Universitatea Politehnica Timişoara

Lecture 8. FlexRay

Event-driven and time-driven communication. Communication cycles: static segments and dynamic segments. Frame format. Physical layer.

Motivation for developing FlexRay

- Stringent safety, reliability and real-time requirements of the communication system for safety-critical driver assistance applications

- Although already established in the automotive field, CAN is unable to fulfil these requirements since it is focused on event-driven communication
  - Lack of redundant mechanism to assure fault tolerance
  - Communication speed limited to 500kbit/s in series production
  - Time-Triggered CAN did not get the expected interest

History

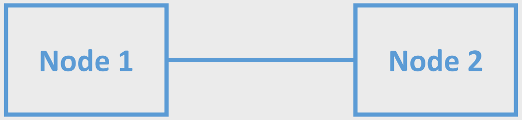
- The FlexRay Consortium was founded in 2000 by Daimler Crysler and BMW which joined with Motorola and Philips to develop a new deterministic and fault tolerant communication standard
- Version 3.0.1 of the FlexRay specification was published in 2010 and work on the ISO standardization began
- ISO 17458 (published in 2013) currently provides the FlexRay specification

Communication characteristics

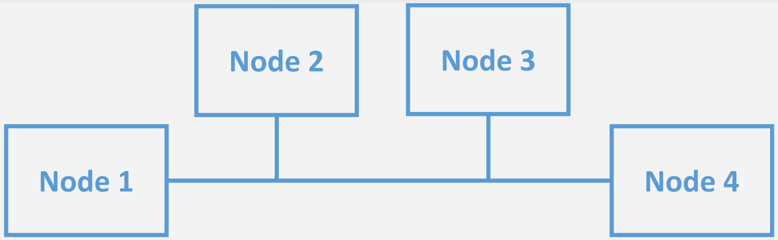
- FlexRay communication can be achieved using different topologies: line topology, passive/active star topology
- Each FlexRay node has two communication channels to enable redundancy
- When redundancy is not required channels may be used to increase data rate to 20Mbit/s
- Based on a time-triggered communication principle
- Optional support for dynamic message transmission
- All communication is performed based on a communication schedule

