Ethernet based

ASC Logging Format

Specification

Version 1.4.2 of 2018-09-20

Status	Completed
Publisher	Vector Informatik GmbH

Document Management

Revision list

Version	Date	Editor	Section	Changes, comments
1.0.0	2010-08-12	Jr	All	Created
1.1.0	2012-03-12	Wwi	4	AFDX logging format added
1.1.1	2012-05-07	Wwi	4	AFDX-flag bit enumeration updated
1.2.0	2013-04-23	Jr	3	Added Ethernet status and error event
1.3.0	2014-03-04	Wwi	4	AFDX status and bus statistic events added
1.3.1	2014-03-04	Wwi	All	Header description separated
1.3.2	2014-12-08	Jmi	4	Ethernet statistic added
1.4.0	2016-06-22	Jr	4	Added extended logging events
1.4.1	2016-12-19	Mom	all	CI and layout
1.4.2	2018-09-20	Jr	4.2	Fixed <channel> is always dec

VECTOR

Contents

1 Disclaimer 4

2 Overview 4

3 Header Section 5

4 Ethernet Format 6

4.1 Example 6

4.2 Logging Events 6

4.3 Symbols 9

5 AFDX Format 11

5.1 Example 11

5.2 Logging Events 11

5.3 Symbols 13

1 Disclaimer

Severability clause - Restrictions for the usage of Vector logging data formats outside of Vector products

The format specification / access functions for the Vector BLF and ASC logging data formats are made available under the restrictions and conditions cited hereafter.

Please note that Vector Informatik GmbH neither gives any guarantee nor assumes any liability beyond compulsory legal regulations for the BLF or ASC logging format respectively as well as for the access functions to the single objects.

Vector Informatik GmbH disclaims all liability for errors which might be contained in the access functions or the format specification itself.

Vector Informatik GmbH does neither provide support for the integration into your software nor for problems occurring inside your software on the customer side.

Beyond that Vector Informatik GmbH reserves the right to change the BLF or ASC data format respectively anytime without prior notification. Therefore, the compatibility of the format is not ensured.

2 Overview

The document specifies the format of Ethernet and AFDX events in the CANoe/CANalyzer ASC logging.

3 Header Section

A log file in ASCII format starts with a header consisting of four lines.

Format	data <DOW> <MMM> <DD> <FullTime> <YYYY> base <number format> timestamps <interpretation> <activation> internal events logged // version <tool version>
Example	Date Don Aug 12 12:08:18.345 2010 base hex timestamps absolute internal events logged // version 7.2.0

The symbols in the header lines have the following meaning:

- date
  - <DOW>: abbreviated (three characters) alphanumeric day of week
  - <MMM>: abbreviated (three characters) alphanumeric month
  - <DD>: day (01 .. 31) of month
  - <FullTime>[1]: standard time display
  - <YYYY>: year
- base
  - Numbers may be represented either in hexadecimal or decimal notation.
  - <number format> = {**hex**|**dec**}
- timestamps
  - The timestamps are either written absolute to the start of the measurement or relative to the preceding event.
  - <interpretation> = {**absolute**|**relative**}
- internal events logged
  - Internal events may be either logged (<activation> = {}) or omitted (<activation> = {**no**}).

[1] Starting with CANalyzer/CANoe 8.2 SP2 the standard time is stored in milliseconds. This can be disabled by setting the flag ASCII_Format_Milliseconds = 0 in section [SYSTEM] of the CAN.INI file.

