Layer Setting Services and Protocol
(LSS)

CiA Draft Standard Proposal 305

Version 1.0

Date: 31.05.00

DIRECTORY

1 Scope

This document contains the protocol specification of the Layer Setting Services (LSS) for CANopen.

2 Normative references

/1/ ISO 11898: Road vehicles - Interchange of digital information – Controller area network (CAN) for high-speed communication, 1993-11-15

/2/ CiA DS 301: CANopen - Application Layer and Communication Profile, Version 4.0, 1999-06-16

3 Abbreviations and definitions

3.1 Abbreviations

COB Communication Object. A unit of transportation in an CAN network. Data must be send across a CAN network inside a COB. A COB can contain at most 8 bytes of data.

COB-ID Each COB is uniquely indentified in a CAN network by a number called the COB Identifier (COB-ID). The COB-ID determines the priority of the COB for the MAX sub-layer.

LMT Layer Management. Functions to inquire and change the settings of certain parameters of the local layers on a CAL module.

LSS Layer Setting Services. Functions to inquire and change the settings of certain parameters of the local layers on a CANopen module.

MAC Medioum Access Control. Äone of the sub-layers of the Data Link Layer in the CAN Reference Model that controls who gets access to the medium to send a message.

NMT Network Management. One of the service elements of the application layer in the CAN Reference Model. The NMT serves to configure, initialise, and handle errors in a CAN network.

3.2 Definition

LSS offers the possibility to inquire and change the settings of certain parameters of the local layers on a CANopen module with LSS Slave capabilities by a CANopen module with LSS Master capabilities via the CAN Network.

The following parameters can be inquired and/or changed by the use of LSS:

- Node-ID of the CANopen Slave
- Bit timing parameters of the physical layer (baud rate)
- LSS address (/2/ Identity Object, Index 1018H)

By using LSS a LSS Slave can be configured for a CANopen network without using any devices like DIP-switches for setting the parameters. There are several solutions available for LSS Slaves with and without a unique LSS-address or non-volatile storage.

3.2.1 LSS Objects and Attributes

LSS functionality is modelled using two objects. The LSS Master object exists exactly once in a CANopen network supporting LSS. The LSS Master configures layer parameters of connected CAN modules by the use of LSS Slave objects residing on the individual modules.
Communication between LSS Master and LSS Slaves is accomplished by the LSS protocol.

3.2.1 LSS Master Object

The module that configures other modules via a CANopen network is called the LSS Master. There may be only one LSS Master in a network. The LSS Master has no attributes.

3.2.2 LSS Slave Object

The module that is configured by the LSS Master via a CANopen network is called the LSS Slave. The number of LSS Slaves in a network is not limited. The LSS Slave has the following attributes:

- **LSS Address**
  An LSS Slave is identified by an LSSAddress. An LSS Address consists of a vendor-id, a product-code, a revision-number and a serial-number. The vendor-id and product-code are numerical numbers. The revision-number contains the major an minor revision as numerical number. The serial-number is coded as a numerical number too. They adhere to the following syntax:

<LSS-ADDRESS>	::=	<vendor-id><product-code><revision-number><serial-number>
<vendor-id>	::=	'UNSIGNED32'
<product-code>	::=	'UNSIGNED32'
<revision-number>	::=	'UNSIGNED32'
<serial-number>	::=	'UNSIGNED32'

  A <vendor-id> is assigned to module suppliers by CiA. A <product-code>, <revision-number> and a <serial-number> are assigned by the module supplier. For LSS-Addresses the following conditions must be met:

  - The LSS address is identical to the CANopen identity object.
  - The LSS address of a LSS Slave can be inquired.
  - There exists no other LSS Slave in the world with the same <LSS-Address>

- **LSS Mode**
  For activate LSS functionality, the device state of all devices must be set to the stop state. The LSS-Master have to locate on the same device which resides the NMT-Master. The LSS mode distinguishes between the LSS configuration phase and the operation phase of the module. In configuration mode all LSS services, in operation mode only the switch mode services are available. Any module not explicitly put into configuration mode is in operation mode.

3.3 LSS Modes and Services

LSS services can be functionally grouped in three areas:

- The switch mode services provide a way to logically connect the LSS Master and LSS Slave(s) for configuration purposes. They change the LSS mode attribute of the LSS Slave (see figure 1).
- The configuration services perform the actual task of configuring the layer parameters of an LSS Slave. The configuration services are only available in configuration mode.
- The inquiry services provide a way for the LSS Master to determine layer parameters. The inquiry services are available only in configuration mode.