Passive network topologies

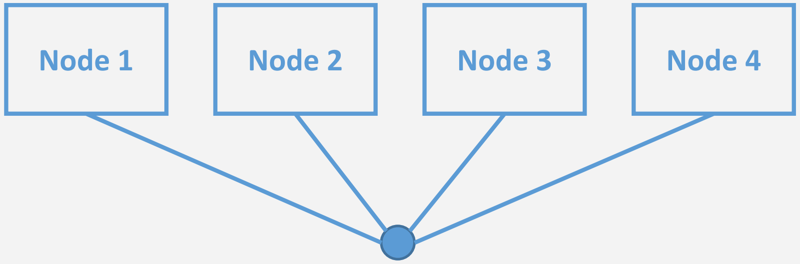
- Point-to-point connection



- Line topology

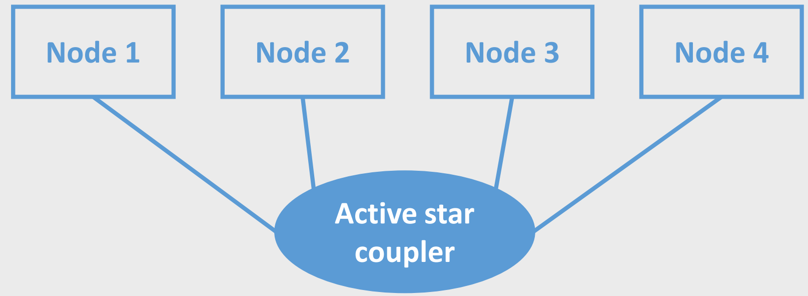


- Passive star topology

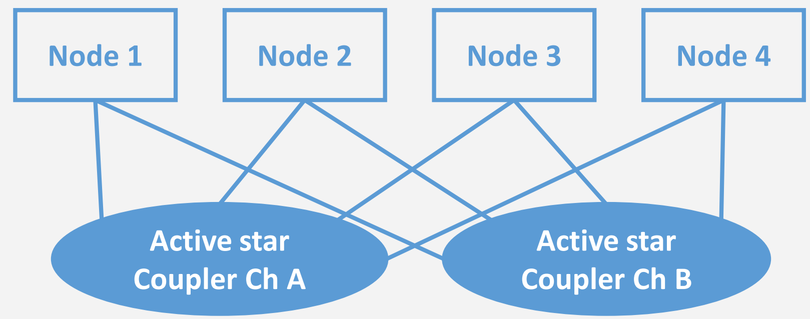


Active network topologies

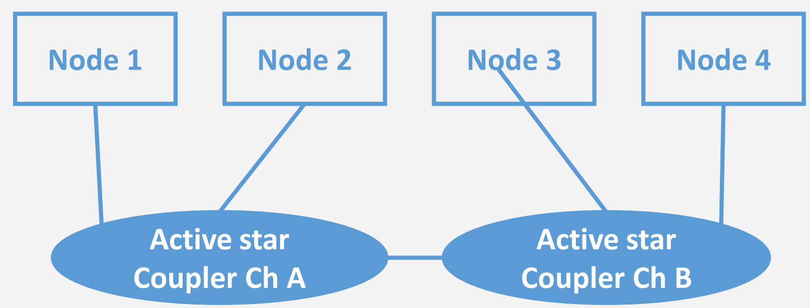
- Active star topology



- Derived topologies
  - Active star with redundant bus

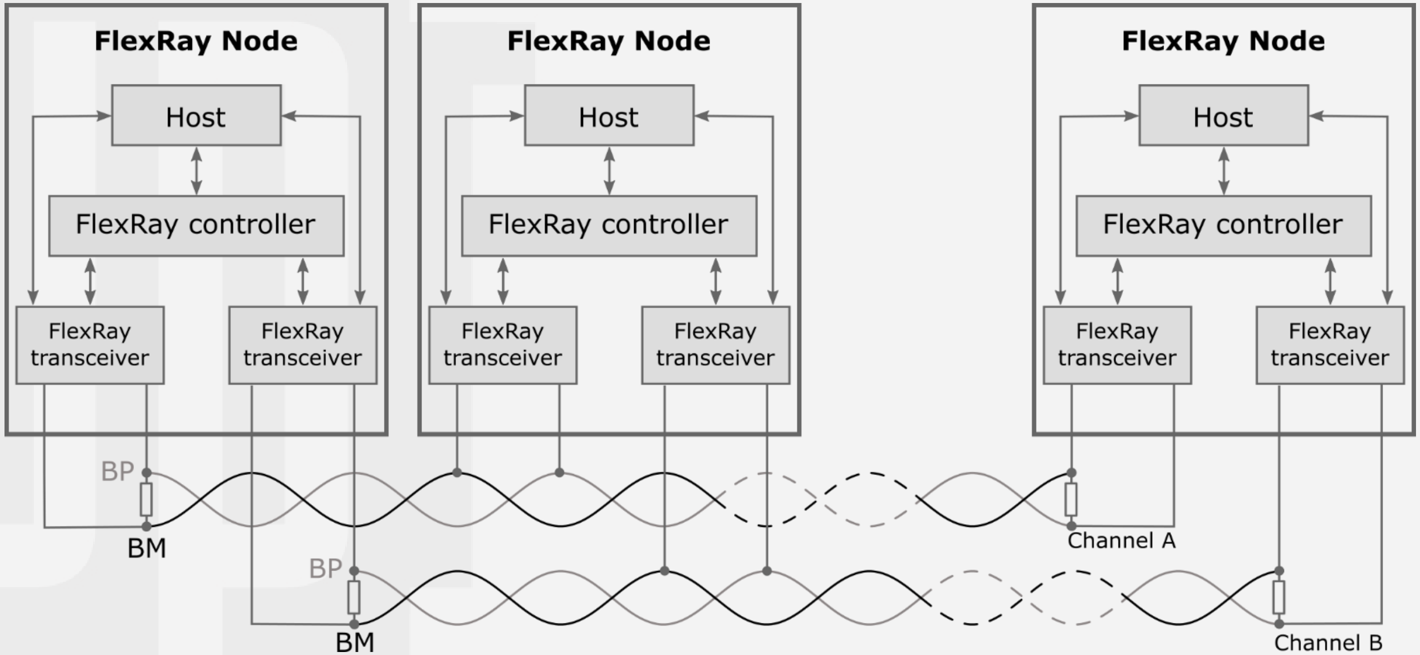


  - Cascaded active star topology



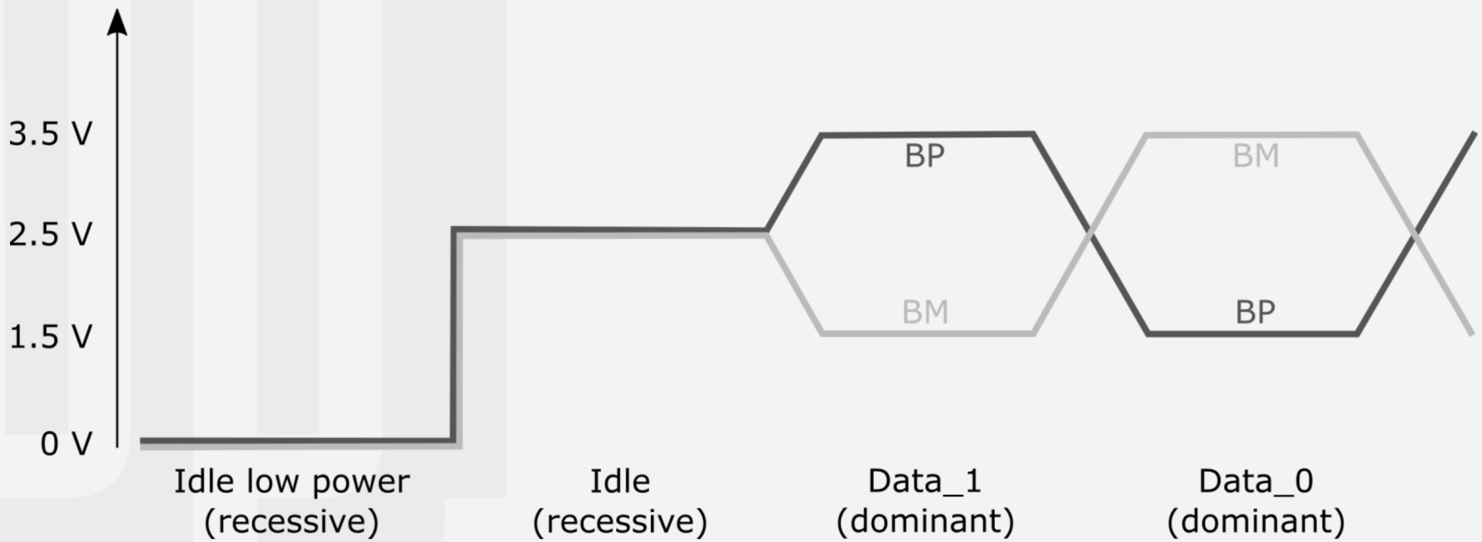
FlexRay OSI layers

- The FlexRay specification covers the Data link and physical layers of the OSI model
- A FlexRay node is consists of an ECU that is connected to the communication line through a FlexRay controller and two FlexRay transceivers (channel redundancy)
- The controller handles the Data link layer of the FlexRay protocol
- The transceivers is responsible with the physical signalling on the bus



FlexRay bus

- FlexRay uses differential signal transmission and NRZ encoding
- Two communication lines: Bus Plus (BP) and Bus Minus (BM)
- Termination registers with values between 80 and 110 ohm must be used
- The physical layer defines four bus levels
  - Dominant – differential voltage not equal to 0 V (Data_1, Data_0)
  - Recessive – differential voltage of 0V (Idle & low power)



