

Beispiele

Hex-kodiert	Reelle Zahl
00 00 00 00 00 00	0.0
81 00 00 00 00 00	1.0
82 00 00 00 00 00	2.0
81 00 00 00 00 80	-1.0
86 00 00 00 00 00	32.0
86 00 00 00 00 80	-32.0
86 00 00 00 00 40	48.0
86 00 00 00 00 C0	-48.0
13 38 A6 42 1A 8B	1179.77

Hinweis

Es können weder "not a number" (undefinierte Werte) noch "plus unendlich" oder "minus unendlich" gespeichert werden.

5.14 Dateiformat

5.14.1 Übersicht

Das Dateiformat von FAMOS enthält neben den eigentlichen Daten in binärer Form zusätzliche Informationen über die gespeicherten Daten.

Die einzelnen Informationen werden in der Datei in so genannten Keys abgelegt. Diese bestehen aus einer Folge von ASCII-Zeichen oder Binärdaten. Ein Key beginnt mit dem Zeichen "]" (ASCII 124), gefolgt von zwei Buchstaben, die zur Identifikation dienen. Danach folgen, jeweils durch Kommata voneinander getrennt, die zum Key gehörenden Parameter. Das Ende jedes Keys bildet ein Semikolon.

Das Dateiformat von FAMOS ist im Vergleich zur Version 3.2 modifiziert worden. Dateien im bisherigen Format bleiben jedoch selbstverständlich lesbar.

Mit dem modifizierten Dateiformat stehen neben dem Speichern von einkanaligen reellen Datensätzen und komplexen Daten zusätzlich folgende Möglichkeiten zur Verfügung.

- Mehrere Datensätze und Texte sind zusammen in einer Datei speicherbar.
- Die Kanäle können im Multiplex (also zeitlich sortiert) oder blockweise (nach Kanälen sortiert) in die Datei geschrieben werden.
- Kanäle und Texte können zu Datengruppen zusammengefasst werden, um zusammenhängende Messgrößen zu strukturieren.
- Kanälen, Texten und Datengruppen können Kommentare zugeordnet werden.
- Die maximale Länge aller Texte, wie Kanalnamen, Kommentare und Einheiten beträgt 255 Zeichen.
- Digitale und analoge Daten können in einer Datei gemischt werden.
- Jeder Kanal oder auch eine ganze Kanalgruppe kann eine eigene Zeitspur haben.

Das Dateiformat ist offen für Erweiterungen, vorgesehen ist z. B.

- die effektive Speicherung von Matrizen,
- die Speicherung von Messungen, die sich in ihrer Abtastzeit, Startzeit und der Anzahl der Samples unterscheiden können, im gleichen Datensatz,
- die Speicherung von Markern, die Zeiger auf markante Stellen im Datensatz darstellen.

Ab Famos 6.1 können Dateien größer als 2GByte werden. Insbesondere kann der binäre Datenblock (|CS) so groß werden. Der Datenbereich für einen einzigen Kanal ist jedoch noch immer auf 2 GByte beschränkt. Insbesondere ist jeder Buffer (|Cb) auf diese Größe beschränkt. Große Dateien entstehen also i.a., wenn mehrere Kanäle (jeder in einem eigenen Buffer) hintereinander in der Datei stehen.

5.14.2 Aufbau eines Keys

1. |**XX**,
Senkrechter Strich (ASCII 124) und zwei Buchstaben. Der erste Buchstabe ist entweder ein C oder ein N. C steht dabei für "critical" (Key muss verstanden werden, um Daten korrekt interpretieren zu können), N für "noncritical". Der 2. Buchstabe erläutert die Bedeutung des Keys. Dann folgt ein Komma.
2. Die Version des Keys, beginnend bei 1. Eine ganze Zahl, wiederum durch Komma beendet.
3. Länge des Keys, die Anzahl der Bytes nach dem folgenden Komma bis zum abschließenden ';'. Das Komma zählt nicht mehr mit, das ';' auch nicht. Damit ist es wie die Länge eines Namen die Anzahl der eingeschlossenen Zeichen. Die Länge des Keys ist i.a. eine ganze Zahl. In einigen Ausnahmefällen eine "große ganze Zahl". Beim entsprechenden Key ist vermerkt, falls die Ausnahme zutrifft.
4. Weitere Parameter, jeweils durch Kommas getrennt.
5. Der letzte Parameter wird durch ein Semikolon beendet.

