LIN protocol description

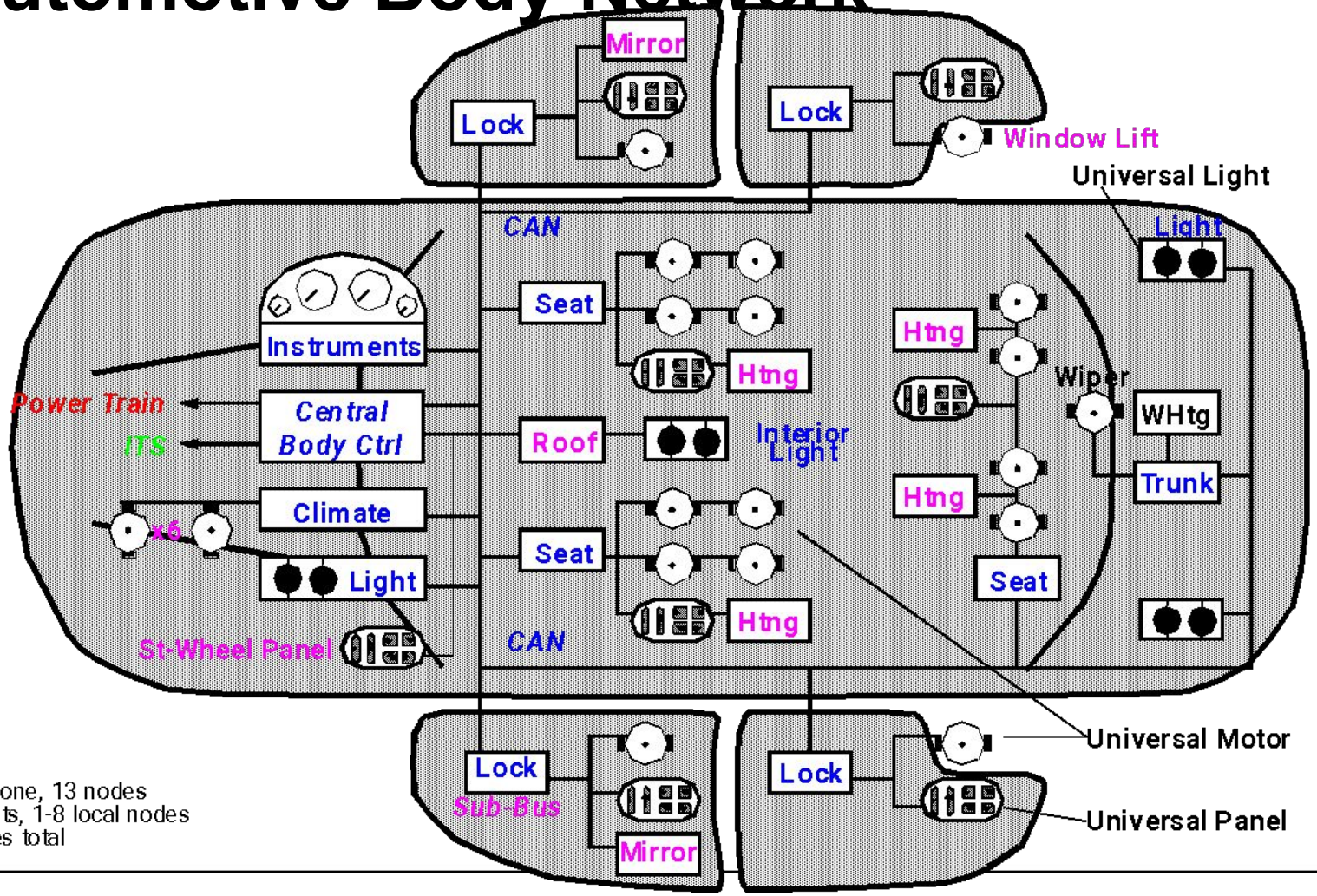
*

TSPG – 8/16 bit Products Division

MOTOROLA intelligence everywhere™

digital dna™

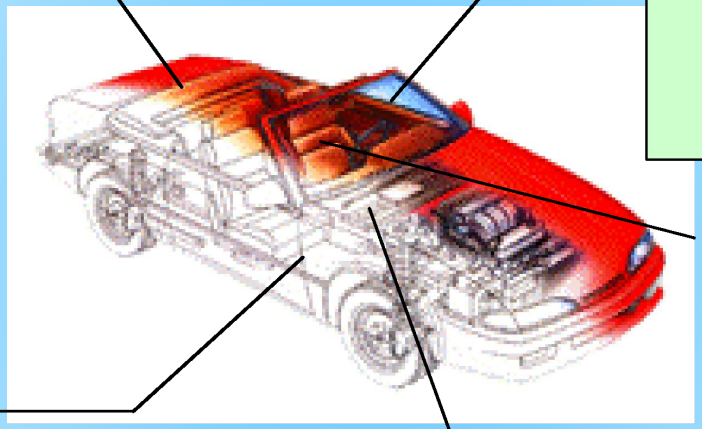
Automotive Body Network



Typical LIN Applications

Roof:
(high amount of wiring)
Rain Sensor, Light Sensor, Light Control, Sun Roof
…
(Rain Sensor needs to be interrogated every 10-20ms)

Steering Wheel:
(very many controls are going to be positioned on the steering wheel)
Cruise Control, Wiper, Turning Light, …
Optional: Climate Control, Radio, Telephone, etc.

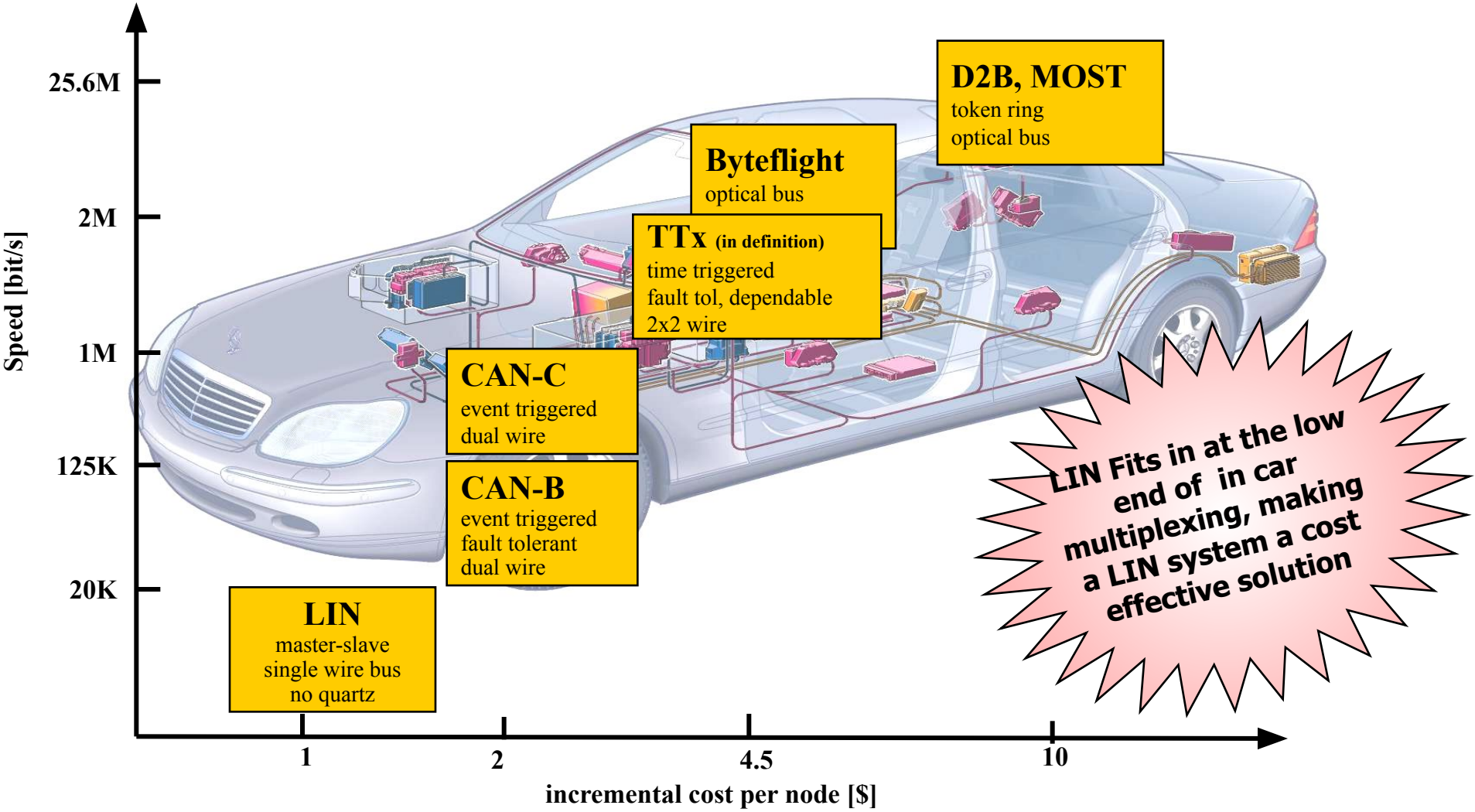


Seat:
many Seat Position Motors, Occupancy Sensor, Control Panel

•Door/window/seat:
Mirror,Central ECU, Mirror, Switch, Window Lift,
Seat Control Switch, Door Lock, etc.