Bus access principle

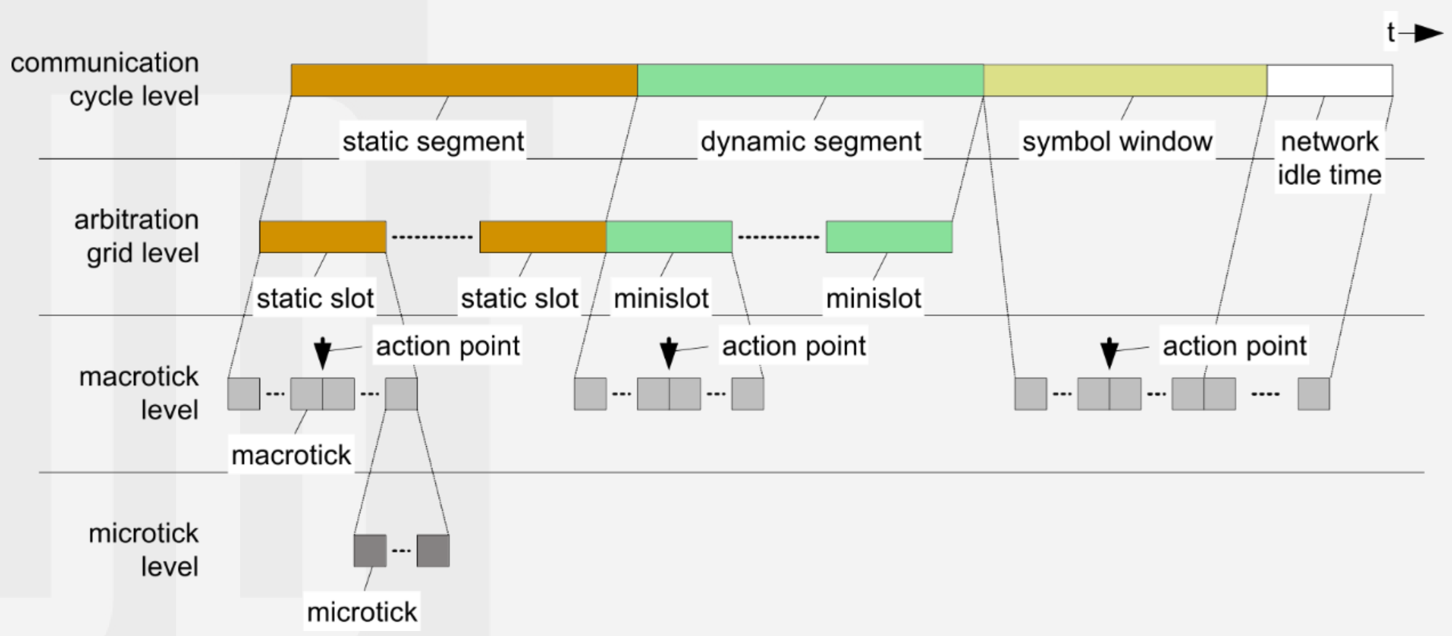
- FlexRay uses two methods for granting bus access to nodes
  - TDMA (Time Division Multiple Access)
  - FTDMA (Flexible Time Division Multiple Access)

- The TDMA method uses a communication schedule which is split into time slots
- Each FlexRay node has one or more slots assigned in which it is granted access to the bus
- The communication schedule is repeated periodically by all nodes

- The FTDMA method is used to implement a dynamic segment which reserves a specific slot in the communication cycle that can be used by any node to transmit messages in a non-deterministic manner

Communication cycle

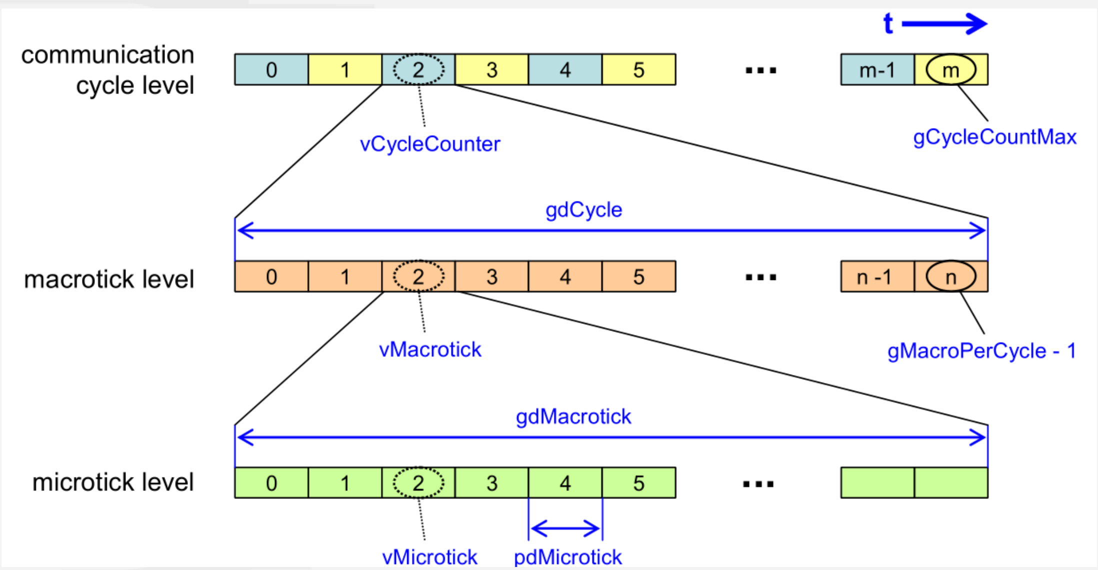
- A communication cycle must consist of at least two time segments: static segment and network idle time (NIT)
- Optional segments: dynamic segment and symbol window



(After FlexRay specification 3.0.1)

