technica
engineering

TECMP

Technically Enhanced Capture Modules Protocol

USER MANUAL

November 2020

Document version: 1.5

CONTENT

GENERAL INFORMATION 4
1.1 Introduction 4
1.2 Purpose of the document 4
1.3 Available resources 4
1.4 TECMP functions 5
1.4.1 Functions and Scope 5
1.5 Abbreviations 5
TECMP Feature Specification 7
2.1 Dedicated EtherType 7
2.2 Data format 8
2.2.1 TECMP Header 9
2.2.2 Specification of Data Flags and Source Data 11
2.2.2.1 TECMP Ethernet Data (Data Type = Ethernet II) 12
2.2.2.2 TECMP LIN Data (Data Type = LIN) 12
2.2.2.3 TECMP CAN Data (Data Type = CAN Data) 13
2.2.2.4 TECMP CAN-FD Data (Data Type = CAN-FD Data) 13
2.2.2.5 TECMP Flexray Data (Data Type = Flexray Data) 14
2.2.2.6 TECMP UART / RS232 ASCII (Data Type = UART / RS232 ASCII) 15
2.2.2.7 TECMP Analog (Data Type = Analog) 16
2.2.3 Packaging IVN messages 17
2.3 Status message 17
2.3.1 Status Configuration message 18
2.3.2 Status Bus/IVN message 19
2.3.3 Status Capture Module message 22
2.4 Control message 23
2.5 Time Synchronization 24
2.6 TECMP Header fields summary 24
2.6.1 Message Type 24
2.6.2 Data Type 24
2.6.3 CM ID 25
2.6.4 Control Message ID 25
2.6.5 CM Type 26

3 Contact 26
4 Changelog 27

GENERAL INFORMATION

1.1 Introduction

The following protocol description targets the Capture Module (CM) from Technica Engineering. The CM is currently available in five different optimized products for which dedicated user manuals are also available:

- CM Ethernet Combo
- CM CAN Combo
- CM LIN Combo
- CM 100 High
- CM 1000 High

1.2 Purpose of the document

This document describes the functionality of the Technica Engineering Capture Module interface. The functions are accessible via the Technically Enhanced Capture Modules Protocol (TECMP) published by Technica Engineering and described in this document.

The interface between the CM and computer/logger is established via standard Ethernet solutions. In addition, some CMs are also equipped with IEEE 100Base-T1 interfaces for this purpose. The time synchronization is done via IEEE 802.1AS. Cascading CMs is possible using selected models only. For additional details, please consult the respective user manuals.

1.3 Available resources

Technica Engineering provides dissector plugins for Wireshark in order to display the data in the Wireshark Toolchain. The newest version of Wireshark (3.4.0) provides a native TECMP dissection. For prior Wireshark versions without native TECMP dissection, the plugin must be added.
An overview of the TECMP packet dissection inside Wireshark is provided in the next illustration.

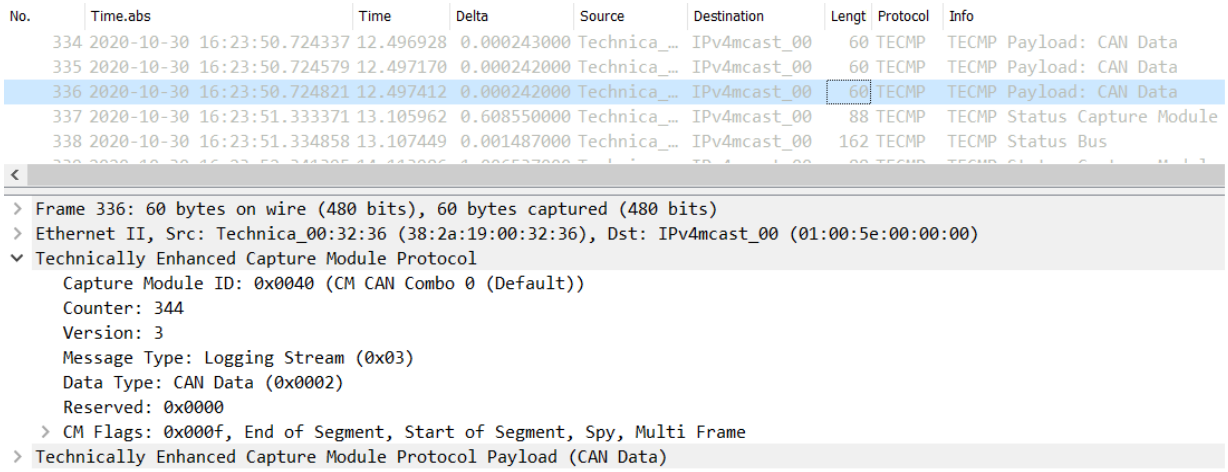

Figure 1: Example of TECMP packet dissection inside Wireshark (CAN)

1.4 TECMP functions

The following TECMP functionalities are covered by this specification:

- Data transport from the CM to the Ethernet Capture Interfaces (Connected to a network, computer or logger)
- Remote monitoring and control
- Time synchronization of the CMs

1.4.1 Functions and Scope

The TECMP protocol is defined by the following keypoints:

- A specific data format
- A protocol for data transmission
- A protocol for the transmission of status and control messages
- A time synchronization technique

An overview of a system setup comprising multiple CMs and a logger module is represented in Figure 2.

