

INTERNATIONAL STANDARD

IEEE Std C37.111™

NORME INTERNATIONALE



**Measuring relays and protection equipment –
Part 24: Common format for transient data exchange (COMTRADE) for power
systems**

**Relais de mesure et dispositifs de protection –
Partie 24: Format commun pour l'échange de données transitoires (COMTRADE)
dans les réseaux électriques**



THIS PUBLICATION IS COPYRIGHT PROTECTED
Copyright © 2013 IEEE

All rights reserved. IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by the Institute of Electrical and Electronics Engineers, Inc.

Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the IEC Central Office.

Any questions about IEEE copyright should be addressed to the IEEE. Enquiries about obtaining additional rights to this publication and other information requests should be addressed to the IEC or your local IEC member National Committee.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue
New York, NY 10016-5997
United States of America
stds.info@ieee.org
www.ieee.org

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigendum or an amendment might have been published.

Useful links:

IEC publications search - www.iec.ch/searchpub

The advanced search enables you to find IEC publications by a variety of criteria (reference number, text, technical committee,...).

It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available on-line and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary (IEV) on-line.

Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.



IEEE

IEC 60255-24

Edition 2.0 2013-04

INTERNATIONAL STANDARD

IEEE Std C37.111™

NORME INTERNATIONALE



**Measuring relays and protection equipment –
Part 24: Common format for transient data exchange (COMTRADE) for power
systems**

**Relais de mesure et dispositifs de protection –
Partie 24: Format commun pour l'échange de données transitoires (COMTRADE)
dans les réseaux électriques**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX

XB

ICS 29.120.70

ISBN 978-2-83220-766-6

**Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.**



Documents

Document Sharing Hub

CONTENTS

FOREWORD.....	5
INTRODUCTION.....	7
1 Scope.....	8
2 Normative references	8
3 Terms and definitions	8
4 File and data storage.....	9
4.1 Categories of files	9
4.1.1 General	9
4.1.2 Executable files	10
4.1.3 Text files	10
4.1.4 Data files	10
4.2 Critical/non-critical data.....	10
4.3 Data representation.....	11
4.3.1 General	11
4.3.2 Binary data	11
4.3.3 ASCII data.....	11
4.4 Data field delimiters and lengths.....	11
4.4.1 General	11
4.4.2 Carriage return/line feed delimiter <CR/LF>.....	11
4.4.3 Comma delimiter	12
4.4.4 Field lengths.....	12
4.5 Floating point notation for ASCII data.....	12
4.6 Methods of accessing data in files.....	13
4.6.1 General	13
4.6.2 Random access files.....	13
4.6.3 Sequential files.....	13
4.7 Primary to secondary ratios.....	14
5 COMTRADE files.....	14
5.1 General	14
5.2 Header file (.HDR).....	14
5.3 Configuration file (.CFG)	15
5.4 Data file (.DAT)	15
5.5 Information file (.INF)	15
6 Header file.....	15
6.1 General	15
6.2 Content	16
6.3 Filenames	16
6.4 Format	16
7 Configuration file	16
7.1 General	16
7.2 Content	16
7.3 Filenames	17
7.4 Format	17
7.4.1 General	17

7.4.2	Station name, identification and revision year	17
7.4.3	Number and type of channels	18
7.4.4	Analog channel information	18
7.4.5	Status (digital) channel information.....	20
7.4.6	Line frequency.....	20
7.4.7	Sampling rate information	20
7.4.8	Date/time stamps.....	21
7.4.9	Data file type	22
7.4.10	Time stamp multiplication factor	22
7.4.11	Time information and relationship between local time and UTC	22
7.4.12	Time quality of samples	23
7.5	Missing data in configuration files	24
7.6	Configuration file layout.....	24
8	Data file.....	24
8.1	General	24
8.2	Content	24
8.3	Data filenames	24
8.4	ASCII data file format	25
8.5	Example ASCII data sample	26
8.6	Binary data files	26
8.7	Example of binary data sample.....	28
9	Information file	28
9.1	General	28
9.2	Content	28
9.3	Information file filenames	28
9.4	Information file structure.....	28
9.4.1	General	28
9.4.2	Public sections	29
9.4.3	Private sections.....	29
9.5	File characteristics	29
9.6	Section headings.....	30
9.6.1	Public and private section header name formatting rules	30
9.6.2	Public section header naming examples	30
9.6.3	Private section header naming examples.....	30
9.7	Entry line.....	30
9.7.1	General	30
9.7.2	Comment lines.....	31
9.7.3	Value string	32
9.8	Adding, modifying, and deleting information	32
9.8.1	General	32
9.8.2	Deleting information	32
9.8.3	Adding information.....	32
9.9	Public section header and entry line definitions	32
9.10	Public record information section.....	32
9.10.1	General	32
9.10.2	Section header definition	33
9.10.3	Public record information entry line definition.....	33
9.11	Public event information definition	34

9.11.1	General	34
9.11.2	Section heading definition: [Public Event_Information_#n] <CR/LF>	34
9.11.3	Public event information entry line definition	34
9.12	Public file description section	35
9.12.1	General	35
9.12.2	Section heading definition: [Public File_Description] <CR/LF>	35
9.12.3	Public file description entry line definition	35
9.13	Public analog channel section	36
9.13.1	General	36
9.13.2	Section heading definition: [Public Analog_Channel_#n]	36
9.13.3	Public analog channel entry line definition	36
9.14	Public status channel section	36
9.14.1	General	36
9.14.2	Section heading definition: [Public Status_Channel_#n]	36
9.14.3	Public status channel entry line definition	36
9.15	Sample .INF file	37
10	Single File Format COMTRADE (with CFF extension)	38
Annex A (informative)	Sources and exchange media for time sequence data	40
Annex B (informative)	Data exchange sampling rates	43
Annex C (informative)	Sample file	47
Annex D (informative)	Sample program for sampling frequency conversion	53
Annex E (informative)	Example application of conversion factors	56
Annex F (informative)	Sample COMTRADE file with CFF extension (with ASCII data)	58
Annex G (informative)	Sample COMTRADE file with CFF extension (with binary data)	60
Annex H (informative)	Schema for phasor data using the COMTRADE file standard	61
Bibliography		69
Figure 1	– Example of data sample in ASCII format	26
Figure 2	– Example of data sample in binary format	28
Figure B.1	– Typical signal processing	43
Figure B.2	– DSP solution	44
Figure B.3	– Example of sample rate conversion	44
Table B.1	– Frequencies corresponding to ($f_{LCM} = 384 \times f_{base}$) samples/cycle	45
Table B.2	– Frequencies corresponding to ($f_{LCM} = 3200 \times f_{base}$) samples/cycle	45

INTERNATIONAL ELECTROTECHNICAL COMMISSION

MEASURING RELAYS AND PROTECTION EQUIPMENT –

Part 24: Common format for transient data exchange (COMTRADE) for power systems

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation.

IEEE Standards documents are developed within IEEE Societies and Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of IEEE and serve without compensation. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards. Use of IEEE Standards documents is wholly voluntary. IEEE documents are made available for use subject to important notices and legal disclaimers (see <http://standards.ieee.org/IPR/disclaimers.html> for more information).

IEC collaborates closely with IEEE in accordance with conditions determined by agreement between the two organizations. This Dual Logo International Standard was originally an IEEE standard that was adopted by the IEC and has been jointly revised by the IEC and IEEE under the terms of that agreement.

- 2) The formal decisions of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees. The formal decisions of IEEE on technical matters, once consensus within IEEE Societies and Standards Coordinating Committees has been reached, is determined by a balanced ballot of materially interested parties who indicate interest in reviewing the proposed standard. Final approval of the IEEE standards document is given by the IEEE Standards Association (IEEE-SA) Standards Board.
- 3) IEC/IEEE Publications have the form of recommendations for international use and are accepted by IEC National Committees/IEEE Societies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC/IEEE Publications is accurate, IEC or IEEE cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications (including IEC/IEEE Publications) transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC/IEEE Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC and IEEE do not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC and IEEE are not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or IEEE or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees, or volunteers of IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board, for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC/IEEE Publication or any other IEC or IEEE Publications.
- 8) Attention is drawn to the normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that implementation of this IEC/IEEE Publication may require use of material covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. IEC or IEEE shall not be held responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patent Claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility.

International Standard IEC 60255-24/IEEE Std C37.111 has been jointly revised by the Power System Relaying Committee of the IEEE Power and Energy Society¹ in cooperation with IEC Technical Committee 95: Measuring relays and protection equipment, under the IEC/IEEE Dual Logo Agreement.