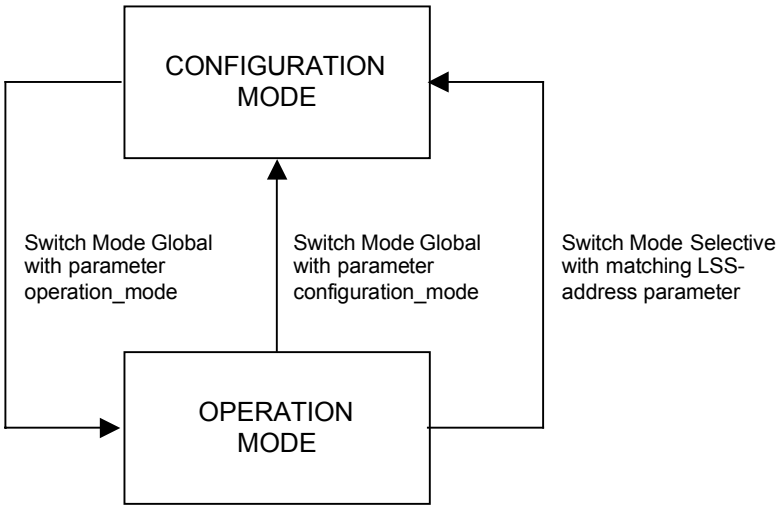


Fig. 1 : LSS modes and switching procedure

The LSS services are described in a tabular form that contains the parameters of each service primitive.

3.4 SWITCH MODE SERVICES

The Switch Mode Services control the mode attribute of a LSS Slave. LSS provides two ways to put a LSS Slave into configuration mode, Switch Mode Global and Switch Mode Selective. Switch Mode Selective switches exactly one LSS Slave into configuration mode. Switch Mode Global switches all LSS Slaves into configuration mode.
Some LSS configuration and inquiry services require that only one LSS Slave is in configuration mode.

Besides the LSS Switch Mode Services there may be other (local and module specific) means to change the mode of an LSS Slave, that are not within the scope of this document.

3.4.1 Switch Mode Global

This service is used to switch all LSS Slaves in the network between operation mode and configuration mode.

Parameter	*Request/Indication*
Argument mode configuration_mode operation_mode	**Mandatory** Mandatory Selection Selection

3.4.2 Switch Mode Selective

This service is used to switch the LSS Slave, whose LSS address attribute equals LSS_address, into configuration mode.

Parameter	*Request/Indication*
Argument LSS_address	**Mandatory** mandatory

3.5 CONFIGURATION SERVICES

The configuration services are available only in configuration mode. Some of the services require that exactly one LSS Slave is in configuration mode.

3.5.1 Configure Node-ID

Through this service the LSS Master configures the NMT-address parameter of a LSS Slave.

Parameter	*Request/Indication*	*Response/Confirmation*
Argument Node-ID	**Mandatory** mandatory	
Remote Result success failure reason		**Mandatory** selection selection optional

This service allows only one LSS Slave in configuration mode. The remote result parameter confirms the success or failure of the service. In case of a failure optionally the reason is confirmed.

3.5.2 Configure Bit Timing Parameters

Through the Configure Bit Timing Parameters service the LSS Master sets the new bit timing on a LSS Slave.

Parameter	*Request/Indication*	*Response/Confirmation*
Argument table_selector table_index Remote Result success failure reason	**Mandatory** mandatory mandatory	 **Mandatory** selection selection optional

By means of the table_selector the bit timing parameter table to be used is specified. In the bit timing parameter table the bit timing parameters for different baud rates are specified. With table_selector value ´0´ the standard CiA bit timing parameter table is referenced. The table_index selects the entry (baud rate) in the selected table (value ‘0’ referes to the highes baud rate).

Standard CiA bit timing table:

Baud Rate	*Tabel_Index*
1000 kBit	0
800 kBit	1
500 kBit	2
250 kBit	3
125 kBit	4
100 kBit	5
50 kBit	6
20 kBit	7
10 kBit	8

This service allows all LSS Slaves in configuration mode. The service has to be followed by an Activate Bit Timing Parameters service to activate the configured parameters. After execution of the Configure Bit Timing Parameters service the node may not execute any remote LSS services except the services Configure Bit Timing Parameters, Activate Bit Timing Parameters and Switch Mode.

The remote result parameter confirms the success or failure of the service. In case of a failure optionally the reason is confirmed.

3.5.3 Activate Bit Timing Parameters

Through the Activate Bit Timing Parameters service the LSS Master activates the bit timing as defined by the Configure Bit Timing Parameters service.

Parameter	*Request/Indication*
Argument switch_delay	**Mandatory** mandatory

The switch_delay parameter specifies the length of two delay periods of equal length, which are necessary to avoid operating the bus with differing bit timing parameters. Each node performs the actual switch of the bit timing parameters ´switch_delay´ milliseconds after the reception of the command. After performing the switch, a node does not transmit any messages before the second time ´switch_delay´ has passed.

Note

Nodes may have different processing times for performing the Activate Bit Timing Parameters command and messages that are transmitted before this command may still be in the receive queue of a node.

This means that a node may still transmit CAN messages with the old bit timing during the duration of the processing delay. Therefore switch_delay has to be longer than the longest processing time of any node in the network to avoid that a node already switches while another node still transmits using the old bit timing parameters. After the time specified by switch_delay has passed the first time, every node must perform the switch during the second duration of switch_delay. Therefore after switch_delay has passed the second time, all nodes are guaranteed to be listening with the new bit timing parameters. The diagram in figure 3 shows the location of the two switch_delay periods.