Anmerkungen

Bei Keys ist Groß/Klein-Schreibung relevant.

Zwischen den Keys sind Leerzeichen (auch CR, LF für die Lesbarkeit) erlaubt.

Keys, die nicht verstanden werden, werden mit Hilfe ihrer Länge beim Lesen einer Datei übersprungen.

Namen, Kommentare dürfen nur lesbare Zeichen ohne 0 zwischendurch enthalten. Texte dürfen max. 255 Zeichen lang sein.

Reellen Zahle im ASCII-Format dürfen max 25 Bytes belegen. Eventuelle Nachkommastellen sind mit Dezimalpunkt (nicht Komma) abzulegen. Punkt und Exponent (E oder e) sind optional.

Alle ganzen Zahlen dürfen maximal 10 Zeichen lang sein. Der Wertebereich muss innerhalb $0 \dots 2^{31}-1$ liegen, also insbesondere nicht negativ.

An wenigen Stellen ist der Datentyp "große ganze Zahl" erlaubt. Diese dürfen maximal 20 Zeichen lang sein. Der Wertebereich beträgt $0 \dots 2^{63} - 1$. Diese Stellen sind durch Angabe des Datentyps besonders gekennzeichnet.

Zwischen dem einleitenden | bis zum Komma hinter der Version des Keys dürfen keine Leerzeichen stehen. Ansonsten sind führende und abschließende Leerzeichen bei Zahlen unter Berücksichtigung der maximalen Zeichenzahl erlaubt.

Alle reellen Zahlen (float, double) in binären Keys müssen gültig sein und im erlaubten Zahlenbereich $-1e35 \dots -1e-35, 0.0, 1e-35 \dots 1e35$ liegen.

Bei Tupel-Daten oder Multiplex-Daten muss jedes teilweise begonnene Tupel komplett in der Datei vorhanden sein, auch wenn am Ende in (einigen) Kanälen nicht mehr alle Samples definiert sind. Damit kann jede Software stets ganze Tupel einlesen und das Tupel-Gefüge wird auch am Ende strikt eingehalten.

Verweis

Die konkrete Beschreibung der verwendeten Zahlenformate für die binäre Abspeicherung der Zahlen finden Sie im Abschnitt "[Datentypen](#)".

5.14.3 Beschreibung der Keys

CF,2,1,1;	
	Dateiformat, Prozessor Dateiformat = 2, Keylänge = 1, Prozessor = 1

CK,1,3,1,Abgeschlossen;	
	Start einer Gruppe von Keys. Folgt immer direkt nach dem CF-Key. Keylänge immer 3.
Abgeschlossen	<p>=1: Diese Key-Gruppe in der Datei ist korrekt abgeschlossen und stimmig.</p> <p>=0: Diese Keygruppe ist nicht korrekt abgeschlossen. Das kann während einer laufenden Messung passieren, wenn schon Messdaten eingetragen wurden, aber der Header noch nicht komplett aktualisiert wurde. Das kann auch passieren, wenn beim Sichern einer Datei der PC abgeschaltet wird.</p> <p>Technik: Datei wird erst mit Abgeschlossen=0 erzeugt. Wenn die Datei dann komplett geschrieben ist, wird Abgeschlossen=1 gesetzt.</p>

NO,1,KeyLang,Ursprung,NameLang,Name, KommLang,Kommentar;	
	Herkunft der Daten. Falls dieser Key überhaupt in der Datei vorhanden ist, muss er gleich hinter dem CK-key stehen.
Ursprung	0= Original, 1= verrechnet
NameLang Name	Name des Erzeugers, z.B IMC-FAMOS
KommLang Kommentar	Datei-Kommentar

CB,1,KeyLang,IndexGruppe,NameLang,Name,KommLang,Kommentar;	
	Gruppen-Definition. Kann mehrmals mit verschiedenem Index hintereinander stehen. Eine Variablen-Gruppe wird definiert.
IndexGruppe	Index der Gruppe, beginnend mit 1 beim ersten CB-Key in der Datei.
NameLang, Name	Name der Datengruppe
KommLang, Kommentar	Kommentar zur Datengruppe