This second edition cancels and replaces the first edition published in 2001 and constitutes a technical revision. The main changes with respect to the previous edition are as follows:

- a) The new edition allows single file format (with extension .CFF) in lieu of four separate files.
- b) The single file with .CFF extension contains four sections of information corresponding to .CFG, .INF, .HDR, and .DAT. The DAT section is either in ASCII or Binary.
- c) The following additional data file types are also supported: binary32 (using 4 bytes to represent integer numbers) and float32 (using 4 bytes to represent real numbers).
- d) The configuration (.CFG) file/section has been modified. Four new fields have been added at the end of the .CFG file/section in two separate lines. Two fields represent the time information and the time difference between local and UTC time, and these two fields comprise one line. Another two fields represent the time quality of samples and comprise the last line of the file/section.
- e) Some of the fields in the Configuration (.CFG) file/section have been designated critical instead of non-critical.
- f) The use of Unicode UTF-8 characters has been added. However and because of the extensive use of the terms ASCII and Text throughout this document, any occurrence of these terms also inherently implies Unicode UTF-8.

The text of this standard is based on the following IEC documents:

FDIS	Report on voting
95/308/FDIS	95/311/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

International standards are drafted in accordance with the ISO/IEC Directives, Part 2.

The IEC Technical Committee and IEEE Technical Committee have decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

¹ A list of IEEE participants can be found at the following URL:
http://standards.ieee.org/downloads/C37/C37.111-2013/C37.111-2013_wg-participants.pdf

INTRODUCTION

The increasing use of digital technology in devices such as protection, oscillograph, measurement, and control apparatus in electric power substations has created the potential for accumulating large numbers of digital records of power system transient events. In addition to these sources of digital data, analog and digital power-system simulators may be used to generate digital records. The users of these records are faced with the problem of having to cope with different formats used by each system to generate, store, and transmit records.

MEASURING RELAYS AND PROTECTION EQUIPMENT –

Part 24: Common format for transient data exchange (COMTRADE) for power systems

1 Scope

This International Standard defines a format for files containing transient waveform and event data collected from power systems or power system models. The format is intended to provide an easily interpretable form for use in exchanging data. The standard is for files stored on currently used physical media such as portable external hard drives, USB drives, flash drives, CD, and DVD. It is not a standard for transferring data files over communication networks.

This standard defines a common format for the data files and exchange medium needed for the interchange of various types of fault, test, and simulation data. The rapid evolution and implementation of digital devices for fault and transient data recording and testing in the electric utility industry have generated the need for a standard format for the exchange of time sequence data. These data are being used with various devices to enhance and automate the analysis, testing, evaluation, and simulation of power systems and related protection schemes during fault and disturbance conditions. Since each source of data may use a different proprietary format, a common data format is necessary to facilitate the exchange of such data between applications. This will facilitate the use of proprietary data in diverse applications and allow users of one proprietary system to use digital data from other systems.

2 Normative references

IEEE Std C37.118TM -2005, *IEEE Standard for Synchrophasors for Power Systems*

IEEE Std C37.232TM -2007, *IEEE Recommended Practice for Naming Time Sequence Data Files*

IEEE Std 260.1TM -1993, *IEEE Standard Letter Symbols For Units of Measurement (SI Units, Customary Inch-Pound Units)*

IEEE Std 280TM -1985 (R1996), *IEEE Standard Letter Symbols for Quantities Used in Electrical Science and Electrical Engineering (DOD)*

IEEE Std 754TM -2008, *IEEE Standard for Floating Point Arithmetic*

ISO 80000-1, *Quantities and units – Part 1: General*

3 Terms and definitions

For the purpose of this document the following terms and definitions apply:

3.1

critical data

any data that are necessary for reproduction of the sample data

3.2

non-critical data

any data in the COMTRADE configuration file which are not absolutely necessary for reproduction of the sample data, and some variables provided in the configuration file that may not be relevant to a particular application

3.3

COMTRADE

Common Format for Transient Data Exchange

format of time sequence data generated by various sources for exchange purpose

Note 1 to entry: This note applies to the French language only.

3.4

electro-magnetic transient program

EMTP

programs that produce time sequence data by analyzing mathematical models of the power system, unlike the devices that record actual power system events

Note 1 to entry: Electromagnetic transient simulation programs can provide many different test cases for a relay, because of the use of the case with which the input conditions of the study can be changed.

Note 2 to entry: This note applies to the French language only.

3.5

skew

time difference between sampling of channels within the sample period of a record for an analog-to-digital converter

EXAMPLE: In an eight-channel device with one analog-to-digital (A/D) converter without synchronized sample and held running at a 1 ms sample rate, the first sample will be at the time represented by the **timestamp**; the sample times for successive channels within each sample period could be up to 125 μ s behind each other. In such cases the skew for successive channels will be 0; 125; 250; 375 μ s...; etc.

3.6

time sequence data

TSD

type of electronic data file where each data item in the file corresponds to an instant of time that is identified by an explicit or implicit time tag, such as transient data records, event sequences, and periodic data logs

Note 1 to entry: This note applies to the French language only.

4 File and data storage

4.1 Categories of files

4.1.1 General

Files stored on digital devices and media consist of bytes representing a combination of alphabetic, numeric, symbol, punctuation, and other formatting characters. Depending on the format, a byte, part of a byte, or more than one byte, may be represented by a letter, number, or symbol (e.g., "A," "3," or "+"). There are three general classes of files used on computer systems: executable files, text files, and data files. The use of the file determines the category.

4.1.2 Executable files

Executable files contain a sequence of instructions suitable for processing by a computer. Computer programs are stored as executable files (.EXE). COMTRADE does not define executable files.

4.1.3 Text files

Text files imply data in human-readable form. A text file may be used for control of a computer program if the format is rigidly specified. COMTRADE text files use the character representation specified in ANSI X3.4-1986 [B1]². This is often called “ASCII format” or “text (.TXT) format” by word processor programs. Characters from the Unicode UTF-8 Standard are also allowed. Any occurrence of the terms ASCII or text in this document also inherently implies Unicode UTF-8.

COMTRADE defines one freeform ASCII text file intended for strictly human interpretation, the header file. COMTRADE also defines three files in which the format is rigidly controlled and which are both human- and computer-readable—the configuration file, the information file, and the ASCII form of the data file.

Most word processors can save text files in two or more formats. The text format contains only the characters actually typed, including punctuation and standard formatting characters such as carriage return/line feed. Other formats contain special characters, specific to the particular word processor being used. The text format shall be used for the text files in a COMTRADE record to eliminate word processor-specific characters or codes. Programs intended to read COMTRADE files only require use of the typed characters that most word processor programs can read or print.

If no command exists in the word processor to save the file in this format, an alternative method is to use the print functions to print the text to disk to create the file.

4.1.4 Data files

Data files may contain numeric data, text data, or both. The data may be stored in either binary or ASCII format. Fields within ASCII format data files use defined text separated by commas, or some other common delimiter. As such, they are both human- and machine-readable. Most word processors cannot format, read, or write data files in binary form. However, many spreadsheet and data processing programs can read binary data files, if the format is known. Binary numbers must be processed by application-specific software to be easily interpreted by humans. COMTRADE defines one binary file, the binary form of the data file. Binary data are generally used when large amounts of data are to be stored because this uses less storage space (e.g., three bytes of binary data can represent numbers from 0 to 16 777 215 whereas three bytes of ASCII data can only represent numbers from 0 to 999). ASCII numbers have the advantage of being interpreted by humans and by standard computer hardware and software.

4.2 Critical/non-critical data

Some of the data in the configuration file are not absolutely necessary for reproduction of the sample data, and some variables provided in the configuration file may not be relevant to a particular application. Such data is described as non-critical and may be omitted. However, the position normally occupied by such variables shall be maintained in order to maintain the integrity of the file. If data are described as non-critical in any clause of this standard, the position may be left empty and the corresponding data separator retained following the

² This is a reference to the Bibliography.

preceding data separator with no intervening characters or spaces. Any data that are necessary for reproduction of the sample data are termed critical. If such data are missing, the file may be unusable.

4.3 Data representation

4.3.1 General

Data are stored in files as series of binary digits or bits. Each bit can be either a 1 or a 0. The bits are organized in groups of eight bits called bytes. When a computer reads the data in a file, it reads the data as a series of bytes.