Climate:
many Small Motors
Control Panel

MUX Standards (Costs and Speeds)



*
TSPG – 8/16 bit Products Division

MOTOROLA intelligence everywhere™

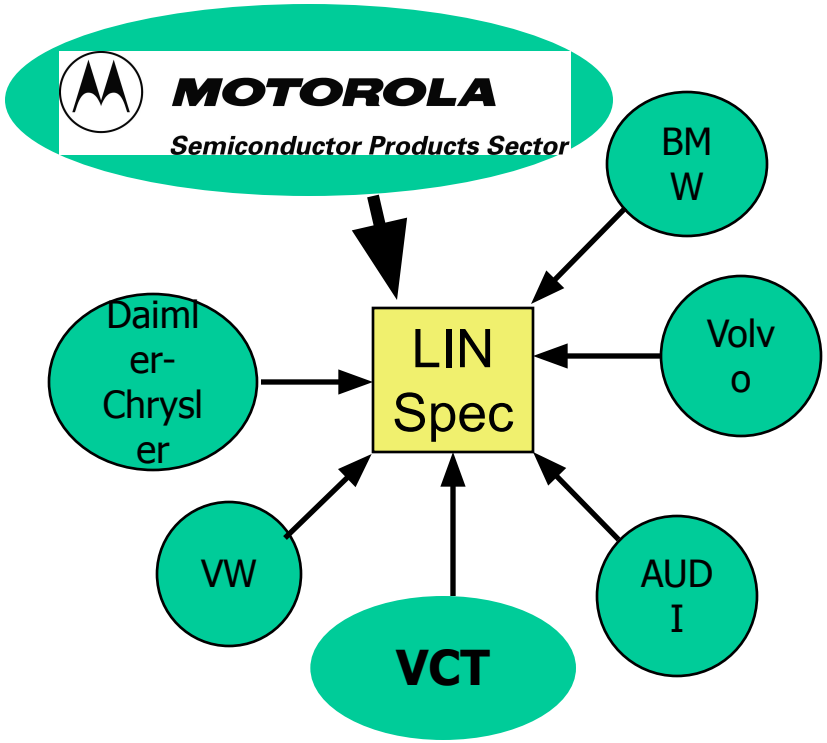
digital dna™

LIN Consortium

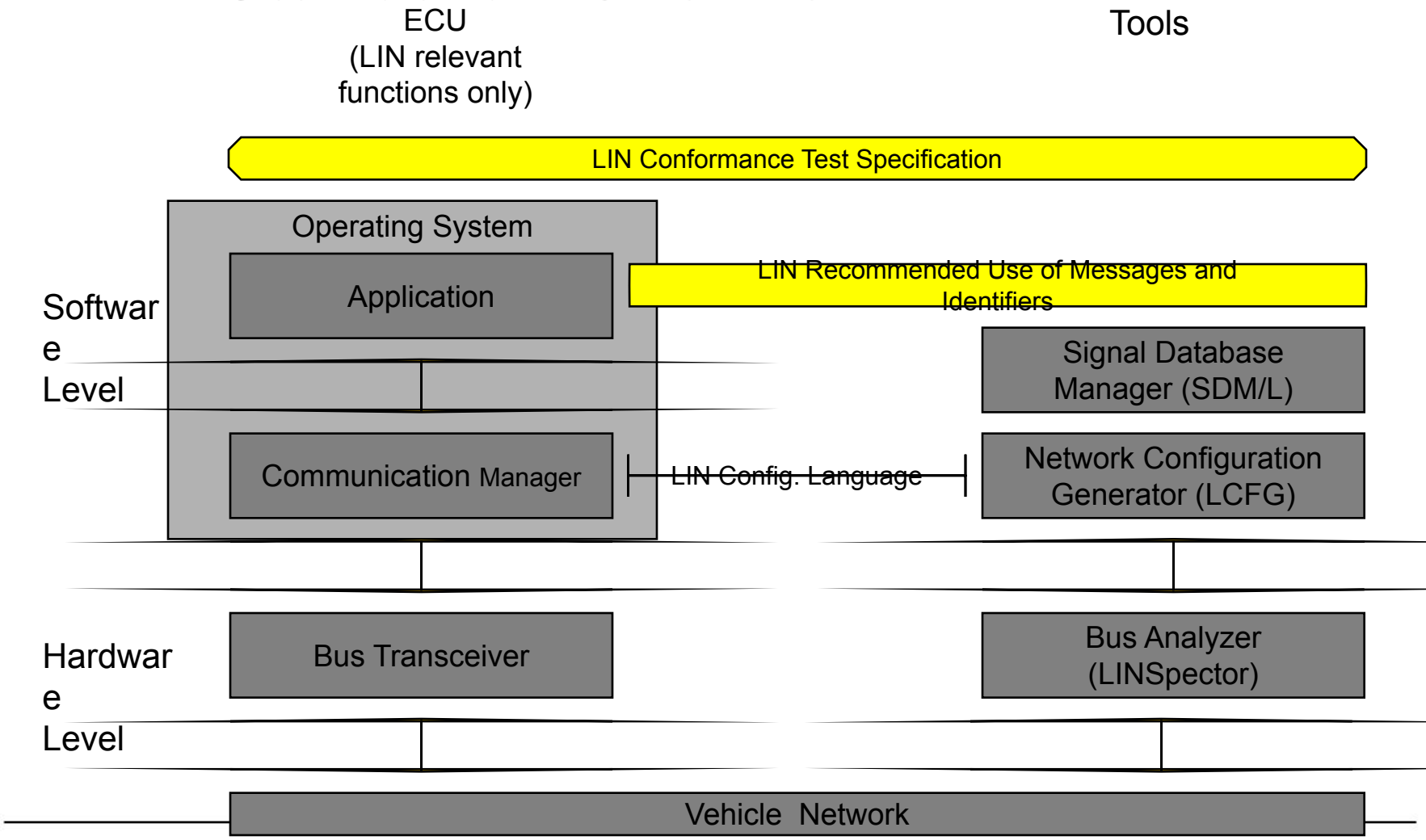
Consortium formed in 1998.
Five Car manufacturers
ONE Semiconductor Supplier (Motorola)
One tool Supplier (VCT)

Specification finalised on 02/02/00
Official Launch at SAE March '00
Open Specification.

Motorola Ready to support LIN with extensive
device families and new parts already in the discussion/ spec finalization loop.
First dedicated LIN part available Q3 '00



LIN Standard - Overview



*
TSPG – 8/16 bit Products Division

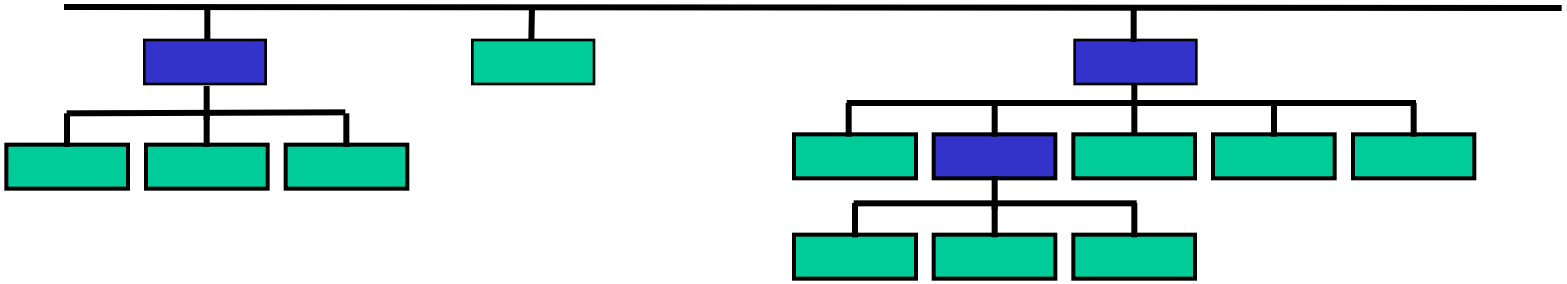
MOTOROLA intelligence everywhere™

digital dna™

Hierarchical Network Structure

Flat Network

Hierarchical Network



- Subnets are necessary to reduce Busload on main Bus
- Solution examples:

•CAN

+Automotive Standard Bus

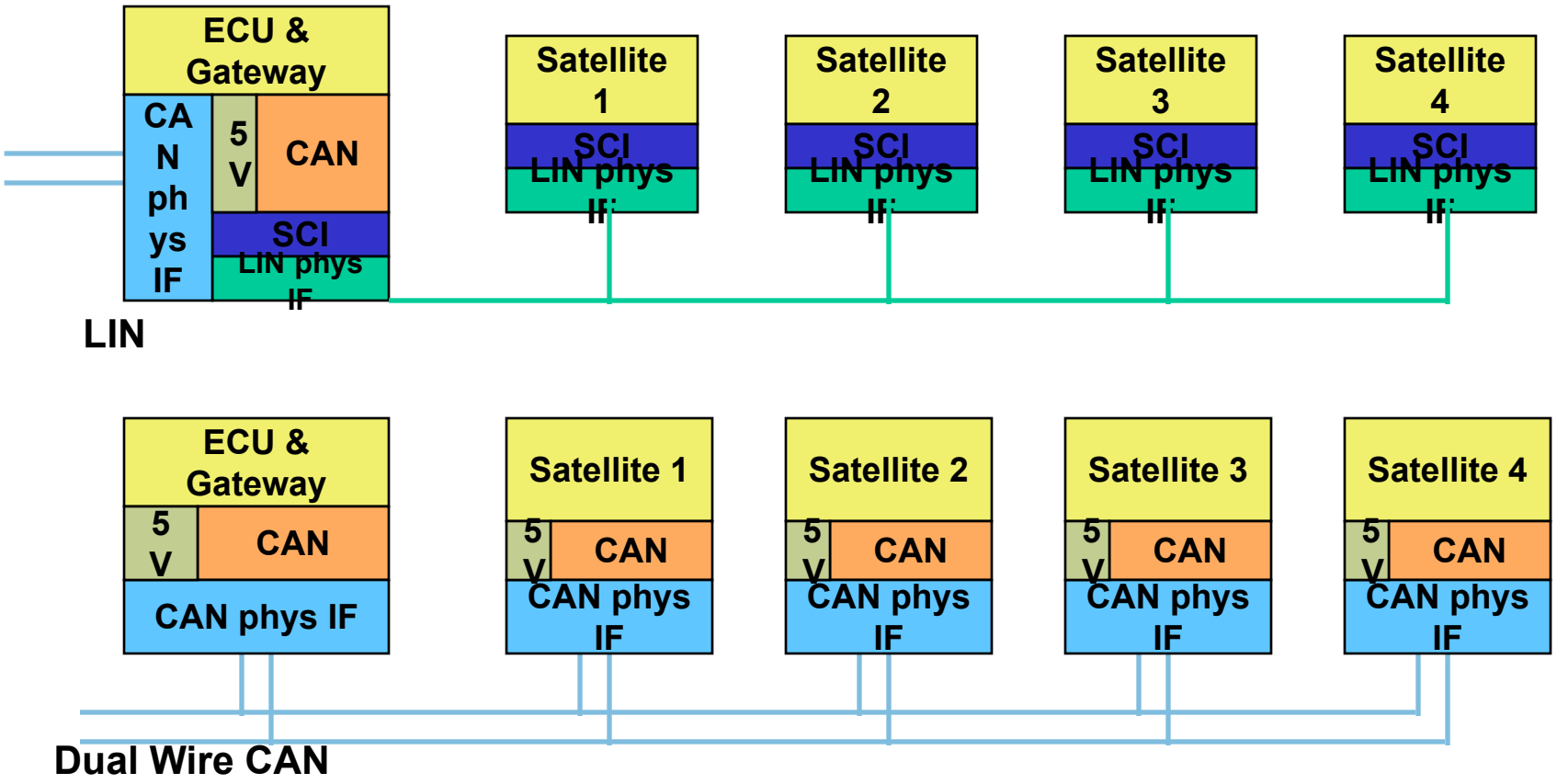
+Compatible with Main Bus

-Expensive (Die Size/ Dual Wire)

•Serial Sub Bus

- no standard Bus System
- not compatible with Main Bus
+ inexpensive
+ SCI-Based: Interface exists even on cheap devices
+ Interface can easily be reconstructed by ASIC or CPLD
+ Protocol can be done in Software