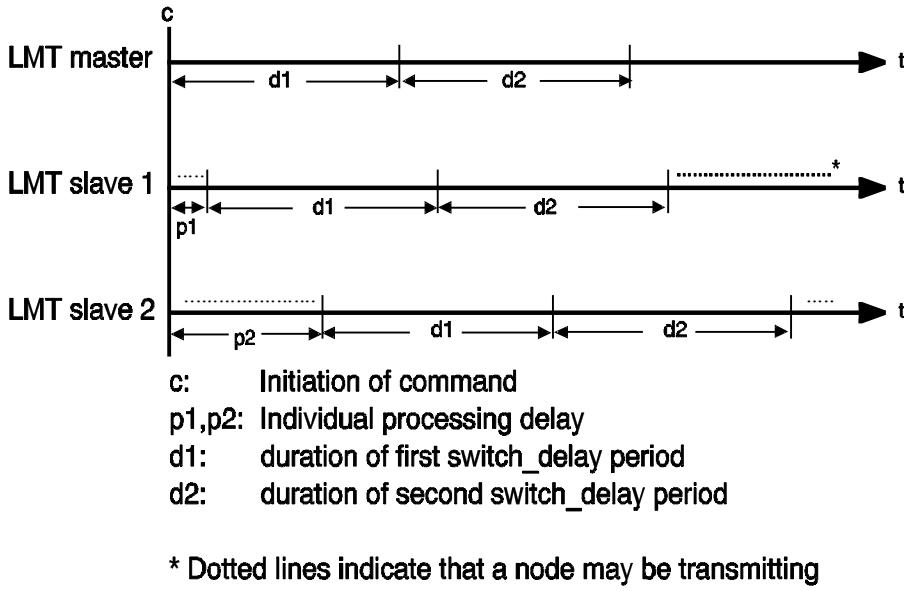


Fig. 2 : Definition of the two switch_delay periods

3.5.4 Store Configured Parameters

The Store Configured Parameters service is used to actuially store the configured parameters into non-volatile storage.

Parameter	*Request/Indication*	*Response/Confirmation*
Argument	**Mandatory**	
Remote Result success failure reason		**Mandatory** selection selection optional

The remote result parameter confirms the success or failure of the service. In case of a failure optionally the reason is confirmed.

3.6 INQUIRY SERVICES

The inquiry services are available only in configuration mode.

3.6.1 Inquire LSS Address

This service allows to determine the LSS-address parameters of a LSS Slave in configuration mode.

Parameter	*Request/Indication*	*Response/Confirmation*
Argument **Remote Result** LSS_address vendor-id product-code revision-number serial-number failure reason	**Mandatory**	 **Mandatory** selection mandatory mandatory mandatory mandatory selection optional

Exactly one LSS slave may be in configuration mode when this service is executed. The remote result parameter confirms the LSS address of the LSS Slave in configuration mode or the failure of the service. In case of a failure optionally the reason is confirmed.

3.7 Identification Services

3.7.1 LSS Identify Remote Slaves

Through this service, the LSS Master requests all LSS slaves, whose LSS address meets the LSS_Address_sel to identify themselves through the 'LSS Identify Slave' service. LSS_Address_sel consists of a fixed manufacturer and product name and a span of serial numbers. This service is unconfirmed.

Parameter	*Request/Indication*
Argument LSS_Address_sel	Mandatory mandatory

3.7.2 LSS Identify Slave

Through this service, an LSS Slave indicates, that it is a Slave with an LSS address within the LSS_Address_sel of an 'LSS Identify Remote Slave' service executed prior to this service. The service is unconfirmed.

Parameter	*Request/Indication*
Argument	**Mandatory**

3.8 LSS Protocol Perspective

The LSS Protocol is executed between the LSS Master and each of the LSS Slaves to implement these services.

3.8.1 LSS Slave Synchronisation

Since in the LSS Protocol all LSS Slaves use the same COB to send information to the LSS Master, there must be only one LSS Slave at a time that communicates with the LSS Master. For all protocols the LSS Master takes the initiative, a LSS 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 LSS service outstanding at a time, the synchronisation is established.

3.8.2 LSS Protocol Descriptions

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

Requesting Messages (from LSSMaster) using COB-ID 2021. Response Messages (from LSSSlave) using COB-ID 2020.

LSS uses command specifiers to identify the commands. Command specifiers from 0 - 07fh are reserved for use by LMT, respectively LSS. The range from 0 – 03fh are reserved for use by LMT services. The range from 040h – 07fh are reserved for use by standard LSS services. Command specifiers from 080h - 0ffh are free for application specific purposes and may only be used with at most one LSS Slave in configuration mode.