CT,1,KeyLang,IndexGruppe, NameLang,Name,TextLang,Text, KommLang,Kommentar;		
	Definition eines Textes	
	IndexGruppe	0: Keiner Gruppe zugeordnet. 1..n: Index der übergeordneten Gruppe, siehe CB-Key
	NameLang Name	Kennung, Titel des Textes
	TextLang Text	Inhalt des Textes
	KommLang Kommentar	Kommentar zum Text

CG,1,KeyLang,AnzahlKomponenten,Feldtyp,Dimension;		
	Definition eines Datenfeldes, Beginn einer Gruppe von zusammengehörigen Komponenten. Achtung. Das ist keine FAMOS-Gruppe (siehe CB-key!), sondern nur eine Zusammenfassung innerhalb de Datei, was hier auch als Gruppe bezeichnet wird.	
	AnzahlKomponenten	Die in dieser Gruppe aufgelisteten Komponenten: Anzahl der folgenden CC-Keys innerhalb dieser Gruppe. Bei digitalen Daten gelten alle Bits zusammen als eine Komponente (also 16 Bit = 1 Komponente). >= Dimension
	FeldTyp	= 1: reelle Zahl, reeller äquidistanter Datensatz, können auch mehrere sein. Auch Timestamp Ascii. = 2: XY, x monoton steigend, (mehrere) reelle Zeitdaten mit gemeinsamer Zeitspur, für Zeitspur Typ along nutzen! = 3: 3 XY, Kennlinie, evtl. mehrere y zu einer x oder mehrere x zu einer y. Keine Anforderungen an Monotonie. Ab hier nur ein Paar, nicht mehrere wie oben! = 4: Komplex, Real-, Imaginärteil = 5: Komplex, Betrag, Phase = 6: Komplex, Betrag in dB, Phase
	Dimension	Je nach Feldtyp = 1: für Typ 1 = 2: für Typ 2 .. 6

CD,1,KeyLang,dx,kalibriert,EinheitLang, Einheit, 0,0,0;		
	Ältere Variante für Standard-Anwendungen. Entweder CD,1 oder CD,2 nutzen. Skalierung, gültig für alle Komponenten des Feldes. (x-Achse bei einkomponentigen Daten, Parameter bei zweikomponentigen Komponenten)	
	dx	Abstand zwischen 2 Samples, bei >1 Komponente für Parameter
	kalibriert	= 1: die Zeitbasis ist kalibriert. = 0: nicht kalibriert
	EinheitLang Einheit	Einheit dieser Achse

CD,2,KeyLang,dx,kalibriert,EinheitLang, Einheit, Reduktion, IsMultiEvents, SortiereBuffer, x0, PretriggerVerwendung;	
	Allgemeinere Variante für kompliziertere Datentypen. Skalierung, gültig für alle Komponenten des Feldes. (x-Achse bei einkomponentigen Daten, Parameter bei zweikomponentigen Komponenten)
dx	Abstand zwischen 2 Samples, bei mehr als 1 Komponente Abstand für Parameter
kalibriert	= 1: die Zeitbasis ist kalibriert. = 0: nicht kalibriert
EinheitLang Einheit	Einheit dieser Achse
x0	x0 der Daten in FAMOS. Bei äquidistanten Daten x-Koordinate des ersten Messwertes, also relative Pretriggerzeit, Startfrequenz. Bei XY- und komplexen Daten Startwert des Parameters. Das feste x0 für alle Messungen. Vor allem bei 2komponentigen Daten. Parameter0 und speziell bei FFT das f0. Das ist dann nicht der Pretrigger. Bedeutung je nach PretriggerVerwendung. Falls nicht benötigt, auf 0 setzen.
PretriggerVerwendung	PretriggerVerwendung (Wie wird der Pretrigger aus Cb verwendet?). Bei Multischuss-Daten kann auch im Cb-key der Pretrigger für jedes Triggerereignis stehen. Ob dieser Eintrag aus Cb benutzt wird oder das x0 aus CD,2, entscheidet dieser Eintrag:: 0: keine (z. B. bei Matrix) x0 wird in x0 des Datensatzes kopiert. Bei Rainflow-Matrix. 1: Pretrigger aus Cb in x0 kopieren. Normale 1komponentige Zeitdaten. (auch individuell pro Event). Bei normalen Zeitdaten. Das ist der Normalfall, wenn man selbst Dateien schreibt. Dann ist dieses x0 = 0 und der Pretrigger kommt in das x0 von Cb. 2: Bei XY-Daten als Offset für die Zeitspur, die dann integer ist, also mit Faktor und Offset skalierbar ist (auch individuell pro Event). x0 wird in Datensatz->x0 kopiert. Bei transitional recording 3: In z0 bei Multi-FFT (auch individuell pro Event). x0 wird in z0 des Datensatzes kopiert. 4: Bei Time Stamp Ascii Daten als Offset für die Zeitspur, die dann integer ist, also mit Faktor und Offset skalierbar ist (auch individuell pro Event)