Sub-Network: LIN vs. CAN



Cost Factors: CAN Module
Crystal
2nd Wire / Connector

Dual Wire Interface
5V supply for bus

*
TSPG – 8/16 bit Products Division

MOTOROLA intelligence everywhere™

digital dna™

SubNets

- Necessary to reduce Busload on main Bus
- Solutions
  - **CAN**
    - \+ **Automotive Standard Bus**
    - \+ **Compatible with Main Bus**
    - \- **Expensive (Die Size/ Dual Wire)**
  - **Serial Sub Bus**
    - \- **no standard Bus System**
    - \- **not compatible with Main Bus**
    - \+ **inexpensive**
    - \+ **SCI-Based: Interface exists even on cheap devices**
    - \+ **Interface can easily be reconstructed by ASIC or CPLD**

Sub Bus Concept

- Basic Requirements:
  - **Satisfy Need for a Standard for Sub Busses**
  - **Cost driven: The solution must be cheaper than CAN**
  - **Reliability: Same Level as CAN expected**
  - **Long Term Solution**
  - **Logical Extension to CAN**
  - **Scalable: Capability to extend Systems with additional nodes**
  - **Lowering Cost of Satellite nodes:**
    - **No Crystal or Resonator**
    - **Easy implementation**
    - **Simple State Machines**
  - **Low Reaction Time (100 ms max)**
  - **Predictable Worst Case Timing**

MOTOROLA intelligence everywhere™

digital dna™

LIN Concept

- Technical Solution
  - Low cost single-wire implementation (enhanced ISO 9141)
  - Speed up to 20Kbit/s (limited for EMI-reasons)
  - Single Master / Multiple Slave Concept
    - → No arbitration necessary
  - Low cost silicon implementation based on common UART/SCI interface hardware
    - → Almost any Microcontroller has necessary hardware on chip
  - Self synchronization without crystal or ceramics resonator in the slave nodes
    - → Significant cost reduction of hardware platform
  - Guaranteed latency times for signal transmission (Predictability)

Master / Slave Protocol

- Master Task
  - **Determines order and priority of messages.**
  - **Monitors Data and check byte and controls the error handler.**
  - **Serves as a reference with its clock base (stable clock necessary)**
  - **Receives Wake- Up Break from slave nodes**

- Slave Task
  - **Is one of 2-16 members on the bus**
  - **Receives or transmits data when an appropriate ID is sent by the master.**
  - **The node serving as a master can be slave, too!**

Master / Slave Protocol

- Master
  - **has control over the whole Bus and Protocol**
    The master controls which message at what time is to be transferred over the bus. It also does the error handling.
    To accomplish this the master
    - **sends Sync Break**
    - **sends Sync Byte**
    - **sends ID-Field**
    - **monitors Data Bytes and Check Byte, and evaluates them on consistance**
    - **receives WakeUp Break from slave nodes when the bus is inactive and they request some action.**
    - **serves as a reference with it's clock base (stable clock necessary)**

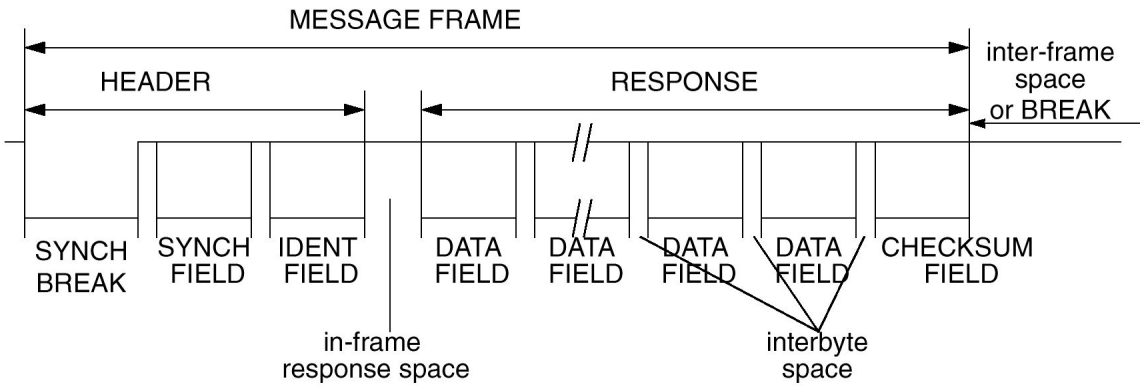
Master/Slave Protocol

- Slave
  - **Is one of 2-16 Members on the Bus and receives or transmits Data when an appropriate ID is sent by the master.**
    - **Slave snoops for ID.**
    - **According to ID, slave determines what to do.**
      - **either receive data**
      - **or transmit data**
      - **or do nothing.**
    - **When transmitting the slave**
      - **sends 1, 2, 4, or 8 Data Bytes**
      - **sends Check-Byte**
    - **The node serving as a master can be slave, too!**

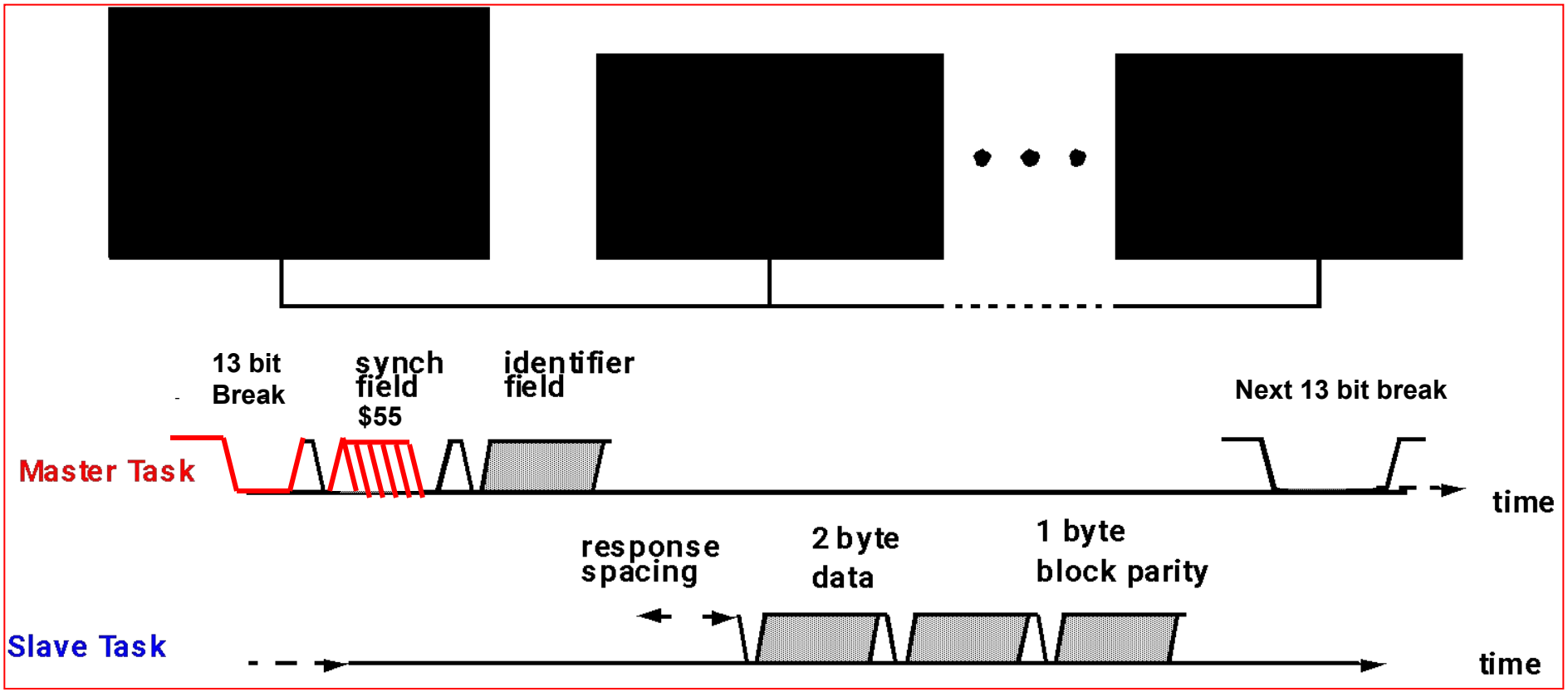
… protocol offers message timing predictability

Time Triggered Approach

- Message Length is known
  - **Number of transmitted data bytes is known**
    → minimum length can be calculated
  - **Each Message has length budget of 140% of it's minimum length**
    → maximum allowed length is known
    → distance between beginning of two messages