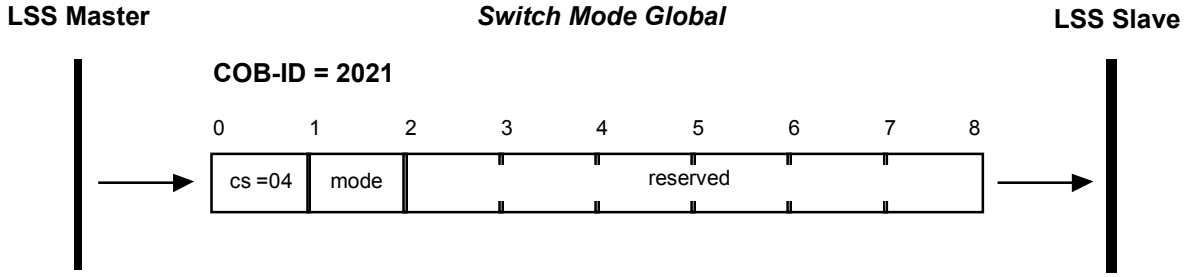
In the description of the COB data format, bytes are numbered from zero to and including seven. Bits within a byte are numbered from zero to and including seven. Byte zero is transmitted first, byte seven is transmitted last. Within a byte, bit zero is the least significant bit, bit seven is the most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

3.9 SWITCH MODE PROTOCOLS

3.9.1 Switch Mode Global

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



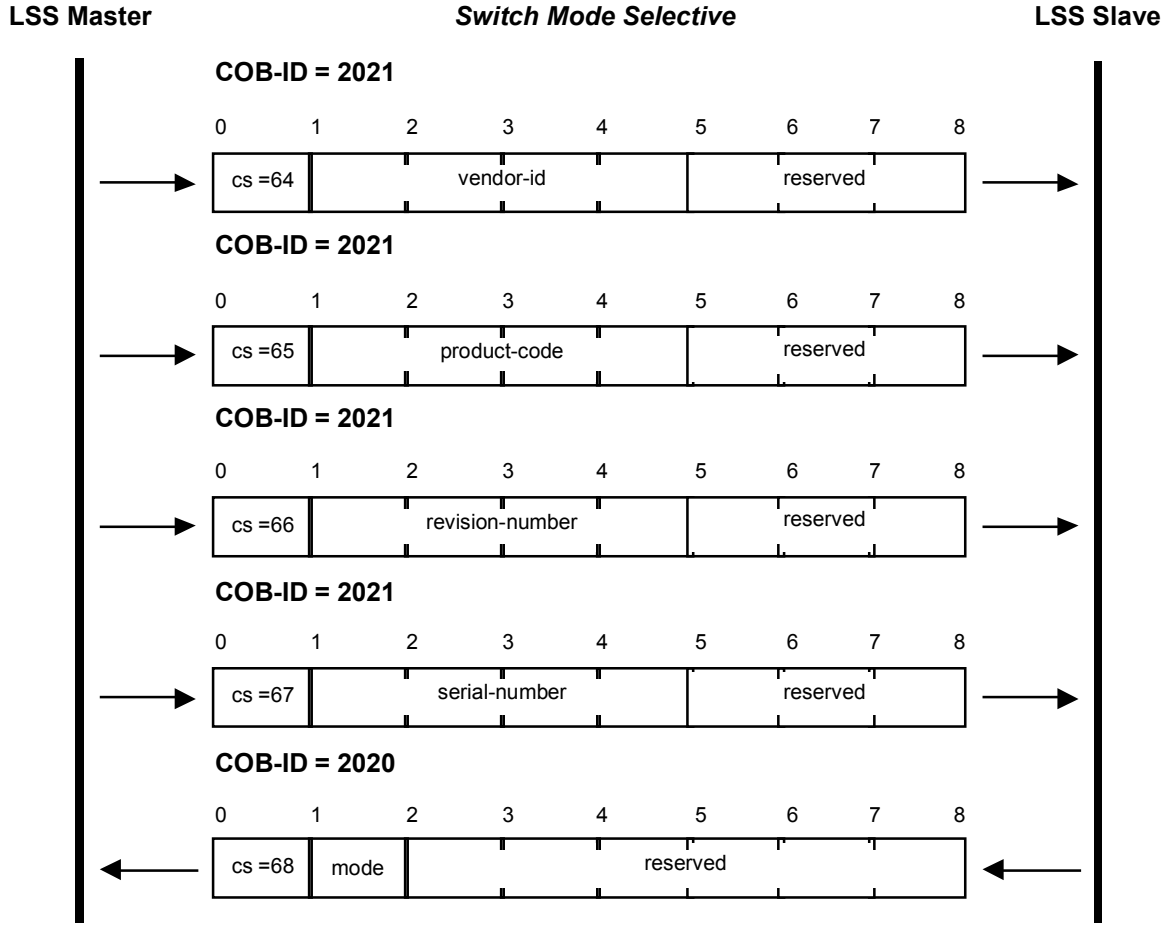
- **cs**:
  LSS command specifier
  04 for Switch Mode Global

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

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

3.9.2 Switch Mode Selective

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



- **cs**:
  LSS command specifiers;
  64 to 68 for Switch Mode Selective

- **vendor-id**:
  Vendor name part of the LSS address, with respect to index 1018h, subindex 1

- **product-code**:
  Product name part of the LSS address, with respect to index 1018h, subindex 2

- **revision-number**:
  Revision part of the LSS address, with respect to index 1018h, subindex 3