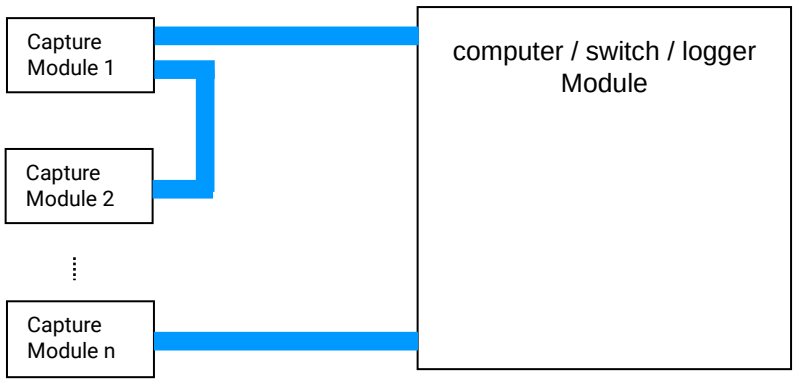

Figure 2: Overview of a system setup

1.5 Abbreviations

Abbreviation	Definition / Explanation
CM	Capture Module
TECMP	Technically Enhanced Capture Modules Protocol
CAN	Controller Area Network
EoS	End of Segment
SoS	Start of Segment
Eth	Ethernet
FR	FlexRay

Abbreviation	Definition / Explanation
HW	Hardware
LIN	Local Interconnect Network
PTP	Precision Time Protocol
Control Message	A method to trigger events. Events can be triggered by a CM or can be received by a CM. Some Control Message can be configured
Status Message	A dedicated message sent by the CM in direction of the Ethernet Capture interface (network, computer, logger)
IVN	In-Vehicle Network

Table 1: Abbreviations

TECMP Feature Specification

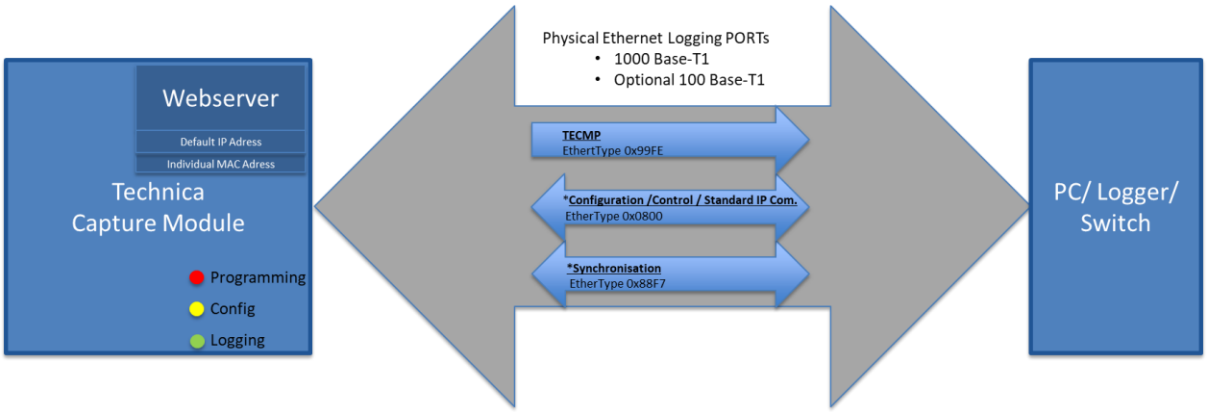

Figure 3: Logical/physical Interfaces

Figure 3 shows an overview of the physical and logical interfaces between the CM and a computer or another Ethernet network components like a datalogger or a switch. For details of the physical interfaces consult the user manual of the specific CM.

CMs may provide one or more additional interfaces which can be used to connect CMs with each other in a cascading structure. Thanks to this feature, the functionality of a CM can be extended with another CM.

The main task of the CMs is recording the data of in-vehicle communication systems via an own logging network based on Ethernet. The additional features such as time synchronization between CM or the ability to connect multiple CMs together are available to support the user in this task. In this way, a user can set up specific logging solutions fitting his needs. This is even possible, in selected scenarios, to use a regular computer instead of a data logger.

2.1 Dedicated EtherType

The CMs support the official TECMP Ethertype (0x99FE). This EtherType is used to transmit the IVN-data collected by the CM to an endpoint connected via Ethernet. The device can be a computer, an Ethernet switch for network connection, or a data logger module. These devices are called “sinks” later in this document. This document defines the data format in which the collected data are transmitted to the defined sinks. Additional metadata like timestamp and other information related to the source data are also covered in this document.

The main use case of CMs is to capture all IVN data and convert it to an Ethernet based packet format. This enables the user to have a kind of logging and tracing network which connects a number of CMs and computers or logging devices via a standard ethernet network. This logging network is scalable and is based on IT standards.

No additional transport protocol is required.

2.2 Data format

Each message recorded on a bus and to be transported on the logging path is provided with its own TECMP header.

The logging data is encapsulated in a standard transport layer frame. This requires the embedding in a standard Ethernet frame according to IEEE 802.3.

Data of non-Ethernet communication systems as well as of Ethernet-based systems are transported in Ethernet frames and identified using TECMP. Ethernet frames are transported without changes to them or their headers. All messages are left intact. This ensures that analysis tools can easily access the unchanged data, if support of TECMP was added or data was converted.

The data format is assigned as follows:

PRE/ SFD	DST MAC	SRC MAC	Outer 802.1Q Header (optional)	Inner 802.1Q Header (optional)	Ether Typ 0x99FE	TECMP Header	Source Data	FCS
8 bytes	6 bytes	6 bytes	4 bytes	4 bytes	2 bytes	28 bytes	Up to n bytes (n <= 1500 bytes)	4 byte

Figure 4: Data Format of a TECMP message

PRE / SFD	Preamble and Start Frame Delimiter according to IEEE Standard 802.3.
DST MAC	The destination MAC address must be assigned the multicast address: 01: 00: 5E: 00: 00: 00
SRC MAC	The source MAC address carries the MAC address of the CM. The MAC addresses are unique for each delivered hardware.
VLAN 802.1Q:	optional VLAN fields are provided for possible future extensions.
• Outer 802.1Q header (optional)	Tagging according to IEEE 802.1Q must be supported by all components. The value of the tag must be configurable in the CM.
• Inner 802.1Q header (optional)	Tagging, according to IEEE 802.1Q, must be supported by all components. The value of the tag must be configurable in the CM.
EtherType	0x99FE: TECMP - Identifies this packet as TECMP packet
TECMP header	see 2.2.1
Source Data	see 2.2.2
FSC - CRC	CRC checksum, calculated according to IEEE Standard 802.3

Table 2: TECMP message fields description

2.2.1 TECMP Header

NOTE: In general, all unspecified number types are UINT and encoded in big-endian. The header structure is identical for all messages that are logged via the CM.