4.3.2 Binary data

The eight bits in a byte can be organized in 256 different combinations. They can be used, therefore, to represent the numbers from 0 to 255. If larger numbers are needed, several bytes can be used to represent a single number. For example, 2 bytes (16 bits) can represent the numbers from 0 to 65 535. When the bytes are interpreted in this fashion, they are known as binary data. Several different formats are in common use for storage of numeric data in binary form. This standard supports three of these formats. The supported formats are 16 and 32 bit integer numbers defined according to the two's complement system (hereinafter, referred to as “binary” and “binary32” data respectively), and 32 bit real numbers defined according to the IEEE Std 754™-2008 (hereinafter, referred to as float32 data). The float32 data type format is intentionally listed in this binary data subclause for convenience even though the format is not a straight binary count.

4.3.3 ASCII data

As an alternative to a byte representing the numbers 0 to 255, a byte can be used to represent 256 different symbols. ASCII is a standard code of symbols that match 128 of the combinations of eight binary bits. For example, the byte 01000001 represents an uppercase “A” while 01100001 represents a lowercase “a.” With 128 different combinations, it is possible to represent all of the keys on the keyboard plus many other special symbols. The remainder of the 256 combinations available from an eight-bit format are used for drawing and other special characters. To represent a number in ASCII format requires one byte for each digit of the number. For example, 4 bytes are needed to represent the number 9 999 in ASCII format. When the bytes are interpreted in this fashion, they are known as ASCII data.

4.4 Data field delimiters and lengths

4.4.1 General

Data fields within a file or within a subset of data in a file shall be separated from the other data fields so that they may be extracted for reading or manipulation. For instance, written text uses a space as a word delimiter. Computer files use a variety of delimiters. In the binary form of COMTRADE data files, the only delimiter is a strict definition of the length and position of each data variable, and a byte count of the position within the file is necessary to determine the limit of any data entry. On the other hand, the ASCII files defined by COMTRADE use the comma and the carriage return/line feed as data separators. This permits the use of variable field lengths, but means that these characters cannot be used within any data entry. Leading spaces or zeroes are allowed in ASCII numeric fields provided the permitted maximum character count is not exceeded.

4.4.2 Carriage return/line feed delimiter <CR/LF>

COMTRADE uses the symbol <CR/LF> to represent a data separator terminating a set of data. The delimiter is the combination of two ASCII formatting characters:

CR = carriage return takes the cursor or insertion point back to the beginning of the current line and is identified by the hexadecimal value 0D.

LF = line feed moves the cursor or insertion point to a new line below the current line and is identified by the hexadecimal value 0A.

The symbols “<” and “>” surrounding the CR/LF are used to delineate the delimiter from the neighbouring text within this standard and are not part of the delimiter.

Historically, operating systems use LF to indicate a new line but not all of them do. Others may use a variety of other characters for indicating new lines. It is important to note that in COMTRADE <CR/LF> is defined as a separator and not as a new line indicator because the main intent is to exchange transient data between users and across operating systems.

4.4.3 Comma delimiter

The comma is used as a delimiter for data entries within the COMTRADE configuration (.CFG), information (.INF), ASCII format data (.DAT), and combined format data (.CFF) files.

4.4.4 Field lengths

Field lengths are specified for many alphabetic or numeric variables in the COMTRADE standard. These limitations were specified to simplify reading lines of data containing many variables. For integer numeric variables, the maximum field length is one character longer than required to hold the maximum value for that field. This extra character space is allowed for a leading minus for signed numbers and to allow the application of simple programming techniques that automatically print the leading space, even for unsigned numbers.

4.5 Floating point notation for ASCII data

Real numbers may be stored in several ways. Numbers of limited range can be entered as a numeric string of ASCII characters with a decimal point. For larger or smaller numbers, any reasonable limit on string length leads to a loss of resolution. In such cases, it is desirable to store the number in a format allowing use of a representation of the significant digits (mantissa) and a multiplier (exponent) format. Spreadsheets and other mathematical programs often use floating point notation to represent such numbers. COMTRADE allows the use of floating point notation (Kreyszig [B6]) to represent real numbers in the .CFG and .DAT files. The terms exponential notation or scientific notation are sometimes used for this form and interpretations of the form vary. Since programs designed to read COMTRADE files must be able to recognize and interpret numbers represented in this format, one single format is defined here. The numbers shall be interpreted and displayed as follows.

A signed floating point value consists of an optional sign (+ or –) and a series of decimal digits containing an optional decimal point, followed by an optional exponent field that contains the character “e” or “E” followed by an optionally signed (+ or –) integer exponent. The exponent is a factor of base 10, so 3E2 means 3 multiplied by 100 (10²) or 300. Correct interpretation of negative numbers and negative exponents requires the inclusion of the negative sign. For positive numbers or exponents the sign is optional and is assumed positive if absent.

The format shall be written as:

[±]d[d][.][d][d][d][d][E[±]d[d][d]]

where

- Square brackets surround any optional item.

- “d” represents any numeral between 0 and 9.
- At least one numeral must appear in the field.
- If the decimal point appears, at least one numeral shall appear to the left and right.
- The character “e” or “E” represents “exponential” with base 10.
 - If the exponential sign appears, it must be followed by at least one numeral
 - The intervening plus/ minus sign is optional if positive, but must be “+” or “–” not “±.”
- The numeric value following “E” must be an integer.

Examples:

Acceptable

1E2 (= 100)

1.23E4 (= 12 300)

0.12345E-5 (= 0.0000012345)

–1.2345E2 (= –123.45)

Unacceptable

.123 (one numeral must precede decimal)

123E (at least one numeral must follow “E”)

±0.123E±4 (plus/minus signs make the value indeterminate)

0.123 E4 (space before “E” not allowed)

4.6 Methods of accessing data in files

4.6.1 General

The two different methods used to access text and data files are sequential or random access. In general, text files are sequential access and data files are either sequential or random access.

4.6.2 Random access files

Data within random access files can be retrieved or stored in any random sequence. The access time for each record is independent of the location of the data. Each data field has a specific address that can be used for reading or writing. COMTRADE does not recommend the use of random access files.

4.6.3 Sequential files

Sequential files are accessed by reading or writing each data field in sequence. Individual data fields have no specific address and their position in the file is relative to the other variables. The exact byte-count position in the file is dependent on the length of the preceding variables. COMTRADE uses sequential files.

4.7 Primary to secondary ratios

The devices used to measure and record events on a high voltage system are not capable of accepting the high voltage and high currents of the power system directly. These devices are built to accept inputs in more manageable and less dangerous levels, termed secondary quantities. Voltage transformers and current transformers [B5] are used to reduce the voltage and current signals on the power system to these lower values. The transformer ratios are chosen so that when the power system is running at the rated or nominal primary value, the secondary value is at the nominal secondary value. The ratio is specified in primary-secondary order, the convention being that the primary is closest to the source of power. Primary ratings are available for all common voltages and load values on the power system. Thus, for a current transformer applied to a feeder and rated at 800:5, the secondary current will be at the nominal 5 A value only when the primary load current is 800 A. Lower values of load result in correspondingly lower values of secondary current.

For three-phase applications, voltage transformers are normally rated in phase-to-phase voltage values rather than phase-to-ground. The output of a voltage transformer rated at 345 kV:120 V will be 120 V phase-to-phase (70 V phase-to-ground) only when the primary system phase-to-phase voltage is 345 kV. The term line-to-line is used interchangeably with phase-to-phase, and similarly line-to-ground instead of phase-to-ground.

5 COMTRADE files

5.1 General

Each COMTRADE record has a set of up to four files associated with it (see Clause 4.). Each of the four files carries a different class of information. The four files are as follows:

- a) header;
- b) configuration;
- c) data; and
- d) information.

All files in the set shall have the same name, differing only by the extensions that indicate the type of files.

Filenames are in the form “name.extension” [B3]. The “name” portion is the title used to identify the record (e.g., FAULT1 or TEST_2). The “extension” portion of the filename is used to identify the type of file and is known as the extension: .HDR for the header file, .CFG for the configuration file, .DAT for data file(s), and .INF for the information file. The filenames should follow IEEE Std C37.232TM-2007. However, users and manufacturers should take appropriate care to restrict the filename length so that the files can be copied using available operating systems and CD/DVD writing technologies.

It is also possible to have all of the four files as separate sections in a single COMTRADE file with extension .CFF. This single file format is described in Clause 10. It must be possible to get the four files mentioned above from the single file or vice-versa by using a conversion program.

5.2 Header file (.HDR)

The header file is an optional ASCII text file created by the originator of the COMTRADE data, typically through the use of a word processor program. The data is intended to be printed and read by the user. The creator of the header file can include any information in any order desired. Examples of information to include are given in 6.2. The header file format is ASCII.