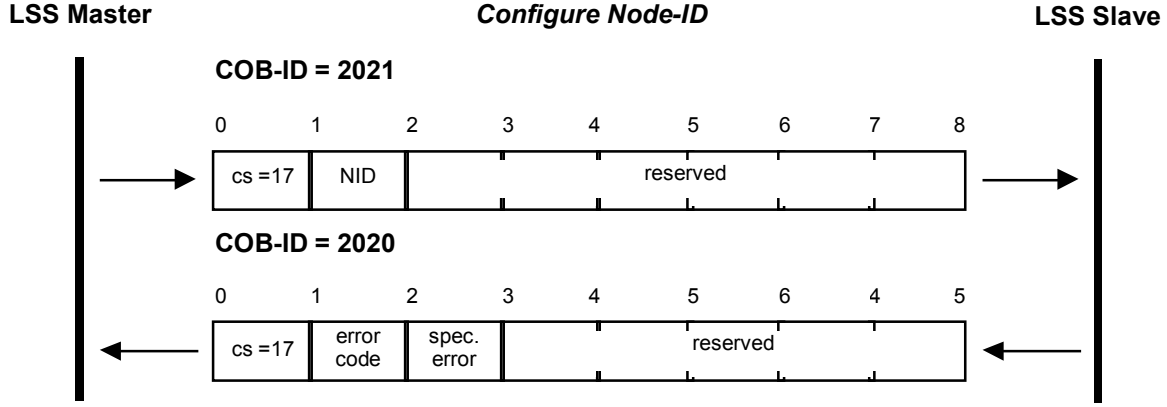
- **serial_number**:
  Serial number part of the LSS address, with respect to index 1018h, subindex 4

- **mode**:
  The actual LSS mode of the Slave
  0: operation mode
  1: configuration mode

3.10 CONFIGURATION PROTOCOLS

3.10.1 Configure Node-ID Protocol

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



- **cs**:
  LSS command specifier
  17 for Configure Node-ID

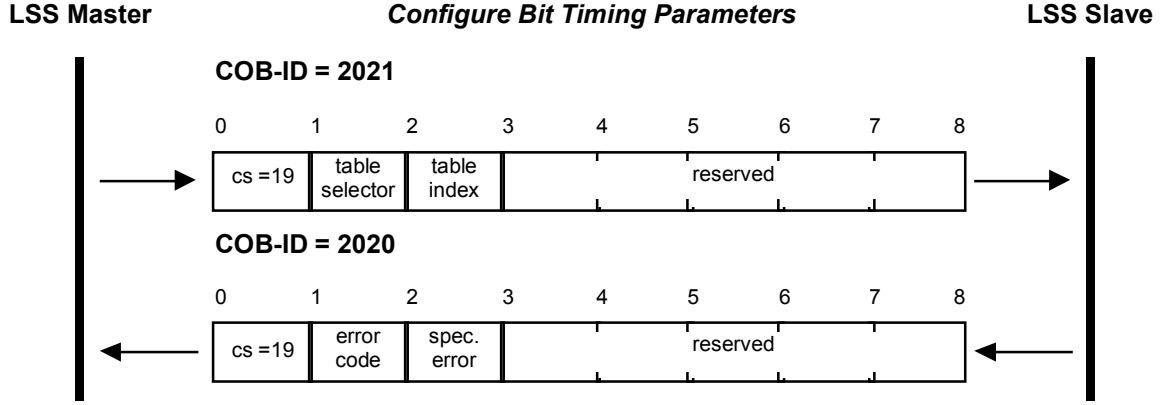
- **NID**:
  The new Node-ID to configure, see /2/

- **error_code**:
  0 : protocol successfully completed
  1 : Node-ID out of range
  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

3.10.2 Configure Bit Timing Parameters Protocol

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



- **cs**:
  LSS 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

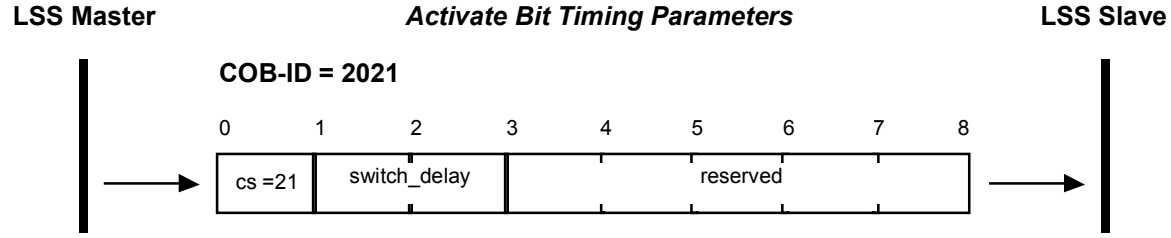
- **table_index**:
  selects the entry (bit timeing parameters) in the selected table; see 3.5.2 for valid indices when using the standard CiA bit timings (table_selector = 0)
- **error_code**:
  0: protocol successfully completed
  1: bit timing not supported
  2..254: reserved for further use by CiA
  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.