4 Ethernet Format

4.1 Example

The following Ethernet logging contains a header and some Ethernet events.

```
Date Don Aug 12 12:08:18 2010
base hex  timestamps absolute
internal events logged
// version 7.2.0
Begin Triggerblock Don Aug 12 12:08:18 2010
   0.000000 Start of measurement
   0.000000  ETH 2 Tx
3c:FFFFFFFFFFFF4000000000040806000108000604000140000000000004C0A8000100000000000000C0A800010000000
0000000000000000000000000000000
   0.000010  ETH 2 Tx
3c:FFFFFFFFFFFF4000000000050806000108000604000140000000000005C0A8000200000000000000C0A800020000000
0000000000000000000002C000000
   0.000020  ETH 2 Tx
3c:FFFFFFFFFFFF4000000000050806000108000604000140000000000005C0A8000200000000000000C0A8000100000C0
00000DC4E860C38F9C90B22482211
   0.000030  ETH 2 Tx
3c:400000000004400000000000508004500002C000100004006F977C0A80002C0A800010017001769EBE5680000000
06002E000E74F0000020405B47400
   0.000040  ETH 2 Tx
3c:400000000005400000000000408060001080006040002400000000004C0A8000140000000000005C0A8000200000C0
00000DC4E860C38F9C90B22482211
   0.000050  ETH 2 Tx
3c:400000000005400000000000408004500002C000100004006F977C0A80001C0A8000200170017F49BC42069EBE56
96012E0002E820000020405B48877
   0.000060  ETH 2 Tx
3c:400000000004400000000000508004500002800020000400 6F97AC0A80002C0A800010017001769EBE569F49BC42
15010E420421F0000400000000004
End TriggerBlock
```

4.2 Logging Events

The section contains the Ethernet events of the CANoe/CANalyzer ASC logging format. See section 0 for an explanation of the symbols.

Ethernet Packet Event	
Ethernet packet received or transmitted on a Ethernet channel	
Format	<Time> ETH <Channel> <Dir> <DataLen>:<Data>
Format Extended Since 9.0 SP3	<Time> ETH <Channel> <Dir> <DataLen>:<Data> [HwCh:<HardwareChannel>] [FCS:<FrameChecksum>] [Length:<FrameDuration>]
Example	0.000000 ETH 2 Tx 3c:FFFFFFFFFFFF4000000000040806000108000604000140000000000004C0A8000100000000000000C0A8000100000000000000000000000000000000000000000000
	2.908192 ETH 2 Rx 3c:FFFFFFFFFFFF200000000001F00101020304050607080 90A0B0C0D0F000000000000000000000000000000000000000000000000000000000000000000000000 HwCh:1 FCS:b0dc9ba4 Length:5764

Ethernet link status	
Status of the Ethernet link	
Format	<Time> ETH <Channel> STAT [Link:Link_up|Link_down] [LinkSpeed:<bitrate>Mbit/s] [Physical:IEEE802.3|BroadR-Reach] [Duplex:Full|Half] [MDI:Crossover|Direct] [Connector:RJ45|D-Sub] [BRClockMode:Master|Slave]
Format Extended Since 9.0 SP3	<Time> ETH <Channel> STAT [Link:Link_up|Link_down] [LinkSpeed:<bitrate>Mbit/s] [Physical:IEEE802.3|BroadR-Reach] [Duplex:Full|Half] [MDI:Crossover|Direct] [Connector:RJ45|D-Sub] [BRClockMode:Master|Slave][HwCh:<HardwareChannel>]
Example	0.002404 ETH 1 STAT Link:Link_up LinkSpeed:100Mbit/s Physical:IEEE802.3 Duplex:Full MDI:Crossover Connector:RJ45

Ethernet error	
Erroneous Ethernet frame received	
Format	<Time> ETH <Channel> <Dir>Er <ErrorCode> <FrameChecksum> <DataLen>:<Data>
Format Extended Since 9.0 SP3	<Time> ETH <Channel> <Dir>Er <ErrorCode> <FrameChecksum> <DataLen>:<Data> [HwCh:<HardwareChannel>] [Length:<FrameDuration>]
Example	1.465409 ETH 2 RxEr 02 00001234 3c:00000000000100000000002FF0100010203040506070809 0A0B0C0D0E0F101112 131415161718191A1B1C1D1E1F2021000000000000000000000000000

