



ASAM

Association for Standardization of
Automation and Measuring Systems

ASAM MDF

Measurement Data Format

Programmers Guide

Version 4.2.0

Date: 2019-09-30

Base Standard

© by ASAM e.V., 2019

Disclaimer

This document is the copyrighted property of ASAM e.V.
Any use is limited to the scope described in the license terms. The license terms can be viewed at www.asam.net/license

Table of Contents

Foreword	7
1 Introduction	8
1.1 Overview	8
1.2 Conventions.....	8
1.3 Motivation	8
1.4 Scope	8
1.5 File Extension	9
2 Relations to Other Standards	10
2.1 Backward Compatibility to Earlier Releases	10
2.1.1 MDF Versions	10
2.1.2 Version History.....	10
2.1.3 Version Handling.....	11
2.1.4 Rules to Ensure MDF Compatibility between Versions.....	12
2.2 Compliance to ASAM ODS Standard	14
3 Introduction	15
4 MDF General Block Format	17
4.1 Overview of Block Types Used	17
4.2 Definition of Data Types Used And Mapping to ASAM Data Types	18
4.3 General Block Setup	19
4.3.1 Header section.....	20
4.3.2 Link section.....	20
4.3.3 Data section.....	20
4.3.4 Description of Members	21
4.4 General Rules	21
4.4.1 String Encoding, Byte Order and Alignment	21
4.4.2 Naming Rules	22
4.4.3 Identification of Channels	23
4.4.4 Linking of Blocks	26
4.4.5 Data storage	29
4.4.6 Synchronization Domains.....	30
4.5 The File Identification Block IDBLOCK	31
4.5.1 Block Structure of IDBLOCK	32
4.5.2 Unfinalized MDF	33
4.6 The Header Block HDBLOCK	37
4.6.1 Block Structure of HDBLOCK.....	37
4.6.2 Contents of the Meta Data Block for the Header Comment	39
4.6.3 Best Practice for Unique Identification of Related Measurement Files..	41
4.7 The Meta Data Block MDBLOCK	43

4.7.1	Block Structure of MDBLOCK	43
4.8	The Text Block TXBLOCK.....	44
4.8.1	Block Structure of TXBLOCK	44
4.9	The File History Block FHBLOCK.....	44
4.9.1	Block Structure of FHBLOCK	44
4.9.2	Contents of the Meta Data Block for the File History Comment.....	46
4.10	The Channel Hierarchy Block CHBLOCK	46
4.10.1	Block Structure of CHBLOCK.....	46
4.10.2	Contents of the Meta Data Block for the Hierarchy Comment	49
4.11	The Attachment Block ATBLOCK	50
4.11.1	Block Structure of ATBLOCK	50
4.11.2	Contents of the Meta Data Block for the Attachment Comment.....	51
4.12	The Event Block EVBLOCK	52
4.12.1	Block structure of EVBLOCK.....	53
4.12.2	Common Use Cases for Events	61
4.12.3	Example For Events.....	62
4.12.4	Contents of the Meta Data Block for the Event Comment.....	63
4.12.5	Event Signals	64
4.12.6	Documentation of Lost Time Synchronization	69
4.13	The Data Group Block DGBLOCK	70
4.13.1	Block Structure of DGBLOCK	70
4.13.2	Contents of the Meta Data Block for the Data Group Comment	71
4.14	The Channel Group Block CGBLOCK.....	71
4.14.1	Block Structure of CGBLOCK	72
4.14.2	Contents of the Meta Data Block for the Channel Group Comment.....	75
4.14.3	Remote Master Link	76
4.14.4	Variable Length Signal Data (VLSD) CGBLOCK.....	76
4.14.5	Example using a VLSD CGBLOCK.....	78
4.15	The Source Information Block SIBLOCK.....	79
4.15.1	Block Structure of SIBLOCK	80
4.15.2	Contents of the Meta Data Block for the Source Comment	81
4.16	The Channel Block CNBLOCK.....	82
4.16.1	Block Structure of CNBLOCK.....	82
4.16.2	Data Structure for Channel Data Type "CANopen Date"	93
4.16.3	Data Structure for Channel Data Type "CANopen Time"	94
4.16.4	Contents of the Meta Data Block for the Channel Comment	94
4.16.5	Contents of the Meta Data Block for the Channel Unit	96
4.17	The Channel Conversion Block CCBLOCK	96
4.17.1	Block Structure of CCBLOCK.....	97
4.17.2	CCBLOCK – 1:1 Conversion.....	100
4.17.3	CCBLOCK – Linear Conversion	100
4.17.4	CCBLOCK - Rational conversion	100
4.17.5	CCBLOCK –Algebraic Conversion (MCD-2 MC Text formula)	101
4.17.6	CCBLOCK – Value to Value Table With Interpolation	101
4.17.7	CCBLOCK – Value to Value Table Without Interpolation.....	102
4.17.8	CCBLOCK – Value Range to Value Table	102
4.17.9	CCBLOCK – Value to Text / Scale Conversion Table	103
4.17.10	CCBLOCK – Value Range to Text / Scale Conversion Table.....	104
4.17.11	CCBLOCK – Text To Value Table	106