Data Transmission

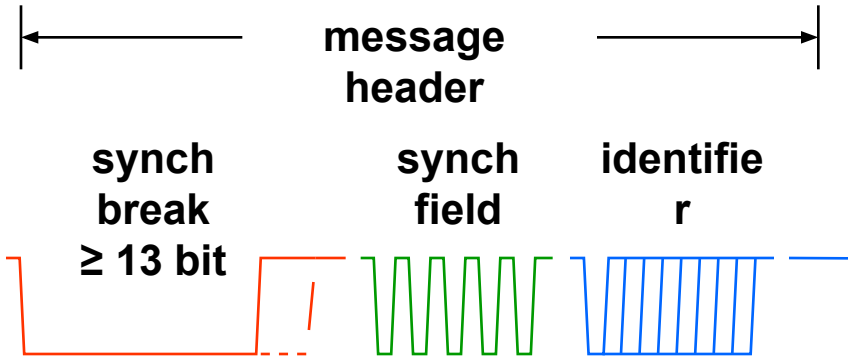


Message Frame

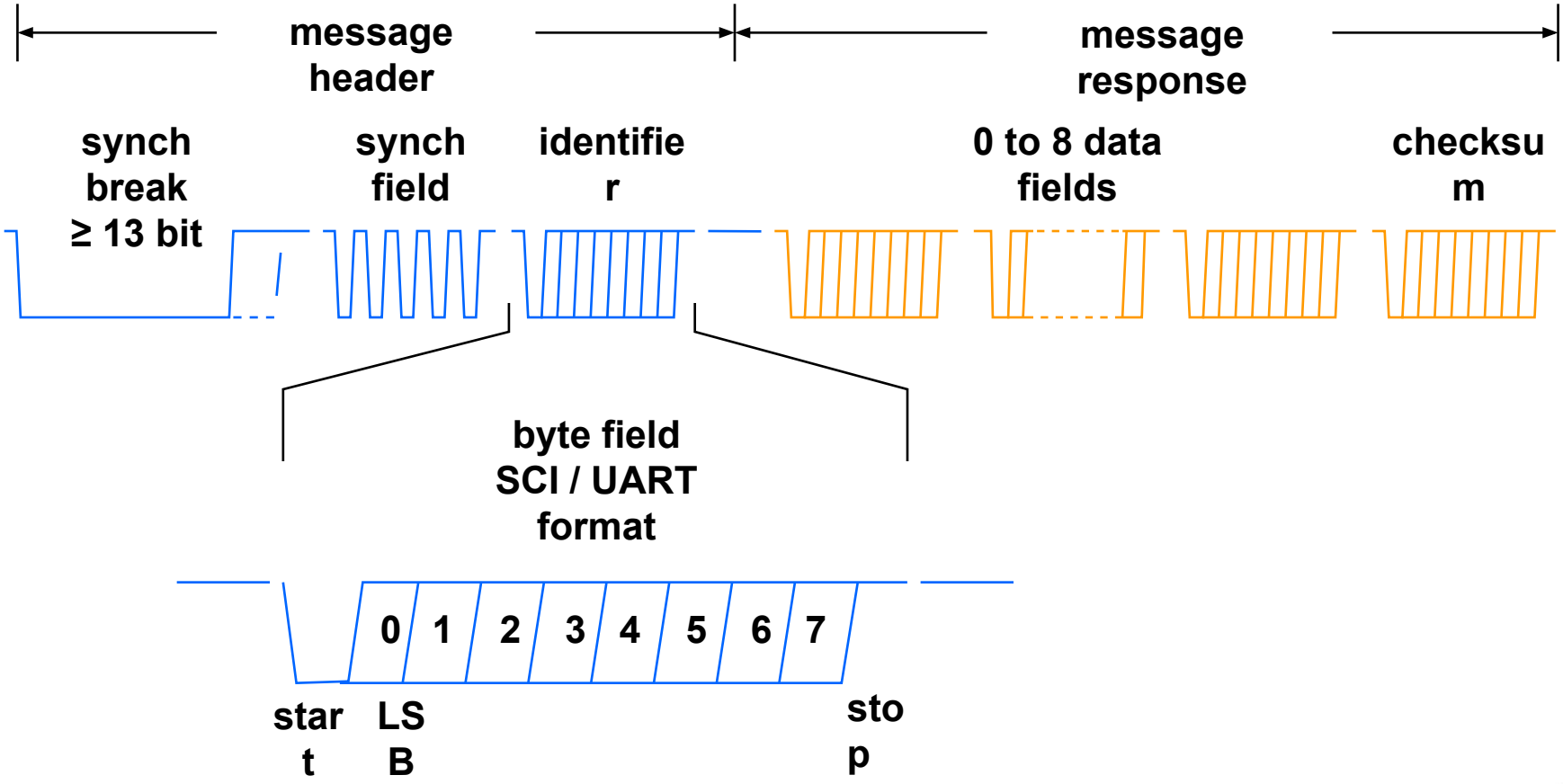
- Synch Byte:
  - **Specific Pattern for Determination of Time Base (Determination of the time between two rising edges)**
  - **A Synch Byte precedes any Message Frame**
- ID-Field:
  - **Message Identifier: Incorporates Information about the sender, the receiver(s), the purpose, and the Data field length. Length 6 Bit. 4 classes of 1/2/4/8 Data Bytes. The length coding is in the 2 LSB of the ID-Field. Each class has 16 Identifiers. A total of 64 Message Identifiers are possible.**
  - **2 Parity Bits protect this highly sensitive ID-Field.**

Identifier

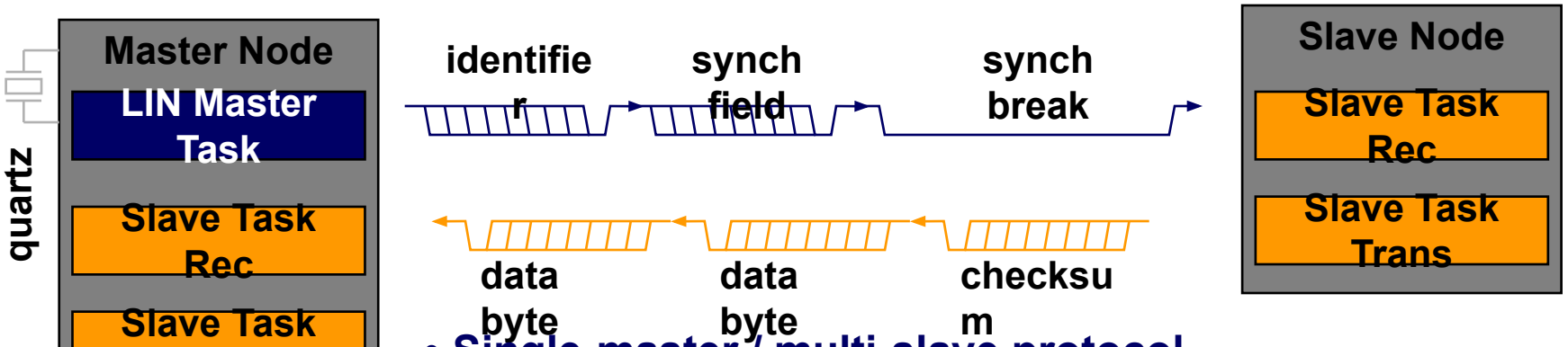
- The identifier field is sent by the master node to all LIN nodes
- This identifier normally contains one of 64 different values and includes 2 parity bits in the 8 bit data
- The identifier is normally associated with a collection of signals that are subsequently transmitted on the LIN bus
- In a specific case this can initiate SLEEP mode in the LIN slave nodes – in this case no further data is transmitted on the LIN bus



LIN Message Frame

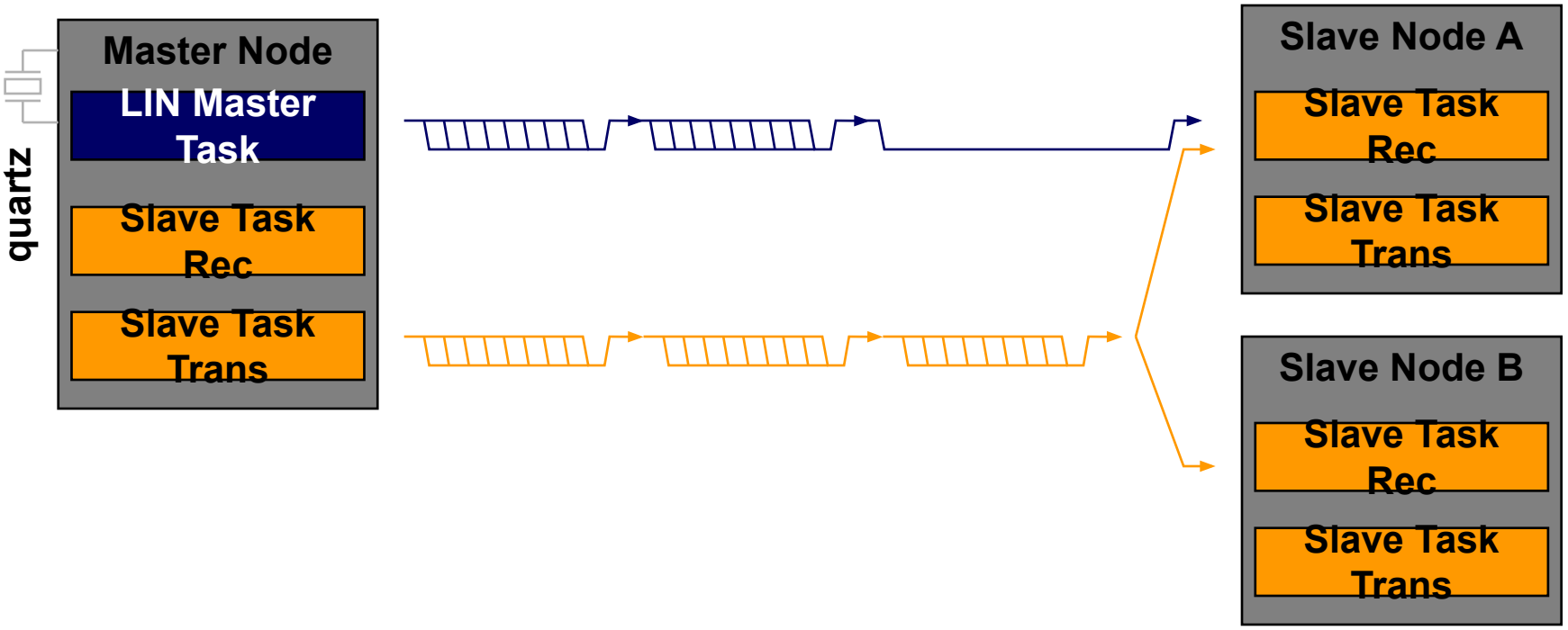


LIN Communication - Data from Slave to Master

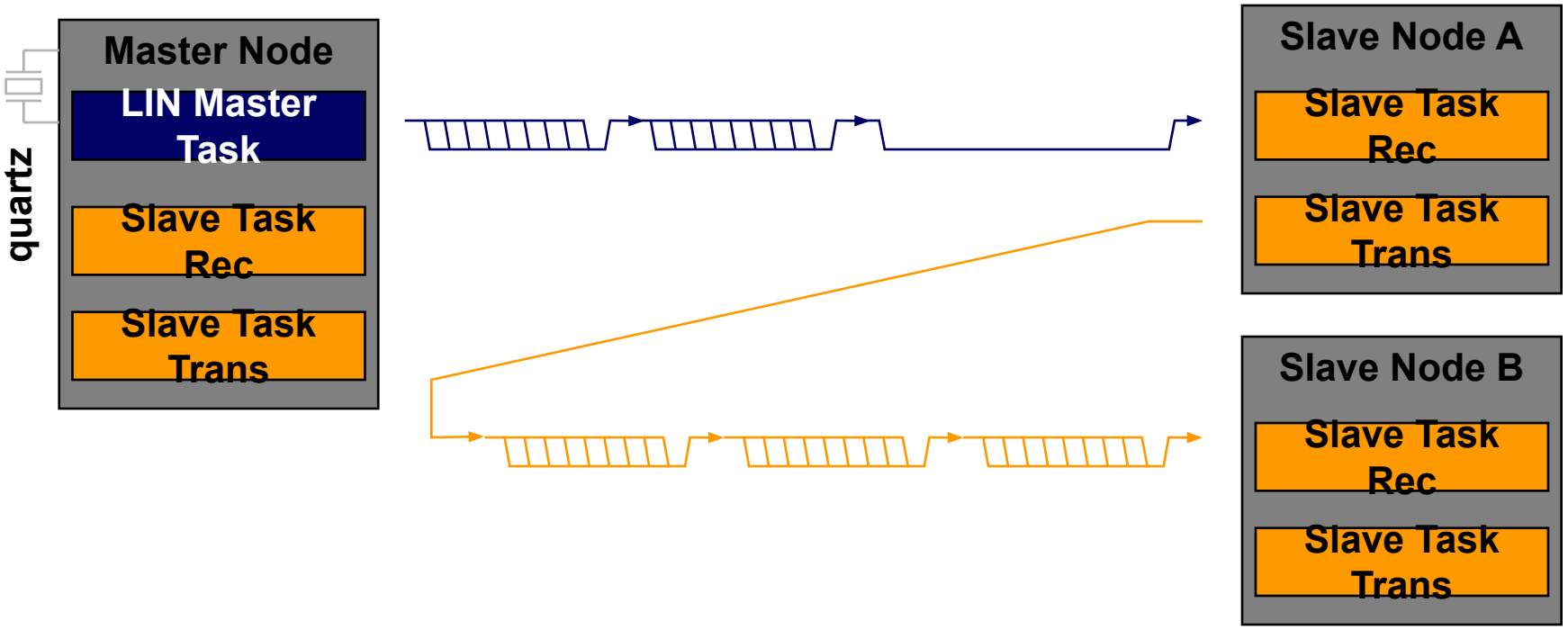


- Single-master / multi-slave protocol
- Time triggered, no arbitration
- Identifier denotes message content, not physical address
- Multicast messages
- Baud rate synchronization through protocol
- Power saving sleep mode

