



## **CarAPI message format specification**

**IAV GmbH**

**Revision pre 0.4 (171)**  
**October, 2002 (21.10.02 10:39)**

1	General Concept .....	3
1.1.1	Message format .....	4
1.1.2	Message body format .....	4
1.1.3	Message list format .....	4
1.1.4	Message block format.....	4
1.1.5	Message ID Types .....	4
1.1.6	Sample message .....	5
2	Messages and Message contents .....	6
2.1	General messages .....	6
2.1.1	Device Enumeration Messages .....	6
2.1.2	Connection Enumeration Messages .....	7
2.1.3	Broadcast Registration Messages .....	8
2.1.4	Message Buffer Space Management .....	9
2.2	µC messages .....	10
2.2.1	µC Flash messages .....	10
2.3	Power messages.....	11
2.3.1	Ignition State Messages.....	11
2.3.2	Battery Level Messages.....	11
2.3.3	Power Switch Messages .....	12
2.4	HID messages .....	13
2.4.1	Key Messages .....	13
2.4.2	Dial Messages .....	13
2.5	Audio Messages .....	14
2.5.1	Audio Selector messages .....	14
2.5.2	Audio Volume messages .....	15
2.5.3	Audio Balance messages .....	16
2.5.4	Audio Fader messages .....	17
2.5.5	Audio Bass messages .....	18
2.5.6	Audio Treble messages .....	19
2.5.7	Audio Loudness messages .....	20
2.5.8	Audio Tuner messages .....	21
2.6	CAN messages .....	24
2.6.1	CAN general messages .....	24
2.6.2	CAN messages for physical connections .....	25
2.6.3	CAN messages for logical connections .....	27
2.6.4	CAN messages for data transfer .....	32
2.7	KLINE messages .....	34
2.7.1	KLINE general messages .....	34
2.7.2	KLINE messages for physical connections .....	35
2.7.3	KLINE messages for logical connections .....	37
2.7.4	KLINE messages for data transfer.....	41
3	Message overview.....	43
4	Revision History .....	45



## 1 General Concept

- Messages are encoded in message blocks.
- An outer wrapper contains a link ID that identifies the application that sent the message and that may receive a confirmation message. A broadcast message may use -1 as Link ID to specify that a message is sent to all active communication links (applications)
- A message block contains the ID of the source/target unit, an application defined message handle, an ID identifying the message, a general status field (used by confirmation messages to indicate a commands execution status) and the message parameters.
- The message parameters are wrapped by a length field to allow for easy extension of the parameters and the main message block as well.
- There are three types of messages:
  - o command messages (CMD) specifying a command to execute,
  - o confirmation messages (CON) that are sent back upon completion of a command and
  - o event messages (EVT) that are sent when a hardware event occurs
- Unit IDs identify a specific functional instance. The existing units can be enumerated to get a list of Units and their capabilities
- Message IDs are unique
- Messages are combined of three basic types:
  - o A Value is coded as a byte that specifies the type of value, a byte that specifies the number of the following bytes that compose the value in big-endian format. Only used bytes are encoded (e.g. the unsigned 32bit value 0x00123456 is coded as 0x05 0x03 0x56 0x34 0x12 – 05 = type, 03 = bytes used for data, 56 34 12 = data)
  - o A structured Block is composed of a leading escape byte (value 0xff) followed by a length value that holds the number of bytes in the rest of the block. The rest of the block contains any number of the three types
  - o An unstructured block is composed of a leading escape byte (value 0xfe) followed by a length value that holds the number of bytes in the rest of that block. The rest of that block contains unstructured data

## 1.1.1 Message format

Name	Type	Description	Notes
Message	Structured Block	Contains message body	Envelope for message blocks

## 1.1.2 Message body format

Name	Type	Description	Notes
Link ID	Unsigned value	Application Link ID	Use -1 for broadcast
Message list	Structured Block	Consists of one or more message blocks	

## 1.1.3 Message list format

Name	Type	Description	Notes
Message Block	Structured block	Contains single message block	
...	Structured Block	May contain additional message blocks	

## 1.1.4 Message block format

Name	Type	Description	Notes
Device ID	Unsigned value	ID of object sending/receiving message	
Message Handle	Unsigned value	Application defined handle	Returned in _CON
Message ID	Unsigned value	ID of message	should be unique
Status	Unsigned value	Used in _CON messages for the result	0 = success 1 = more data
Parameters	Structured Block	Holds message specific parameters	

## 1.1.5 Message ID Types

Type	Description
CMD	command messages sent by application
CON	sent by driver in response to a command message, contains result and other requested information
EVT	event message sent by driver



## 1.1.6 Sample message

(queries audio volume and balance settings)

Data	Hierarchy level / Description	Value
0f 01 32	length:	50
05 01 01	link id:	1
0f 01 2c	block list length:	44
0f 01 13	block length:	19
05 01 33	unit id:	0x33
05 04 a0 e5 86 fd	message handle:	0xfd86e5a0
05 04 09 01 03 04	message id:	0x04030109 (Audio Volume)
00	status:	unused (0)
0f 01 00	parameter length:	0
	parameter body:	empty
0f 01 13	block length:	19
05 01 33	unit id:	0x33
05 04 a0 e5 86 fd	message handle:	0xfd86e5b0
05 04 09 02 03 04	message id:	0x04030209 (Audio Balance)
00	status:	unused (0)
0f 01 00	parameter length:	0
	parameter body:	empty

## 2 Messages and Message contents

### 2.1 General messages

#### 2.1.1 Device Enumeration Messages

##### MSG\_Unit\_EnumDevices\_CMD

###### Description

Requests a list of all devices and their properties in the given class. The block header member 'Device ID' must be 0.

###### Parameters:

Name	Type	Description
Device Class	Unsigned value	Describes the class that should be enumerated. The device class may contain a class, a subclass and a type field and only the devices that match the given class specification are enumerated.

##### MSG\_Unit\_EnumDevices\_EVT

###### Description

Contains a device description after an application has started a device enumeration process.