5.3 Configuration file (.CFG)

The configuration file is an ASCII text file intended to be read by a computer program and, therefore, must be saved in a specific format. The configuration file contains information needed by a computer program in order to properly interpret the data (.DAT) file. This information includes items such as sample rates, number of channels, line frequency, channel information, etc.

One field in the first line of the configuration file identifies the year of the COMTRADE standard revision with which the file complies (e.g., 1991, 1999, 2013, etc.). If this field is not present or it is empty, then the file is assumed to comply with the original issue of the standard (1991). The configuration file also contains a field that identifies whether the companion data file is stored in ASCII or binary format. Details of the exact content and format of the configuration file are given in Clause 7.

The configuration file can be created with a word processing program or by a computer program that creates the configuration file from the data that is the source of the transient record. The program that creates the configuration file must save the data in ASCII text file format.

5.4 Data file (.DAT)

The data file contains the value for each input channel for each sample in the record. The number stored for a sample is a scaled version of the value presented to the device that sampled the input waveform. The stored data may be zero-based, or it may have a zero offset. Zero-based data spans from a negative number to a positive number (e.g., –2000 to 2000). Zero-offset numbers are all positive with a positive number chosen to represent zero (e.g., 0 to 4000, with 2000 representing zero). Conversion factors specified in the configuration file defines how to convert the data values to engineering units. The data file also contains a sequence number and time stamp for each set of samples.

In addition to data representing analog inputs, inputs that represent on/off signals are also frequently recorded. These are often referred to as digital inputs, digital channels, digital sub-channels, event inputs, logic inputs, binary inputs, contact inputs, or status inputs. In this standard, this type of input is referred to as a status input. The state of a status input is represented by a number “1” or “0” in the data file.

The data files may be in ASCII, binary, binary32, or float32 format—a field in the configuration files indicates which format is used. A detailed description of the data file format is given in Clause 8.

5.5 Information file (.INF)

The information file is an optional file containing extra information that, in addition to the information required for minimum application of the data set, file originators may wish to make available to users. The format provides for public information that any user can read and use, and private information that may be accessible only to users of a particular class or manufacturer. The information file is described in detail in Clause 9.

6 Header file

6.1 General

The header file is an ASCII text file for the storage of supplementary narrative information, provided for the user to better understand the conditions of the transient record. The header file is not intended to be manipulated by an applications program.

6.2 Content

Examples of information that may be included in the header file are as follows:

- a) description of the power system prior to disturbance;
- b) name of the station;
- c) identification of the line, transformer, reactor, capacitor, or circuit breaker that experienced the transient;
- d) length of the faulted line;
- e) positive and zero-sequence resistance, reactance, and capacitance;
- f) mutual coupling between parallel lines;
- g) locations and ratings of shunt reactors and series capacitors;
- h) nominal voltage ratings of transformer windings, especially the potential and current transformers;
- i) transformer power ratings and winding connections;
- j) parameters of the system behind the nodes where the data was recorded (equivalent positive- and zero-sequence impedance of the sources);
- k) description of how the data was obtained, whether it was obtained at a utility substation or by simulating a system condition on a computer program such as an electro-magnetic transient program (EMTP);
- l) description of the anti-aliasing filters used;
- m) description of analog mimic circuitry; and
- n) the phase sequencing of the inputs.

6.3 Filenames

Header filenames shall have the .HDR extension to distinguish them from the configuration, data, and information files in the same set and to serve as a convention that is easy to remember and identify.

6.4 Format

The header file shall be a freeform ASCII text file of any length.

7 Configuration file

7.1 General

The configuration file is an ASCII text file that provides the information necessary for a human or a computer program to read and interpret the data values in the associated data files. The configuration file is in a predefined, standardized format so that a computer program does not have to be customized for each configuration file.

7.2 Content

The configuration file shall have the following information:

- a) station name, identification of the recording device, and COMTRADE standard revision year;
- b) number and type of channels;
- c) channel names, units, and conversion factors;
- d) line frequency;

- e) sample rate(s) and number of samples at each rate;
- f) date and time of first data point;
- g) date and time of trigger point;
- h) data file type;
- i) time stamp multiplication factor;
- j) time code and local code; and
- k) time quality of the samples.

7.3 Filenames

Configuration filenames shall have the .CFG extension to distinguish them from header, data, and information files in the same set and to serve as a convention that is easy to remember and identify.

7.4 Format

7.4.1 General

The configuration file is an ASCII text file in a standardized format. It must be included with every file set to define the format of the data file.

The file is divided into lines. Each line shall be terminated by a carriage return and line feed. Commas are used to separate fields within a line. The data separator comma is required even if no data is entered into a field. Since commas, carriage returns, and line feeds are used as data separators, they are not legal characters within any field. For example, a channel name such as “Pacific West, Line number two” shall be interpreted as two separate fields. The use of data separators allows the field length to be variable so that leading or padding zeroes or spaces are not required. However, because some programming languages reserve a leading character position for a minus sign, programs intended to read COMTRADE files shall be written to tolerate at least one leading space in fields. The information in each line of the file must be listed in the exact order shown in 7.4.2 to 7.4.12. The lines must appear in the exact order shown in 7.6. Deviations from this format will invalidate the file set.

7.4.2 Station name, identification and revision year

The first line of the configuration file shall contain the station name, the recording device identification, and the COMTRADE standard revision year.

station_name,rec_dev_id,rev_year<CR/LF>

where

- | | |
|---------------------|---|
| station_name | is the name of the substation or the location of the substation or the place where the files have been recorded. Critical, alphanumeric, minimum length = 0 characters, maximum length = 64 characters. |
| rec_dev_id | is the identification number or name of the recording device. Critical, alphanumeric, minimum length = 0 characters, maximum length = 64 characters. |
| rev_year | is the year of the standard revision, e.g. 2013, that identifies the COMTRADE file version. Critical, numeric, minimum length = 4 characters, maximum length = 4 characters. rev_year can only adopt three particular values: 1991, 1999 and 2013, corresponding to the years of revision of the COMTRADE standard. This field shall identify that the file structure differs from the file structure requirement in the IEEE Std C37.111 TM -1999 and IEEE Std C37.111 TM -1991 COMTRADE standard. Absence of the field or an empty field is interpreted to mean that the file complies with the 1991 |

version of the standard.

7.4.3 Number and type of channels

This statement contains the number and type of channels as they occur in each data record in the data file:

TT,##A,##D<CR/LF>

where

- TT** is the total number of channels. Critical, numeric, integer, minimum length = 1 character, maximum length = 6 characters, minimum value = 1, maximum value = 999999. TT must equal the sum of ##A and ##D below.
- ##A** is the number of analog channels followed by identifier A. Critical, alphanumeric, minimum length = 2 characters, maximum length = 7 characters, minimum value = 0A, maximum value = 999999A.
- ##D** is the number of status channels followed by identifier D. Critical, alphanumeric, minimum length = 2 characters, maximum length = 7 characters, minimum value = 0D, maximum value = 999999D.

7.4.4 Analog channel information

This group of lines contains analog channel information. There is one line for each analog channel, the total number of analog channel lines shall equal ##A (see 7.4.3). If the analog channel count = 0, then there are no analog channel information lines. The following format shall be used:

An,ch_id,ph,ccbm,uu,a,b,skew,min,max,primary,secondary,PS<CR/LF>

where

- An** is the analog channel index number. Critical, numeric, integer, minimum length = 1 character, maximum length = 6 characters, minimum value = 1, maximum value = 999999. Leading zeroes or spaces are not required. Sequential counter from 1 to total number of analog channels (##A) without regard to recording device channel number.
- ch_id** is the channel identifier. Critical, alphanumeric, minimum length = 1 character, maximum length = 128 characters.
- ph** is the channel phase identification. Non-critical, alphanumeric, minimum length = 0 characters, maximum length = 2 characters.
- ccbm** is the circuit component being monitored. Non-critical, alphanumeric, minimum length = 0 character maximum length = 64 characters.
- uu** are the channel units (e.g., kV, V, kA, A, A RMS, A Peak). Critical, alphabetic, minimum length = 1 character, maximum length = 32 characters. Units of physical quantities shall use the standard nomenclature or abbreviations specified in IEEE Std 260.1TM–1993 or IEEE Std 280TM–1985 (R1996) or ISO 80000-1. Numeric multipliers shall not be included. Standard multiples such as k (thousands), m (one thousandth), M (millions), etc. may be used. The word “NONE” is to be used for unit-less values.
- a** is the channel multiplier. Critical, real, numeric, minimum length = 1 character, maximum length = 32 characters. Standard floating point notation may be used (Kreyszig [B6]).
- b** is the channel offset adder. Critical, real, numeric, minimum length = 1 character, maximum length = 32 characters. Standard floating point notation may be used

(Kreyszig [B6]).

