CAN in Automation (CiA)
International Users and Manufacturers Group e.V.

CONTROLLER

AREA

NETWORK
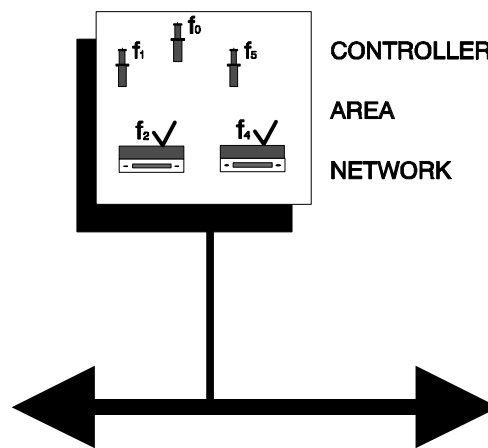
CAN Application Layer for Industrial Applications
CiA/DS205-2
February 1996

LMT Protocol Specification

1. SCOPE

This document contains the protocol specification of the Layer Management (LMT). LMT is part of the CAN Application Layer. This document is part of a set of documents that standardize the CAN Application Layer for Industrial Applications.

2. REFERENCES

/1/: CiA/DS201, CAN in the OSI Reference Model

/2/: CiA/DS205-1, LMT Service Specification

/3/: CiA/DS207, Application Layer Naming Conventions

/4/: CiA/DS102, Version 2.0, CAN Physical Layer for Industrial Applications

3. GENERAL DESCRIPTION

3.1 LMT Protocol Perspective

The Layer Management (LMT) service element in the CAN Reference Model (see /1/) provides the LMT services. The LMT Protocol is executed between the LMT Master and each of the LMT Slaves (see /2/) to implement these services.

3.2 LMT Slave Synchronisation

Since in the LMT Protocol all LMT Slaves use the same COB to send information to the LMT Master, there must be only one LMT Slave at a time that communicates with the LMT Master. For all protocols the LMT Master takes the initiative, a LMT Slave is only allowed to transmit within a confirmed service after it has been uniquely switched into configuration mode. Since there can be atmost one confirmed LMT service outstanding at a time (see /2/), the synchronisation is established.

3.3 LMT Protocol Descriptions

A protocol description specifies the sequence of COB's and their format that are exchanged between the LMT Master and LMT Slave(s) for a particular LMT service.

LMT uses command specifiers to identify the commands. Command specifiers from 0 - 07fh are reserved for use by standard LMT services. Command specifiers from 080h - 0ffh are free for application specific purposes and may only be used with at most one LMT Slave in configuration mode.

In the description of the COB data format, bytes are numbered from zero to and including seven. Bits within a byte are numbered from zero to and including seven. Byte zero is transmitted first, byte seven is transmitted last. Within a byte, bit zero is the least significant bit, bit seven is the most significant bit.

The terms 'lsb' and 'msb' stand for 'least significant byte' and 'most significant byte' respectively and are used to define how an integer number is represented in more than one byte for the LMT Protocol. The order of significance is increasing from lsb to msb.
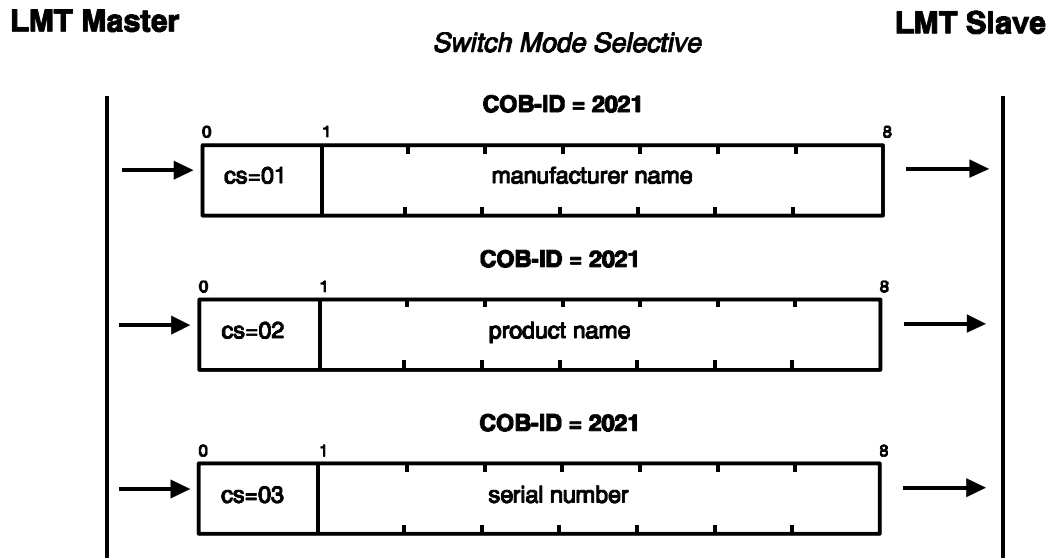
4. SWITCH MODE PROTOCOLS