###### Parameters:

Name	Type	Description
Group ID	Unsigned value	Specifies an object group to which this device belongs.
Function block list	Structured block	Contains a set of function block descriptions coded as structured blocks. A function block is a unit that implements a set of interfaces and exports a set of ports.

A **Function block** description consists of the following elements

Name	Type	Description
Interface type list	Structured block	Contains an array of interface type values. An interface type consists of class, subclass and type information and thus specifies support for a specific set of messages.
Port description list	Structured block	Contains an array of Port description blocks. A port description specifies the function/compatibility of a port and its connections

A **Port description block** consists of the following elements

Name	Type	Description
Port ID	Unsigned value	Identifies the port.
Port type	Unsigned value	

##### MSG\_Unit\_EnumDevices\_CON

###### Description

Terminates a device enumeration process. The block header member 'Device ID' must be 0.

###### Parameters:

Name	Type	Description
Device Class	Unsigned value	Describes the class that should be enumerated. The device class may contain a class, a subclass and a type field and only the devices that match the given class specification are enumerated.

## 2.1.2 Connection Enumeration Messages

### MSG\_Unit\_EnumConnections\_CMD

#### Description

Requests a list of all device connections. The block header member 'Device ID' must be 0.

#### Parameters:

none

### MSG\_Unit\_EnumConnections\_EVT

#### Description

Contains a set of connection statements and is sent after an application has started the connection enumeration process. The block header member 'Device ID' must be 0.

#### Parameters:

Name	Type	Description
Connection description list	Structured block	Contains a set of connection descriptions coded as structured blocks.

A **connection description** consists of the following elements

Name	Type	Description
Source port	Structured block	Contains a port specification containing the source port.
Destination port	Structured block	Contains a port specification containing the destination port.

A **port specification** consists of the following elements

Name	Type	Description
Device ID	Unsigned value	Identifies the device
Port ID	Unsigned value	Identifies the port

### MSG\_Unit\_EnumConnections\_CON

#### Description

Terminates a connection enumeration process. The block header member 'Device ID' must be 0.

#### Parameters:

none

## 2.1.3 Broadcast Registration Messages

### MSG\_Unit\_RegisterClient\_CMD

#### Description

Requests that the unit sends broadcasts to this application. A unit may send broadcasts to all applications even if there was no RegisterClient message.

#### Parameters:

none

### MSG\_Unit\_RegisterClient\_CON

#### Description

Confirms the broadcast activation.

#### Parameters:

None

### MSG\_Unit\_UnregisterClient\_CMD

#### Description

Requests that the unit stops sending broadcasts to this application. The application may however still receive broadcast from that unit.

#### Parameters:

none

### MSG\_Unit\_UnregisterClient\_CON

#### Description

Confirms the broadcast deactivation.

#### Parameters:

none



## 2.1.4 Message Buffer Space Management

### MSG\_MessageBuffer\_SetSize\_CMD

#### Description

Requests that the Buffer space for messages on the used link ID is adjusted to the given size.

#### Parameters:

Name	Type	Description
BufferSize	Unsigned value	Requested buffer size

### MSG\_MessageBuffer\_SetSize\_CON

#### Description

Confirms that the Buffer space for messages on the used link ID was adjusted to the given size.

#### Parameters:

Name	Type	Description
BufferSize	Unsigned value	Requested buffer size



## 2.2 $\mu$ C messages

### 2.2.1 $\mu$ C Flash messages

#### CARMSG\_Flash\_Write\_CMD

##### Description

Requests that the attached data block is written to the specified memory location

##### Parameters:

Name	Type	Description
Address	Unsigned value	Destination Address of data block
Length	Unsigned value	Number of bytes in data block
Data	Unstructured block	Data block

#### CARMSG\_Flash\_Write\_CON

##### Description

indicates that a data block was written to the specified memory location (check status field for operation results)

##### Parameters:

Name	Type	Description
Address	Unsigned value	Address where the data was written

## 2.3 Power messages

For proper operation of the power management on the device, the application should send a MSG\_Unit\_RegisterClient\_CMD message to the device.

### 2.3.1 Ignition State Messages

#### CARMSG\_Ignition\_State\_CMD

##### Description

Queries the state of the ignition key

##### Parameters:

None

#### CARMSG\_Ignition\_State\_CON

##### Description

indicates the state of the ignition key

##### Parameters:

Name	Type	Description
State	Unsigned value	Current state of the ignition key

#### CARMSG\_Ignition\_State\_EVT

##### Description

indicates that the state of the ignition key has changed

##### Parameters:

Name	Type	Description
State	Unsigned value	New state of the ignition key

### 2.3.2 Battery Level Messages

#### CARMSG\_Battery\_Level\_CMD

##### Description

Queries the level of the battery

##### Parameters:

None

#### CARMSG\_Battery\_Level\_CON

##### Description

indicates that the battery level has changed

##### Parameters:

Name	Type	Description
Level	Unsigned value	New battery level (Percent where 0% means 'critical power level')

#### CARMSG\_Battery\_Level\_EVT

##### Description

indicates that the battery level has changed

##### Parameters:

Name	Type	Description
Level	Unsigned value	New battery level (Percent where 0% means 'critical power level')

## 2.3.3 Power Switch Messages

### CARMSG\_PowerSupply\_SwitchOff\_CMD

#### Description

Queries for a power off after the given delay

#### Parameters:

Name	Type	Description
Delay	Unsigned value	Delay before power off in ms

### CARMSG\_PowerSupply\_SwitchOff\_CON

#### Description

Confirms the power off command without waiting for the timeout to elapse. This message is sent only if a delay has been specified.