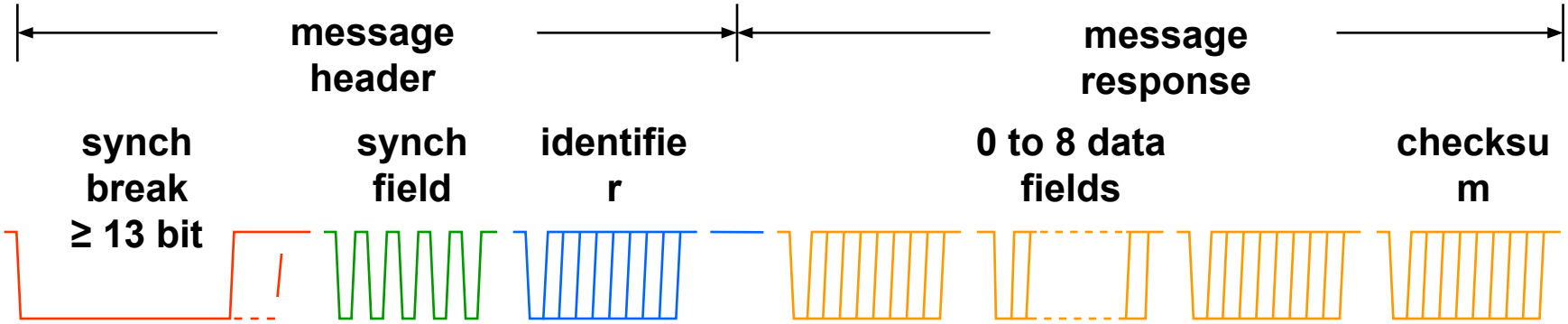
LIN Communication - Data from Master to Slave(s)



LIN Communication - Data from Slave to Slave



LIN Message Frame



Synchronisation on frame

Identifier byte

Synchronisation field

Message

*
TSPG – 8/16 bit Products Division

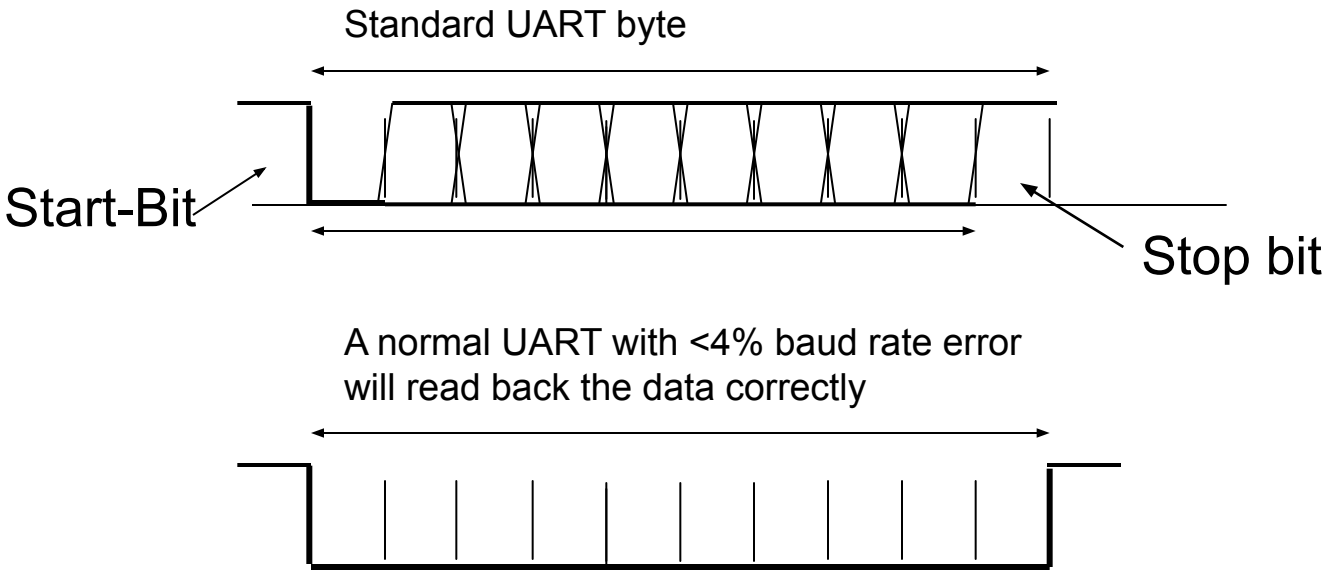
MOTOROLA intelligence everywhere™

digital dna™

Frame Synchronisation (1)

Initial conditions: +/- 4% baud rate accuracy relative the transmitting source

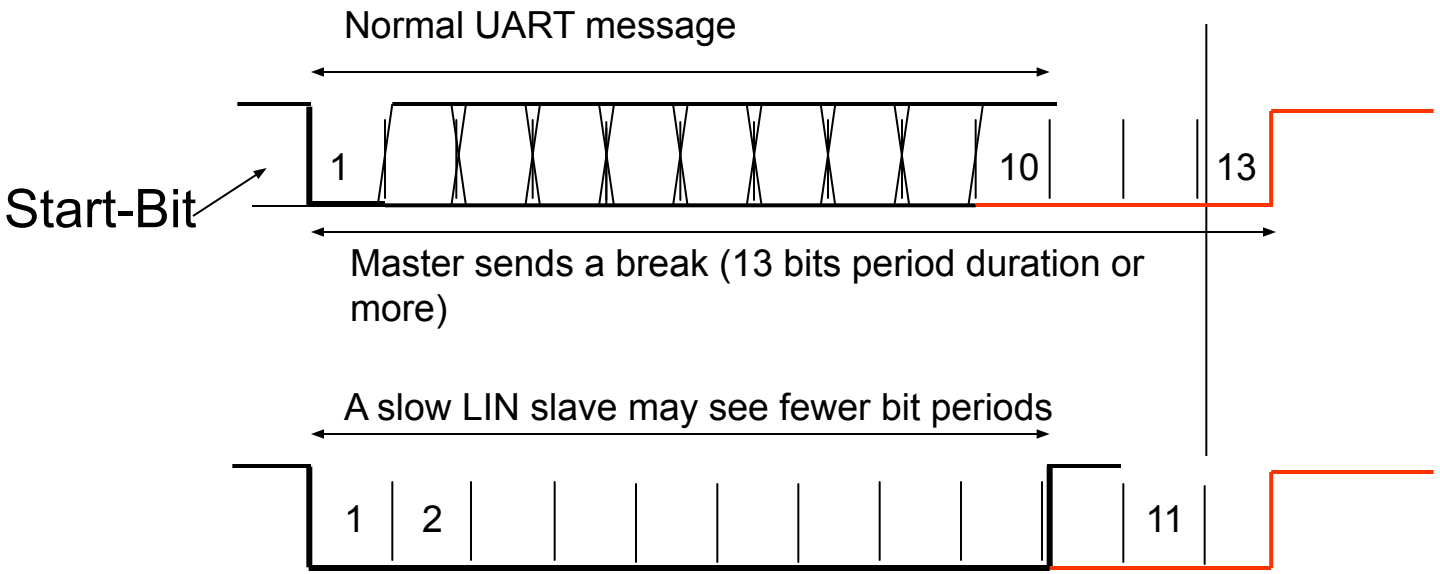
A standard transmission of data will require matched send and receiver baud rates



Frame Synchronisation (2)

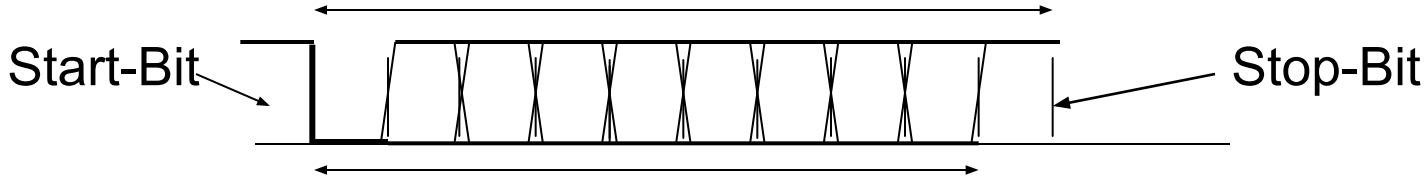
Initial conditions: +/- 15% baud rate accuracy relative the the LIN master transmitting the synchronisation frame

A synch break must be at least 13 bit periods in duration to allow for this initial variation in oscillator accuracy within the LIN slave