4.1 Switch Mode Global

This protocol is used to implement the Switch Mode Global service.

LMT Master   *Switch Mode Global*   **LMT Slave**

COB-ID = 2021

0	1	2	8
cs=04	mode	reserved	
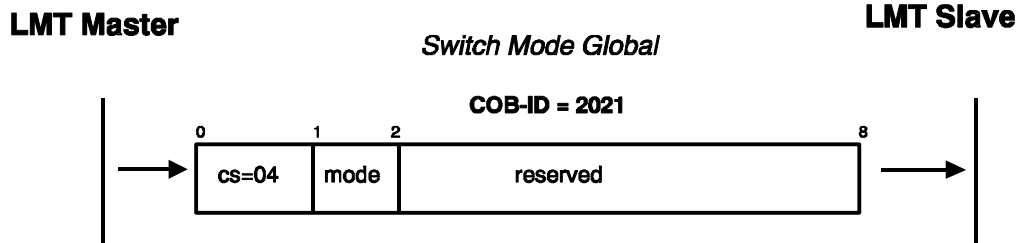
- **cs**: LMT command specifier
  04 for Switch Mode Global

- **mode**: The LMT mode to switch to:
  - 0: switches to operation mode
  - 1: switches to configuration mode

- **reserved**: reserved for further use by CiA.

4.2 Switch Mode Selective

This protocol is used to implement the Switch Mode Selective service.

LMT Master   *Switch Mode Selective*   **LMT Slave**

COB-ID = 2021

0	1	8
cs=01	manufacturer name	

COB-ID = 2021

0	1	8
cs=02	product name	

COB-ID = 2021

0	1	8
cs=03	serial number	

- **cs**: LMT command specifiers
  01 to 03 for Switch Mode Selective
- **manufacturer_name**: The manufacturer name part of the LMT address, see /3/
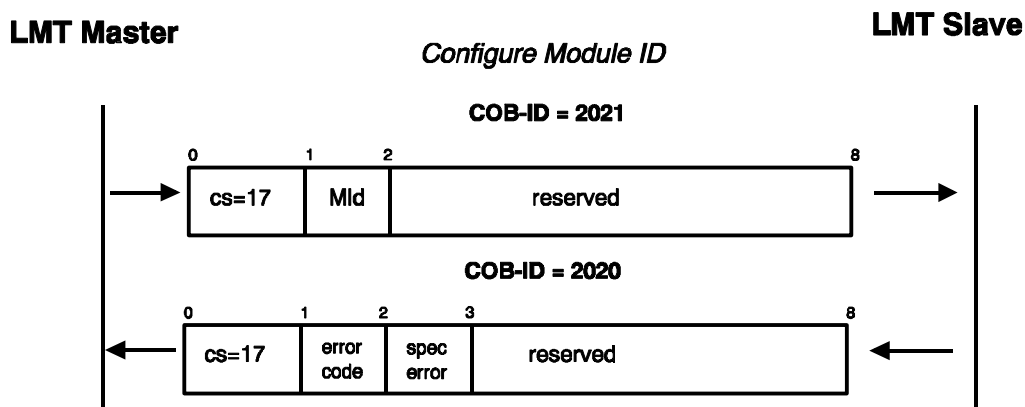- **product_name**: The product name part of the LMT address, see /3/