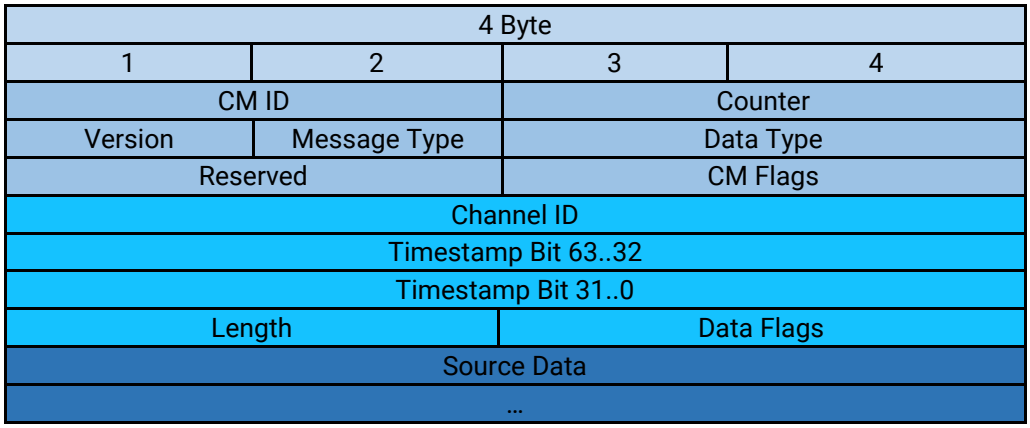

Figure 5: Overview of a vehicle data frame in a TECMP frame

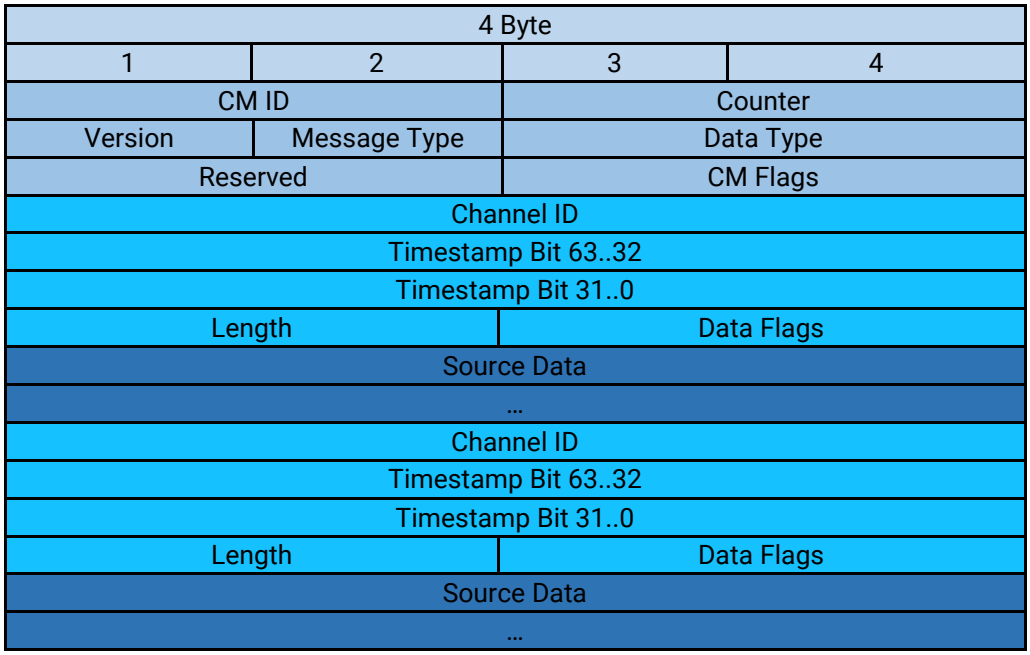

Figure 6: Overview of multiple vehicle data frames packed in a TECMP frame

***Note**: Packing multiple vehicle frames is only done for the same Data Type.*

Field Name	Data Type	Description
CM ID	UINT16	Each CM can be configured with a unique ID. This ID can be used to identify different Modules if they are used in the same network (see Table 23).
Counter	UINT16	Counter is incremented for each TECMP frame sent from a CM. The counter starts again from 0x0000 when reaching the maximum value 0xFFFF. If a CM has several physical interfaces to the sink (Ethernet links), a separate counter is implemented on each interface. The CM ID field is the same for all interfaces in this case.

Field Name	Data Type	Description
Version	UINT8	Used version of the TECMP protocol; This UINT8 uniquely identifies the version.
Message Type	UINT8	Description of the payload function, whether status, configuration or bus data (see Table 21)
Data Type	UINT16	Description of the Type of Source Data (e.g., CAN, LIN, Flexray, Ethernet, Undefined, Voltage, Current, …). These types are preconfigured and fixed (see Table 22)
Reserved	UINT16	Reserved, pre-filled with 0x0000
CM Flags	UINT16	Provides additional information from the CM: 0b0000 0000 0000 0001 EoS (End of Segment; used to signal a last segment of segmented information.) 0b0000 0000 0000 0010 SoS (Start of Segment: used to signal a first segment of segmented information Note: For single segment transmitted in a TECMP frame, the CM Flags EoS and SoS are always set to '1'. 0b0000 0000 0000 0100 SPY (SPY flag can be used to trigger additional actions on the sink.) Note: For messages from recording on-board communication, the value for the flag is always '1'.) 0b0000 0000 0000 1000 Multi-Frame (a CM that can send multiple Channel IDs in a TECMP frame must set this flag) 0b0000 0000 0001 0000 - reserved 0b0000 0000 0010 0000 - reserved 0b0000 0000 0100 0000 - reserved 0b0000 0000 1000 0000 - reserved 0b0000 0001 0000 0000 - reserved 0b0000 0010 0000 0000 - reserved 0b0000 0100 0000 0000 - reserved 0b0000 1000 0000 0000 - reserved 0b0001 0000 0000 0000 - reserved 0b0010 0000 0000 0000 - reserved 0b0100 0000 0000 0000 - reserved 0b1000 0000 0000 0000 CM overflow (Module has lost data)
Channel ID	UINT32	ID that uniquely identifies the log data / bus / link on the IVN.
Timestamp	UINT64	Timestamp is the time in past 1ns since Thursday, January 1, 1970, 00:00 UTC. The resolution is 1ns. Bit 63: • 1: time is asynchronous, synchronization with the timing master failed • 0: time is synchronous, synchronization with the timing master successful Bit 62: Reserved The timestamp value 0x0000000000000000 is not allowed.
Length	UINT16	Length of Source Data Note: bytes of the length and Data Flags fields are ignored.
Data Flags	16 Bitfield	Provides additional information from the source network: The Data Flags refer to the messages transmitted on the IVN (100Base-T1, 1000Base-T1, FlexRay, CAN, CAN-FD, LIN). 0b0000 0000 0000 0001 IVN specific please see in the corr. IVN section 0b0000 0000 0000 0010 IVN specific please see in the corr. IVN section