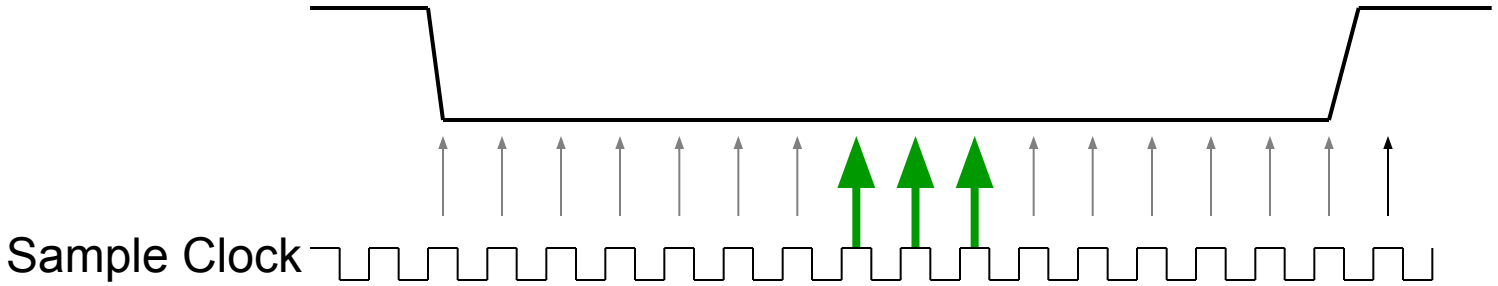


Bit-Synchronisation

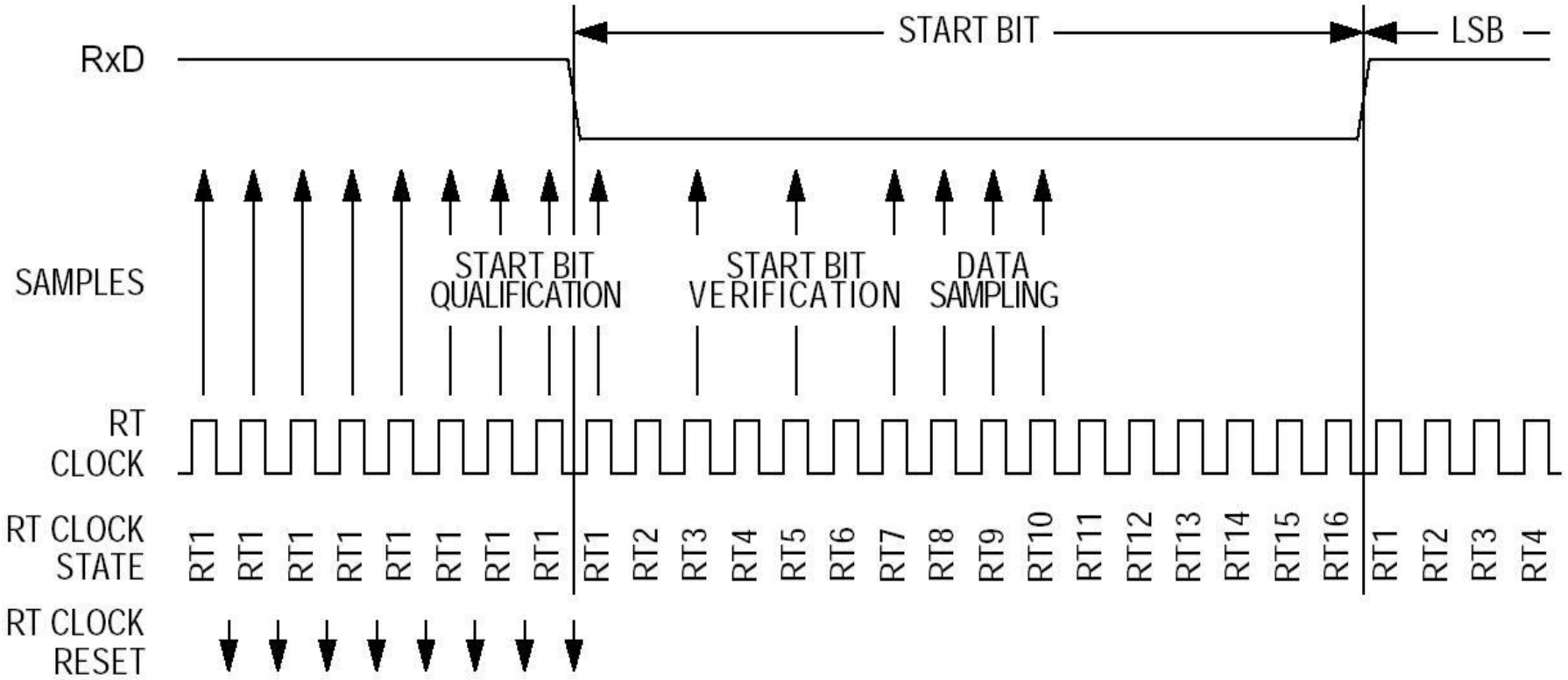
- A start bit transition to a low logic level (dominant) indicates a start of a byte, least significiant first and completing with a logic high level (resessive) bit to indicate the STOP bit



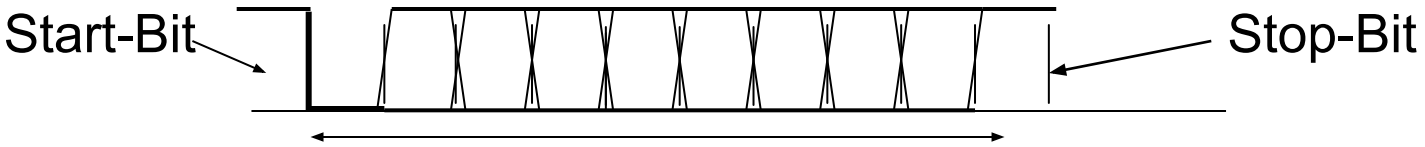
Data is sampled in the middle of the bit field:



Bit Sampling

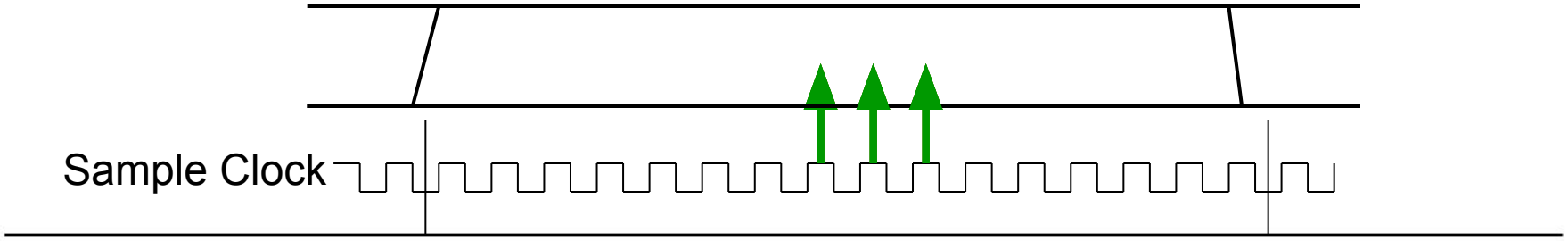


Bit-Synchronisation



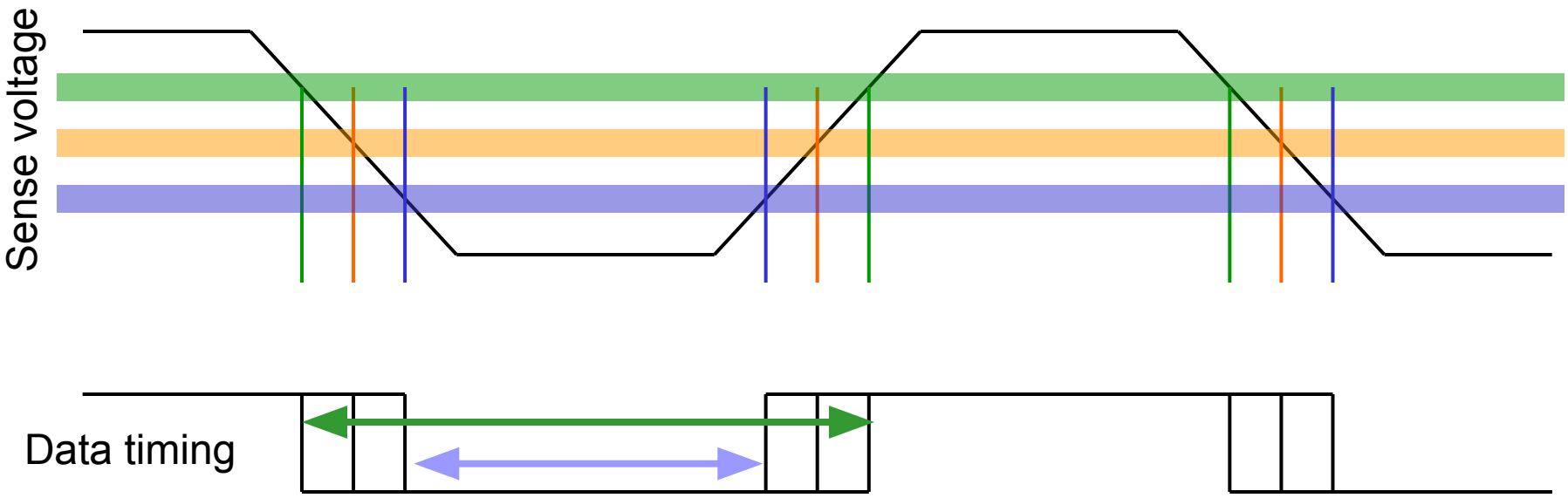
After recognition of a Low level in the start bit, the data is sampled at a rate 16 times the bit rate expected. The middle 3 samples must all agree for an error free reception of the data.