#### Parameters:

None

### CARMSG\_PowerSupply\_SwitchOff\_EVT

#### Description

Indicates that the PC will be switched off after the given delay

#### Parameters:

Name	Type	Description
Delay	Unsigned value	Delay before power off in ms



## 2.4 HID messages

### 2.4.1 Key Messages

#### CARMSG\_Key\_Down\_EVT

##### Description

indicates that the key has been pressed

##### Parameters:

none

#### CARMSG\_Key\_Up\_EVT

##### Description

Indicates that the key has been released

##### Parameters

Name	Type	Description
Duration	Unsigned value	Measured time between key down and key up events in ms (may be zero if unavailable or larger than one second)

### 2.4.2 Dial Messages

#### CARMSG\_Dial\_TurnLeft\_EVT

##### Description

indicates that the dial has been turned left one step

##### Parameters:

Name	Type	Description
Steps	Unsigned value	Number of steps the dial has been turned in that direction since the last event

#### CARMSG\_Dial\_TurnRight\_EVT

##### Description

indicates that the dial has been turned right one step

##### Parameters:

Name	Type	Description
Steps	Unsigned value	Number of steps the dial has been turned in that direction since the last event

## 2.5 Audio Messages

### 2.5.1 Audio Selector messages

#### CARMSG\_AudioSelector\_QueryCaps\_CMD

##### Description

Requests a description of the devices capabilities.

##### Parameters:

none

#### CARMSG\_AudioSelector\_QueryCaps\_CON

##### Description

Describes an input selectors capabilities

##### Parameters:

Name	Type	Description
NumPositions	Unsigned value	Number of input sources

#### CARMSG\_AudioSelector\_Position\_CMD

##### Description

Requests that the given source is selected. To query the current position, the parameter may be omitted.

##### Parameters:

Name	Type	Description
Position	Unsigned value	Desired source selection (optional)

#### CARMSG\_AudioSelector\_Position\_CON

##### Description

Indicates that a new input source has been selected

##### Parameters:

Name	Type	Description
Position	Unsigned value	Actual/New selected source

## 2.5.2 Audio Volume messages

### CARMSG\_AudioVolume\_QueryCaps\_CMD

#### Description

Requests a description of the devices capabilities.

#### Parameters:

none

### CARMSG\_AudioVolume\_QueryCaps\_CON

#### Description

Describes a volume attenuators capabilities

#### Parameters:

Name	Type	Description
MinVolume	Signed value	Minimum Volume level
MaxVolume	Signed value	Maximum Volume level

### CARMSG\_AudioVolume\_Position\_CMD

#### Description

Requests that the given volume level is set. To query the current level, the parameter may be omitted.

#### Parameters:

Name	Type	Description
Volume	Signed value	Desired volume level (optional)

### CARMSG\_AudioVolume\_Position\_CON

#### Description

indicates that a new volume level has been set

#### Parameters:

Name	Type	Description
Volume	Signed value	Actual/New Volume level

## 2.5.3 Audio Balance messages

### CARMSG\_AudioBalance\_QueryCaps\_CMD

#### Description

Requests a description of the devices capabilities.

#### Parameters:

none

### CARMSG\_AudioBalance\_QueryCaps\_CON

#### Description

Describes a balance attenuators capabilities

#### Parameters:

Name	Type	Description
Left	Signed value	Leftmost position
Right	Signed value	Rightmost position
Center	Signed value	Center position (optional)

### CARMSG\_AudioBalance\_Position\_CMD

#### Description

Requests that the given balance position is set. To query the current position, the parameter may be omitted.

#### Parameters:

Name	Type	Description
Position	Signed value	Desired balance level (optional)

### CARMSG\_AudioBalance\_Position\_CON

#### Description

Indicates that a new balance level has been set

#### Parameters:

Name	Type	Description
Position	Signed value	Actual/New balance level



## 2.5.4 Audio Fader messages

### CARMSG\_AudioFader\_QueryCaps\_CMD

#### Description

Requests a description of the devices capabilities.

#### Parameters:

none

### CARMSG\_AudioFader\_QueryCaps\_CON

#### Description

Describes a fader attenuators capabilities

#### Parameters:

Name	Type	Description
Front	Signed value	Foremost position
Rear	Signed value	Hindmost position
Center	Signed value	Center position (optional)

### CARMSG\_AudioFader\_Position\_CMD

#### Description

Requests that the given balance position is set. To query the current position, the parameter may be omitted.

#### Parameters:

Name	Type	Description
Position	Signed value	Desired fader level (optional)

### CARMSG\_AudioFader\_Position\_CON

#### Description

Indicates that a new balance level has been set

#### Parameters:

Name	Type	Description
Position	Signed value	Actual/New fader level

## 2.5.5 Audio Bass messages

### CARMSG\_AudioFilterBass\_QueryCaps\_CMD

#### Description

Requests a description of the devices capabilities.

#### Parameters:

none

### CARMSG\_AudioFilterBass\_QueryCaps\_CON

#### Description

Describes a bass filters capabilities

#### Parameters:

Name	Type	Description
Min	Signed value	Min position (maximal attenuation)
Max	Signed value	Max position (minimal attenuation, max boost)
Center	Signed value	Center position
Center Frequency	Unsigned value	Center frequency

### CARMSG\_AudioFilterBass\_Position\_CMD

#### Description

Requests that the given filter position is set. To query the current position, the parameter may be omitted.

#### Parameters:

Name	Type	Description
Position	Signed value	Desired filter position (optional)

### CARMSG\_AudioFilterBass\_Position\_CON

#### Description

indicates that a new filter position has been set

#### Parameters:

Name	Type	Description
Position	Signed value	Actual/New filter position

## 2.5.6 Audio Treble messages

### CARMSG\_AudioFilterTreble\_QueryCaps\_CMD

#### Description

Requests a description of the devices capabilities.

#### Parameters:

none

### CARMSG\_AudioFilterTreble\_QueryCaps\_CON

#### Description

Describes a treble filters capabilities

#### Parameters:

Name	Type	Description
Min	Signed value	Min position (maximal attenuation)
Max	Signed value	Max position (minimal attenuation, max boost)
Center	Signed value	Center position
Center Frequency	Unsigned value	Center frequency

### CARMSG\_AudioFilterTreble\_Position\_CMD

#### Description

Requests that the given filter position is set. To query the current position, the parameter may be omitted.