Ethernet statistics	
Erroneous Ethernet frame received	
Format	<Time> ETH <Channel> BUSSTATISTIC [HwRxPkts:<RxPacketCountHW> HwTxPkts:<TxPacketCountHW> HwRxError:<RxErrorCountHW> HwTxError:<TxErrorCountHW> HwRxBytes:<RxByteCountHW> HwTxBytes:<TxByteCountHW> HwRxNoBuffer:<RxNoBufferHW> HwSQIValue:<SQIValueHW>]

Ethernet statistics	
Format Extended Since 9.0 SP3	<Time> ETH <Channel> BUSSTATISTIC [HwRxPkts:<RxPacketCountHW> HwTxPkts:<TxPacketCountHW> HwRxError:<RxErrorCountHW> HwTxError:<TxErrorCountHW> HwRxBytes:<RxByteCountHW> HwTxBytes:<TxByteCountHW> HwRxNoBuffer:<RxNoBufferHW> HwSQIValue:<SQIValueHW>] [HwCh:<HardwareChannel>]
Example	0.161924 ETH 1 BUSSTATISTIC HwRxPkts:1 HwTxPkts:0 HwRxError:0 HwTxError:0 HwRxBytes:72 HwTxBytes:0 HwRxNoBuffer:0 HwSQIValue:6

Ethernet Packet Event Forwarded	
Ethernet packet which was forwarded by interface hardware. The interface hardware must support this feature, i.e. VN5640 in operation mode Ethernet switch.	
Format Extended Since 9.0 SP3	<Time> ETH <Channel> TxFwd <DataLen>:<Data> [HwCh:<HardwareChannel>] [FCS:<FrameChecksum>] [Length:<FrameDuration>]
Example	10.111046 ETH 1 TxFwd 4a:FFFFFFFFFFFF200000000200F00400010203040506070809 0A0B0C0D0F000102030405060708090A0B0C0D0F000102030405060708090A0B0C0D0F00010203040506070809 0A0B0C0D0F HwCh:1 FCS:16871a3 Length:6880

Ethernet Error Forwarded	
Erroneous Ethernet frame which was forwarded by interface hardware. The interface hardware must support this feature, i.e. VN5640 in operation mode Ethernet switch.	
Format Extended Since 9.0 SP3	<Time> ETH <Channel> TxErFwd <ErrorCode> <FrameChecksum> <DataLen>:<Data> [HwCh:<HardwareChannel>] [Length:<FrameDuration>]
Example	11.711114 ETH 1 TxErFwd 06 12345678 67:FFFFFFFFFFFF200000000200F00600010203040506070809 0A0B0C0D0F000102030405060708090A0B0C0D0F000102030405060708090A0B0C0D0F00010203040506070809 0A0B0C0D0F001020304050607080 90A0B0C0D0F001020304050607080 90A0B0C0D0F001020304050607080 90A0B0C0D0F HwCh:13 Length:920

4.3 Symbols

Explanation of symbols used in descriptions above.

Symbol	Width in chars		Meaning	Range	Example (hex mode)	Special
	hex	dec				
<Channel>	-	1 … 3	Application channel	1…255	1	1)
<FullTime>	15	15	a string that represents a time in the current format. hh:mm:ss.ms am|pm	00:00:00.000 … 12:60:60.999	01:13:17.123 pm	Different range in German version: 00:00:00.000 … 23:60:60.999 In German version 'am' and 'pm' are not used. Therefore the width is only 12 chars.
<Data>	>0	-	Data of Ethernet packet as Hex string. Starting with the target Ethernet MAC ID and ending with data or padding bytes (without Ethernet packet CRC). Each data byte has two hex characters.	(0…FF_{16})*	ED002FB2 C0 FF 23 …	2)
<DataLen>	3	4	Length of data in bytes (packet length).	0…1518	60	
<Dir>	2 or 4		Direction of transmission	Rx, Tx, TxRq	Rx	3)
<Time>	-	>=11	Absolute or relative time in seconds		1234.567890	usually 6 decimal places
<ErrorCode>	1		Error code for Rx errors 1 – Data length error 2 – Invalid CRC 4 – Invalid Data received 8 – Collision detected	0..15		

