Mode (NRM)

HDLC не може да работи със зацикляне. HDLC предоставя избор между три:

- Normal Response Mode (NRM),
- Asynchronous Response Mode (ARM),
- Asynchronous Balanced Mode (ABM).

Режими на предаване

- Normal Response Mode (NRM) – режим на нормален отговор – вторичната станция не може да комуникира с първичната, докато не получи разрешение от нея;
- Asynchronous Response Mode (ARM) – режим на асинхронен отговор – всяка вторична станция може да бъде инициатор на обмен, без да е необходимо разрешение от първичната;
- Asynchronous Balanced Mode (ABM) – асинхронен балансиран режим – изисква всички станции да бъдат конфигурирани като комбинирани, така че, в зависимост от ситуацията, да могат да приемат ролята на първични или на вторични. В такава мрежа всяко устройство може да предизвика обмен по всяко време, без да е необходимо разрешение.

Въпроси ?

Благодаря за вниманието !