- **serial_number**: The serial number part of the LMT address, see /3/

5. CONFIGURATION PROTOCOLS

5.1 Configure NMT Address Protocols

Configure Module ID Protocol

This protocol is used to implement the Configure NMT Address service for the module-ID part of the NMT address.

- **cs**: LMT command specifier
    17 for Configure Module ID

- **MId**: The new module_id to configure, see /3/

- **error_code**:
    0:              protocol successfully completed
    1 ... 254:   reserved for further use by CiA
    255:          implementation specific error occured.

- **specific_error_code**: If error_code equals 255, specific_error_code gives an implementation specific error code, otherwise it is reserved for further use by CiA.
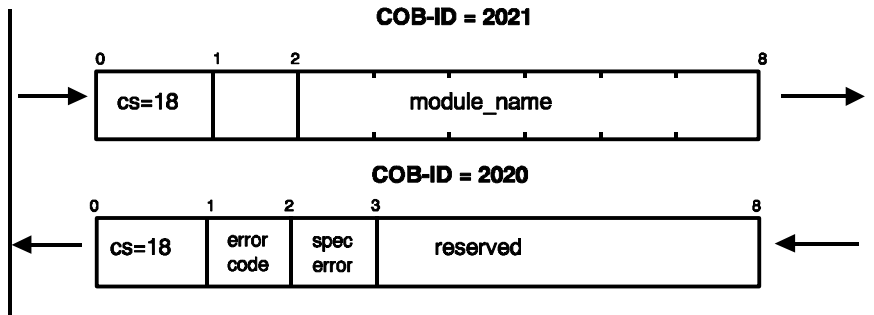
- **reserved**: reserved for further use by CiA

Configure Module Name Protocol

This protocol is used to implement the Configure NMT Address service for the module-name part of the NMT address.

LMT Protocol Specification

LMT Master **LMT Slave**

Configure Module Name

COB-ID = 2021

0	1	2		8
cs=18		module_name		

COB-ID = 2020

0	1	2	3		8
cs=18	error code	spec error	reserved		

- **cs**: LMT command specifier
  18 for Configure Module Name

- **module_name**: the new module_name to configure, see /3/

- **table_index**: index for the bit timing to use.

- **error_code**:
  0:              protocol successfully completed
  1 . . 254:   reserved for further use by CiA
  255:          implementation specific error occured

- **specific_error_code**: If error_code equals 255, specific_error_code gives an implementation specific error code, otherwise it is reserved for further use by CiA.

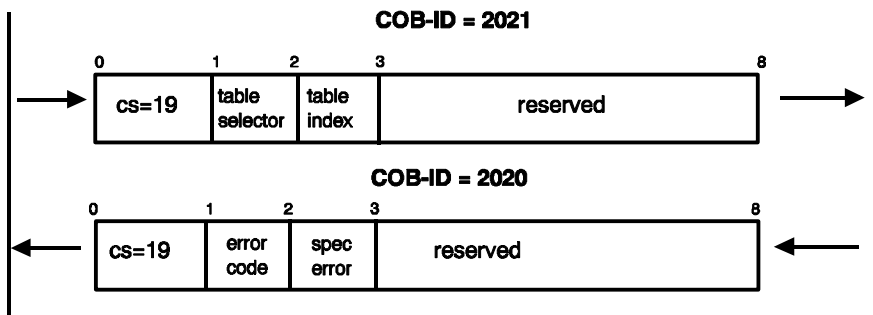- **reserved**: reserved for further use by CiA.