Symbol	Width in chars		Meaning	Range	Example	Special
<Frame Checksume>	-	8	Frame checksum	0..FFFFFFFF$_{16}$	00001234	
<RxPacketCountHW>	> 0		Rx-frames detected by HW in epoch	Any integer	1	
<TxPacketCountHW>	> 0		Tx-frames detected by HW in epoch	Any integer	0	
<RxErrorCountHW>	> 0		Rx-error frames detected by HW	Any integer	0	
<TxErrorCountHW>	> 0		Tx-error frames detected by HW	Any integer	0	
<RxByteCountHW>	> 0		Rx-bytes detected by HW in epoch	Any integer	72	
<TxByteCountHW>	> 0		Tx-bytes detected by HW in epoch	Any integer	0	
<RxNoBufferHW>	> 0		Rx-frames with lost/dropped data buffer in epoch	Any integer	0	
<SQIValueHW>	1		Value for the Quality of the BroadR-Reach connection 0 – ErrorOccuring 1 – NoMargin 2 - Marginal 3 - Acceptable 4 - Good 5 – Excellent 6 - not available	0...6	6	
<HardwareChannel>	1..2		Hardware channel of the Application Channel. Only available, if interface hardware supports more than 1 hardware channel in configured operation mode, i.e. Ethernet switch mode.	1..16	1	
<MessageDuration>	> 0		Frame duration [in nanoseconds].		768000	

1) Number of supported channels depend of CANalyzer/CANoe version.

2) hexadecimal always

3) TxRq not used yet

5 AFDX Format

5.1 Example

The following AFDX logging contains AFDX events.

```
date Tue Jan 21 01:06:22 pm 2014
base hex  timestamps absolute
internal events logged
// version 8.2.0
Begin Triggerblock Tue Jan 21 01:06:22 pm 2014
   0.000000 Start of measurement
   0.000000 AFDX 2 STAT Line:A Status:Link_up LinkSpeed:10Mbit/s
   0.002274 AFDX 1 STAT Line:A Status:Link_up LinkSpeed:100Mbit/s Physical:IEEE802.3
Duplex:Full MDI:Direct Connector:RJ45
   0.002624 AFDX 1 STAT Line:A Status:Link_up LinkSpeed:100Mbit/s Physical:IEEE802.3
Duplex:Full MDI:Direct Connector:RJ45 Line:B Status:Link_up LinkSpeed:100Mbit/s
Physical:IEEE802.3 Duplex:Full MDI:Crossover Connector:RJ45
   1.000117  AFDX 1 BUS Line:A Flag:127 Dur: 1000000 LC:2 LS:100 LL:0 HwRxPkts:0 HwTxPkts:0
HwRxErr:0 HwTxErr:0 HwRxB:0 HwTxB:0 RxPkts:0 TxPkts:0 Inv:0 Drop:0 Lost:0
   1.000117  AFDX 1 BUS Line:B Flag:127 Dur: 1000000 LC:2 LS:100 LL:0 HwRxPkts:0 HwTxPkts:0
HwRxErr:0 HwTxErr:0 HwRxB:0 HwTxB:0 RxPkts:0 TxPkts:0 Inv:0 Drop:0 Lost:0
   1.000117  AFDX 2 BUS Line:A Flag:127 Dur: 1000000 LC:2 LS:10 LL:0 HwRxPkts:0 HwTxPkts:0
HwRxErr:0 HwTxErr:0 HwRxB:0 HwTxB:0 RxPkts:0 TxPkts:0 Inv:0 Drop:0 Lost:0
   1.627267  AFDX 1 Tx      1    0    0
6b:030000007E0502000002262008004500005C000040000111EAA90A022600E0E07E057E057E05004800000000000
0030303030000012300000123000001230000012300000000000000000000000000000000000000000000000000
0000000000000000000000000000000
   1.627271  AFDX 1 Rx      2  811    0
6b:030000007E0502000002262008004500005C000040000111EAA90A022600E0E07E057E057E05004800000000000
0030303030000012300000123000001230000012300000000000000000000000000000000000000000000000000
0000000000000000000000000000000
   1.633557  AFDX 1 Tx      1    0 1891
6b:030000007E0502000002262008004500005C000040000111EAA90A022600E0E07E057E057E05004800000000000
0030303030000012300000123000001230000012300000000000000000000000000000000000000000000000000
0000000000000000000000000000001
   1.633561  AFDX 1 Rx      2  811 1891
6b:030000007E0502000002262008004500005C000040000111EAA90A022600E0E07E057E057E05004800000000000
0030303030000012300000123000001230000012300000000000000000000000000000000000000000000000000
0000000000000000000000000000001
End TriggerBlock
```