Field Name	Data Type	Description
		0b0000 0000 0000 0100 IVN specific please see in the corr. IVN section 0b0000 0000 0000 1000 IVN specific please see in the corr. IVN section 0b0000 0000 0001 0000 IVN specific please see in the corr. IVN section 0b0000 0000 0010 0000 IVN specific please see in the corr. IVN section 0b0000 0000 0100 0000 IVN specific please see in the corr. IVN section 0b0000 0000 1000 0000 IVN specific please see in the corr. IVN section 0b0000 0001 0000 0000 TBD 0b0000 0010 0000 0000 TBD 0b0000 0100 0000 0000 TBD 0b0000 1000 0000 0000 TBD 0b0001 0000 0000 0000 TBD 0b0010 0000 0000 0000 CRC error (CAN, Flexray Frame CRC (trailer Segment: 24bits - 15), Ethernet). This bit indicates that a CRC error was found during logging. 0b0100 0000 0000 0000 TX (CM as sender). This bit indicates a request to the CM to transmit the current message on the specified interface. This bit will not be set in Status messages nor in logged messages. 0b1000 0000 0000 0000 Overflow (loss of previous message (s) per IVN)
Source Data		Data transferred from the vehicle IVN (source data).

Table 3: Structure and description of the TECMP Header

Note Counter:

- One or more CM IDs can be accepted for each TECMP-HW interface (logger or computer or the data sink). Each CM ID must have its own counter
- If a CM / adapter has multiple HW interfaces to the data sink, each HW interface must have its own counter. At the sink, a CM ID can arrive several times, whereby the counters are independent.

Note timestamp bit 63:

- If a time slave does not receive a 1AS packet from the master for more than 250 ms, this bit must be set to indicate that synchronization was lost.

For efficient transmission, multiple recorded messages can be packaged in a transport layer frame (see 2.2.3).

2.2.2 Specification of Data Flags and Source Data

In order to simplify the data processing and to transmit data as efficiently as possible, data are transferred in the same coding as in the “original” vehicle's internal buses (IVN). The recorded data are marked with a corresponding timestamp and transferred to the other sinks (e.g. data logger) or to the logging network.The coding details of the data are specified in the next chapters.

In order to use the logging network effectively, smaller data packets can be aggregated into a larger logging packet. The maximum length is limited to 1526 bytes.

2.2.2.1 TECMP Ethernet Data (Data Type = Ethernet II)

Ethernet packets of the size of standard frames (up to 1500 bytes) are completely transferred to the Ethernet payload of the newly generated packet, from MAC Destination up to and including FCS.

Example: A recorded Ethernet Message on an IVN Link.
"Source data" comes from a transmission of an Automotive Ethernet Packet (Ethernet II):
(The payload is an standard Ethernet frame)

Source Data						
DST MAC	SRC MAC	Outer 802.1Q Header (optional)	Inner 802.1Q Header (optional)	Ether Type	Payload	FCS
6 byte	6 byte	4 byte	4 byte	2 byte	<= 1500 byte	4 byte

Figure 7: Structure of an Ethernet Packet (Ethernet II)

In this case the EtherType of the Ethernet Header is the entry of the message to be recorded from the source network.

2.2.2.2 TECMP LIN Data (Data Type = LIN)

Field Name	Data Type	Description
LIN ID	UINT8	Received LIN ID
Payload Length	UINT8	Length of the LIN payload (1,2,4 or 8 bytes)
Payload	UINT8 [0-8]	Bytes of payload (up to 8 bytes), depending on the Payload Length.
Checksum	UINT8	Received LIN checksum

Data Flags (Data Type = LIN)

Information regarding LIN error condition:

15..3	2	1	0
Reserved	No Slave Response	Parity Error	Collision Error

- Collision Error: A '1' value in this bit indicates if there was a collision during the reception of the LIN frame. In this case, the received payload is not valid.
- Parity Error: A '1' value in this bit indicates if there is a parity error in the Protected ID.
- No Slave Response: A '1' value in this bit indicates that there was no response from a slave on a Request from a master

2.2.2.3 TECMP CAN Data (Data Type = CAN Data)

Field Name	Data Type	Description
CAN ID	UINT32	Bit 31: • 0: 11 bit ID; • 1: 29 bit ID; Bit 28-0 received CAN ID.
Payload Length	UINT8	Length of the CAN payload (in bytes).
Payload	UINT8 [0-8]	Bytes of the payload (up to 8 bytes), depending on the Payload Length.

Data Flags (Data Type = CAN Data)

Information regarding CAN reception condition:

15..4	3	2	1	0
Reserved	ERR	IDE	RTR	ACK

- ACK: A '1' value in this bit indicates whether there was an acknowledgement during the reception of the CAN frame.
- RTR: A '1' value in this bit indicates if the CAN frame is a remote frame. In this case, the LENGTH field will be 0x00, and there is no any payload.
- IDE: A '1' value in this bit indicates extended CAN-Identifier (29 bits).
- ERR: A '1' value in this bit indicates a Bus Error – "Error Frame"

REMARK: ERR: '1' value in this bit indicates an Error on the bus. (6 x rec or 6 dom) .
In this case no payload is transported and the Payload Length will be set to 0.