#### Parameters:

Name	Type	Description
Position	Signed value	Desired filter position (optional)

### CARMSG\_AudioFilterTreble\_Position\_CON

#### Description

Indicates that a new filter position has been set

#### Parameters:

Name	Type	Description
Position	Signed value	Actual/New filter position

## 2.5.7 Audio Loudness messages

### CARMSG\_AudioFilterLoudness\_QueryCaps\_CMD

#### Description

Requests a description of the devices capabilities.

#### Parameters:

none

### CARMSG\_AudioFilterLoudness\_QueryCaps\_CON

#### Description

Describes a loudness filters capabilities

#### Parameters:

Name	Type	Description
Min	Signed value	Min position (no loudness correction)
Max	Signed value	Max position (maximal bass boost)
Center	Signed value	Center position

### CARMSG\_AudioFilterLoudness\_Position\_CMD

#### Description

Requests that the given filter position is set. To query the current position, the parameter may be omitted.

#### Parameters:

Name	Type	Description
Position	Signed value	Desired filter position (optional)

### CARMSG\_AudioFilterLoudness\_Position\_CON

#### Description

Indicates that a new filter position has been set

#### Parameters:

Name	Type	Description
Position	Signed value	Actual/New filter position

## 2.5.8 Audio Tuner messages

### CARMSG\_AudioTuner\_QueryCaps\_CMD

#### Description

Queries the capabilities of the tuner module.

#### Parameters:

none

### CARMSG\_AudioTuner\_QueryCaps\_CON

#### Description

Displays the capabilities of the tuner module.

#### Parameters:

Name	Type	Description
Area list	Structured block	One or more area definitions

An **Area list** contains the following elements

Name	Type	Description
Area description	Structured block	Area description
...	Structured block	Additional area descriptions

An **Area description** contains the following elements

Name	Type	Description
Area code	Unsigned value	Identifies the area. Possible values are 1: Europe 2: USA 3: East 4: Japan
Band list	Structured block	Contains a list of supported bands

A **Band list** contains the following elements

Name	Type	Description
Band description	Structured block	Band description
...	Structured block	Additional band descriptions

A **Band description** consists of the following elements

Name	Type	Description
Band type	Unsigned value	Identifies band. Possible values are: 1: FM (all frequencies divided by 100kHz) 2: SW 3: MW (all frequencies divided by 10kHz) 4: LW 5: WX
Minimum frequency	Unsigned value	Minimum frequency (uses frequency divisor used in current area, FM 100,00Mhz is reported as 1000)
Maximum frequency	Unsigned value	Maximum frequency (uses frequency divisor used in current area)
Frequency resolution	Unsigned value	Frequency resolution (uses frequency divisor used in current area, 50kHz is reported as 5)

## CARMSG\_AudioTuner\_SelectArea\_CMD

### Description

Requests selection of a specific area

### Parameters:

Name	Type	Description
Area	Unsigned value	ID of area to select

## CARMSG\_AudioTuner\_SelectArea\_CON

### Description

Confirm selection of a specific area.

### Parameters:

Name	Type	Description
Area	Unsigned value	Selected area ID

## CARMSG\_AudioTuner\_SelectBand\_CMD

### Description

Requests selection of a specific band

### Parameters:

Name	Type	Description
Band	Unsigned value	ID of band to select

## CARMSG\_AudioTuner\_SelectBand\_CON

### Description

Confirm selection of a specific area.

### Parameters:

Name	Type	Description
Band	Unsigned value	Selected band ID

## CARMSG\_AudioTuner\_SetFrequency\_CMD

### Description

Selects a frequency

### Parameters:

Name	Type	Description
Frequency	Unsigned value	Frequency to be set (uses frequency divisor used in current area)

## CARMSG\_AudioTuner\_SetFrequency\_CON

### Description

Confirms selection of a frequency

### Parameters:

Name	Type	Description
Frequency	Unsigned value	Frequency that has been set (uses frequency divisor used in current area)
Signal Strength	Unsigned value	Signal strength
Flags	Bit field	Additional info: Bit 0: Stereo Bit 1: Traffic info available

## CARMSG\_AudioTuner\_ScanBand\_CMD

### Description

Starts a band scan

### Parameters:

Name	Type	Description
Minimum frequency	Unsigned value	Minimum frequency (uses frequency divisor used in current area, FM 100,00Mhz is reported as 1000)
Maximum frequency	Unsigned value	Maximum frequency (uses frequency divisor used in current area)
Step	Unsigned value	Frequency resolution for scan (uses frequency divisor used in current area)

## CARMSG\_AudioTuner\_ScanBand\_EVT

### Description

Provides band scan results.

### Parameters:

Name	Type	Description
Frequency info list	Structured block	One or more Frequency info elements

A **Frequency info** contains the following elements

Name	Type	Description
Frequency	Unsigned value	Current frequency
Signal strength	Unsigned value	Signal strength
Flags	Bit field	Additional info: Bit 0: Stereo Bit 1: Traffic info available

## CARMSG\_AudioTuner\_ScanBand\_CON

### Description

Confirms and ends a band scan

### Parameters:

none

## 2.6 CAN messages

### 2.6.1 CAN general messages

#### CARMSG\_IOCAN\_QueryCaps\_CMD

##### Description

Queries the capabilities of the CAN module.

##### Parameters:

none

#### CARMSG\_IOCAN\_QueryCaps\_CON

##### Description

Displays the capabilities of the CAN module.