INT,1,KeyLang,Tag,Monat,Jahr,Stunden,Minuten,Sekunden;

Triggerzeit

Die Sekunden sind eine reelle Zahl, alle anderen ganze Zahlen.

Tag 1 .. 31

Monat 1..12

Jahr >= 1980

Stunden 0 .. 23

Minuten 0 .. 59

Sekunden >= 0.0 .. <= 60.0 mit max. 7 Nachkommastellen, =60 wegen Rundungsfehlern zugelassen

|NT,2,KeyLang,Tag,Monat,Jahr,Stunden,Minuten,Sekunden,Zeitzone,Sommerzeit;

Triggerzeit

Die Sekunden sind eine reelle Zahl, alle anderen ganze Zahlen. Die angegebene Zeit ist die lokale Zeit

Tag 1 .. 31

Monat 1..12

Jahr >= 1980

Stunden 0 .. 23

Minuten 0 .. 59

Sekunden >= 0.0 .. <= 60.0 mit max. 7 Nachkommastellen, =60 wegen Rundungsfehlern zugelassen

Zeitzone: Zeitversatz zur UTC; Angabe in Minuten

Lokale_Zeit = UTC_Zeit + Zeitzone

Beispiel:

= 60 (UTC+1h) für Deutschland bei Standard-Zeit,

= 120 für Deutschland bei Sommerzeit

= -360 (UTC-6h) für New York

Sommerzeit:

0: Es ist nicht bekannt, ob die angegebene lokale Zeit Sommer- oder Winterzeit ist

1: Die angegebene lokale Zeit ist Standardzeit

2: Die angegebene lokale Zeit ist Sommerzeit

|CZ,1,KeyLang,dz,dzkali,z0,z0kali,EinheitLang,Einheit, SegmentLang;

Skalierung der z-Achse bei Segmenten

Immer vorhanden bei Segmenten (2dim-Matrizen)

Der CZ-key steht bei einfachen Datentypen i. a. nicht in der Datei.

dz

Abstand zwischen 2 Segmenten. > 0.0. Z-Koordinate einer Zeile einer Matrix ist $z = z_0 + i * dz$, $i \geq 0$.

dzkali

Ist dz kalibriert ? 1 (ja, der Standard), 0 (nein)

z0

Offset der z-Koordinate. Z-Koordinate der ersten Zeile

z0kali

Ist z0 kalibriert ? 1 (ja, der Standard), 0 (nein)

EinheitLang, Einheit

Einheit der Daten in z-Richtung

SegmentLang

Anzahl der Samples in 1 Segment (1 Zeile), ist fest für die ganze Gruppe. Wenn > 0, dann Daten segmentiert. Wenn = 0, dann nicht. Dann gilt trotzdem die z-Koordinate.

|CC,1,KeyLang,KomponentenIndex,AnalogDigital;

Start einer Komponente (component)

KomponentenIndex

= 1: reell
Einzelwert
Realteil
Betrag
Betrag in dB

		y von XY y von XY, x monoton Timestamp Ascii = 2 : x von XY, x monoton x von XY Imaginärteil Phase
	AnalogDigital	= 1 : analog und Timestamp Ascii = 2 : digital