Timing hierarchy

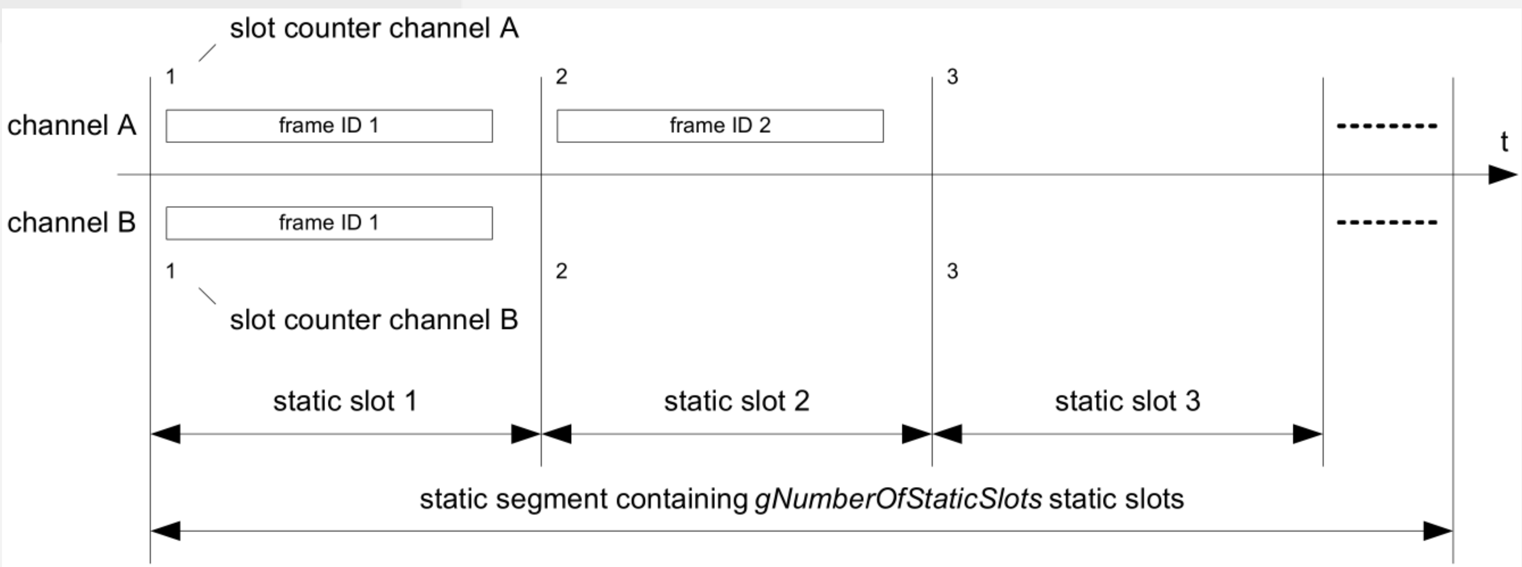
- Macroticks
  - A communication cycle consists of an integer number of macroticks
  - Established as a result of the cluster-wider clock synchronization
  - Always consist of an integer number of microticks
- Microticks
  - Derived from the internal node oscillator
  - Node’s internal time granularity



(After FlexRay specification 3.0.1)

Static segment

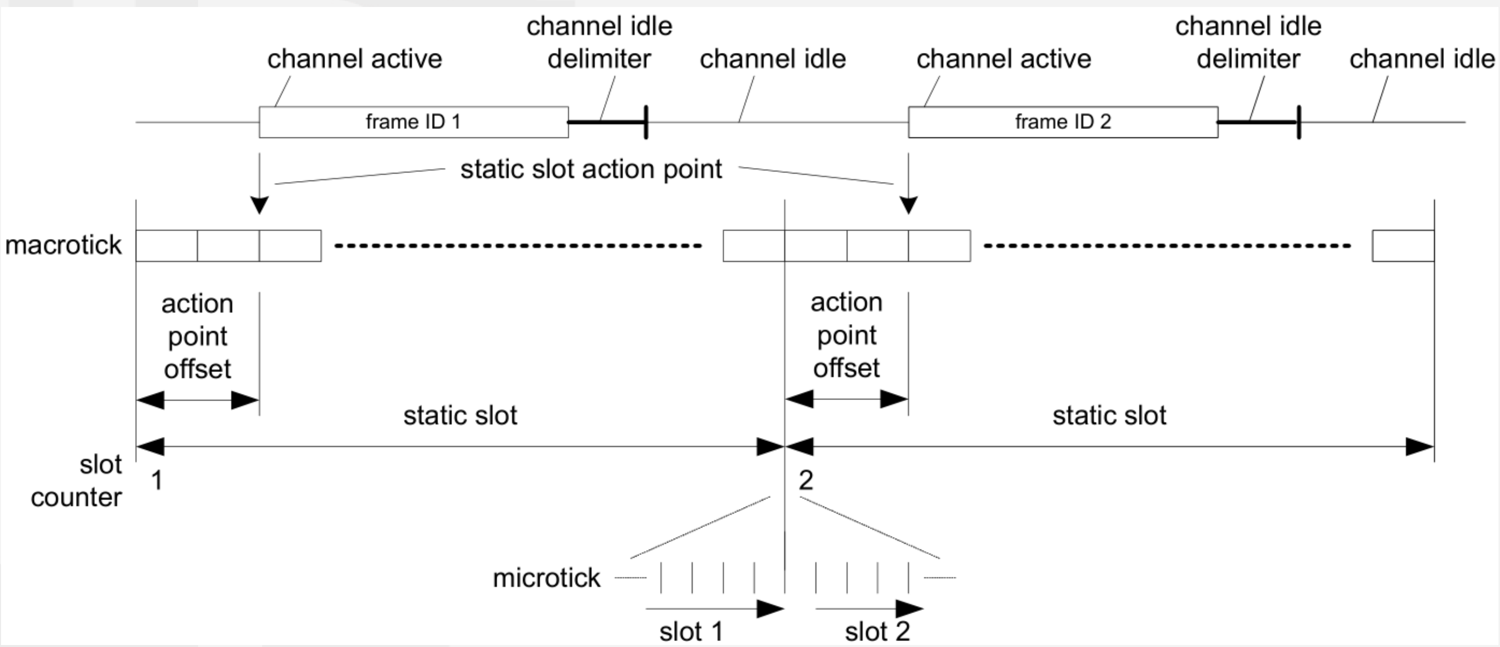
- Assures equidistant data transmission
- Organized in slots of equal length
- Nodes transmit messages in the static slots assigned to them
- A maximum of 1023 static slots can be defined on each channel
- Local counters are used to monitor static slot precedence