##### Parameters:

Name	Type	Description
PhysConnCaps list	Structured block	List of Physical Connection Caps elements

**Physical Connection Caps** contains the following elements

Name	Type	Description
PhysConnectionType	Unsigned value	Type of connection 1: raw mailbox access
PhysConnProperties	Structured block	Physical connection properties
LogConnCaps list	Structured block	List of Logical Connection Caps

**Physical Connection Properties** contains the following elements

For physical connection type 1 (raw mailbox access):

Name	Type	Description
MaxBusSpeed	Unsigned value	Maximum bus speed in kBit/s

**Logical Connection Caps** contains the following elements

Name	Type	Description
LogConnectionType	Unsigned value	Type of connection For PhysConnType 1 (raw mailbox access) - 1: message based frame transmission - 2: ring buffer based frame transmission
LogConnFlags	Bit field	Supported logical connection flags (see CARMSG_IOCAN_LogicalConnect_CMD)
TransmissionFlags	Bit field	Supported message transmission flags (see CARMSG_IOCAN_Send_CMD)



## 2.6.2 CAN messages for physical connections

### CARMSG\_IOCAN\_PhysicalConnect\_CMD

#### Description

Requests the setup of a physical connection.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
PhysConnType	Unsigned value	Type of connection 1: raw mailbox access
PhysConnParameters	Structured block	Parameters for the connection

#### Physical connection parameters

for Connection Type 1 (raw mailbox access)

Name	Type	Description
Bit rate	Unsigned value	Bus speed in kBit/s
Bit timing	Structured block	Default results in sampling at 75% of the bit time

**Bit timing** contains the following elements

Name	Type	Description
Prop_Seg	Unsigned value	Length of Propagation Time Segment in Time Quantum's ( $t_q$ ). The Length of the Synchronization Segment is 1 $t_q$ .
Phase_Seg1	Unsigned value	Length of Phase Buffer Segment 1
Phase_Seg2	Unsigned value	Length of Phase Buffer Segment 2
ResyncJumpWidth	Unsigned value	Length of Resynchronization Jump Width

### CARMSG\_IOCAN\_PhysicalConnect\_CON

#### Description

Confirms the request for a physical connection.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier

### CARMSG\_IOCAN\_PhysicalConnect\_EVT

#### Description

Indicates that a physical connection was established.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that was connected
StackParameters	Structured block	

#### StackParameters

For PhysConnType 1 (direct mailbox) StackParameters is an empty struct.

**CARMSG\_IOCAN\_PhysicalDisconnect\_CMD****Description**

Requests termination of a physical connection.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that shall be disconnected

**CARMSG\_IOCAN\_PhysicalDisconnect\_CON****Description**

Confirms request for termination of a physical connection.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that shall be disconnected

**CARMSG\_IOCAN\_PhysicalDisconnect\_EVT****Description**

Indicates termination of a physical connection.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that was disconnected
DisconnectReason	Unsigned value	1: Hardware triggered (Bus-Off) 2: User mode triggered 3: other
Disconnect Information	Unsigned value	information about bus-off condition

### 2.6.3 CAN messages for logical connections

#### CARMSG\_IOCAN\_LogicalConnect\_CMD

##### Description

Requests the setup of a logical connection within a given physical connection.

##### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	Physical connection identifier
LogConnID	Unsigned value	logical connection identifier
LogConnType	Unsigned value	Type of connection For physical connection type: 1 (raw mailbox access): - 1: message based frame transfer - 2: ring buffer based frame transfer
LogConnParams	Structured block	Parameters for requested logical connection

**LogConnParams** contains the following elements

for LogConnType 1 (direct mailbox access with message based frame transfer):

Name	Type	Description
Flags	Bit field	Additional information Bit 0: Send enable Bit 1: Recv enable Bit 2: Acknowledge messages (receive) Bit 3: 11 bit mode Bit 4: 29 bit mode Bit 5: Sort send queue by ID priority Bit 6: Send timestamps Bit 7: Recv timestamps
ReceiveTimeout	Unsigned value	Receive timeout (0 is no timeout, most of the received frames will be sent in a separate message). The receive timer is started when the first new frame is received. The driver will wait the specified time for additional frames. The driver may send a group of frames before the timeout expires, when a certain number of frames have been collected.
MessageFilterList	Structured block	List of MessageFilter blocks

for LogConnType 2 (direct mailbox access with ringbuffer based frame transfer):

Name	Type	Description
Flags	Bit field	Additional information Bit 0: Send enable Bit 1: Recv enable Bit 2: Acknowledge messages (receive) Bit 3: 11 bit mode Bit 4: 29 bit mode Bit 5: Sort send queue by ID priority Bit 6: Send timestamps Bit 7: Recv timestamps
TransmitRingbuffer	Structured block	Describes the transmit ring buffer
ReceiveRingbuffer	Structured block	Describes the receive ring buffer
ReceiveTimeout	Unsigned value	Receive timeout in ms (0 is no timeout) before signaling
ReceiveLimit	Unsigned value	Maximum incoming frames before signaling, disregard of timeout
MessageFilterList	Structured block	List of MessageFilter blocks

**Ring buffer:**

Name	Type	Description
Address	Pointer	Base address of the ring buffer. The ring buffer must be valid until a logical disconnect EVT for the connection was received. For the ring buffer layout see header file
Element Count	Unsigned value	Number of frames that can be stored in the ring buffer
Element Size	Unsigned value	Size of a frame element in the ring buffer

**Ring buffer layout for direct mailbox access:**

```

struct framebuffer
{
    unsigned _int32    Flags
    unsigned _int32    CAN-ID
    unsigned _int8     Framedata[8]
    unsigned _int64    Timestamp
}

struct
{
    unsigned _int32    Read_Element
    unsigned _int32    Write_Element
    framebuffer        Frames[MyMaxFrameCount]
}

```

Ring buffer structure elements are aligned on addresses dividable by 4.

Flags uses the same values as SendFlags in CARMSG\_IOCAN\_Send\_CMD.

The Read\_Element is adjusted by the component that reads from the ring buffer and identifies the next element to read. The Write\_Element is adjusted by the component that writes to the ring buffer and identifies the next element to write.

When Read\_Element reaches the value of Write\_Element, the ring buffer is empty. When Write\_Element reaches the value of Read\_Element, the ring buffer is full.

Ring buffer elements use native machine byte alignment (endian).

**MessageFilter:**

Name	Type	Description
Flags	Bit field	Bit 0: Ignore frames Bit 1: Filter on RTR flag (RTR mask) Bit 2: RTR flag filter value (RTR code)
Identifier Mask	Bit field	masks identifier bits (0: ignore bit position)
Identifier Value	Bit field	Value of the identifier
Data filter	Structured block	List of max. 8 databytes to be filtered

**Data filter:**

Name	Type	Description
Databyte Mask	Bit field	masks databyte bits (0: ignore bit position)
Databyte Value	Bit field	acceptance filter code

**Filtering** is done using the following formula:

if ((Value BITWISEAND FilterMask) EQUAL (FilterValue BITWISEAND FilterMask))  
then value passes the filter. Otherwise the value does not pass the filter.  
Flags bit 0 inverts the action.