5.3   Configure Bit Timing Parameters Protocol

This protocol is used to implement the 'Configure Bit Timing Parameters' service

LMT Master **LMT Slave**

Configure Bit Timing Parameters

COB-ID = 2021

0	1	2	3		8
cs=19	table selector	table index	reserved		

COB-ID = 2020

| 0	1	2	3		8
cs=19	error code	spec error	reserved		

- **cs**: LMT command specifier

19 for Configure Bit Timing Parameters

- **table_selector**: selects which bit timing parameters table has to be used
  - 0: standard CiA bit timing table (see /4/)
  - 1..127: reserved for further use by CiA
  - 128..255: may be used for manufacturer specific bit timings

- **table_index**: selects the entry (bit timeing parameters) in the selected table; see /4/ for valid indices when using the standard CiA bit timings (table_selector = '0')

- **error_code**:
  - 0: protocol successfully completed
  - 1: bit timing not supported
  - 2..254: resrerved for further use by CiA
  - 255: implementation specific error occured

- **specific_error_code**: if error_code equals 255, specific_error_code gives an implementation specific error code, otherwise it is reserved for further use by CiA.

- **reserved**: reserved for further use by CiA.

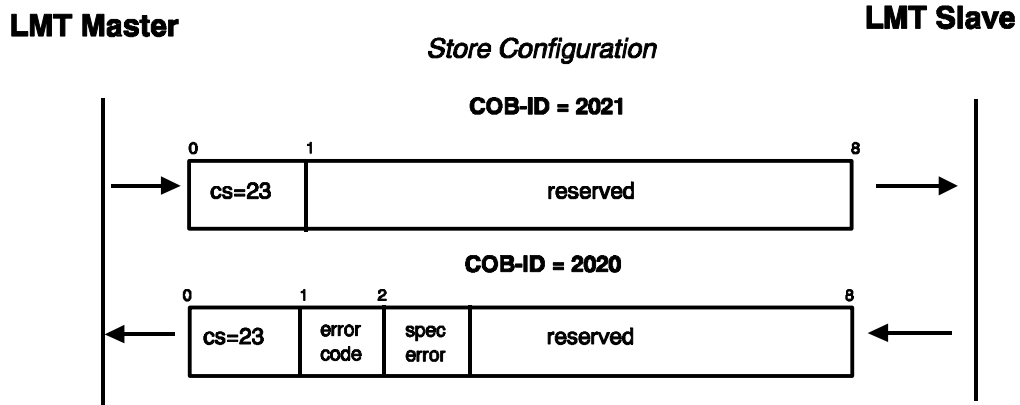## 5.4   Activate Bit Timing Parameters Protocol

This protocol is used to implement the Activate Bit Timing Parameters service.
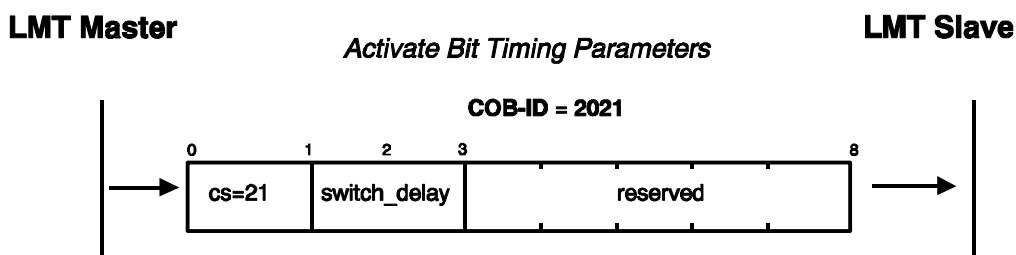
- **cs**: LMT command specifier
  21 for Activate Bit Timing Parameters

- **switch_delay**: The duration of the two periods of time to wait until the bit timing parameters switch is done (first period) and before transmitting any CAN message with the new bit timing parameters after performing the switch (second period) The time unit of switch delay is 1 ms.

- **reserved**: reserved for further use by CiA.

February 1996
LMT Protocol Specification

5.5 Store Configuration Protocol

This protocol is used to implement the Store Configured Parameters service.