The channel conversion factor is $ax+b$. The stored data value of x , in the data (.DAT) file, corresponds to a sampled value of $(ax+b)$ in units (uu) specified above. The rules of mathematical parsing are followed such that the data sample “ x ” is multiplied by the gain factor “ a ” and then the offset factor “ b ” is added. Manipulation of the data value by the conversion factor restores the original sampled values. See Annex E for an example.

- skew** is the channel time skew (in μs) from start of sample period. Critical, real number, minimum length = 1 character, maximum length = 32 characters. Standard floating point notation may be used (Kreyszig [B6]).
- The field provides information on time differences between sampling of channels within the sample period of a record. For example, in an eight-channel device with one A/D converter without synchronized sample and held running at a 1 ms sample rate, the first sample will be at the time represented by the **timestamp**; the sample times for successive channels within each sample period could be up to 125 μs behind each other. In such cases the skew for successive channels will be 0; 125; 250; 375...; etc.
- min** is the range minimum data value (lower limit of possible data value range) for data values of this channel. Critical, numeric (integer or real), minimum length = 1 character, maximum length = 13 characters, minimum value = $-3.4028235\text{E}38$, maximum value = $3.4028235\text{E}38$.
- max** is the range maximum data value (upper limit of possible data value range) for data values of this channel. Critical, numeric (integer or real), minimum length = 1 character, maximum length = 13 characters, minimum value = $-3.4028235\text{E}38$, maximum value = $3.4028235\text{E}38$. Note: $\text{max} \geq \text{min}$ always.
- primary** is the channel voltage or current transformer ratio primary factor. Critical, real, numeric, minimum length = 1 character, maximum length = 32 characters.
- secondary** is the channel voltage or current transformer ratio secondary factor. Critical, real, numeric, minimum length = 1 character, maximum length = 32 characters.
- P or S** is the primary or secondary data scaling identifier. The character specifies whether the value received from the channel conversion factor equation $ax+b$ will represent a primary (P) or secondary (S) value. Critical, alphabetic, minimum length = 1 character, maximum length = 1 character. The only valid characters are: p,P,s,S.

The data in the data file, the channel conversion factors, and the channel units can refer to either primary or secondary units. So, a 345 kV to 120 V transformer for a channel in which the units are kV will have the primary factor of 345 and a secondary factor of 0.12 (345, 0.12). The primary or secondary variable (PS) is provided as a means to calculate the equivalent primary or secondary values in applications where the primary or secondary value is desired and the alternate value is provided. If the data originate in an environment that has no primary/secondary relationship such as an analog power system simulator, the primary-secondary ratio shall be set to 1:1. With the determination of the primary (P) or secondary (S) values from the $ax+b$ equation, the user can then determine the values required for analysis or playback.

Value required	Setting of variable PS	
	P (provides primary values)	S (provides secondary values)
Primary	Use value	Multiply by primary value and divide by secondary value
Secondary	Divide by primary value and multiply by secondary value	Use value

7.4.5 Status (digital) channel information

This group of lines contains the status channel information. There is one line for each status channel. The total number of status channel lines shall equal ##D (see 7.4.3). If the status channel count = 0, then there are no status channel information lines. The following format shall be used:

Dn,ch_id,ph,ccbm,y<CR/LF>

where

- Dn** is the status channel index number. Critical, integer, numeric, minimum length = 1 character, maximum length = 6 characters, minimum value = 1, maximum value = 999999. Leading zeroes or spaces are not required. Sequential counter ranging from 1 to total number of status channels (##D) without regard to recording device channel number.
- ch_id** is the channel name. Critical, alphanumeric, minimum length = 1 character, maximum length = 128 characters.
- ph** is the channel phase identification. Non-critical, alphanumeric, minimum length = 0 characters, maximum length = 2 characters.
- ccbm** is the circuit component being monitored. Non-critical, alphanumeric, minimum length = 0 characters, maximum length = 64 characters.
- y** is the normal state of status channel (applies to status channels only), that is, the state of the input when the primary apparatus is in the steady state condition. The normal state of status channel does not carry information regarding the physical representation of the status signal, whether there is a clean contact (open or closed) or a voltage (live or dead). The purpose is to define whether a 1 represents the normal or abnormal state. Critical, integer, numeric, minimum length = 1 character, maximum length = 1 character, the only valid values are 0 or 1.

7.4.6 Line frequency

The line frequency shall be listed on a separate line in the file:

If<CR/LF>

where

- If** is the nominal frequency in Hz (for example, 50, 60, or 16.7 for train applications) of the network or sub-network from which samples have been obtained. Critical, real, numeric, minimum length = 0 characters, maximum length = 32 characters. Standard floating point notation may be used (Kreyszig [B6]).

7.4.7 Sampling rate information

This subclause contains information on the sample rates and the number of data samples at a given rate.

For files with one or multiple predetermined sample rates, the information comprises one line with the total number of sampling rates followed by a line for each sample rate including the number of the last sample at this sample rate. There shall be one line of sample rate and end sample number information for each sampling rate within the data file. For files with continuously variable sample periods, such as event-triggered files, the sample rate information comprises two lines: one line with a zero signifying that there are no fixed sample periods or rates, and a second line including a zero signifying that the sample period is not fixed, and the number of the last sample in the data file.

nrates<CR/LF>
samp,endsamp<CR/LF>

where

- nrates** is the number of sampling rates in the data file. Critical, integer, numeric, minimum length = 1 character, maximum length = 3 characters, minimum value = 0, maximum value = 999.
- samp** is the sample rate in Hertz (Hz). Critical, real, numeric, minimum length = 1 character, maximum length = 32 characters. Standard floating point notation may be used (Kreyszig [B6]).
- endsamp** is the last sample number at the sample rate. Critical, integer, numeric, minimum length = 1 character, maximum length = 10 characters, minimum value = 1, maximum value = 9999999999.

Note that, if **nrates** and **samp** are zero, the **timestamp** in the data file becomes critical and **endsamp** must be set to the number of the last sample in the file. When both the **nrates** and **samp** variable information and the **timestamp** information is available, use of **nrates** and **samp** variables is preferred for precise timing.

7.4.8 Date/time stamps

There are two date/time stamps in the configuration file. The first one is for the time of the first data value in the data file. The second one is for the time of the trigger point. They shall be displayed in the following format:

dd/mm/yyyy,hh:mm:ss.ssssss<CR/LF>

dd/mm/yyyy,hh:mm:ss.ssssss<CR/LF>

where

- dd** is the day of month. Critical, integer, numeric, minimum length = 1 character, maximum length = 2 characters, minimum value = 01, maximum value = 31.
- mm** is the month. Critical, integer, numeric, minimum length = 1 character, maximum length = 2 characters, minimum value = 01, maximum value = 12.
- yyyy** is the year. Critical, integer, numeric, minimum length = 4 characters, maximum length = 4 characters, minimum value = 1900, maximum value = 9999. All 4 characters of the year shall be included.
- The variables **dd**, **mm**, and **yyyy** are grouped together as one field, the numbers being separated by the “slash” character with no intervening spaces.
- hh** is the hour. Critical, integer, numeric, minimum length = 2 characters, maximum length = 2 characters, minimum value = 00, maximum value = 23. All times are to be shown in 24 h format.
- mm** are the minutes. Critical, integer, numeric, minimum length = 2 characters, maximum length = 2 characters, minimum value = 00, maximum value = 59.
- ss.ssssss** are the seconds. Critical, decimal, numeric, resolution = down to 1 ns resolution, minimum length = 9 characters (microseconds), maximum length = 12 characters (nanoseconds), minimum value = 00.000000, maximum value = 59.999999999.

All values for the date and time are to be preceded and padded by zeros, as required. If any data for the time and date stamp is missing, field separator commas/<CR/LF> may follow each other without intervening characters, or the correctly formatted field may be filled with numeric values replaced by zeros.

7.4.9 Data file type

The data file type shall be identified as an ASCII, binary, binary32, or float32 file by the file type identifier in the following format:

ft<CR/LF>

where

ft is the file type. Critical, alphabetic, non-case sensitive, minimum length = 5 characters, maximum length = 8 characters.