A stop bit is expected after 1 start bit and 8 data bits in a typical message



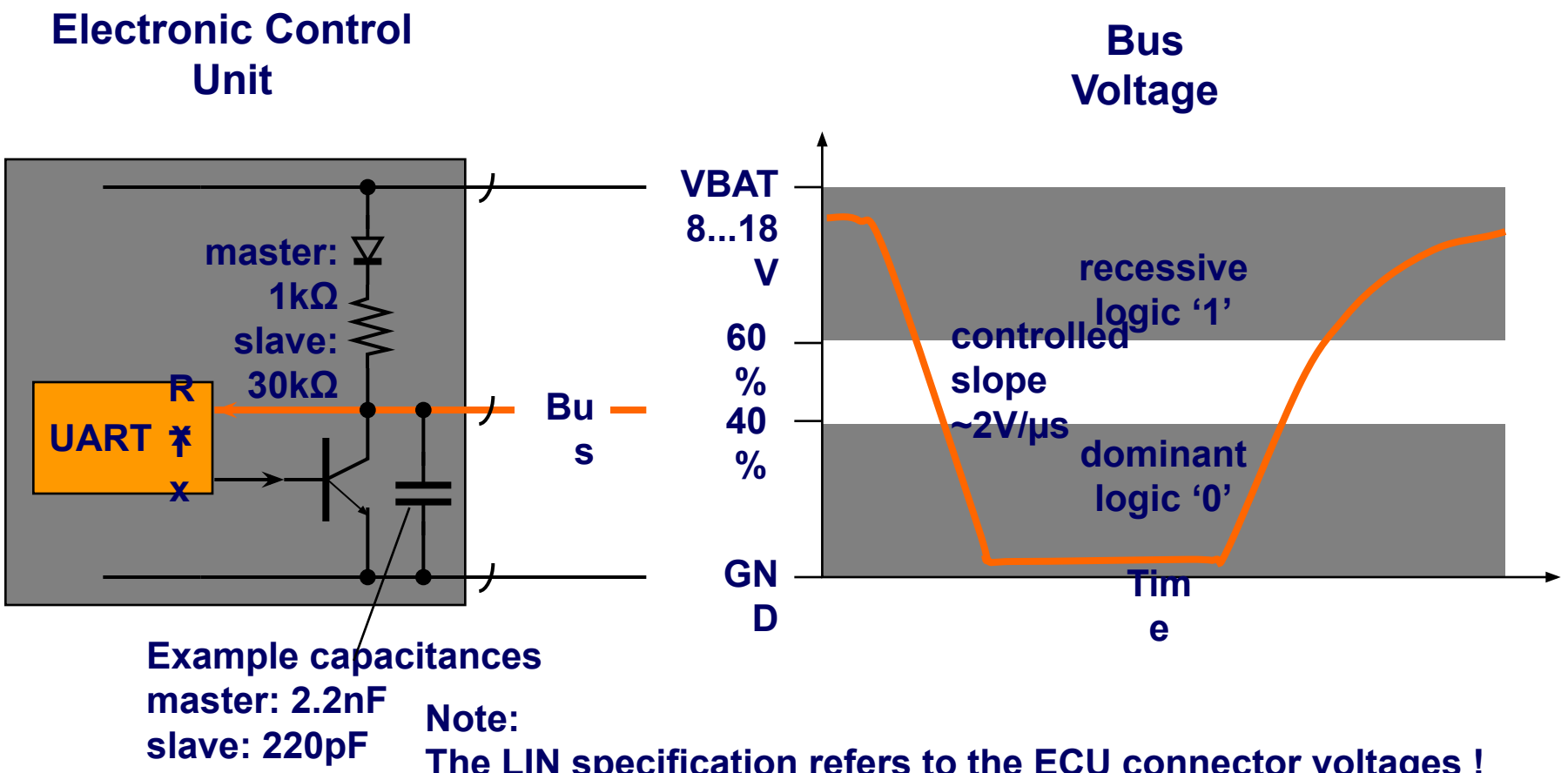
Taking account of Ground-Shift

The detection point for data transitions can be affected by voltage references. Ground shift can change this reference by a significant amount, affecting the bit timing of the data



Available bit sampling zone can reduce worst case bit width to around 40us at 20k baud
This affects the overall baud rate tolerance required for safe LIN communications

LIN Physical Interface

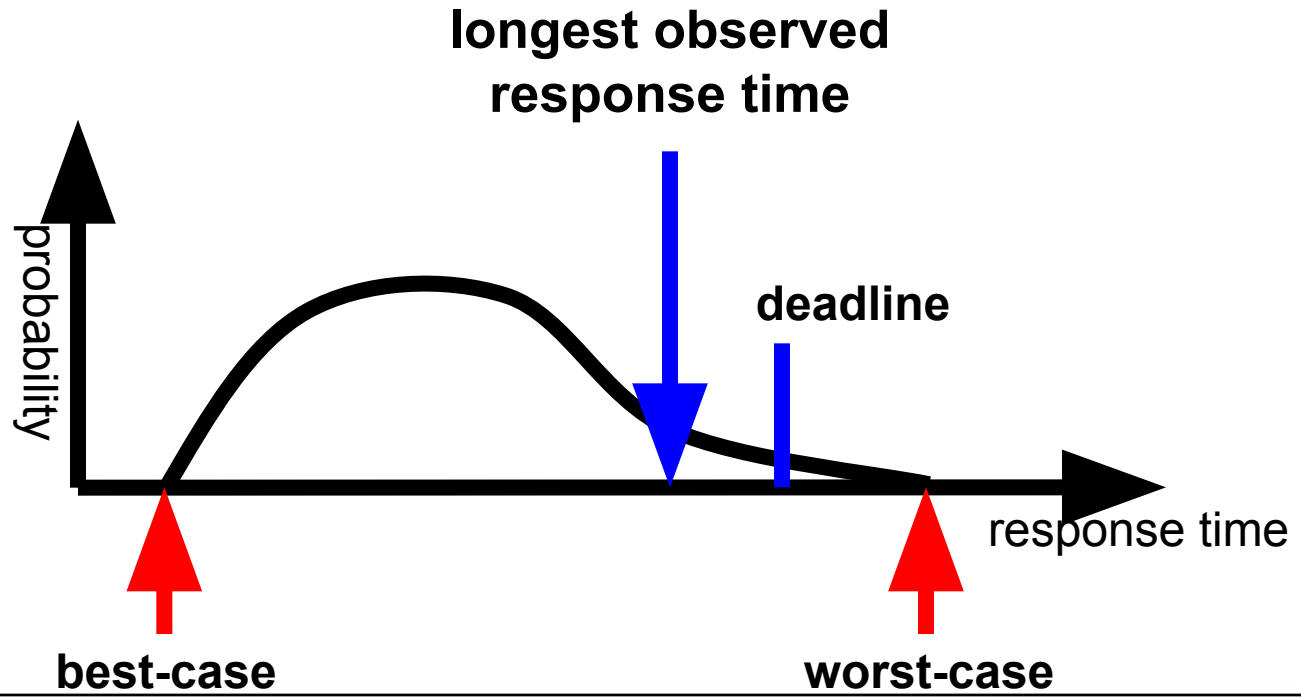


Note:
The LIN specification refers to the ECU connector voltages !

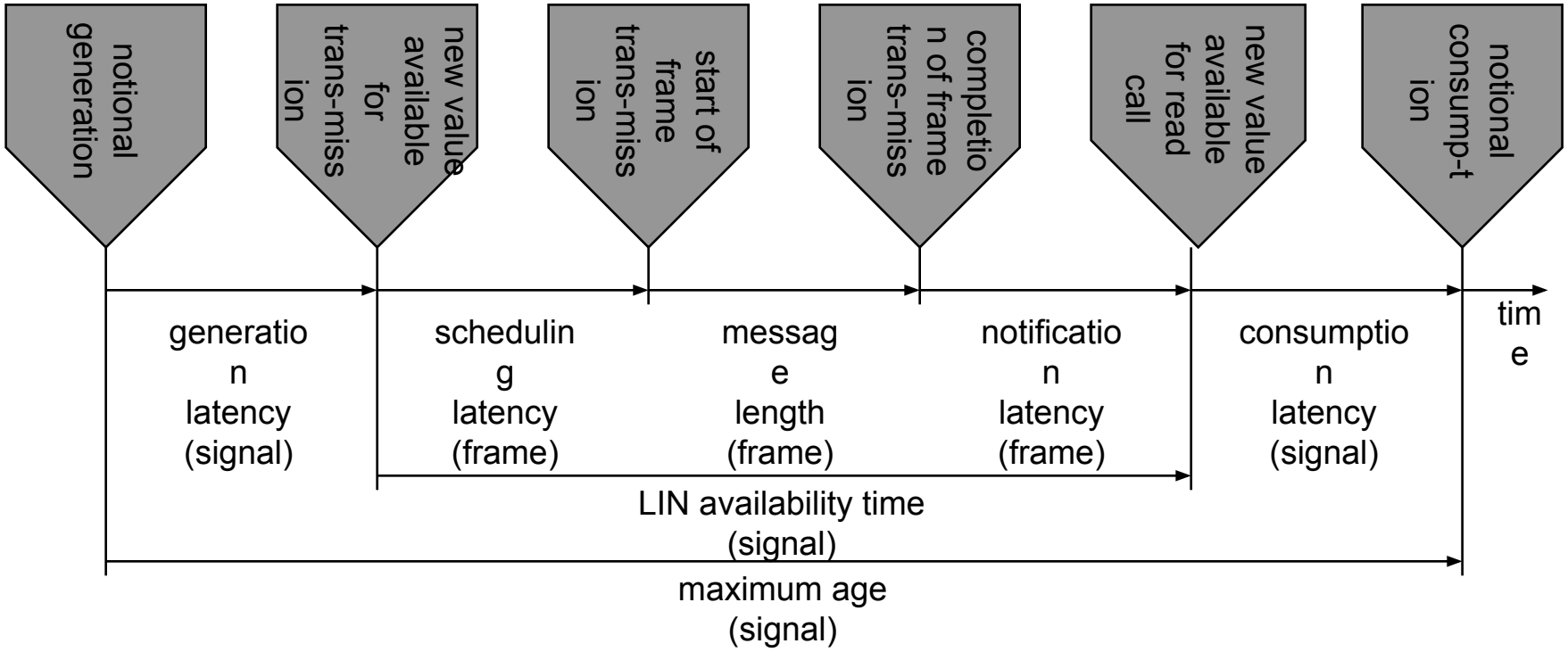
Examination of whether the Deadline is met

- Signal based messaging with static latency analysis ensures that all signals meet defined minimum latency times

- Drives need for complete configuration tool support to ensure guaranteed timing of all signals in a LIN network