(FlexRay specification 3.0.1)

Static slot

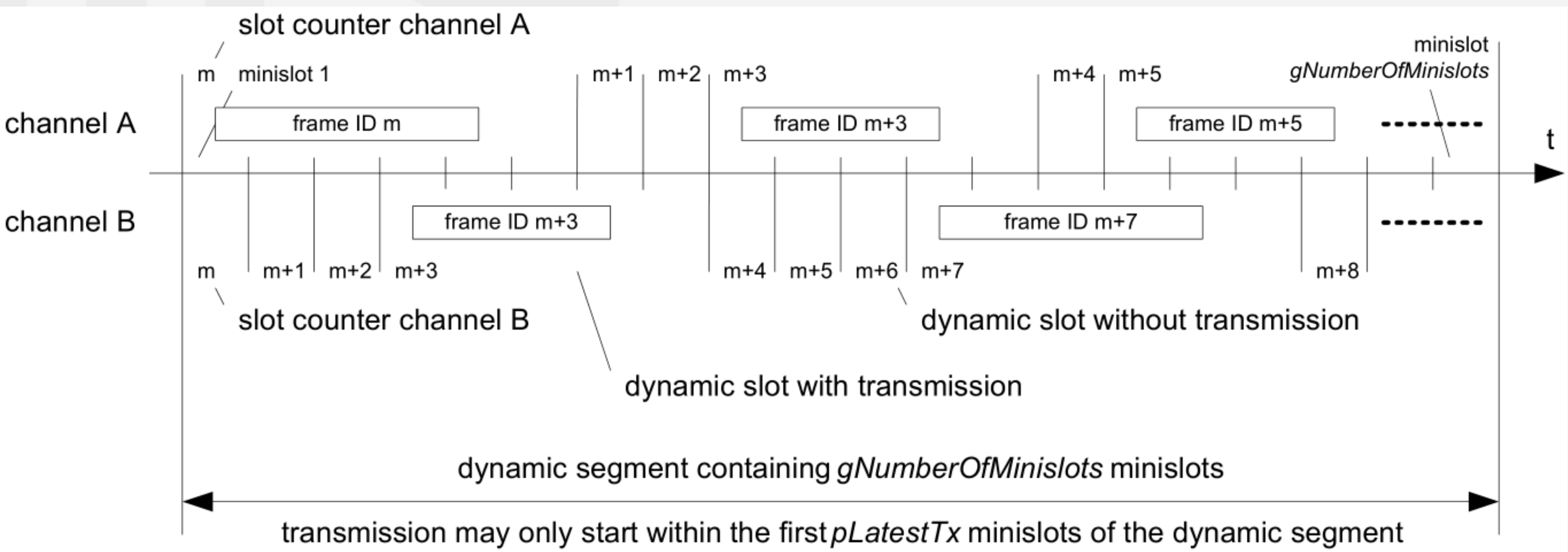
- The length of a static slot is determined by
  - The longest FlexRay message
  - Largest transmission delay (up to 2,5 μs)
  - Clock skew between nodes
- A static slot consists of four segments: Action Point Offset, FlexRay Frame, Channel Idle Delimiter (11 recessive bits) & Channel Idle



(FlexRay specification 3.0.1)

Dynamic segment

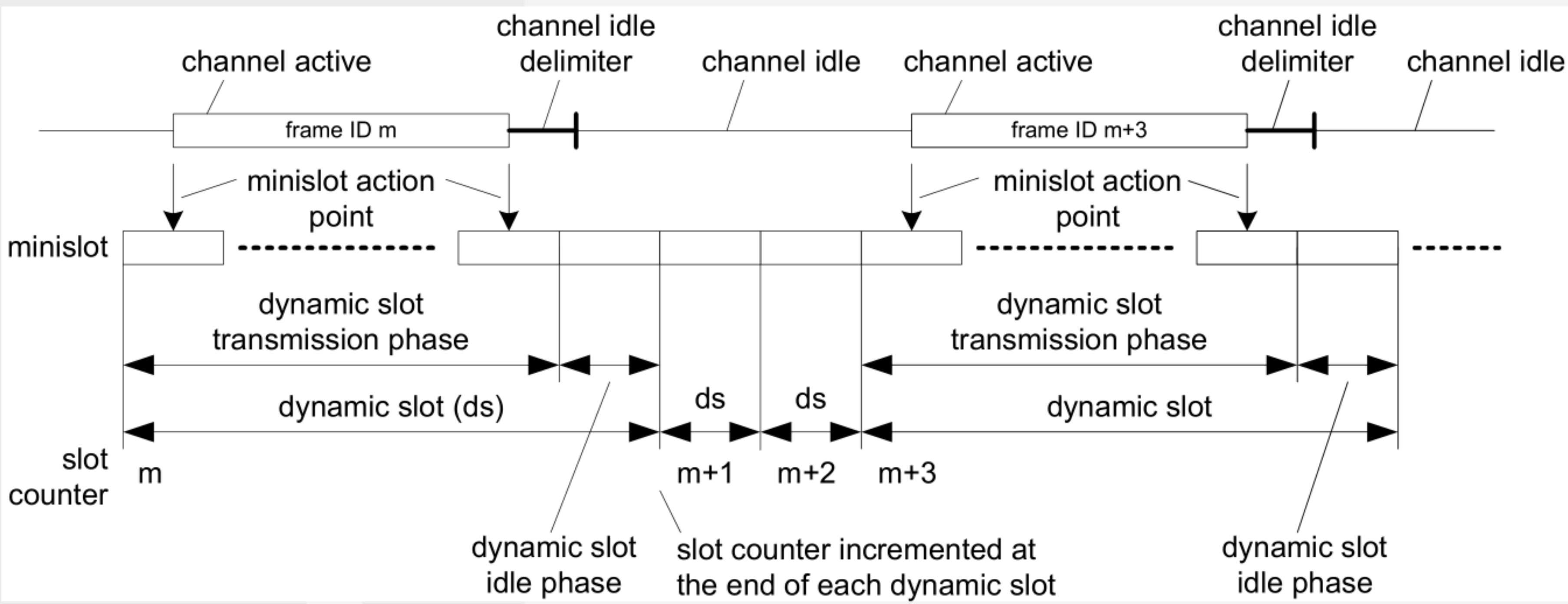
- Used to transmit on-event messages
- Always has the same length and follows the static segment
- Also based on a schedule but transmission is done only if the corresponding events occur
- Split in minislots which are occupied by Dynamic slots used to send messages until no space is left to send other dynamic messages