7.4.10 Time stamp multiplication factor

This field shall be used as a multiplication factor for the time stamp (**timestamp**) field in the data file(s) to allow for long duration recordings to be stored in COMTRADE format. The time stamp has a base unit of microseconds or nanoseconds depending on the definition of the date/time stamp in the CFG file. The elapsed time from the first data sample in a data file to the sample marked by any time stamp field in that data file is the product of the time stamp for that data sample and the time multiplier in the configuration file (**timestamp*timemult**).

timemult<CR/LF>

where

timemult is the multiplication factor for the time differential (timestamp) field in the data file. Critical, real, numeric, minimum length = 1 character, maximum length = 32 characters. Standard floating point notation may be used (Kreyszig [B6]).

7.4.11 Time information and relationship between local time and UTC

This line contains time zone information for the date/time stamps in 7.4.8 and the location of the recorder. The line is composed of two fields: the time code field and the local code field.

The time code is the same as the time code defined in IEEE Std C37.232TM-2007. The field is used to specify the time difference between local time and UTC (Coordinated Universal Time scale without offset, that is, with neither local time zone or daylight saving offset). The field is restricted to a maximum of six (6) formatted characters. The first character is a sign character and is followed by up to five (5) characters for indicating the time difference (up to two (2) digits for the hours followed by the letter “h” followed by two (2) digits for the minutes). The last three (3) characters are required only when fractional hours are in use. Examples are shown below:

- “-4” means the time difference is minus 4 h (minus means time is behind UTC),
- “+10h30” means the time difference is plus 10 h and 30 min (half hour time zone),
- “-7h15” means the time difference is minus 7 h and 15 min, and
- “0” means the time difference is 0 (local time is UTC).

The time difference reflects whether standard time or daylight savings time was in effect at the time of the recording.

The local code is defined as the time difference between the local time zone of the recording location and UTC. If the recording device is not set to UTC, time code and local code will be the same. However, if the recording device is set to UTC, the fields will be different: local code will provide the local time zone information and the time code will be zero (“0”) irrespective of the location of the recording device. Local code will be zero (“0”) only when the local time zone is UTC.

In addition, there is a special situation in which a COMTRADE file is created by using data from different stations in different time zones, and it is imperative that in such situation the time code be set to UTC and the local code be set to “x”, which means that the local code field is not applicable.

time_code, local_code<CR/LF>

where

time_code is the same as the time code defined in IEEE Std C37.232-2007. Critical, alphanumeric, minimum length = 1 character, maximum length = 6 characters.

local_code is the time difference between the local time zone of the recording location and UTC and is in the same format as **time_code**. Critical, alphanumeric, minimum length = 1 character, maximum length = 6 characters.

7.4.12 Time quality of samples

The time quality of the samples shall be identified by the time quality identifier in the following format:

tmq_code, leapsec<CR/LF>

where

tmq_code is the time quality indicator code of the recording device's clock. It is an indication of synchronization relative to a source and is similar to the time quality indicator code as defined in IEEE Std C37.118™. Critical, hexadecimal, minimum length = 1 character, maximum length = 1 character. The time quality value used shall be the quality at the time of time stamp.

4-bit time quality indicator code

BINARY	HEX	VALUE (worst case accuracy)
1111	F	Fault--clock failure, time not reliable
1011	B	Clock unlocked, time within 10 s
1010	A	Clock unlocked, time within 1 s
1001	9	Clock unlocked, time within 10 ⁻¹ s
1000	8	Clock unlocked, time within 10 ⁻² s
0111	7	Clock unlocked, time within 10 ⁻³ s
0110	6	Clock unlocked, time within 10 ⁻⁴ s
0101	5	Clock unlocked, time within 10 ⁻⁵ s
0100	4	Clock unlocked, time within 10 ⁻⁶ s
0011	3	Clock unlocked, time within 10 ⁻⁷ s
0010	2	Clock unlocked, time within 10 ⁻⁸ s
0001	1	Clock unlocked, time within 10 ⁻⁹ s
0000	0	Normal operation, clock locked

leapsec is the leap second indicator. It indicates that a leap second may have been added or deleted during the recording resulting in either two pieces of data having the same Second of Century time stamp or a missing second. Critical, integer, numeric, minimum length = 1 character, maximum length = 1 character. The only valid values are:

- 3 = time source does not have the capability to address leap second,
- 2 = leap second subtracted in the record,

- 1 = leap second added in the record, and
- 0 = no leap second in the record.

7.5 Missing data in configuration files

The configuration file format provides for the fact that some data may be unavailable. However, it is understood that lack of some critical data can make the file set unusable. Some data are therefore specified as noncritical and some as critical. A lack of critical data in the configuration file renders the file set invalid and as not conforming to the standard. A lack of non-critical data in the configuration file does not render the file non-conforming and does not make the file set unusable. When data are missing, the data separators follow each other with no intervening characters unless otherwise specified elsewhere in this clause. Programs intended to read COMTRADE files shall be written to tolerate data separators immediately following each other with no intervening spaces (null fields).

7.6 Configuration file layout

```
station_name,rec_dev_id,rev_year<CR/LF>
TT,##A,##D<CR/LF>
An,ch_id,ph,ccbm,uu,a,b,skew,min,max,primary,secondary,PS<CR/LF>
An,ch_id,ph,ccbm,uu,a,b,skew,min,max,primary,secondary,PS<CR/LF>
An,ch_id,ph,ccbm,uu,a,b,skew,min,max,primary,secondary,PS<CR/LF>
An,ch_id,ph,ccbm,uu,a,b,skew,min,max,primary,secondary,PS<CR/LF>
Dn,ch_id,ph,ccbm,y<CR/LF>
Dn,ch_id,ph,ccbm,y<CR/LF>
lf<CR/LF>
nrates<CR/LF>
samp,endsamp<CR/LF>
samp,endsamp<CR/LF>
dd/mm/yyyy,hh:mm:ss.ssssss<CR/LF>
dd/mm/yyyy,hh:mm:ss.ssssss<CR/LF>
ft<CR/LF>
timemult<CR/LF>
time_code, local_code<CR/LF>
tmq_code, leapsec<CR/LF>
```

8 Data file

8.1 General

The data file contains the data values that are scaled representations of the sampled event. The data must conform exactly to the format defined in the configuration file so that the data can be read by a computer program. The data file type (**ft**) field defined in the configuration file specifies the file type. For binary data files **ft** is set to binary, binary32, or float32. For ASCII data files **ft** is set to ASCII.

8.2 Content

The data file contains the sample number, time stamp, and data values of each channel for each sample in the file. In ASCII data files, the data for each channel within a sample are separated from the succeeding channel data by a comma. This is commonly called “comma delimited format.” Sequential samples are separated by a <CR/LF> between the last channel data value in a sample and the sample number of the succeeding sample. In binary, binary32, or float32 files, there are no separators between the data for each channel within a sample or between sequential sample periods. No other information is contained in the data file.

8.3 Data filenames

Data filenames shall have the .DAT extension to distinguish them from header, configuration, and information files in the same set and to serve as a convention that is easy to remember

and identify. The filename itself shall be the same for header, configuration, data, and information files to associate all of the files.

Appropriate medium for storage and exchange of data files should be used depending on the file size. It is strongly recommended to use the binary, binary32, or float32 formats for large data files.

8.4 ASCII data file format

The ASCII data file shall be divided into rows and columns. The number of data rows varies with the length of the recording and thus affects the length of the file. Each row shall be divided into $TT+2$ columns where TT is the total number of channels, analog and status, in the recording; the other two columns are for the sample number and time stamp. The number of columns is dependent upon the recording system and also affects the file length. Field lengths specified for ASCII data files are maximum values and are not fixed lengths. All numeric characters, including sign notation, shall fit within the field length limits.

- a) The first column contains the sample number.
- b) The second column is the time stamp for the data of that sample number.
- c) The third set of columns contains the data values that represent analog information.
- d) The fourth set of columns contains the data for the status channels.
- e) The next row (line) begins with the next sample number followed by the next data set.
- f) An ASCII end of file (EOF) marker ("1A" HEX) shall be placed immediately following the carriage return/line feed (<CR/LF>) of the last data row of the file.