**CARMSG\_IOCAN\_LogicalConnect\_CON****Description**

Confirms the request for a logical connection.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	Physical connection identifier
LogConnID	Unsigned value	logical connection identifier

**CARMSG\_IOCAN\_LogicalConnect\_EVT****Description**

Indicates that a logical connection was established.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	Physical connection identifier
LogConnID	Unsigned value	logical connection identifier
ConnectionType	Unsigned value	(unused)
ConnectionParameters	Unsigned value	(unused)

**CARMSG\_IOCAN\_LogicalParams\_CMD****Description**

Requests the modification of a logical connections parameters.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
LogConnParams	Structured block	Parameters for requested logical connection

For a description of **LogConnParams** see CARMSG\_IOCAN\_LogicalConnect\_CMD

**CARMSG\_IOCAN\_LogicalParams\_CON****Description**

Confirms the request to modify a logical connections parameters.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOCAN\_LogicalStatus\_CMD

### Description

Requests status information of logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOCAN\_LogicalStatus\_CON

### Description

Confirms the request for status information of logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOCAN\_LogicalStatus\_EVT

### Description

Contains status information for a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
Status	Structured block	Status information block

Status:

Name	Type	Description
Status	Unsigned Value	1: Error Active (normal operation) 2: Error Passive 3: Error Bus Off 4: Buffer overrun
Rx ErrorCounter	Unsigned Value	Current value of rx error count
Tx ErrorCounter	Unsigned Value	Current value of tx error count
Error Code	Unsigned Value	Contains details about the occurred error

## CARMSG\_IOCAN\_LogicalDisconnect\_CMD

### Description

Requests termination of a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOCAN\_LogicalDisconnect\_CON

### Description

Confirms request for termination of a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOCAN\_LogicalDisconnect\_EVT

### Description

Indicates termination of a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DisconnectReason	Unsigned value	Reason for disconnect

## 2.6.4 CAN messages for data transfer

### CARMSG\_IOCAN\_Send\_CMD

#### Description

Requests to send the given data block(s).

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DataBlocks	Structured block	List of DataBlock elements to send

**DataBlock** contains the following elements

For LogConnType 1 (direct mailbox access with message based frame transfer):

Name	Type	Description
ID	unsigned value	CAN message ID
SendFlags	Bit field	(connection type specific) Bit 0: RTR Bit 1: Single Shot Transmission Bit 2: Return Confirmation
Data	Unstructured block	Data bytes

For LogConnType 2 (direct mailbox access with ring buffer based frame transfer):

DataBlocks is an empty struct. The frames to be transmitted must be written to the ring buffer before sending the message.

### CARMSG\_IOCAN\_Send\_CON

#### Description

Indicates the transmission of the provided data blocks if specified by SendFlags in CARMSG\_IOCAN\_Send\_CMD

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DataConfirmList	Structured block	List of DataConfirm elements

**DataConfirm** contains the following elements

For LogConnType 1 (direct mailbox access with message based frame transfer):

Name	Type	Description
ID	unsigned value	CAN message ID
XmitComplete	Timestamp	Frame transmission start timestamp, 64-bit value representing the number of 100-nanosecond intervals since January 1, 1601 (see Win32 FILETIME structure). The timestamp may not be perfectly synchronized with the system clock.

For LogConnType 2 (direct mailbox access with ring buffer based frame transfer):

DataConfirmList is an empty struct. The message indicates that the transmit ring buffer was empty. Transmit time stamps are written to the frames timestamp element in the ring buffer.



## CARMSG\_IOCAN\_Recv\_EVT

### Description

Indicates that one or more data blocks were received.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DataBlocks	Structured block	List of Datablock elements received

**DataBlock** contains the following elements

For LogConnType 1 (direct mailbox access with message based frame transfer):

Name	Type	Description
ID	Unsigned value	CAN message ID
RecvFlags	Bit field	(connection type specific) Bit 0: RTR
Data	Unstructured block	Data bytes
RecvComplete	Timestamp	Frame receive complete timestamp

For LogConnType 2 (direct mailbox access with ring buffer based frame transfer):

DataBlocks is an empty struct. The message indicates that the receive ring buffer contains frames and that either the receive timeout has expired or the number of received frames reached the specified receive limit.

## 2.7 KLINE messages

### 2.7.1 KLINE general messages

#### CARMSG\_IOKLINE\_QueryCaps\_CMD

##### Description

Queries the capabilities of the KLINE module.

##### Parameters:

none

#### CARMSG\_IOKLINE\_QueryCaps\_CON

##### Description

Displays the capabilities of the KLINE module.

##### Parameters:

Name	Type	Description
PhysConnCaps list	Structured block	List of Physical Connection Caps elements

**Physical Connection Caps** contains the following elements

Name	Type	Description
PhysConnectionType	Unsigned value	Type of connection 1: local bus access
PhysConnProperties	Structured block	Physical connection properties
LogConnCaps list	Structured block	List of Logical Connection Caps

**Physical Connection Properties** contains the following elements

For physical connection type 1 (local bus access):

PhysConnProperties is an empty struct

**Logical Connection Caps** contains the following elements

Name	Type	Description
LogConnectionType	Unsigned value	Type of connection For PhysConnType 1 (local bus access) - 1: KWP71
SupportedBusSpeeds	Bit field	Supported bus speeds Bit 1: baudrate auto detect Bit 2: 2400 baud Bit 3: 4800 baud Bit 4: 9600 baud Bit 5: 10400 baud
LogConnFlags	Bit field	Supported logical connection flags (see CARMSG_IOKLINE_LogicalConnect_CMD)
TransmissionFlags	Bit field	Supported message transmission flags (see CARMSG_IOKLINE_Send_CMD)

## 2.7.2 KLINE messages for physical connections

### CARMSG\_IOKLINE\_PhysicalConnect\_CMD

#### Description

Requests the setup of a physical connection.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
PhysConnType	Unsigned value	Type of connection 1: local bus access
PhysConnParameters	Structured block	Parameters for the connection

#### Physical connection parameters

for Connection Type 1 (raw mailbox access):

LogConnParams is an empty struct

### CARMSG\_IOKLINE\_PhysicalConnect\_CON

#### Description

Confirms the request for a physical connection.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier

### CARMSG\_IOKLINE\_PhysicalConnect\_EVT

#### Description

Indicates that a physical connection was established.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that was connected
StackParameters	Structured block	

#### StackParameters

For PhysConnType 1 (local bus access) StackParameters is an empty struct.