(FlexRay specification 3.0.1)

Dynamic slot

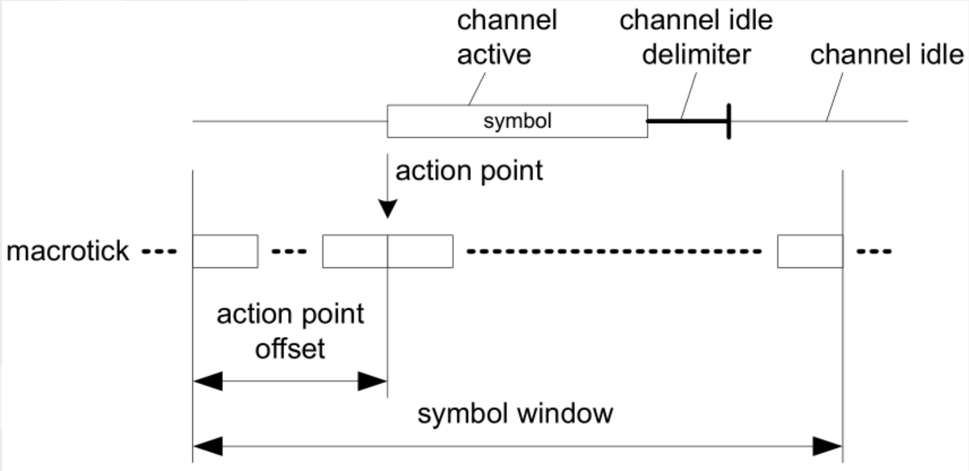
- Similar to static slots
- Messages can have different payload sizes
- An additional field – Dynamic Trailing Sequence (DTS) – is added to assure alignment to minislot timings



(FlexRay specification 3.0.1)

Symbol window

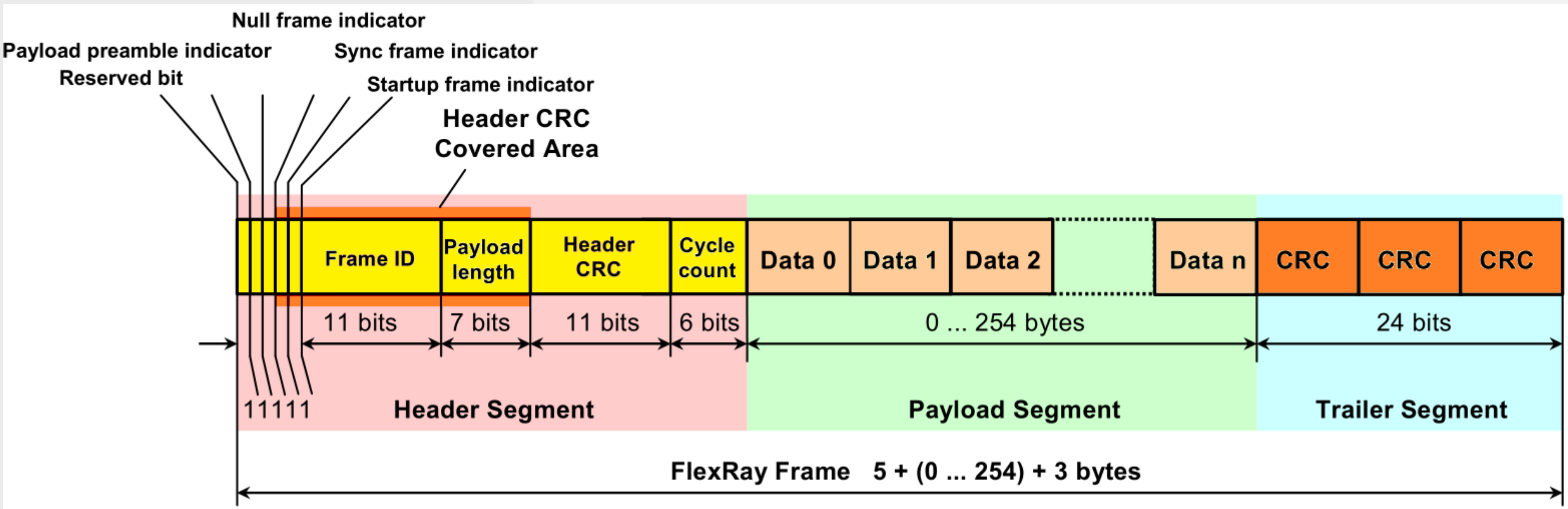
- Used to transmit special bit patterns called symbols
- Symbols are used for establishing communication and network maintenance
- Flexray Symbols:
  - Collision avoidance symbol – used by coldstart nodes to start communication
  - Media test symbol – used to test the bus guardian
  - Wake-up symbol – used to initiate the wake-up process



(FlexRay specification 3.0.1)