Each data sample record shall consist of integers arranged as follows:

n, timestamp, A₁, A₂, ..., A_k, D₁, D₂, ..., D_m

where

- | | |
|--|--|
| n | is the sample number. Critical, integer, numeric, minimum length = 1 character, maximum length = 10 characters, minimum value = 1, maximum value = 9999999999. |
| timestamp | is the time stamp. Non-critical if nrates and samp variables in .CFG file are nonzero, critical if nrates and samp variables in .CFG file are zero. Integer, numeric, minimum length = 1 character, maximum length = 13 characters. Base unit of time is microseconds or nanoseconds depending on the definition of the date/time stamp in the CFG file. The elapsed time from the first data sample in a data file to the sample marked by any time stamp field is the product of the time stamp and the time multiplier in the configuration file (timestamp * timemult). When both the nrates and samp variable information are available and the timestamp information is available, the use of nrates and samp variables is preferred for precise timing. |
| A₁ ... A_k | are the analog channel data values separated by commas. Non-critical, numeric (integer or real), minimum length = 1 character, maximum length = 13 characters, minimum value = -3.4028235E38, maximum value = 3.4028235E38. Missing analog values must be represented by data separators immediately following each other with no spaces (null fields). |
| D₁ ... D_m | are the status channel data values separated by commas. Non-critical, integer, numeric, minimum length = 1 character, maximum length = 1 character. The only valid values are 0 or 1. No provision is made for tagging missing status data and in such cases the field must be set to 1 or to 0. The last data value in a sample shall be terminated with carriage return/line feed. |

8.5 Example ASCII data sample

Figure 1 shows an example data sample as specified in this standard. It has six analog values and six status values. It is taken from Annex C.

5, 667, -760, 1274, 72, 61, -140, -502,0,0,0,0,1,1 <CR/LF>

IEC 918/13

Figure 1 – Example of data sample in ASCII format

8.6 Binary data files

The binary, binary32, and float32 data files use the same basic structure as that used for the ASCII data files, with the exception that status channel data are compacted as described below. The format is sample number, time stamp, data value for each analog channel, and grouped status channel data for each sample in the file. No data separators are used; the data within a binary sample record is not separated by commas and the end of a sample record is not marked by carriage return/line feed characters. The data file is a continuous stream of data. Data translation is determined by sequential position within the file. If any data element is missing or corrupt, the sequence of variables will be lost and the file may be unusable. No provision is made for recovery under these circumstances.

Data are stored in binary format, but for convenience the values are shown in hexadecimal here. The data are not stored as ASCII representations of hexadecimal numbers. In the binary case, when storing a two-byte (16 bit) word, the least significant byte (LSB) of the data is stored first, then the most significant byte (MSB). The two-byte data value “1234” will be stored in “3412” format. In the binary32 and float32 cases, when storing a four-byte (32 bit) word, the least significant byte (LSB) of the word is stored first, then the next to least significant byte, then the next to most significant byte, then the most significant byte (MSB). The four-byte data value “12345678” will be stored in “78563412” format. The bits within a byte are numbered zero (least significant) to seven (most significant).

The sequential data in a binary data file represent the following:

- Sample number and time stamp data are stored in unsigned binary form of four bytes each.
- Analog channel sample data are stored as follows: for binary or binary32 data files the data is stored in two's complement binary format of two or four bytes each. A data value of zero is stored with each one of the bytes set to 00 hexadecimal, and -1 is stored with each one of the bytes set to FF. The maximum positive value is obtained when the most significant bit is set to a 0 and the rest of the bits are each set to a 1, and the maximum negative value is the complement of the maximum positive value. As for the float32 data files, the data is stored according to the IEEE Std 754™-2008. The maximum negative values of the binary, binary32, and float32 formats are reserved to mark missing data.
- Status channel sample data in the binary, binary32, and float32 formats are stored in groups of two bytes for each 16 status channels, with the least significant bit of a word assigned to the smallest input channel number belonging to that group of 16 channels. Thus, bit 0 of status word 1 (S1) is the status of digital input number 1, while bit 1 of status word 2 (S2) is the status of digital input number 18. No provision is made for marking missing status data, but a bit set to 1 or to 0 must be included to maintain the integrity of the word.

The length of the file will vary with the number of channels and the number of samples in the file. The number of bytes required for each scan in the file will be:

$$(A_k \times N) + (2 \times \text{INT}(D_m/16)) + 4 + 4$$

where

- A_k is the number of analog channels,
 N is the number of bytes per sample (two for binary and four for binary32 and float32),
 D_m is the number of status channels,
 $\text{INT}(D_m/16)$ is the number of status channel divided by 16 and rounded up to the next integer, and
 $4 + 4$ represents 4 bytes each for the sample number and the time stamp.

Each data sample record shall consist of numeric values arranged as follows:

n timestamp $A_1 A_2 \dots A_k S_1 S_2 \dots S_m$

where

- n** is the sample number. Critical, integer, numeric, minimum length = 4 bytes, maximum length = 4 bytes, minimum value = 00000001 in hexadecimal, maximum value = FFFFFFFF.
- timestamp** is the time stamp. Non-critical if **nrates** and **samp** variables in .CFG file are nonzero, critical if **nrates** and **samp** variables in .CFG file are zero. Minimum length = 4 bytes, maximum length = 4 bytes, minimum value = 00000000 in hexadecimal, maximum value = FFFFFFFE. Missing time stamp values shall be replaced by placing the value FFFFFFFF in the field to maintain the integrity of the file structure. Base unit of time is microseconds or nanoseconds depending on the definition of the date/time stamp in the CFG file. The elapsed time from the first data sample in a data file to the sample marked by any time stamp field is the product of the time stamp and the time multiplier in the configuration file (**timestamp** * **timemult**). When both the **nrates** and **samp** variable information and the **timestamp** information are available, the use of **nrates** and **samp** variables is preferred for precise timing.
- $A_1 \dots A_k$** are the analog channel data values. Non-critical, numeric (integer or real), length is fixed at 2 bytes for binary data files and 4 bytes for binary32 and float32 data files. Missing analog values must be represented by placing the corresponding maximum negative value in the field.
- $S_1 \dots S_m$** are the status channel data values in 2 bytes (16 bits) for each 16 or part of 16 status channels. Non-critical, integer unsigned binary format, minimum length = 2 bytes, maximum length = 2 bytes, minimum value = 0000 in hexadecimal, maximum value = FFFF. No provision is made for tagging missing status data and in such cases the bit may be set to 1 or to 0.

If the number of status channels is not integrally divisible by 16, the higher channels shall be padded with 0 bits.

Example:

For a set of six status inputs (0,0,0,0,1,1) as shown for the ASCII data file in 8.5.:

- write these status inputs as a binary number (110000), recognizing that the channels are listed low bits first in the ASCII data file;
- then pad the number out to a 16 bit number (0000 0000 0011 0000).
- translate this to a hexadecimal value (00 30).
- the data is then stored in LSB/MSB format (30 00).

8.7 Example of binary data sample

Figure 2 shows an example of a data sample as specified in this standard. It has six analog values and six status values. It is the binary equivalent of the ASCII sample shown in 8.5.

05 00 00 00 9B 02 00 00 08 FD FA 04 48 00 3D 00 74 FF 0A FE 30 00

IEC 919/13

Figure 2 – Example of data sample in binary format

9 Information file

9.1 General

The information file (.INF) is an optional file. The .INF file provides for the exchange of information regarding the event recorded in the COMTRADE record that may enable enhanced manipulation or analysis of the data. This optional information is stored in a separate file to allow full backwards and forwards compatibility between current and future programs that utilize COMTRADE files. Any program reading data from information files shall be able to recognize any public section header, entry, or other data defined in this standard, and take any action in response to that data. Programs not recognizing certain data shall not alter that data in any way.

The file format is similar to the Windows™ .INI file format. Most programming languages now include functions for writing and reading from these files. Many programmers and users are familiar with the structure of these files.

Some of the sections in the information file duplicate information stored in the .CFG configuration file. The .CFG and .DAT files are the primary COMTRADE files and any data for which a variable is defined in either of these two files must be stored in the appropriate file even if duplicated in the .INF information file.

9.2 Content

The information file is an ASCII text file that is in a computer-readable specified format. The file contains both information readable by the general user and information specific to a given class of users which may be unreadable to the general user. These two types of information are classed as public and private, respectively, and reside in separate sections of the file. Data stored in the information file shall be stored in a public section whenever a suitable section is defined. If a suitable predefined public section is not available, a private section may be used. The entries shall conform exactly to the format defined below so that the data can be read by a computer program.

9.3 Information file filenames

Information filenames shall have the .INF extension to distinguish them from header, configuration, and data files of the same set and to serve as a convention that is easy to remember and identify. The filename itself shall be the same as for the header, configuration, and data files with which it is associated.

9.4 Information file structure

9.4.1 General

The information file is divided into sections. Each section consists of a header line followed by a number of entry lines. There is no limit to the number of sections but there shall be at least one section per file. No data shall reside outside of a section. Each section is identified by a unique section header line. All data belong to the nearest section header above it in the file.