CP,1,KeyLang,BufferReferenz,Bytes,Zahlenformat,SignBits, Maske,Offset, DirekteFolgeAnzahl,AbstandBytes;		
	Pack-Information zu dieser Komponente	
	BufferReferenz	Eine Nummer > 0, die die zugehörige Buffer-Beschreibung im Cb-Key identifiziert. Alle Buffer, die diese Nummer haben, enthalten Daten zu diesem Kanal.
	Bytes	Bytes für einem Messwert (1..8)
	Zahlenformat	1 unsigned byte 2 signed byte 3 unsigned short 4 signed short 5 unsigned long 6 signed long 7 float 8 double 9 imc Devices Transitional Recording 10 Timestamp Ascii 11 2-Byte-Word digital 13 6byte unsigned long: wie unsigned long, aber 6 statt 4 Bytes. MS Byte ebenfalls an höchster Adresse. Verwendung nur für streng monoton wachsende Zeitspuren
	SignBits	Anzahl der signifikanten Bits, 0 < SignBits <= Bits in Zahl
	Maske	Standard = 0 Bei digitalen (binäre 0 oder 1) Daten auch = 0. Sonst bei analogen Daten eine Maske mit allen ungültigen, nicht zu beachtenden Bits. Beispielsweise 3, wenn die beiden untersten Bits zu ignorieren (herauszushiften) sind. Maske wird dezimal ohne Vorzeichen angegeben (0.. 65534). Mindestens 1 Bit muss gültig sein.
	Offset	Offset des ersten Samples im binären Datenblock, >= 0. Der Offset zählt ab dem ersten gültigen Messwert im Buffer. Der Offset ist ein logischer Offset, also auch im Ringbuffer bei nach vorn überlaufenden Daten immer >= 0.
	DirekteFolgeAnzahl	So viele Samples von diesem Kanal stehen direkt hintereinander. = 1 wenn keine anderen Kanäle zwischendurch. Immer >= 1.
	AbstandBytes	So viele Bytes Abstand vom letzten Sample dieses Kanals bis zum nächsten. Abstand zwischen den Sample-Gruppen = 0 wenn keine anderen Kanäle zwischendurch. Immer >= 0. Bei Daten mit AbstandBytes > 0 (also mit Datenpaketen) muss in den entsprechenden Buffern bzw. im Sample-Key ausreichend Platz für eine ganze Anzahl von Paketen sein. Auch wenn das letzte Paket unvollständig ist (die Daten am Ende fehlen), muss ausreichend Platz dafür vorhanden sein.

Cb,1,KeyLang,AnzahlBufferInKey,BytesInUserInfo, BufferReferenz, IndexSamplesKey, OffsetBufferInSamplesKey, BufferLangBytes, OffsetFirstSampleInBuffer, BufferFilledBytes, 0, x0, AddZeit, UserInfo, NeuesEvent, [BufferReferenz, ...];		
	Buffer-Beschreibung, Mehrere Buffer können in diesem Key aufgelistet sein. Jeder Buffer kann als Ringbuffer organisiert sein. Bei Ringbuffer darf kein Messwert auseinander gebrochen werden (am Ende des Buffer darf also nicht 1 Byte einer 2Byte-Zahl stehen und das nächste Byte am Anfang des Buffers).	
	AnzahlBufferInKey	So viele Buffer werden in diesem Key beschrieben. ≥ 0 .
	BytesInUserInfo	So lang ist die UserInfo.
	Die folgenden Angaben wiederholen sich für jeden Buffer!	
	BufferReferenz	Eine Nummer > 0 , die eine Folge von Buffern für den CP-Key identifiziert. Alle Buffer, die Messdaten zu denselben Kanälen enthalten, haben dieselbe Buffer-Referenz.
	IndexSampleKey	Im angegebenen CS-Key sind die Daten vorhanden
	OffsetBufferInSamplesKey:	Ab dem Beginn des CS-Keys sind so viele Bytes zu überlesen, bis der Buffer beginnt. Datentyp "große ganze Zahl"
	BufferLangBytes:	So viele Bytes ist der Buffer lang, ≥ 0 . Im zugehörigen CS-Key muss ausreichend Platz sein!
	OffsetFirstSampleInBuffer	Der Offset des ersten Messwertes im Buffer, die Bytes des vom Start des Buffers an gezählt. Bei einem Ringbuffer ist $\text{OffsetFirstSampleInBuffer} \geq 0$ und $< \text{BufferLangBytes}$, sonst gleich 0. Datentyp "große ganze Zahl"
	BufferFilledBytes	So viele gültige Bytes stehen im Buffer. $0 \leq \text{BufferFilledBytes} \leq \text{BufferLangBytes}$
	x0	Bei Beginn eines (neuen) Triggerereignisses wird diese Zeit als Zeitoffset gegenüber dem Trigger für diesen Schuss (Triggerereignis) verstanden. > 0 Posttrigger, < 0 Pretrigger. = 0 erlaubt. Angaben in der Einheit entsprechend CD-key. Bei CD,1 wird x0 immer benutzt (und kommt in x0 des Datensatzes), bei CD,2 nur entsprechend PretriggerVerwendung in CD,2
	AddZeit	Im Normalfall = 0. Nur bei mehreren Triggerereignissen: Bei Beginn eines Triggerereignisses wird diese Zeit zu der Zeit aus NT dazuaddiert, um die absolute Triggerzeit für diesen Schuss zu erhalten. Angabe als reelle Zahl, aber immer wird die Zahl in Sekunden verstanden.
	UserInfo.	Binäre Info für den Ersteller der Datei.
	Neues Event	0: kein neues Event. Die Messwerte werden direkt hinter die Messwerte des vorangegangenen Buffers derselben Nummer aufgenommen. 1: neues Event, AddX0, AddZeit gelten. Ein neues Ereignis von Messdaten.