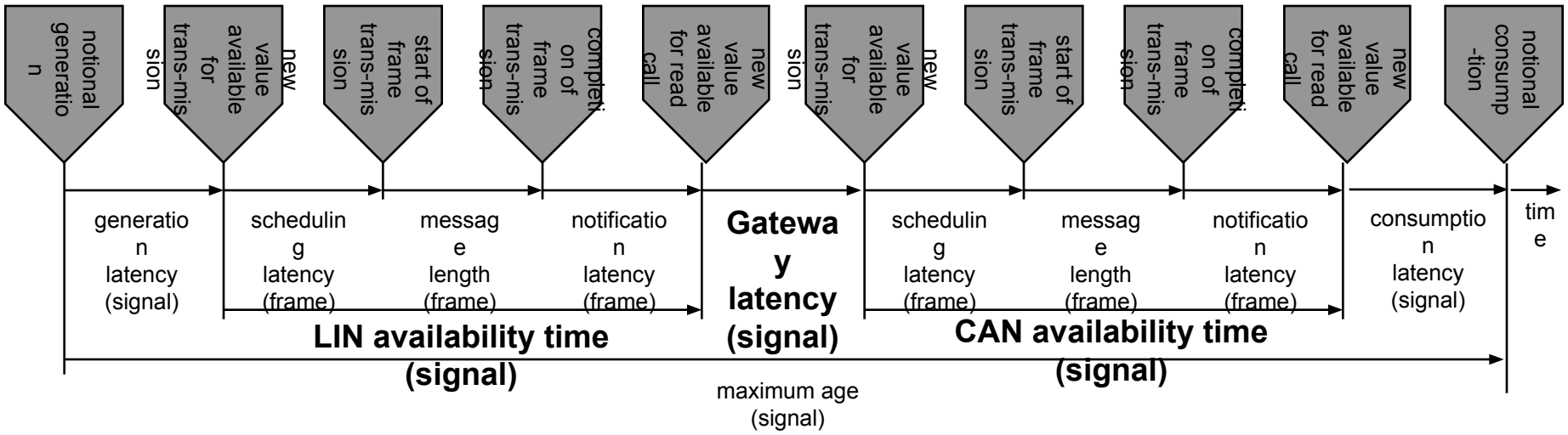
Message latency



MOTOROLA intelligence everywhere™

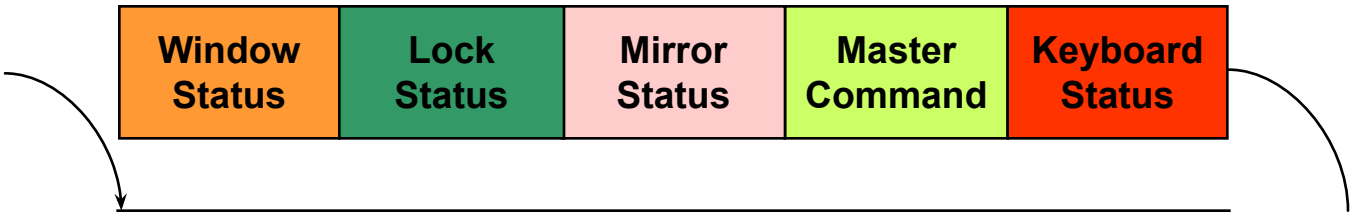
digital dna™

Message latency across a network

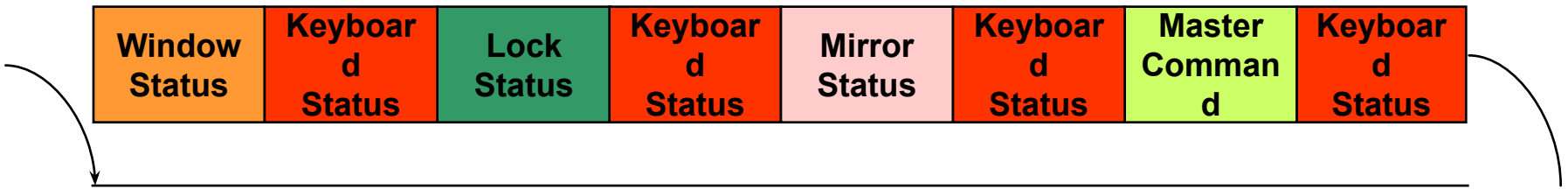


Latency optimisation with LIN

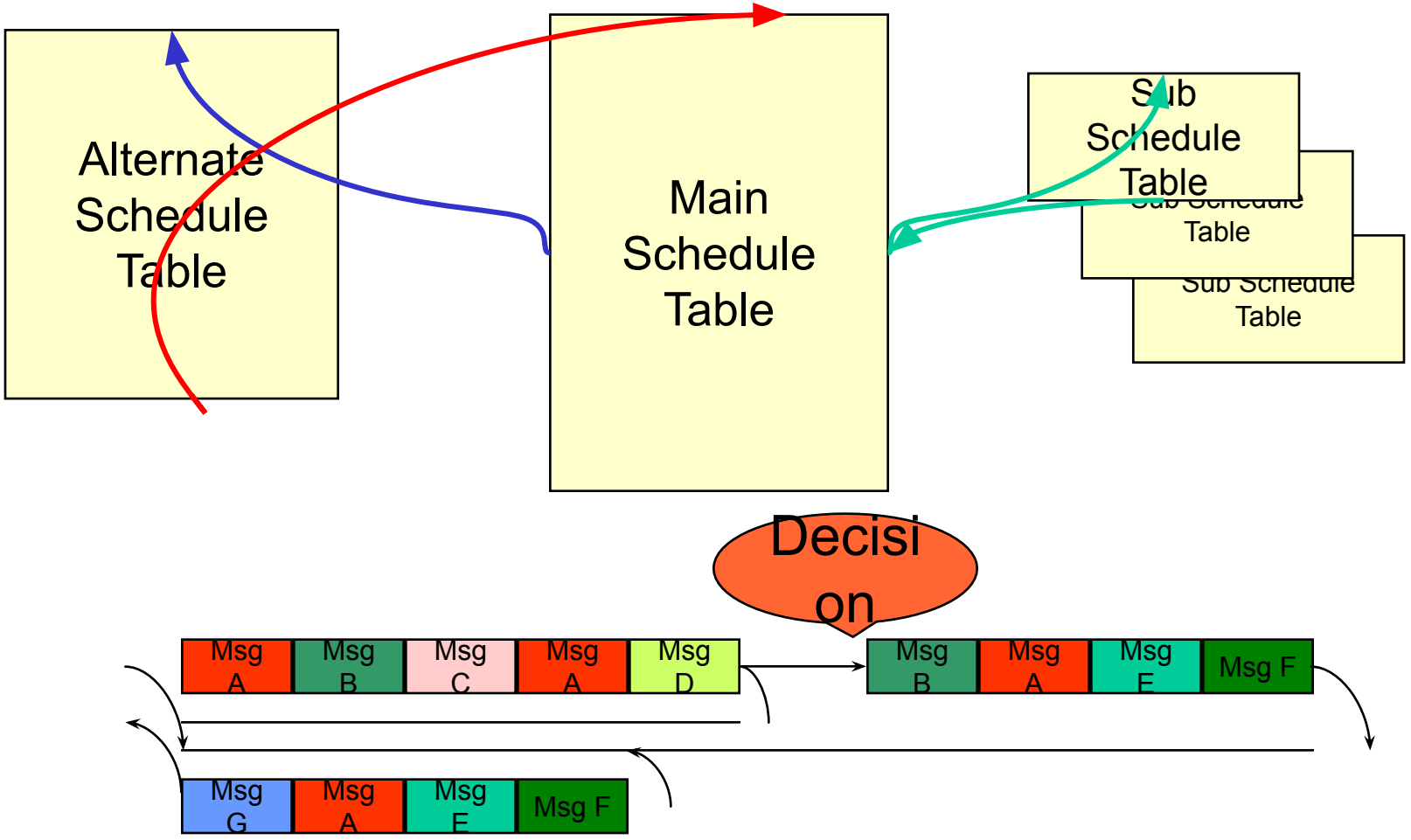
Basic schedule



Alternate schedule for low latency signals from a keyboard



Variables Scheduling



Event Triggered Message

- Problem
  - **Specific node communication required but this takes up too much time for all network messages**
- Solution : Event Triggered frame:
  - **Header is sent out**
    1. **normal case: no answer**
    2. **Rare response: only one node responds**
    3. **Very rare response : several nodes respond simultaneously**
- Cases 1 and 3 are exceptions that should be addressed at the application design.
- Event triggered messaging is complementary to the regular signal based messaging scheme

Further information

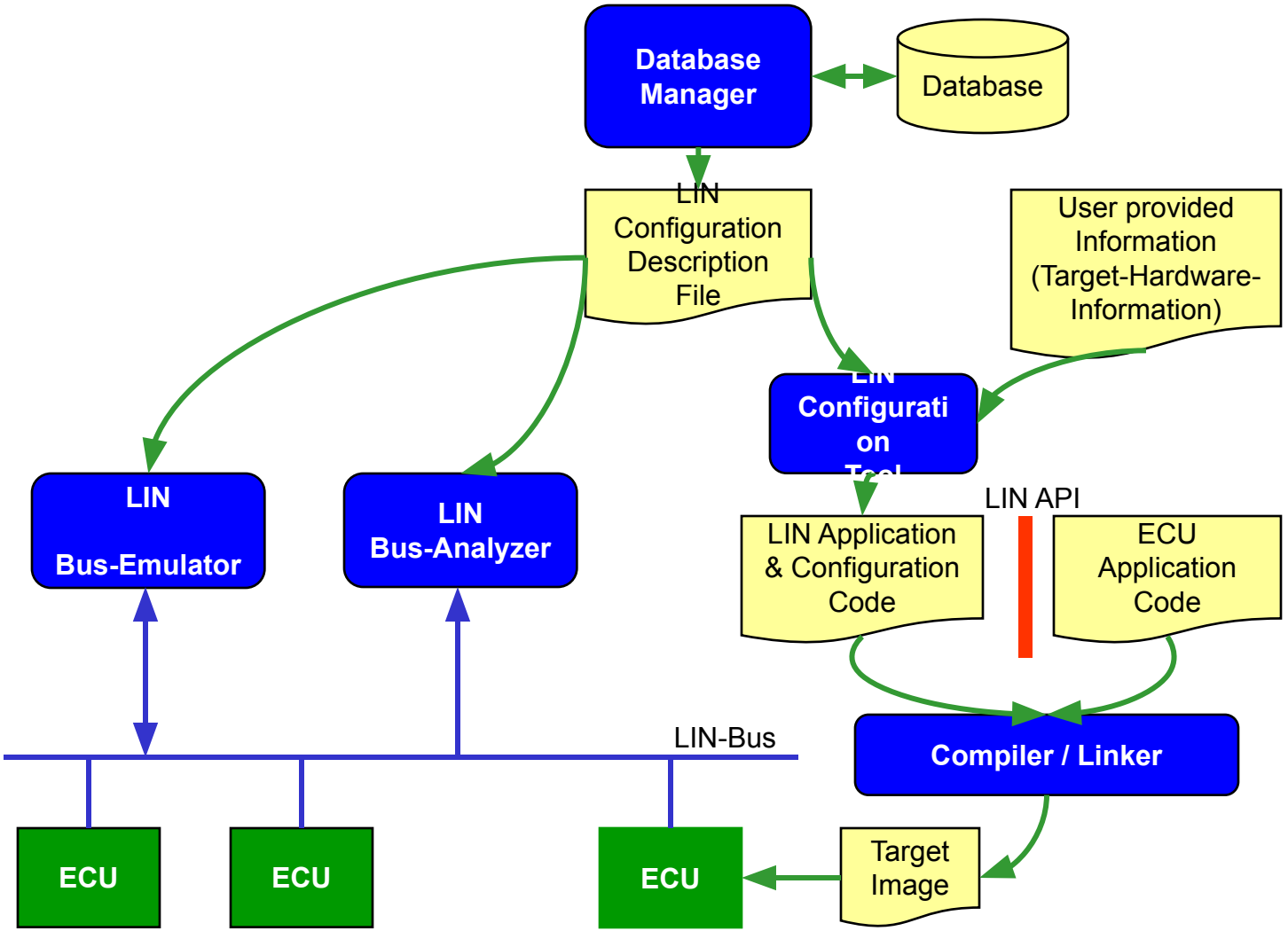
- Consortium

http://www.lin-subbus.org

MOTOROLA intelligence everywhere™

digital dna™

LIN Development Flow



*
TSPG – 8/16 bit Products Division

MOTOROLA intelligence everywhere™

digital dna™

LIN Configuration Description File

- Includes all essential information of network signals, latency periods, cycle times, nodes affected
- Input file serves as a development interface for a node
- LIN Application Generator
  - **LIN-Emulator**
  - **LIN Analyser**

MOTOROLA intelligence everywhere™

digital dna™

The Workflow

- **Data Input**
  - Definition of objects
  - Definition of relations between the objects
- **Data Processing**
  - Signal Packing (Frame Editor/Frame Compiler)
  - Timing Analysis
- **Data Output**
  - Configuration file generation
  - Various optional customer-defined post-operations