2.2.2.4 TECMP CAN-FD Data (Data Type = CAN-FD Data)

Field Name	Data Type	Description
CAN ID	UINT32	Bit 31: • 0: 11 Bit ID; • 1: 29 Bit ID; Bit 28-0 received CAN ID
Payload Length	UINT8	Length of the CAN payload (in bytes) (>8 .. 12,16,20,24,32,48,64 Byte)
Payload	UINT8 [0-64]	Bytes of the payload (up to 64 bytes), depending on the Payload Length.

Data Flags (Data Type = CAN-FD Data)

Information regarding CAN reception Condition:

15..5	4	3	2	1	0
Reserved	BRS	ERR	IDE	ESI	ACK

- ACK: A '1' value in this bit indicates whether there was an acknowledgement during the reception of the CAN frame.
- ESI: A '1' value in this bit indicates that the error node is active
- IDE: A '1' value in this bit indicates that the CAN frame ID is extended format (29 bits).
- ERR: A '1' value in this bit indicates a Bus Error – "Error Frame"

- BRS: A '1' value in this bit indicates that the alternate bitrate is used for the data phase of the CAN-FD frame.

REMARK: ERR: '1' value in this bit indicates an Error on the bus. (6 x rec or 6 x dom) .
In this case no payload is transported and the Payload Length will be set to 0.

2.2.2.5 TECMP Flexray Data (Data Type = Flexray Data)

Field Name	Data Type	Description
Cycle	UINT8	Cycle in which the frame was received.
Frame ID	UINT16	Received frame ID if Message Type = "Logging Stream" Transmitted frame ID if Message Type = "Replay data"
Payload Length	UINT8	Length of Flexray Frame Payload for a specific ID
Payload	UINT8 [0-254]	Bytes of the payload (up to 254 bytes), depending on the Payload Length.

Data Flags (Data Type = FlexRay Data)

Information regarding FlexRay reception condition:

15..6	5	4	3	2	1	0
Reserved	CAS	PPI	WUS	Sync	SF	NF

- Null Frame (NF):
  - '1': it is a Null Frame,
  - '0': it is not a Null Frame
- Startup Frame (SF)
  - '1': it is a Startup Frame,
  - '0': it is not a Startup Frame
- Sync Frame (Sync):
  - '1': it is a Sync Frame,
  - '0': it is not a Sync Frame
- Wakeup Symbol (WUS):
  - '1': it is a WUS, no data payload is carried, the payload length is set to 0,
  - '0': it is not a WUS
- Payload Preamble Indicator (PPI):
  - '1': the payload of the frame contains either a Network Management Vector (static segment) or a message ID (dynamic segment),
  - '0': the payload of the frame does not contain neither a Network Management Vector (static segment) nor a message ID (dynamic segment)
- Collision Avoidance Symbol (CAS):
  - '1': it is a CAS,
  - '0': it is not a CAS

2.2.2.6 TECMP UART / RS232 ASCII (Data Type = UART / RS232 ASCII)

Several data symbols of an RS232 interface are collected in a TECMP Data Block. In this case, only the user data (7 - 8 bit length per data symbol) is transmitted in one byte (possibly filled with '0 bits').
A maximum of 1000 RS232 data symbols may be transmitted in one TECMP Data Block.
The real timestamping for the timestamp (field Timestamp in the TECMP Header) takes place when the first symbol arrives at the RS232 line. Configuration of the interface (bitrate, symbol length, parity bit, ...) takes place in the CM. Settings relevant to the recording are stored in the Data Flags.

Field Name	Data Type	Description
Payload	UINT8	RS-232 message (1 byte per RS-232 symbol)
...	..	...
Payload n	UINT8	

Data Flags (Data Type = UART/RS232 ASCII**)**

15..4	3	2	1	0
reserved	DL3	DL2	DL1	ParityError

DL3 to DL1:

Bit 3 to bit 1 indicates the symbol length on RS232:

RS232 with 7 bit: 010
RS232 with 8 bit: 011

If a parity error occurs on the RS232 interface, the bit Parity Error is set, the message is written completely and then the TECMP packet is sent.
If the parity error bit is set, the receiver knows that the last RS232 message in the TECMP packet had a parity error.

2.2.2.7 TECMP Analog (Data Type = Analog)

Several measured values of an analog interface are collected in a TECMP Data Block. A maximum of 700 measured values may be transmitted in one TECMP Data Block.

The time recording for all measured values (field Timestamp in the TECMP Header) takes place when the first analog value is acquired. Configuration of the interface (voltage, current, sample time, ...) takes place in the sample. Settings relevant to the recording are stored in the Data Flags.

Field Name	Data Type	Description
Sample 0	UINT16	Contains the first value of the recording
...	UINT16	...
Sample N	UINT16	Contains the nth value of the measurement transmitted in a TECMP frame.

Data Flags (Data Type = Analog)

Information regarding analog threshold: this means that at least one sample has exceeded or fallen below the reference points (setting required in the sample).

15	14...11	10,9	8,7	6 ...4	3, 2	1,0
Reserved	Sample Time	Reserved	Factor	Reserved	Unit	Threshold

Threshold
Bit 0: '1' means: set upper threshold exceeded
Bit 1: '1' means: set lower threshold below

Bit 2, 3 Unit
00 '1' V

Bit 7...8 Factor
00 0,1
01 0,01
10 0,001
11 0,0001

Bit 11...14 Sample Time
0000 0 (Configuration sample as comparator)
0001 2500ms
0010 1000ms
0011 500ms
0100 250ms
0101 100ms
0110 50ms
0111 25ms
1000 10ms
1001 5ms
1010 2,5ms
1011 1,0ms

2.2.3 Packaging IVN messages

As explained in the beginning of this document, it is very useful to transfer at once several IVN messages packed in one frame to the attached sinks. However, not every message has its own complete TECMP Header here. The first 12 bytes of the TECMP Header are sent only once at the beginning of the packed IVN messages. The remaining header is appended to each message logged.