FlexRay frame

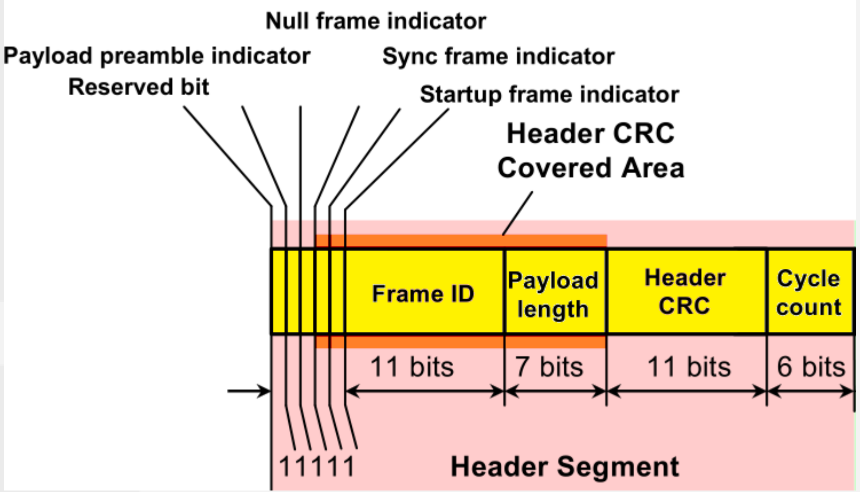
- Consists of three parts: header, payload and trailer



(FlexRay specification 3.0.1)

FlexRay frame - Header

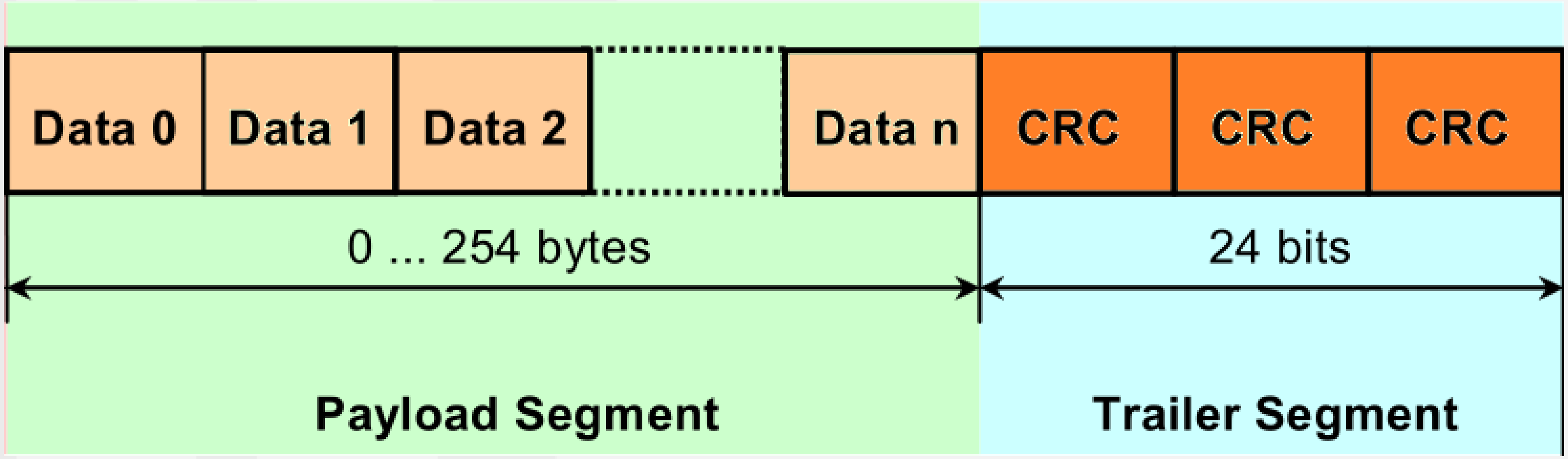
- Indicator bits – specify the message type
  - Startup frame indicator – sent in the static segment as a startup frame
  - Sync frame indicator – sent in the static segment for synchronization
  - Null frame indicator – specifies if payload is valid or not
  - Payload Preamble Indicator – specifies if a network management vector is sent in the payload
- ID – Frame identifier
- Payload length – in words up to 127 → up to 254 bytes
- CRC – Computed over the ID, payload length and some indicator bits
- Cycle count – the current communication cycle count (counts up to 63)



(FlexRay specification 3.0.1)