LMT Master   **LMT Slave**

Store Configuration

COB-ID = 2021

0	1	8
cs=23	reserved	

COB-ID = 2020

| 0	1	2	8
cs=23	error code	spec error	reserved

- **cs**: LMT command specifier
  23 for Store Configuration

- **error_code**:
    0:              protocol successfully completed,
    1:              store configuration is not supported,
    2 . . 254:   reserved for further use by CiA,
    255:            implementation specific error occured.

- **specific_error_code**: If error_code equals 255, specific_error_code gives an implementation specific error code, otherwise it is reserved   for   further use by CiA.

- **reserved**: reserved for further use by CiA.

- DS205-2 p. 8 -

6.   INQUIRY PROTOCOLS

6.1   Inquire LMT Address Protocols

These protocols are used to implement the Inquire LMT Address service. To implement the service, each of the following three protocols has to be executed.

Inquire Manufacturer Name Protocol
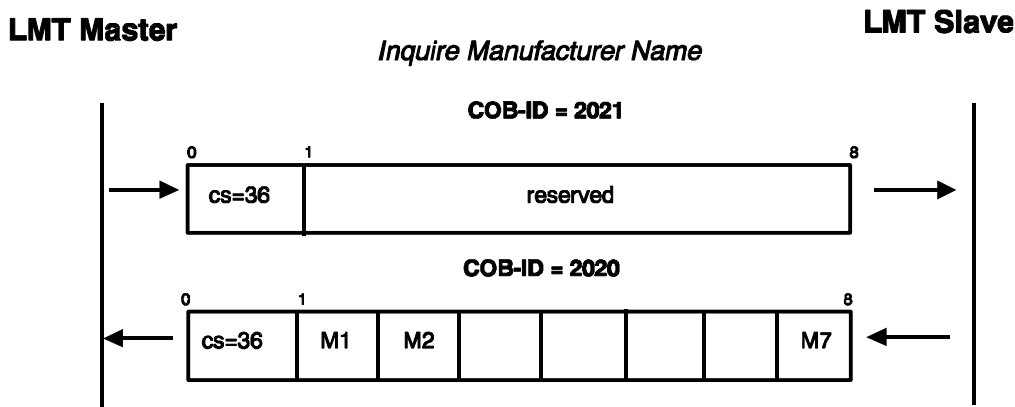
- **cs**: LMT command specifier
    36 for Inquire Manufacturer Name

- **M1 - M7**: The manufacturer_name (see /3/) of the selected module or error code. If M1 is a valid <alpha-num>, the response contains the name. If M1 is 0ffh, M2 contains the error code, M3 contains the reason if valid for the error code.

- **reserved**: reserved for further use by CiA.

Inquire Product Name Protocol
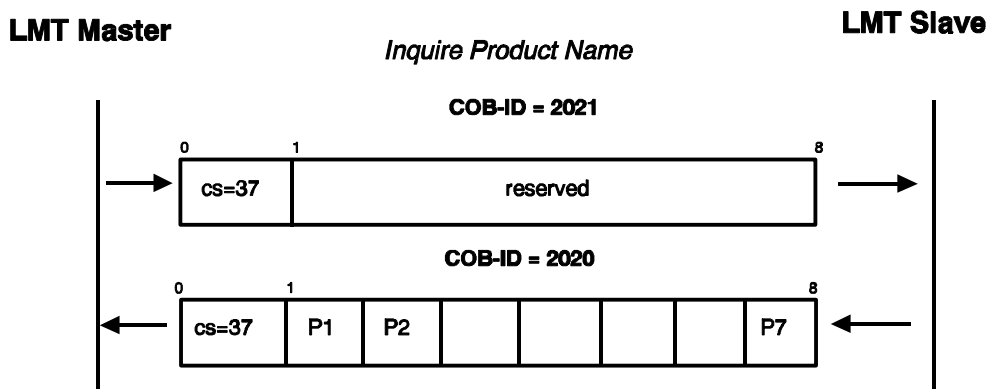
- **cs**: LMT command specifier
    37 for Inquire Product Name

- **P1 - P7**: The product_name (see /3/) of the selected module or error code. If P1 is a valid <alpha-num>, the response contains the name. If P1 is 0ffh, P2 contains the error code, P3 contains the reason if valid for the error code.