Format (Example with a single CAN-message):

Figure 8: Packaging of a single CAN message

Format (Example with multiple CAN-messages):

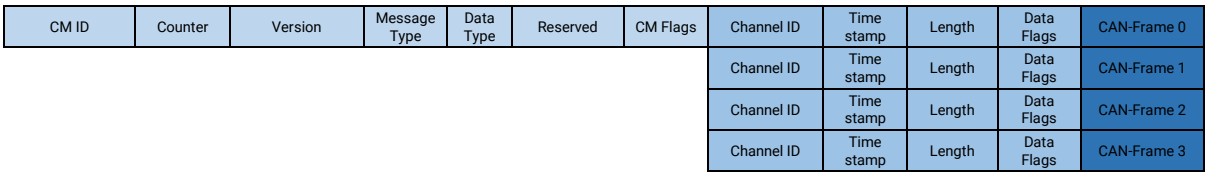

Figure 9: Packaging of multiple CAN messages

If, despite the packaging of several messages, the minimum size of a 64-byte Ethernet frame cannot be reached, the CM inserts appropriate zero padding at the end of the message.

2.3 Status message

Three different Status messages are currently supported by CMs:

- Status Configuration
- Status Bus/IVN
- Status CM

The Status messages use the following TECMP Header.

Figure 10: Status messages TECMP overview

The maximal length of Status messages is 1400 Bytes. If the payload to be transferred does not fit in a single Status message, the payload will be segmented and transferred in multiple and successive Status messages of the same type, each carrying one segment.

2.3.1 Status Configuration message

Status Configuration messages are sent cyclically every 60 seconds after the logging link has been established. The payload is depicted in the next illustration.

4 Byte			
0	1	2	3
Vendor ID	CM Version	CM Type	Reserved
Vendor Data Length (VDL)		CM ID	
Serial Number			
Vendor Data			

Table 4: Status Configuration message payload

Field Name	Length	Description
Vendor ID	1 byte	Vendor ID - Technica Engineering: 0x0C
CM Version	1 byte	Versioning of the CM
CM Type	1 byte	Type of CM (e.g.: CAN,FR,Ethernet,LIN) , see Table 25
Reserved	1 byte	For future extensions
Vendor Data Length	2 byte	Length of the “Vendor Data” field (in Bytes)
CM ID	2 byte	Configured Capture Module ID
Serial Number	4 byte	Serial number of the CM
Vendor Data	n byte	Technica Engineering specific CM configuration

Table 5: Status Configuration message message fields description

Any Capture Module HW variant can be easily identified based on the notified Serial Number and Capture Module version.

CM ID	Counter	Version	Message Type	Data Type	Reserved	CM Flags	Channel ID	Time stamp	Length	Data Flags
Vendor ID	CM Version	CM Type	Reserved	Vendor Data Length	CM ID	Serial Number	Vendor Data			

Table 6: Status Configuration message overview

CM Version:
Version of the Capture Module. Used to distinguish different Capture Module variants / revisions.

Vendor ID:

Technica Engineering	0x0C

CMType:
See Table 25.

Capture Module Serial Number:
This field is 4 bytes long. In case that the correct serial number for the device is not saved in the memory, the last 4 bytes of the MAC address will be shown as serial number

Vendor Data: Multiple configuration parameters are exchanged with the CM in the form of a serialized JSON configuration file. The format of the Vendor Data field is depicted and described in the next two illustrations.

4 Byte			
0	1	2	3
Version	Reserved	Config Message ID	
Total Length (Bytes)			
Total number of segments		Segment number	
Length of segment (Bytes)		Segment Data	...
...			

Table 7: Vendor Data overview

Field Name	Data Type	Description
Version	UINT8	Version number of the Status Configuration message vendor data format
Reserved	UINT8	Reserved field: 0x00
Config Message ID	UINT16	Identifier incremented by one at every new cyclic transmission
Total Length	UINT32	Sum of the lengths of the individual segments
Total number of segments	UINT16	Number of segments onto which the information is distributed. As many number of Status Configuration messages as segments will be generated per cyclic transmission
Segment number	UINT16	Identification of the present segment: between 0 and the value of the field “Total number of segments”-1
Length of segment	UINT16	Length (x) of the following “Segment Data” field in Bytes
Segment Data	x * UINT8	Serialized configuration

Table 8: Status Configuration message Vendor Data fields description

2.3.2 Status Bus/IVN message

Status Bus/IVN messages are sent cyclically every second after the logging link has been established. CMs with multiple interfaces send the status of all connected IVN’s. The payload is depicted in the next illustration.

4 Byte			
0	1	2	3
Vendor ID	CM Version	CM Type	Reserved
Vendor Data Length (VDL)		CM ID	
Serial Number			
Bus Data			

Table 9: Status Bus/IVN message payload overview

4 Byte
Bus Data Entry 1
Bus Data Entry 2 (optional)
Bus Data Entry 3 (optional
Bus Data Entry 4 (optional)
Optional further Entries ...

Table 10: Bus Data

4 Byte
Channel ID
Messages Total
Errors Total
Vendor Data

Table 11: Bus Data Entry

Field Name	Length	Description
Vendor ID	1 byte	Vendor ID - Technica Engineering: 0x0C
CM Version	1 byte	Versioning of the CM
CM Type	1 byte	Type of CM (e.g.: CAN,FR,Ethernet,LIN)
Reserved	1 byte	For future extensions
Vendor Data Length	2 byte	Length of the "Vendor Data" field per Bus (in Bytes)
CM ID	2 byte	Configured Capture Module ID
Serial Number	4 byte	Serial number of the CM
Channel ID	4 byte	ID that uniquely identifies the log data / bus / link of the IVN on which the Messages/frames were received or sent
Messages Total	4 byte	Messages received on the bus/IVN (Channel ID) since the startup of the CM
Errors Total	4 byte	Erroneous messages received on the bus/IVN (Channel ID) since the start of the Capture Module
Vendor Data	n byte	See description below