3.10.3 Activate Bit Timing Parameters Protocol

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



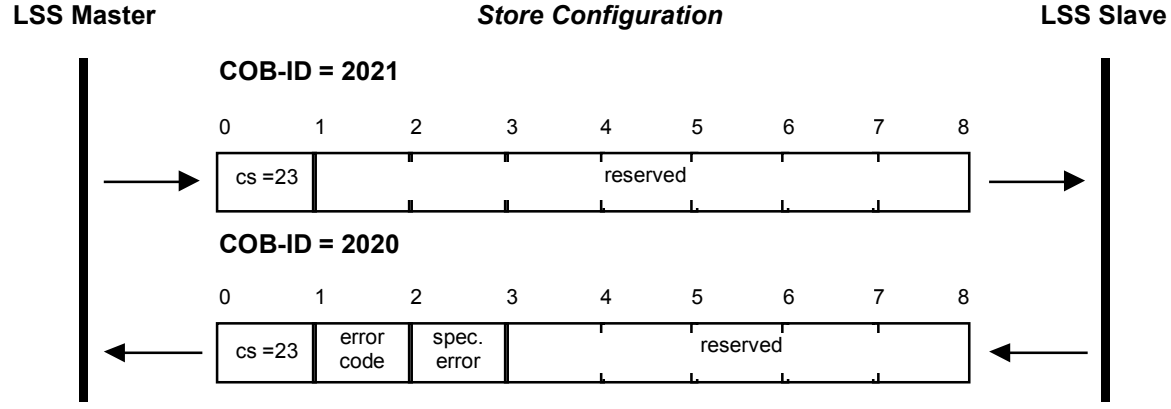
- **cs**:
  LSS command specifier
  21 for Activate Bit Timing Parameters

- **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.

3.10.4 Store Configuration Protocol

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



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

- **error_code**:
  0: protocol successfully completed,
  1: store configuration is not supported,
  2: storage media access error
  3 . . 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.

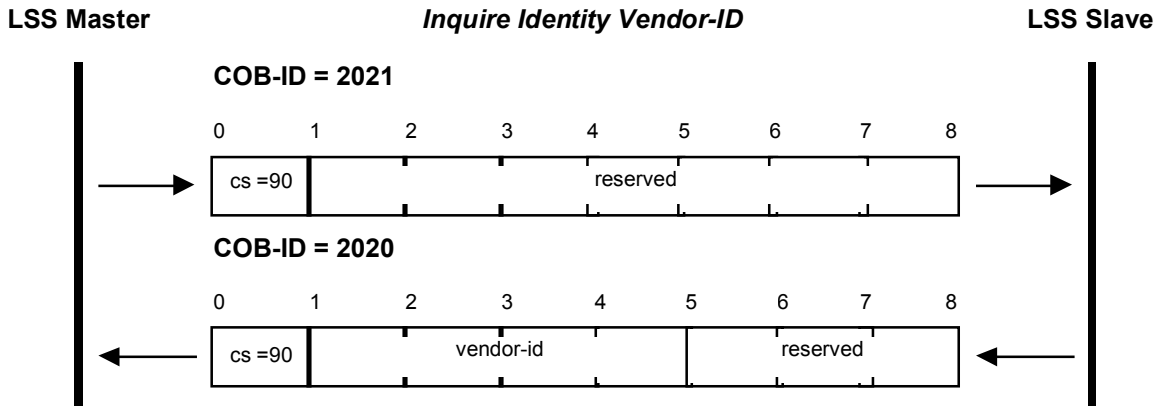
- **reserved**:
  reserved for further use by CiA.

3.11 INQUIRY PROTOCOLS

3.11.1 Inquire LSS Address Protocols

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

3.11.1.1 Inquire Identity Vendor-ID Protocol

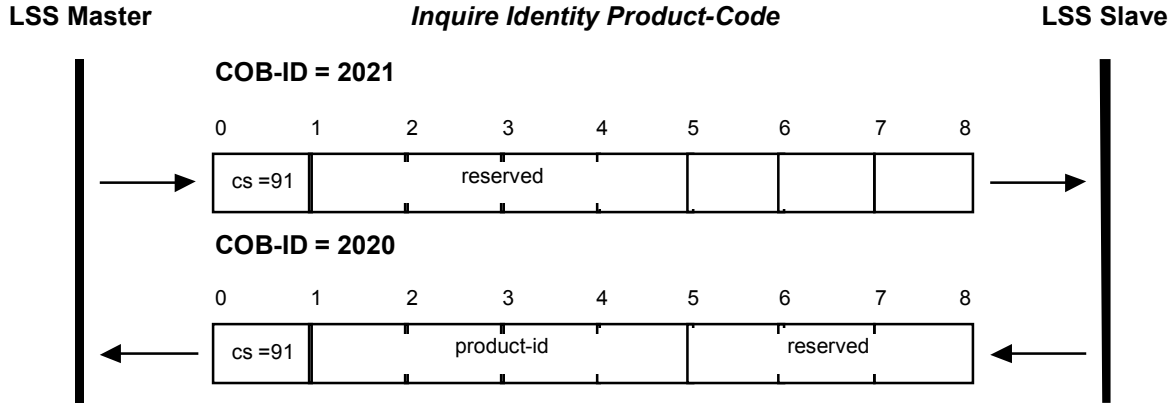


- **cs**:
  LSS command specifier
  90 for Inquire Manufacturer Name

- **vendor-id**:
  The vendor-id (see /2/) of the selected module.