 CR,1,KeyLang,Transformieren,Faktor,Offset,Kalibriert,EinheitLang, Einheit;		
	Wertebereich der Komponente, nur bei analogen, nicht bei digitalen Daten. Bei Timestamp Ascii wird hiermit die Zeitspur skaliert.	
	Transformieren :	= 1 : ja, entsprechend Faktor und Offset, nur bei ganzzahligen Rohdaten. = 0 : nein, evtl. bei ganzzahligen, aber immer bei reellen Rohdaten.
	Faktor Offset	Für die Transformation gilt: $\text{PhysikalDaten} = \text{Rohdaten} * \text{Faktor} + \text{Offset}$
	Kalibriert	Skala ist kalibriert bei 1, nicht kal. bei 0.
	EinheitLang Einheit	Einheit dieser Achse

 ND,1,KeyLang,FarbeR,FarbeG,FarbeB,yMin,yMax;		
	Display-Eigenschaften	
	FarbeR FarbeG FarbeB	Rot-Grün,Blau-Anteil, Bereich 0...255, 0 Minimum, 255 Maximum. (0,0,0) ist schwarz. (255,255,255) ist weiß. Die Farbe, in der die Daten dargestellt werden sollen. Achtung bei unterschiedlichen Ausgabegeräten. Wenn alle auf -1 gesetzt sind, dann ist keine feste Farbe mit diesem Kanal verbunden. Die Farbe wird dann je nach Ausgabegerät gewählt.
	yMin,yMax:	Beide auf 0, falls kein besonderer Darstellungsbereich gewünscht ist. Sonst reelle Zahlen mit $yMin < yMax$, die bei y-Achsen (nur bei reell, XY, Betrag von komplexen) zur festen Skalierung benutzt werden, falls dieser Datensatz als Kurvenverlauf dargestellt wird.

 CN,1,KeyLang,IndexGruppe,0,IndexBit,NameLang,Name,KommLang,Kommentar;		
	Name, Kommentar des Kanals. Bei digitalen Daten für jedes Bit einmal Genau eine von 2 zusammengehörenden Komponenten hat einen Namen.	
	IndexGruppe	0 : Keiner Gruppe zugeordnet. 1..n : Index der übergeordneten Gruppe, siehe CB-Key
	IndexBit	= 0 bei analogen, bei digital = 1 für LSB ... 16 für MSB.
	NameLang Name	Kennung, Titel
	KommLang Kommentar	Kommentar

 CS,1,KeyLang,Index,Rohdaten; Für KeyLang wird der Datentyp "große ganze Zahl" benutzt.		
	Samples, Rohdaten	
	Index	Beginnend mit 1 beim ersten solchen Key in der Datei.
	Rohdaten	Binärer Datenblock. Die Datenmenge kann den Bereich einer 32 bit ganzen Zahl überschreiten.