Table 12: Description of the Status Bus/IVN message payload fields

For one bus/IVN interface:

CM ID	COUNTER	VERSION	Message Type	DATA TYPE	RESERVED	CM Flags	Channel ID	TIME STAMP	LENGTH	Data Flags
VENDOR ID	CM VERSION	CM TYPE	RESERVED	LENGTH	CM ID	Serial Number	Channel ID	MESSAGEs TOTAL	ERRORs TOTAL	VENDOR DATA

Figure 11: Packaging a single Status Bus/IVN in one Status Bus/IVN message

For multiple bus/IVN interfaces:

CM ID	COUNTER	VERSION	Message Type	DATA TYPE	RESERVED	CM Flags	Channel ID	TIME STAMP	LENGTH	Data Flags
VENDOR ID	CM VERSION	CM TYPE	RESERVED	VENDOR DATA LENGTH	CM ID	Serial Number	Channel ID	MESSAGEs TOTAL	ERRORs TOTAL	VENDOR DATA
							Channel ID	MESSAGEs TOTAL	ERRORs TOTAL	VENDOR DATA
							Channel ID	MESSAGEs TOTAL	ERRORs TOTAL	VENDOR DATA

Figure 12: Packaging multiple Status Bus/IVN in one Status Bus/IVN message

Messages Total / Errors Total:
Counted are the absolute received / sent messages and messages with errors. An error increases both the Messages Total and the Errors Total value. Further information can be transmitted in the Vendor Data.

Vendor Data
The Vendor Data field is used to convey specific information in every Capture Module variant. Consequently, the field has different meaning in different Capture Module types. As an example, for both the CM Ethernet Combo and CM 100 High variants:

Vendor Data for CM Eth Combo & CM 100 High	
Bytes	
0	1
Link Status	Link Quality
Linkup Time	

Table 13: Status Bus/IVN message Vendor Data example

Field Name	Data Type	Description
Link Status	UINT8	Flag to indicate whether there is linkUp or not in the corresponding channel: x00: No x01: Yes
Link Quality	UINT8	Quality of the link of the corresponding channel, from 0 to 5: 0x00: LinkDown 0x01: Lowest Link quality … 0x05: Highest Link quality
Linkup Time	UINT16	Time (ms) measured between power up of the board and link up on this port. Done once at startup. Two default reference values are defined based on an internal reference timeout elapsing 500ms after power up. • 0x0000: No linkup detected and the reference timeout did not elapse yet. • 0xFFFF: No linkup detected and the reference timeout already elapsed. For instance: x153 means 339 ms.

Table 14: Link Status, Link Quality and Linkup Time fields description

2.3.3 Status Capture Module message

Status Capture Module messages are sent cyclically every second after the logging link has been established. The payload is depicted in the next illustration.

4 Byte			
0	1	2	3
Vendor ID	CM Version	CM Type	Reserved
Vendor Data length (VDL)		CM ID	
Serial Number			
Vendor Data			

Table 15: Status CM Message payload overview

Field Name	Length	Description
Vendor ID	1 byte	Vendor ID – Technica Engineering: 0x0C
CM Version	1 byte	Versioning of the CM
CM Type	1 byte	Type of Capture Module (eg: CAN-Combo
Reserved	1 byte	For future extensions
Vendor Data Length	2 byte	Length of the “Vendor Data” field (in Bytes)
CM ID	2 byte	Configured ID of the Capture Module
Serial Number	4 byte	Serial number of the Capture Module
Vendor Data	n byte	Specific status information (e.g.: temperature processor, or CM’s error conditions)

Figure 13: Description of the Status CM message payload fields

Vendor Data - for all CM Variants:

Vendor Data - for all CM variants			
Bytes			
0	1	2	3
Reserved	SW Version		
HW Version		Buffer fill level	Buffer overflow
Buffer Size			
Lifecycle			
Lifecycle			
Voltage Integer part	Voltage fractional part	Temperature	

Table 16: Status CM message Vendor Data overview prior to SW version v015.X.010

Vendor Data - for all CM variants			
Bytes			
0	1	2	3
Reserved	SW Version		
HW Version		Buffer fill level	Buffer overflow
Buffer Size			
Lifecycle			
Lifecycle			
Voltage Integer part	Voltage fractional part	Chassis Temperature	Silicon Temperature

Table 17: Status CM message Vendor Data overview from SW version v015.X.010 onwards

Field Name	Data Type	Description
Reserved	UINT8	Reserved
SW Version	3x UINT8	vX.Y.Z, one byte for each part. E.g. v8.6.40 → 0x08 0x06 0x28
HW Version	2x UINT8	vX.Y, one byte for each part. E.g. v3.1 → 0x03 0x01
Buffer fill level	UINT8	Percentage of internal memory currently used. Possible values: 0, 10, 20, 30, 40, 50, 60, 70, 80, 85, 90, 95 100% is never possible because the memory is emptied before it reaches that level.
Buffer overflow	UINT8	Flag to indicate whether there is a memory overflow or not. x00: No, x01: Yes
Buffer Size	UINT32	Size of the internal memory in gigabits.
Lifecycle	UINT64	Time since startup, in ns.
Voltage integer part	UINT8	Voltage value, the part before the comma. E.g: 12,30 V → 12 = 0x0C
Voltage fractional part	UINT8	Voltage value, the part after the comma. E.g: 12,30 V → 30 = 0x1E
Chassis Temperature	SINT8	Temperature of the CM chassis in °C. • x00 up to 0x7E → 0°C up to 126°C • 0x7F → temp ≥ 127°C • 0x80 → Not Available • 0xFF down to 0x81 → -1°C down to -127°C
Silicon Temperature	SINT8	Internal temperature (junction temperature) of the CM processing unit in °C. • x00 up to 0x7E → 0°C up to 126°C • 0x7F → temp ≥ 127°C • 0x80 → Not Available • 0xFF down to 0x81 → -1°C down to -127°C

Table 18: Status CM message Vendor Data fields description from SW version v015.X.010 onwards

2.4 Control message

To control the recording system and all possible connected systems on the logging network (consisting of several loggers, Captures Module and instrumentation adapters) and signaling events in the recorded trace, Control messages are configured in the measurement components. All CMs can be informed about the occurrence of a configurable Control message. Received Control messages must be passed on by all components.