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

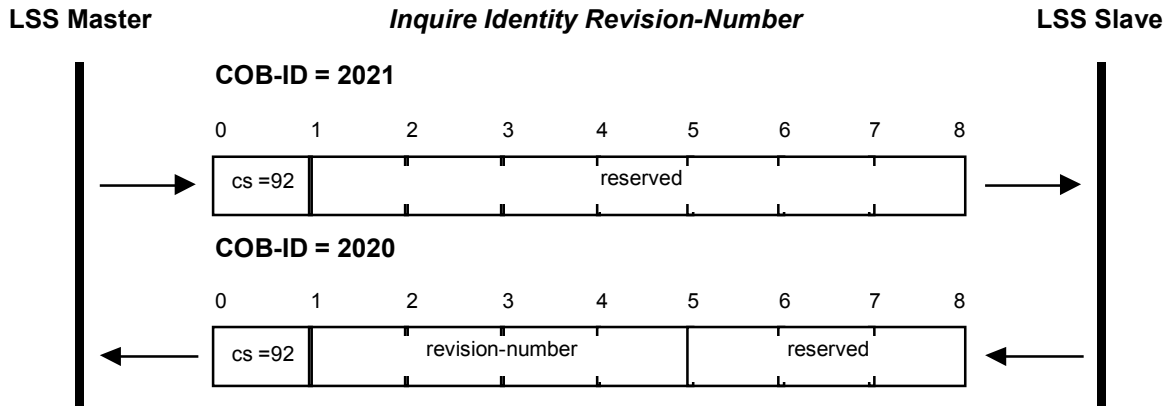
3.11.1.2 Inquire Identity Product-Code Protocol



- **cs**:
  LSS command specifier
  91 for Inquire Product Name
- **product-code**:
  The product-code (see /2/) of the selected module.

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

3.11.1.3 Inquire Identity Revision-Number Protocol

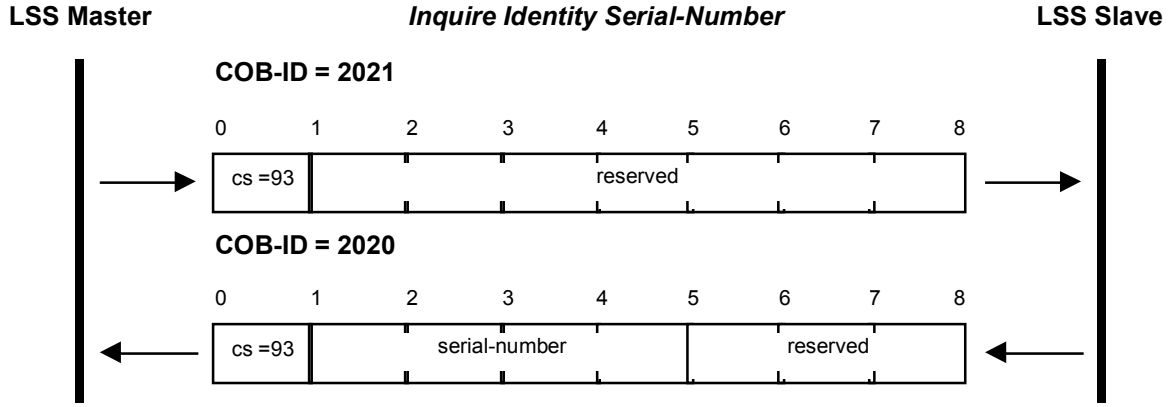


- **cs**:
  LSS command specifier
  92 for Inquire Serial Number

- **revision-number**:
  The revision-number (see /2/) of the selected module.