## CARMSG\_IOKLINE\_PhysicalDisconnect\_CMD

### Description

Requests termination of a physical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that shall be disconnected

## CARMSG\_IOKLINE\_PhysicalDisconnect\_CON

### Description

Confirms request for termination of a physical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that shall be disconnected

## CARMSG\_IOKLINE\_PhysicalDisconnect\_EVT

### Description

Indicates termination of a physical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	ID of connection that was disconnected
DisconnectReason	Unsigned value	1: Hardware triggered (timeout/disconnect) 2: User mode triggered 3: other
Disconnect Information	Unsigned value	

## 2.7.3 KLINE messages for logical connections

### CARMSG\_IOKLINE\_LogicalConnect\_CMD

#### Description

Requests the setup of a logical connection within a given physical connection.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	Physical connection identifier
LogConnID	Unsigned value	logical connection identifier
LogConnType	Unsigned value	Type of connection For physical connection type: 1 (local bus access): - 1: KWP71
LogConnParams	Structured block	Parameters for requested logical connection

**LogConnParams** contains the following elements  
for LogConnType 1 (KWP71):

Name	Type	Description
Bit rate	Unsigned value	Bus speed in kBit/s, 0 means auto detect
NodeID	Unsigned value	ID of Bus node to connect to

### CARMSG\_IOKLINE\_LogicalConnect\_CON

#### Description

Confirms the request for a logical connection.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	Physical connection identifier
LogConnID	Unsigned value	logical connection identifier

### CARMSG\_IOKLINE\_LogicalConnect\_EVT

#### Description

Indicates that a logical connection was established.

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	Physical connection identifier
LogConnID	Unsigned value	logical connection identifier
ConnectionType	Unsigned value	(unused)
ConnectionParameters	Unsigned value	(unused)

**CARMSG\_IOKLINE\_LogicalParams\_CMD****Description**

Requests the modification of a logical connections parameters.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
LogConnParams	Structured block	Parameters for requested logical connection

For a description of **LogConnParams** see CARMSG\_IOKLINE\_LogicalConnect\_CMD

**CARMSG\_IOKLINE\_LogicalParams\_CON****Description**

Confirms the request to modify a logical connections parameters.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOKLINE\_LogicalStatus\_CMD

### Description

Requests status information of logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOKLINE\_LogicalStatus\_CON

### Description

Confirms the request for status information of logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOKLINE\_LogicalStatus\_EVT

### Description

Contains status information for a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
Status	Structured block	Status information block

Status:

Name	Type	Description
Status	Unsigned Value	1: Error Active (normal operation) 2: Error Passive 3: Error Bus Off 4: Buffer overrun
Rx ErrorCounter	Unsigned Value	Current value of rx error count
Tx ErrorCounter	Unsigned Value	Current value of tx error count
Error Code	Unsigned Value	Contains details about the occurred error

## CARMSG\_IOKLINE\_LogicalDisconnect\_CMD

### Description

Requests termination of a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOKLINE\_LogicalDisconnect\_CON

### Description

Confirms request for termination of a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier

## CARMSG\_IOKLINE\_LogicalDisconnect\_EVT

### Description

Indicates termination of a logical connection.

### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DisconnectReason	Unsigned value	Reason for disconnect



## 2.7.4 KLINE messages for data transfer

### CARMSG\_IOKLINE\_Send\_CMD

#### Description

Requests to send the given data block(s).

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DataBlocks	Structured block	List of DataBlock elements to send

**DataBlock** contains the following elements

For LogConnType 1 (KWP71):

Name	Type	Description
SendFlags	Bit field	(connection type specific) Bit 0: Return Confirmation
Data	Unstructured block	Data bytes

### CARMSG\_IOKLINE\_Send\_CON

#### Description

Indicates the transmission of the provided data blocks if specified by SendFlags in CARMSG\_IOKLINE\_Send\_CMD

#### Parameters:

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DataConfirmList	Structured block	List of DataConfirm elements

**DataConfirm** contains the following elements

For LogConnType 1 (KWP71):

Name	Type	Description
XmitComplete	Timestamp	Frame transmission start timestamp, 64-bit value representing the number of 100-nanosecond intervals since January 1, 1601 (see Win32 FILETIME structure). The timestamp may not be perfectly synchronized with the system clock.

**CARMSG\_IOKLINE\_Recv\_EVT****Description**

Indicates that one or more data blocks were received.

**Parameters:**

Name	Type	Description
PhysConnID	Unsigned value	physical connection identifier
LogConnID	Unsigned value	logical connection identifier
DataBlocks	Structured block	List of Datablock elements received

**DataBlock** contains the following elements

For LogConnType 1 (KWP71):

Name	Type	Description
RecvFlags	Bit field	(connection type specific) Bit 0: (unused)
Data	Unstructured block	Data bytes
RecvComplete	Timestamp	Frame receive complete timestamp