 NU,1,KeyLang,KennLang,Kennwort,Daten;		
	Benutzerdefinierter Key.	
	KennLang Kennwort	Identifikator für die User-Daten, systemweit eindeutig zu halten.
	Daten	Binärer Block, beliebig, auch ASCII für Parameter im DSF- Style.

[Cl,1,KeyLang, IndexBlockKey, Zahlenformat, NameLang, Name, Wert, EinheitLang, Einheit, KommentarLang, Kommentar, Zeit;		
	Numerischer Wert, Einzelwert, Zahlenwert mit Einheit	
	IndexBlockKey	Index auf eine Block-Definition, =0 kein Block
	Zahlenformat	1 unsigned byte 2 signed byte 3 unsigned short 4 signed short 5 unsigned long 6 signed long 7 float 8 double 11 LSB in 2byte-Word digital 13 6byte unsigned long
	NameLang	Länge des folgenden Namens
	Name	Kennung, Titel
	Wert	Zahlenwert binär, genauso viele Bytes lang wie Zahlenformat festlegt.
	EinheitLang	Physikalische Einheit
	KommentarLang	Länge des folgenden Kommentars
	Zeit	Entstehungszeit in Sekunden seit 1980, mit Nachkommastellen möglich. Binär, stets 8 Byte double

[Np,1,KeyLang, Property[,Property...];		
	Property: Eine Eigenschaft zu einem Kanal. Form: "PropertyName" "PropertyWert" Datentyp Flags	
	PropertyName:	in Anführungszeichen(""). Falls "-Zeichen innerhalb auftaucht, ist es verdoppelt.
	PropertyWert	in Anführungszeichen(""). Falls "-Zeichen innerhalb auftaucht, ist es verdoppelt.
	Datentyp:	0 String 1 Integer 2 Reell 3 Zeitstempel im DM-Format, Sekunden ab 1.1.1980 4 Aufzählung 5 Bool
	Flags:	2 hide in Editoren 4 read-only in Editoren Die Flags können kombiniert werden (2+4 -> 6)

|Ca,1,KeyLang, AddRef,p1;

Add-Referenz-Key. Ab hier wird jeder gefundenen Referenz (z.B. Buffer-Referenz, IndexSamplesKey, IndexEventListeKey, Index-Block-Key ...) ein Inkrement dazuaddiert. Referenzen gelten auch über diesen Key hinaus. Dieser Key darf hinter dem korrekten Dateianfang überall stehen.

Wird nie dazuaddiert bei

IndexBlockKey = 0 in |CT

KomponentenIndex in |CC

IndexBlockKey = 0 in |CN

IndexMarkerListeKey = 0 in |CN

IndexBit in |CN

AddRef:

Ganze Zahl ≥ 0 , die in allen folgenden Keys den Referenzen hinzuaddiert wird.

p1:

Reserviert= 0

|NE,1,KeyLang, Daten;

Extraktionsvorschrift für Kanäle aus protokollierten Bus-Nachrichten.

Ist einem Timestamp Ascii Kanal zugeordnet. Nur einmal pro Datei. Also nur für Dateien, die maximal 1 Timestamp Ascii Kanal enthalten.

Daten :

Komprimierter binärer Block mit der Vorschrift. Deutung/ Erzeugung ist nur imc intern.

5.14.4 Reihenfolge der Keys

Nach CF			
	muss CK		
	kann NO		
		können CT (Texte),	
		 CB (Gruppendefinitionen),	
		 CI (Einzelwert),	
		 CG (Datenfelder) beliebig hintereinander folgen	
		Nach CG	
			kann: CD (wenn für die gesamte Gruppe gleich)
			kann: NT (wenn für die gesamte Gruppe gleich)
			kann: CZ (wenn für die gesamte Gruppe gleich und es Matrix-Daten sind)
			muss: (das folgende Schema kann sich mehrmals wiederholen)
			 CC
			können CD, NT, CZ die dann für diesen CC und alle folgenden in dieser Gruppe gelten, solange bis wieder undefiniert wird.
			muss CP
			kann CR , (nur bei analogen)
			kann ND
			kann CN , bei digitalen mehrmals hintereinander.
			Die Definition einer Komponente wird mit CS, CC, CG, CT, CB abgeschlossen.
		Weitere CG, CB, CT, CI	