FlexRay frame – Payload and Trailer

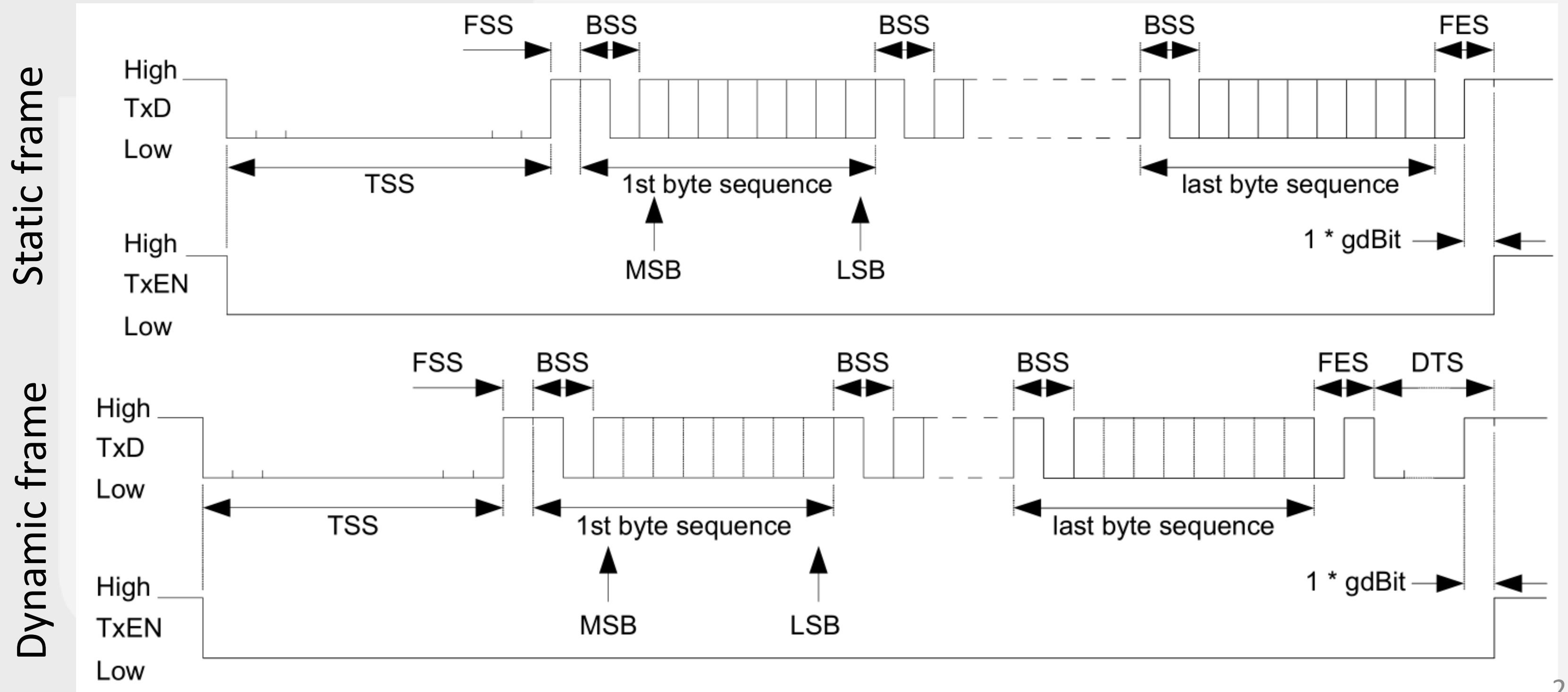
- All messages in the static segment have the same payload size
- A message sent in the static segment can use the first 12 bytes in the payload to send the Network Management Vector
- The trailer CRC sequence is computed over the header and payload



(FlexRay specification 3.0.1)

FlexRay frame encoding

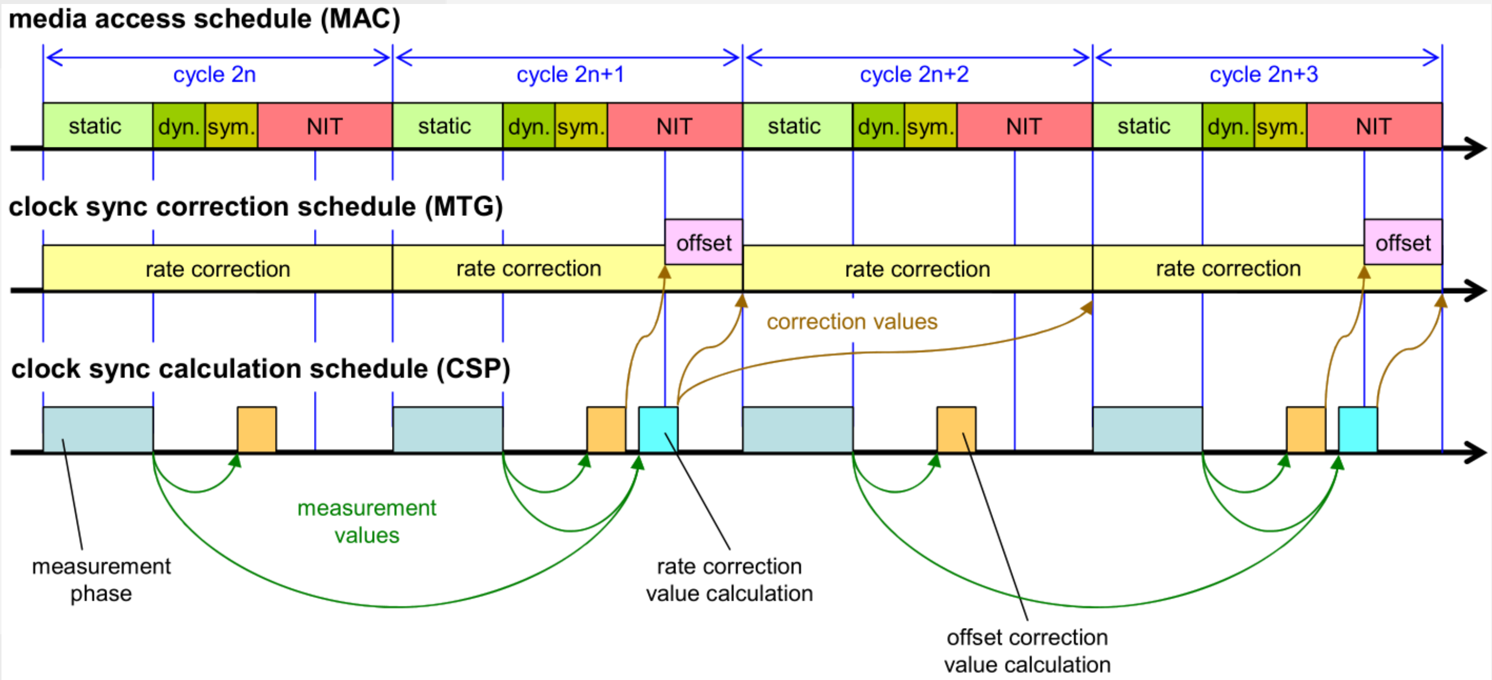
- TSS = Transmit Start Sequence (5-15 low bits)
- FSS = Frame Start Sequence (one high bit)
- BSS = Byte Start Sequence (one high bit followed by one low bit)
- FES = Frame End Sequence (one low bit followed by one high bit)



(FlexRay specification 3.0.1)

FlexRay synchronization

- Synchronization required to assure reliable FlexRay communication
- Communication cycles and slots must begin at the designated time and should always have the same length
- FlexRay nodes regularly correct their local clocks :
  - Phase or offset correction
  - Frequency or rate correction



(FlexRay specification 3.0.1)