Note: Control messages can only be sent out via the logging port of a CM. The configuration port does not support it.

4Byte			
0	1	2	3
CM ID		Control Message ID	

Table 19: Control message payload overview

- **CM ID**: sender of the Control Message (see Table 23)
- **Control Message ID**: Identifier of the specific Control Message (see Table 24)

CM ID	Counter	Version	Message Type	Data Type	Reserved	CM Flags	Channel ID	Time-stamp	Length	Data Flags	CM ID	Control Message ID

Table 20: Control message overview

2.5 Time Synchronization

Time synchronization is required between the CM and other devices on the logging network.

Transmission medium: Ethernet (PTP)
Transmission protocol: PTP
PTP roles supported by Technica CMs:

- Grand master
- Bridge master
- Bridge slave

The time synchronization is provided via the logging interface. This bidirectional communication is therefore coexistent with TECMP communication with the purpose of transmitting logging data.

2.6 TECMP Header fields summary

The supported values of the TECMP header fields are documented in this chapter.

2.6.1 Message Type

Control Message	0	0x0
Status CM	1	0x1
Status Bus/IVN	2	0x2
Logging Stream	3	0x3
Status Configuration	4	0x4
Replay Data	10	0xA

Table 21: Message Type

Note: “Logging Stream” identifies data messages which are received by the CM. “Replay Data” identifies data messages which are sent by the CM. Both frame formats are identical.

2.6.2 Data Type

None (UNDEFINED)	00	0x0000
CAN Data	02	0x0002
CAN-FD Data	03	0x0003
LIN	04	0x0004
Flexray Data	08	0x0008
GPIO	10	0x000A
UART/RS232 ASCII	16	0x0010
UART/RS232 SLA	18	0x0012
Analog	32	0x0020
Analog SLA	33	0x0021
Ethernet II (classical Ethernet frame including UDP -DLT)	128	0x0080

Table 22: Data Type

2.6.3 CM ID

In order to enforce interoperability, the IP addresses of the CMs is defined based on the IP-network 10.104.3.xxx.

Capture Module	**IP Address** 10.104.3.xx	**CM ID** 2 Bytes	**Rotary switch position** (0x)
LIN Combo			
LIN Combo 0	10.104.3.48	48	0
LIN Combo 1	10.104.3.49	49	1
LIN Combo 2	10.104.3.50	50	2
...	...	...	...
LIN Combo 15	10.104.3.63	63	F
CAN Combo			
CAN Combo 0	10.104.3.64	64	0
CAN Combo 1	10.104.3.65	65	1
CAN Combo 2	10.104.3.66	66	2
...	...	...	...
CAN Combo 15	10.104.3.79	79	F
100 High			
100 High 0	10.104.3.96	96	0
100 High 1	10.104.3.97	97	1
100 High 2	10.104.3.98	98	2
...	...	...	...
100 High 15	10.104.3.111	111	F
Eth Combo			
Eth Combo 0	10.104.3.128	128	0
Eth Combo 1	10.104.3.129	129	1
Eth Combo 2	10.104.3.130	130	2
...	...	...	...
Eth Combo 15	10.104.3.143	143	F
1000 High			
1000 High 0	10.104.3.144	144	0
1000 High 1	10.104.3.145	145	1
1000 High 2	10.104.3.146	145	2
...	...	...	...
1000 High 15	10.104.3.159	159	F

Table 23: CM ID

2.6.4 Control Message ID

The type of control message is defined via a 2 Bytes long Control Message ID field.

Control Message ID	**Value**	**Comment**
Logger Ready	0x00 02	This control message shall be sent by the logger to notify the CM that it is ready to receive some data. The control message shall be sent cyclically every second.

Table 24: Control Message ID

2.6.5 CM Type

CM LIN Combo	0x02
CM CAN Combo	0x04
CM 100 High	0x06
CM Eth Combo	0x08
CM 1000 High	0x0A

Table 25: CM Type

3 Contact

If you have any questions regarding this document or our products, please contact us:

Technica Engineering GmbH
Leopoldstr. 236
80807 München
Germany

Fax: +49-89-34290265

Info@technica-engineering.de

www.technica-engineering.de

4 Changelog

Version	Chapter	Description	Date
1.0	All	First release.	10.02.2020
1.1	2.4 and 2.6.1	References to “User Event” replaced with “Control Message”.	12.02.2020
1.2	2.3.2, 2.3.3, 2.6.3 2.2.2.5 2.6.1 2.3	Description of the fields of Vendor Data (Status Bus/IVN messages and Status Capture Module Message) added. CM IP-address list generalized. References to “Length of vendor data” replaced with “Vendor Data Length” (VDL). Added value description to the TECMP Flexray Data flags. Added “Replay Data” Message type and frame format description. Aligned cyclic transmission of status message.	06.04.2020
1.3	2.6.3 2.6.5 2.3 2.3.2 2.3.1	CM ID table (switch position) updated with hexadecimal value. Capture Module naming updated. Cyclic time of Status messages updated. Added upper bound length of 1400 Bytes. Added Linkup time to Status bus/IVN message payload. Added description of Status Configuration message vendor data. Added 1000 High CM ID.	27.05.2020
1.4	2.2.1 2.2.2.4 2.2.2.5	Added restriction on Timestamp. “CRC Error” Data Flag bit detailed for FlexRay. “Tx” Data Flag bit detailed. Added “BRS” bit to the CAN-FD Data frame. Added “CAS” bit to the FlexRay Data frame.	25.08.2020
1.5	1.1 – 2.1 2.3.1 2.3.3	Editorial changes Status Configuration message cycle time updated from 1s to 60s. Field “Temperature” renamed to “Chassis Temperature”. Update from UINT8 to SINT8. Added “Silicon Temperature” field.	13.11.2020