Beispiel

Datei mit einem einzelnen Wert ("Mittelwert", Einheit "V" und Kommentar) und einer Textvariablen ("TxBearbeiter").

```
|CF,2,1,1;|CK,1,3,1,1;
|NO,1,16,1,9,imc-FAMOS,0,;
|CG,1,5,1,1,1;
|CD,1,31,1.0000000000000000E+0,1,0,,0,0,0;
|NT,1,27, 3,11,1995,21,10,12.00000000;
|CC,1,3,1,1;
|CP,1,16,1,8,8,64,0,0,1,0;
|Cb,1,41,1,0,1,1,0,8,0,8,1,0.0000000000000000E+0,0,;
|CR,1,11,0,0,0,1,1,V;
|CN,1,36,0,0,0,10,Mittelwert,13,Meßreihe 0815;
|CT,1,48,0,12,TxBearbeiter,24,E.Mustermann, 23.10.1995,0,;
|CS,1,10,1,...ëQ,î^@;
```

Beispiel

Die Datei enthält eine Datengruppe ("Messung1"), die aus 2 Kanälen ("kanal1","kanal2") der Länge 3 besteht. Es handelt sich um Daten im 1Byte-Integerformat ohne Vorzeichen im Wertebereich 0..10V. Beide Kanäle besitzen eine Abtastzeit von 0.5s und einen Startoffset von 3s. Die y-Einheit ist "V". Die Skalierung für das Kurvenfenster ist mit 0..10V vorgegeben.

```
|CF,2,1,1;|CK,1,3,1,1;
|NO,1,16,1,9,imc-FAMOS,0,;
|CB,1,15,1,8,Messung1,0,;
|CG,1,5,1,1,1;
|CD,1,32,5.0000000000000000E-1,1,1,s,0,0,0;
|NT,1,27, 3,11,1995,21,24, 2.00000000;
|CC,1,3,1,1;
|CP,1,15,1,1,1,8,0,0,1,0;
|Cb,1,41,1,0,1,1,0,3,0,3,1,3.0000000000000000E+0,0,;
|CR,1,49,1,3.921568627450980E-2,0.0000000000000000E+0,1,1,V;
|ND,1,50,-1,-1,-1,0.0000000000000000E+0,1.0000000000000000E+1;
|CN,1,17,1,0,0,6,kanal1,0,;
|CG,1,5,1,1,1;
|CD,1,31,5.0000000000000000E-1,1,0,,0,0,0;
|NT,1,27, 3,11,1995,21,24, 6.00000000;
|CC,1,3,1,1;
|CP,1,15,2,1,1,8,0,0,1,0;
|Cb,1,41,1,0,2,1,3,3,0,3,1,3.0000000000000000E+0,0,;
|CR,1,49,1,3.921568627450980E-2,0.0000000000000000E+0,1,1,V;
|ND,1,50,-1,-1,-1,0.0000000000000000E+0,1.0000000000000000E+1;
|CN,1,17,1,0,0,6,kanal2,0,;
|CS,1,8,1,leclass="Code Example" translate="true">|NO,1,37,1,9,
    imc-FAMOS,20,Erzeugt:E.Mustermann;
|CG,1,5,2,2,2;
|CD,1,31,1.0000000000000000E-1,1,0,,0,0,0;
|NT,1,27,10, 6,1993,19,18,20.00000000;
|CC,1,3,1,1;
|CP,1,16,1,4,7,32,0,0,1,0;
|Cb,1,40,1,0,1,1,0,16,0,16,1,0.0000000000000000E+0,0,;
|CR,1,11,0,0,0,1,1,V;
|CN,1,20,0,0,0,9,E06_6_121,0,;
|CC,1,3,2,1;
|CP,1,16,2,4,5,32,0,0,1,0;
|Cb,1,40,1,0,2,1,16,16,0,16,1,0.0000000000000000E+0,0,;
|CR,1,49,1,1.000000024447218E-1,2.139131538569927E-8,1,1,s;
|CS,1,34,1,^_p@_f@,Û•@H_p@*Hh;
```