5.2 Logging Events

The section contains the AFDX event of the CANoe/CANalyzer ASC logging format. See section 5.3 for an explanation of the symbols.

AFDX received / transmitted AFDX frame	
Format	<Time> AFDX <Channel> <Dir> <AFDX-channel> <Flags> <BAG> <DataLen>:<Data>
Example	0.000000 AFDX 1 Tx 1 64 0 3c:FFFFFFFFFFFF4000000000040806000108000604000140000000000 4C0A8000100000000000 0C0A8000100000000000000000000000000000000000

AFDX link status	
Format	<Time> AFDX <Channel> STAT Line:[A|B] [Status:Link_up|Link_down] [LinkSpeed:<bitrate>Mbit/s] [Physical:IEEE802.3|BroadR-Reach] [Duplex:Full|Half] [MDI:Crossover|Direct] [Connector:RJ45|D-Sub] [Line:B [Status:Link_up|Link_down] [LinkSpeed:<bitrate>Mbit/s] [Physical:IEEE802.3|BroadR-Reach] [Duplex:Full|Half] [MDI:Crossover|Direct] [Connector:RJ45|D-Sub]]
Example	0.003796 AFDX 1 STAT Line:A Status:Link_up LinkSpeed:100Mbit/s Physical:IEEE802.3 Duplex:Full MDI:Direct Connector:RJ45

Note that one line in the log file may contain either status of line A only or line B only or both together.

AFDX bus statistic events	
Format	<Time> AFDX <Channel> BUS Line:[A|B] Flag:<ValidFlags> Dur:<Duration> [mysec] [LC:<LinkStatus> LS:<LinkSpeed>[Mbps] LL:<LinkLosses>] [HwRxPkts:<RxPacketCountHW> HwTxPkts:<TxPacketCountHW> HwRxErr:<RxErrorCount> HwTxErr:<TxErrorCount> HwRxB:<RxByteCount> HwTxB:<TxByteCount>] [RxPkts:<RxPacketCount> TxPkts:<TxPacketCount] [Inv:<InvalidPacketCount>] [Drop:<DroppedPacketCount>] [Lost:<LostPacketCount>]
Example	5.921321 AFDX 1 BUS Line:A Flag:127 Dur:999999 LC:2 LS:100 LL:0 HwRxPkts:0 HwTxPkts:0 HwRxErr:0 HwTxErr:0 HwRxB:0 HwTxB:0 RxPkts:0 TxPkts:0 Inv:0 Drop:0 Lost:0

5.3 Symbols

Explanation of symbols used in descriptions above.