4.17.12CCBLOCK – Text To Text Table.....	107
4.17.13CCBLOCK – Bitfield Text Table.....	108
4.17.14Contents of the Meta Data Block for the Conversion Comment	109
4.17.15Contents of the Meta Data Block for the Conversion Unit	110
4.18 Composition of Channels	110
4.18.1 Structures	110
4.18.2 Arrays	113
4.19 The Channel Array Block CABLOCK	113
4.19.1 Block structure of CABLOCK	113
4.20 Nested Composition of Channels	129
4.20.1 Structures of Composed Signals.....	129
4.20.2 Arrays of Composed Signals.....	129
4.21 The Data Block DTBLOCK	134
4.21.1 Block structure of DTBLOCK.....	134
4.21.2 Data Block Format	134
4.21.3 Sorted and Unsorted Data.....	135
4.21.4 DLBLOCK vs. LDBLOCK vs. single Data Block	136
4.21.5 Reading Signal Values from a Data Record	136
4.21.5.1 Reading the Invalidation Bit.....	136
4.21.5.2 Reading the Signal Value	137
4.21.6 Example 1: Little Endian (Intel) Byte Order	138
4.21.7 Example 2: Big Endian (Motorola) Byte Order.....	139
4.22 The Data Values Block DVBLOCK.....	140
4.22.1 Block structure of DVBLOCK	140
4.23 The Invalidation Data Block DIBLOCK.....	141
4.23.1 Block structure of DIBLOCK.....	141
4.24 The Sample Reduction Block SRBLOCK.....	141
4.24.1 Block Structure of SRBLOCK.....	142
4.24.2 Motivation for Sample Reduction.....	143
4.24.3 Layout of Sample Reduction Records	144
4.25 The Reduction Data Block RDBLOCK.....	146
4.25.1 Block structure of RDBLOCK	146
4.26 The Reduction Values Block RVBLOCK.....	146
4.26.1 Block structure of RVBLOCK	147
4.27 The Reduction Data Invalidation Block RIBLOCK.....	147
4.27.1 Block structure of RIBLOCK.....	147
4.28 The Signal Data Block SDBLOCK	147
4.28.1 Block structure of SDBLOCK	148
4.28.2 Example for Usage of SDBLOCK.....	148
4.29 The Data List Block DLBLOCK.....	149
4.29.1 Block structure of DLBLOCK.....	150
4.30 The List Data Block LDBLOCK.....	155
4.30.1 Block structure of LDBLOCK.....	156
4.31 The Data Zipped Block DZBLOCK.....	160
4.31.1 Block structure of DZBLOCK.....	161
4.31.2 Transposition of Data.....	162
4.32 The Header List Block HLBLOCK.....	164

4.32.1	Block structure of HLBLOCK.....	164
5	MDF Meta Data Format	167
5.1	Content of the MDBLOCK.....	167
5.1.1	XML Versioning.....	167
5.1.2	Standard Tags	168
5.1.3	Standard Attributes	169
5.1.4	Embedding of Markup	169
5.2	Common Properties	170
5.2.1	<e> Tag	170
5.2.2	<tree> Tag	171
5.2.3	<list> Tag.....	172
5.2.4	<elist> Tag.....	173
5.3	Vendor Extensions	173
5.4	Alternative Names	174
5.5	Formula Specifications	174
6	Terms and Definitions	176
7	Symbols and Abbreviated Terms	177
8	Bibliography	179
	Figure Directory	180
	Table Directory	181

Foreword

MDF stands for Measurement Data Format. It is a binary file format to store measured or calculated data for post-measurement processing or long-term conservation. Common sources of the data to be stored are sensors, ECUs or bus monitoring systems. In addition to the plain measurement data, MDF also contains descriptive and customizable meta data within the same file.

The format is organized in loosely coupled binary blocks to ensure high performance reading and writing. The measurement data is stored channel-oriented and organized in records. MDF supports non-equidistant sampling rates and multiple rates per file. Master channels are used for synchronization which can be time, angle, distance or simply index related.

MDF allows storage of raw measurement values and corresponding conversion formulas. It supports special data types and information particularly required in the automotive area, and is closely related to other ASAM standards, e.g. ASAM MCD-2 MC (ASAP2) [1].

The format definition is published in the base standard which describes the following content:

- Version history
- Version handling and compatibility rules
- MDF block structure
- General rules
- List of all binary block types and their contents
- MDF meta data format (XML)

The description how to store information for certain use cases is published in different associated standards. Such use cases are:

- **Naming of Channels and Channel Groups [10]**
This associated standard describes how to set the different names for channels and channel groups for common use cases in order to provide important information to the user and to achieve a unique identification of the channels.
- **Bus Logging [7]**
This associated standard describes how to store the traffic of common bus systems.
- **Measurement Environment [9]**
This associated standard describes how to store information about the measurement environment.
- **Classification Results [8]**
This associated standard describes how to store classification results.

Binary example MDF files are contained in the deliverable of the respective standard or may be requested from the ASAM MDF working group (mdfproject@asam.net).