- **reserved**: reserved for further use by CiA.

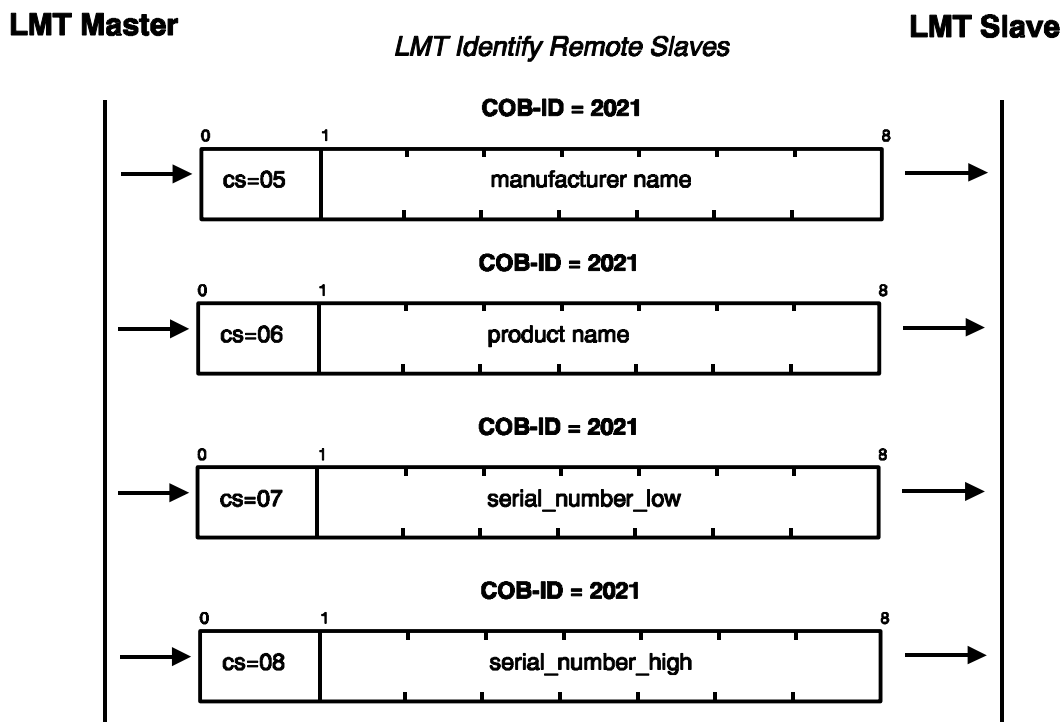**Inquire Serial-Number Protocol**
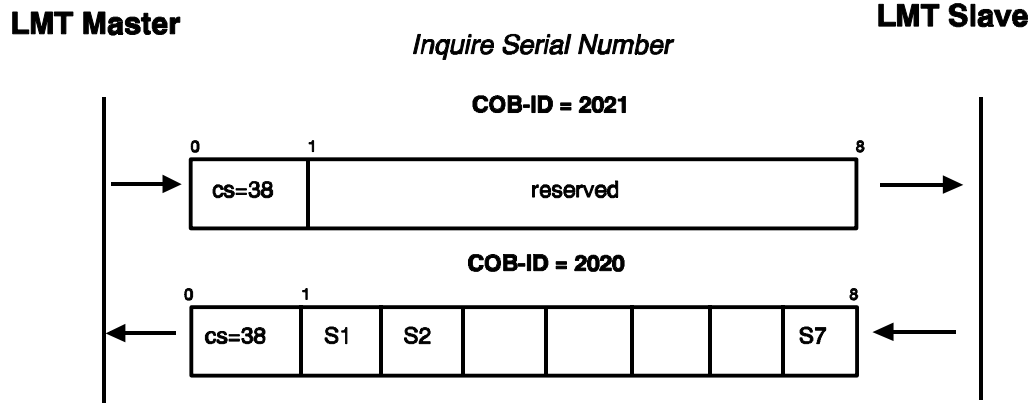
- **cs**: LMT command specifier
    38 for Inquire Serial Number