Symbol	Width in chars		Meaning	Range	Example (hex mode)	Special
	hex	dec				
<BAG>	> 0		Measured time [µsec] since last frame on this VL	Any integer	1025	
<Channel>	1 … 2	1 … 3	Application channel	1…255	1	1)
<Data>	>0	-	Data of AFDX packet as Hex string. Starting with the target Ethernet MAC ID and ending with data or padding bytes (without Ethernet packet CRC) and AFDX-SeqNo. Each data byte has two hex characters.	$(0…FF_{16})$*	ED002FB2 C0 FF 23 …	2)
<DataLen>	3	4	Length of data in bytes (packet length).	0…1518	60	
<Dir>	2 or 4		Direction of transmission	Rx, Tx, TxRq	Rx	3)
<ETH-channel>	1 … 2	1 … 3	Underlying ETH-channel	1 … 255	1	1)
<Flags>	1 … 4	1 … 5	Flags signaling specific status and errors, see below	0 … FFFF	1	4)
<Time>	-	>=11	Absolute or relative time in seconds		1234.567890	usually 6 decimal places
<ValidFlags>	1 … 2	1 … 3	Flags signaling the validity of information blocks	0 … FF	FF	5)
<Duration>	> 0		Time period of recording the values (epoch) in [µsec]	Any integer	1000000	
<LinkStatus>	1		Enumeration of link status	0 … 4	2	6)
<LinkLosses>	> 0		link losses encountered during epoch	Any integer	0	
<RxPacketCountHW>	> 0		Rx-frames detected by HW in epoch	Any integer	100	

Symbol	Width in chars		Meaning	Range	Example (hex mode)	Special
	hex	dec				
<TxPacketCountHW>	> 0		Tx-frames detected by HW in epoch	Any integer	100	
<RxErrorCount>	> 0		Rx-error frames detected by HW	Any integer	0	
<TxErrorCount>	> 0		Tx-error frames detected by HW	Any integer	0	
<RxByteCount>	> 0		Rx-bytes detected by HW in epoch	Any integer	10340	
<TxByteCount>	> 0		Tx-bytes detected by HW in epoch	Any integer	10340	
<RxPacketCount>	> 0		Rx-frames received by CANwin in epoch	Any integer	100	
<TxPacketCount>	> 0		frames sent by CANwin in epoch	Any integer	100	
<InvalidPacketCount>	> 0		invalid frames detected by CANwin in epoch	Any integer	0	
<DroppedPacketCount>	> 0		frames dropped by CANwin in epoch	Any integer	0	
<LostPacketCount>	> 0		frames lost by CANwin in epoch	Any integer	0	

1) Number of supported channels depends on CANalyzer/CANoe version.

2) hexadecimal always

3) TxRq not used yet

4) Status- and error flags for AFDX packets using following bits with meaning:
   Bit 0 – Info: Frame from line-B
   Bit 1 – Info: Frame is redundant
   Bit 2 – Info: Frame is a fragment of a message
   Bit 3 – Info: Packet is a SAP message or part of it
   Bit 4 – Error: Frame occurred on wrong line (A/B)
   Bit 5 – Error: Frame is not invalid AFDX frame (Ethernet header failure)
   Bit 6 – Error: AFDX sequence number is invalid
   Bit 7 – Error: Redundancy error encountered
   Bit 8 – Error: Fragmentation / reassembly error
   Bit 9 – Error: violation of a higher protocol (IPv4, UDP etc.)
   Bit 10 – Info: Packet has been reassembled.
   Bit 11 – Info: Frame has been checked by redundancy manager
   Bit 12 Warning: a constant or value violates AFDX recommendation
   Bit 13 Warning: Frame size does not match expected size from database
   Bit 14 Info: Packet is not defined in database
   Bit 15 Info: Info: Frame has been checked by integrity manager

5) Valid flags for AFDX statistics

Bit 0 – channel is configured
Bit 1 – HW related counters are valid
Bit 2 – CANwin related counters are valid
Bit 3 – link-related info is valid
Bit 4 – invalid packet counter is valid
Bit 5 – lost packet counter is valid
Bit 6 – dropped packet counter is valid
Bit 7 – byte counters are based on CANwin packets, not on HW

6) Link status

0: Unknown: Link status could not be determined
1: Down: Link is down
2: Up: Link is up
3: Negotiate: Link is currently performing auto-negotiation
4: Error: Link is in error state

Purpose, Goal