### 3 Message overview

Specified	Supported	Name	Notes
✓		CARMSG_Unit_EnumDevices_CMD	
✓		CARMSG_Unit_EnumDevices_CON	
✓		CARMSG_Unit_EnumDevices_EVT	
✓		CARMSG_Unit_EnumConnections_CMD	
✓		CARMSG_Unit_EnumConnections_CON	
✓		CARMSG_Unit_EnumConnections_EVT	
		CARMSG_Flash_Read_CMD	
		CARMSG_Flash_Read_CON	
✓		CARMSG_Flash_Write_CMD	
✓		CARMSG_Flash_Write_CON	
✓		CARMSG_Ignition_State_CMD	
✓		CARMSG_Ignition_State_CON	
✓		CARMSG_Ignition_State_EVT	
✓		CARMSG_Battery_Level_CMD	
✓		CARMSG_Battery_Level_CON	
✓		CARMSG_Battery_Level_EVT	
✓		CARMSG_PowerSupply_SwitchOff_CMD	
✓		CARMSG_PowerSupply_SwitchOff_CON	
✓	✓	CARMSG_PowerSupply_SwitchOff_EVT	
✓	✓	CARMSG_Key_Up_EVT	
✓	✓	CARMSG_Key_Down_EVT	
✓	✓	CARMSG_Dial_TurnLeft_EVT	
✓	✓	CARMSG_Dial_TurnRight_EVT	
✓	✓	CARMSG_AudioSelector_QueryCaps_CMD	
✓	✓	CARMSG_AudioSelector_QueryCaps_CON	
✓	✓	CARMSG_AudioSelector_Position_CMD	
✓	✓	CARMSG_AudioSelector_Position_CON	
✓	✓	CARMSG_AudioVolume_QueryCaps_CMD	
✓	✓	CARMSG_AudioVolume_QueryCaps_CON	
✓	✓	CARMSG_AudioVolume_Position_CMD	
✓	✓	CARMSG_AudioVolume_Position_CON	
✓	✓	CARMSG_AudioBalance_QueryCaps_CMD	
✓	✓	CARMSG_AudioBalance_QueryCaps_CON	
✓	✓	CARMSG_AudioBalance_Position_CMD	
✓	✓	CARMSG_AudioBalance_Position_CON	
✓	✓	CARMSG_AudioFader_QueryCaps_CMD	
✓	✓	CARMSG_AudioFader_QueryCaps_CON	
✓	✓	CARMSG_AudioFader_Position_CMD	
✓	✓	CARMSG_AudioFader_Position_CON	
✓	✓	CARMSG_AudioFilterBass_QueryCaps_CMD	
✓	✓	CARMSG_AudioFilterBass_QueryCaps_CON	
✓	✓	CARMSG_AudioFilterBass_Position_CMD	
✓	✓	CARMSG_AudioFilterBass_Position_CON	
✓	✓	CARMSG_AudioFilterTreble_QueryCaps_CMD	
✓	✓	CARMSG_AudioFilterTreble_QueryCaps_CON	
✓	✓	CARMSG_AudioFilterTreble_Position_CMD	
✓	✓	CARMSG_AudioFilterTreble_Position_CON	
✓	✓	CARMSG_AudioFilterLoudness_QueryCaps_CMD	
✓	✓	CARMSG_AudioFilterLoudness_QueryCaps_CON	
✓	✓	CARMSG_AudioFilterLoudness_Position_CMD	
✓	✓	CARMSG_AudioFilterLoudness_Position_CON	
		CARMSG_AudioFilterParametric_QueryCaps_CMD	
		CARMSG_AudioFilterParametric_QueryCaps_CON	
		CARMSG_AudioFilterParametric_Position_CMD	
		CARMSG_AudioFilterParametric_Position_CON	
		CARMSG_AudioFilterParametric_Frequency_CMD	
		CARMSG_AudioFilterParametric_Frequency_CON	
		CARMSG_AudioFilterParametric_QualityFactor_CMD	
		CARMSG_AudioFilterParametric_QualityFactor_CON	
✓	✓	CARMSG_AudioTuner_QueryCaps_CMD	
✓	✓	CARMSG_AudioTuner_QueryCaps_CON	
✓	✓	CARMSG_AudioTuner_SelectArea_CMD	
✓	✓	CARMSG_AudioTuner_SelectArea_CON	
✓	✓	CARMSG_AudioTuner_SelectBand_CMD	
✓	✓	CARMSG_AudioTuner_SelectBand_CON	
✓	✓	CARMSG_AudioTuner_SetFrequency_CMD	
✓	✓	CARMSG_AudioTuner_SetFrequency_CON	
✓	✓	CARMSG_AudioTuner_ScanBand_CMD	
✓	✓	CARMSG_AudioTuner_ScanBand_EVT	
✓	✓	CARMSG_AudioTuner_ScanBand_CON	
		CARMSG_IOCAN_QueryCaps_CMD	
		CARMSG_IOCAN_QueryCaps_CON	
✓	✓	CARMSG_IOCAN_PhysicalConnect_CMD	
✓	✓	CARMSG_IOCAN_PhysicalConnect_CON	
✓	✓	CARMSG_IOCAN_PhysicalConnect_EVT	
✓	✓	CARMSG_IOCAN_PhysicalDisconnect_CMD	
✓	✓	CARMSG_IOCAN_PhysicalDisconnect_CON	
✓	✓	CARMSG_IOCAN_PhysicalDisconnect_EVT	
✓	✓	CARMSG_IOCAN_LogicalConnect_CMD	
✓	✓	CARMSG_IOCAN_LogicalConnect_CON	
✓	✓	CARMSG_IOCAN_LogicalConnect_EVT	
		CARMSG_IOCAN_LogicalParams_REQ	
		CARMSG_IOCAN_LogicalParams_CON	
		CARMSG_IOCAN_LogicalStatus_CMD	
		CARMSG_IOCAN_LogicalStatus_CON	
		CARMSG_IOCAN_LogicalStatus_EVT	
✓	✓	CARMSG_IOCAN_LogicalDisconnect_CMD	



Specified	Supported	Name	Notes
√	√	CARMSG_IOCAN_LogicalDisconnect_CON	
√	√	CARMSG_IOCAN_LogicalDisconnect_EVT	
√	√	CARMSG_IOCAN_Send_REQ	
√		CARMSG_IOCAN_Send_CON	
√	√	CARMSG_IOCAN_Recv_EVT	

## 4 Revision History

Revision	Changes
0.3	Added MSG_Unit_RegisterClient_CMD messages for power management activation
0.4	Added MSG_IOKLINE_XXX messages