- **S1 - S7**: The serial_number (see /3/) of the selected module or error code. If S1 contains a valid BCD-pair (see /3/), the response contains the serial number. If S1 is 0ffh, S2 contains the error code, S3 contains the reason if valid for the error code.

- **reserved**: reserved for further use by CiA.

7.   IDENTIFICATION PROTOCOLS

7.1   LMT Identify Remote Slaves

This protocol is used to implement the 'LMT Identify Remote Slaves' service.

LMT Master   *LMT Identify Remote Slaves*   **LMT Slave**

COB-ID = 2021

cs=05	manufacturer name

COB-ID = 2021

cs=06	product name

COB-ID = 2021

cs=07	serial_number_low

COB-ID = 2021

| cs=08	serial_number_high

- **cs**: LMT command specifier
    05 to 08 for LMT Identify Remote Slaves

- **manufacturer_name**: The manufacturer name part of the LMT Address

- **product_name:** The product name part of the LMT Address

- **serial_number_low:** The lower boundary of the requested serial numbers range

- **serial_number_high:** The higher boundary of the requested serial numbers range

The boundaries are included in the interval. All LMT Slaves with matching manufacturer name and product name whose serial numbers lie within this range, are requested to identify themselves with the LMT Identify Slave service.

7.2   LMT Identify Slave Protocol

This protocol is used to implement the 'LMT Identify Slave' service.

LMT Slave   **LMT Master**

LMT Identify Slave

COB-ID = 2020

0	1						8
cs=09	reserved						
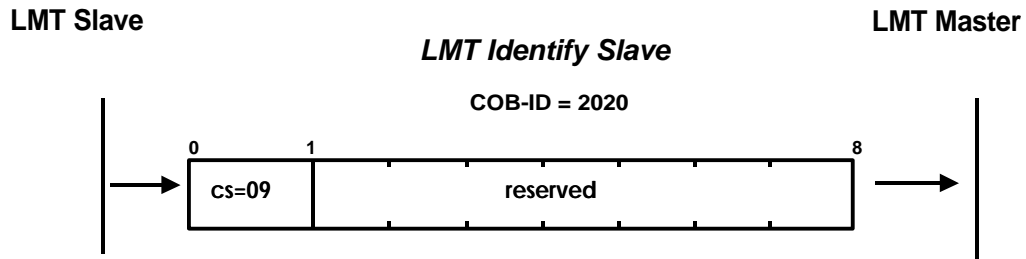
- **cs:** LMT command specifiers
  09 for Identify Slave

- **reserved**: all bytes set to '0'

ANNEX I

IMPLEMENTATION RULES

When implementing the LMT protocols, the following rules have to be followed to guarantee inter-operability. The rules deal with the following implementation aspects:

Invalid COBs

A COB is invalid if it has a COB-ID that is used by the LMT Protocol, but contains invalid parameter values according to the LMT Protocol. This can be caused by errors in the lower layer (see /1/) or implementaton errors. Invalid COBs must be handled locally in an implementation specific way that does not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications. As far as the LMT Protocol is concerned, an invalid COB must be ignored.

Time-Outs

Since COBs may be ignored, the response of a confirmed LMT service may never arrive. To resolve this situation, an implementation may, after a certain amount of time, indicate this to the service user (time-out). A time-out is not a confirm of the LMT service. A time-out indicates that the service has not completed yet. The application must deal with this situation. Time-out values are considered to be implementation specific and do not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications. However, it is recommended that an implementation provides facilities to adjust these time-out values to the requirements of the application.