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

3.11.1.4 Inquire Identity Serial-Number Protocol



- **cs**:
  LSS command specifier
  93 for Inquire Serial Number

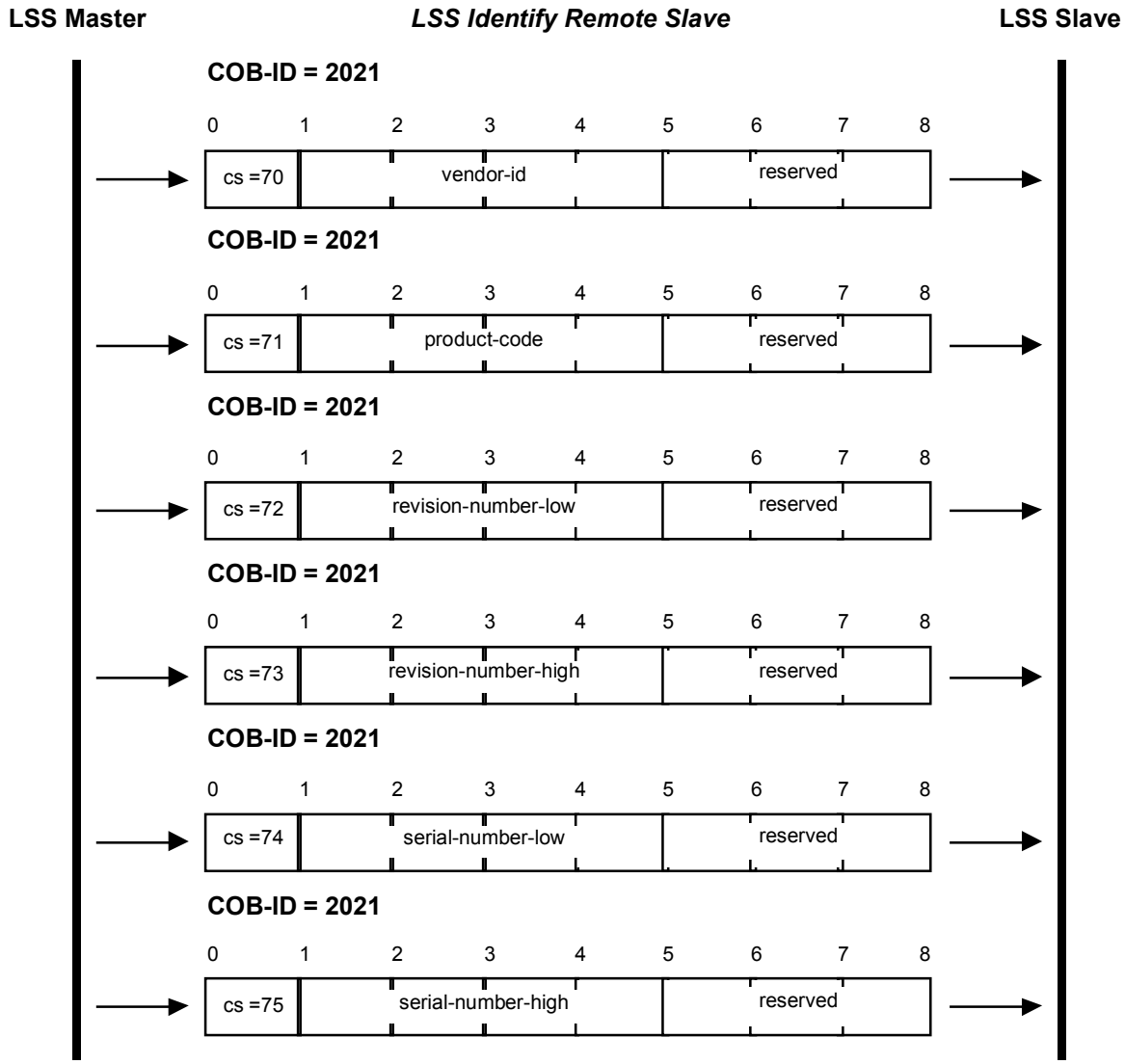
- **serial-number**:
  The serial-number (see /2/) of the selected module.

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

3.12 IDENTIFICATION PROTOCOLS

3.12.1 LSS Identify Remote Slaves

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

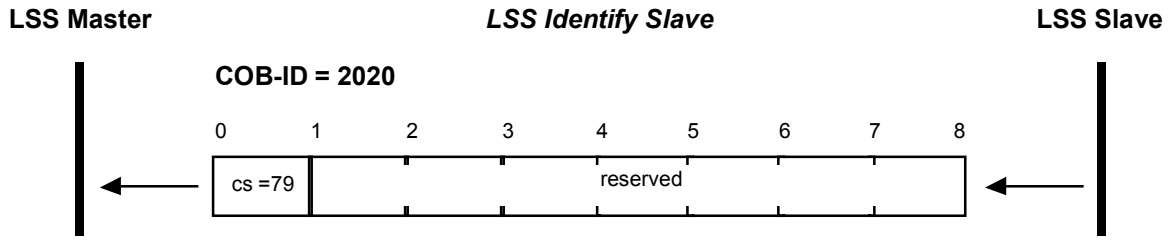


- **cs**:
  LSS command specifier
  70 to 75 for LSS Identify Remote Slaves
- **vendor-id**:
  The manufacturer name part of the LSS Address
- **product-code:**
  The product name part of the LSS Address
- **revision-number-low:**
  The lower boundary of the requested revision numbers range. The Minore range must be set to 0000h.
- **revision-number-high:**
  The higher boundary of the requested revision numbers range. The Minor range must be set to FFFFh.
- **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 LSS Slaves with matching vendor-id and product-code whose major revision-number and serial-numbers lie within the given ranges, are requested to identify themselves with the LSS Identify Slave service.

3.12.2 LSS Identify Slave Protocol

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



- **cs:**
  LSS command specifiers
  79 for Identify Slave

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

3.13 IMPLEMENTATION RULES

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

CAL Layer Management (LMT)

To distinguish between LMT and LSS, all for LSS services used command specifiers are fixed to a range from 040h – 07fh.

Invalid COB's

A COB is invalid if it has a COB-ID that is used by the LSS Protocol, but contains invalid parameter values according to the LSS Protocol. This can be caused by errors in the data link layer (see /1/) or implementaton errors. Invalid COB's must be handled locally in an implementation specific way that does not fall within the scope of the /2/. As far as the LSS Protocol is concerned, an invalid COB must be ignored.

Time-Outs

Since COB's may be ignored, the response of a confirmed LSS 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 LSS 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 /2/. However, it is recommended that an implementation provides facilities to adjust these time-out values to the requirements of the application.