Generically the structure is as follows:

- Public Record Information Section Header (information relating to the whole record)
 - Publicly-Defined Record Information Entry Lines
- Public Event Information Section Header (information relating to a particular channel and sample in the record)
 - Publicly-Defined Event Information Entry Lines
- Public File Description Section Header (information equivalent to .CFG file information relating to the whole record)
 - Publicly-Defined File Description Entry Lines
- Public Analog Channel #1 Section Header (information equivalent to .CFG file information relating to the first analog channel in the record)
 - Publicly-Defined Analog Channel Entry Lines
- Public Analog Channel #n Section Header (information relating to the next analog channel in the record, with a new section for each channel, up to the number of analog channels in the record)
 - Publicly-Defined Analog Channel Entry Lines
- Public Status Channel #1 Section Header (information relating to the first status channel in the record)
 - Publicly-Defined Status Channel Entry Lines
- Public Status Channel #n Section Header (information relating to the next status channel in the record, with a new section for each channel, up to the number of status channels in the record)
 - Publicly-Defined Status Channel Entry Lines
- Private Information Header
 - Privately-Defined Record Information Entry Lines
- Private Information Header
 - Privately-Defined Record Information Entry Lines

9.4.2 Public sections

Public sections contain information in a form that can be used by equipment and/or software made by more than one manufacturer. Specific public section entry lines are defined in this document. Each revision of the standard will update public section variables and include any openly distributed private section entries in use at that time.

9.4.3 Private sections

Private sections contain manufacturer-specific information that is only useful with a specific vendor's software or hardware, or that is in a format unique to that manufacturer. Multiple private sections are allowed per manufacturer, and a single information file may contain private sections from several manufacturers. It is anticipated that manufacturers will generate private sections for specific purposes. If two or more manufacturers use similar private sections, a common form of the private section could be approved for use as public sections in future revisions of this standard.

9.5 File characteristics

Information files shall be in ASCII format as defined in 4.1.3, with the following additional limitations:

- a) leading spaces are not allowed on any line;
- b) file shall not include any user-added end of file (EOF) marker, such as "1A" HEX; and
- c) file length shall not exceed 64K.

9.6 Section headings

9.6.1 Public and private section header name formatting rules

The section name is delimited by square brackets. The section name resides alone on a line. No other data shall reside on the same line as the section name. The line is terminated with a <CR/LF>. The section name shall start with a letter character; a number or a symbol shall not be the first character of a section name. The section name must start with the word “Public” or, for private sections, a word clearly representing the organization to which the section belongs, followed by exactly one space, then followed by any number of words identifying the section. Individual words in proprietary company or organization names or trademarks comprising more than one word shall be concatenated by deleting the space between the words, or, to improve readability, by substituting the underline space character “_” for the space.

Section headings after the first section heading shall be separated from the preceding section header or entry lines by an empty line.

Public section headers shall be meaningful to a power systems engineer with limited computer knowledge.

9.6.2 Public section header naming examples

Examples:

Acceptable:

[Public File_Description] <CR/LF>

Unacceptable:

[Public DataSource]<CR/LF> (Leading space)

[DataSource Public]<CR/LF> (Shall begin with word Public)

9.6.3 Private section header naming examples

Examples:

Acceptable:

[Company1 InputRanges]<CR/LF>

[Company2 IsolatorType] <CR/LF>

Unacceptable:

[Company Name Input Ranges] <CR/LF> (Spaces not allowed in owner identifier)

[12] <CR/LF> (Starts with number)

{Bad Section}<CR/LF> (Wrong bracket style)

[Bad Section<CR/LF> (Missing bracket)

[Bad Section] Extra Data=Not Allowed<CR/LF> (Extra text or entries on line after closing bracket)

9.7 Entry line

9.7.1 General

An entry line must start with one word 3 to 32 characters long followed by an equal (=) sign. The first word is the “Entry Name.” The entry name is a description of the function of the value string that follows. It is analogous to the name of a variable or constant in many programming languages. The entry name shall be meaningful when read in conjunction with the section

name. The entry name need not be fully descriptive. The entry name can contain any printable characters with ASCII values between 33 and 127 decimal. The line shall be terminated with a <CR/LF>.

Examples:

Acceptable:

```
[Public File Description] <CR/LF>
Recording_Device_ID=Unit 123<CR/LF>
```

```
[Company2 Calibration] <CR/LF>
Ch1=2044.5, -7, 1<CR/LF>
Ch2=2046.2, 5.3, 1<CR/LF>
Ch3=2042.0, -0.4, -1<CR/LF>
```

Unacceptable:

```
[Company3 Calibration] <CR/LF>
cl33421thv1st=2044.5,-7,1,2046.2,5.3,1,2042.0,-0.4,-1<CR/LF>      (Entry name not meaningful)
Ch 1= 2044.5, -7, 1<CR/LF>                                         (Extra spaces)
[Company3 Device Type] <CR/LF>(No space between new section header and last section)
```

9.7.2 Comment lines

An entry line prefixed with a semicolon is considered a comment line. Such lines are to be skipped by file reading algorithms and are used for comments or to comment out certain entries. The comment lines may be created by users or by a program. Comment lines shall not be used for extensive documentation or explanations, since this increases file size, file read time, and obscures the file structure to human readers.

When section headings are commented out, all entry lines in that section shall also be commented out. Failing to comment out the entry lines in a section where the heading has been commented out would cause any uncommented entry lines in that section to fall under the previous section heading.

Examples:

Acceptable:

```
[Company2 Calibration] <CR/LF>
; Sequence is gain, offset, polarity<CR/LF>
```

```
Ch1=2044.5, -7, 1<CR/LF>
Ch2=2046.2, 5.3, 1<CR/LF>
;Channel 2 replaced 7/16/95<CR/LF>
Ch3=2042.0, -0.4, -1<CR/LF>
```

Unacceptable:

```
:[Company3 Calibration] <CR/LF>      (Section heading commented out leaving orphan data)
Ch 1 = 2044.5, -7, 1<CR/LF>          (Extra spaces)
;This recorder uses 8 bit data and
has
64 channels, test points on the card
are
high impedance and not galvanically
```

isolated. <CR/LF>

(excessive and wrongly-placed documentation)

9.7.3 Value string

The value string is defined as all characters on an entry line from the equal sign to the end-of-line sequence. Value strings can contain one data item or several data items. Multiple data items are separated by commas. Numeric values shall begin immediately after the equal sign or comma delimiter with no leading space. Text strings that include a space after the equal sign or comma delimiter shall include the space as part of the value. For public sections, this information is specified in this standard. For private sections, the data type, format, and number of items per entry line are defined by the user.

9.8 Adding, modifying, and deleting information

9.8.1 General

Because several programs may write to, modify, and read from the .INF file independently, rules governing the deletion and addition of information are needed to reduce the potential for damage from programs operating without human intervention. Deliberate human intervention via user entry fields can be used to add or delete information from any section. However, this can render the information file unfit for the intended application.

9.8.2 Deleting information

A program cannot delete private sections that it did not create, nor may it modify or delete items from those sections. A program cannot delete public sections or items from those sections. However, items in public sections may be modified or items may be added.

9.8.3 Adding information

Any program may add entries to a public section. A program cannot add entries to a private section that it did not create. The format allows an unlimited number of public and private sections, each with an unlimited number of entries.

9.9 Public section header and entry line definitions

This standard specifies some public section headers and entry lines. If a publicly-defined section header is included, all of the defined entry lines for that section shall be included in the order listed. An entry line in which the equal sign “=” is followed by the line terminating <CR/LF> shall be interpreted as a null string (no characters) or a zero numeric value. If no suitable public format is available, new complementary private section definitions may be created restricting use to the originating manufacturer or user. Future revisions of this standard will document those in commonly accepted use at the time of the revision.

9.10 Public record information section

9.10.1 General

This public data section defines the software that writes the file, describes the COMTRADE event, and indicates the number of public event information sections included in the information file.

```
[Public Record_Information] <CR/LF> (Section heading, shall include brackets)
Source=Value<CR/LF>
Record_Information=Value<CR/LF>
Location=Value<CR/LF>
max_current=Value<CR/LF> (Entry lines)
min_current=Value<CR/LF>
```

max_voltage=Value<CR/LF> (Entry lines)
min_voltage=Value<CR/LF>
EventNoteCount=Value<CR/LF>

9.10.2 Section header definition

The following text string is publicly defined as a section heading for parameters applicable to the whole file:

[Public Record_Information]<CR/LF>

9.10.3 Public record information entry line definition

The following public record information entry lines and entry value variables are publicly defined:

Source=Value<CR/LF>

- An optional entry line providing a place for machine-readable text description of the software that was used to write the record. Value is an alphanumeric string with printable ASCII characters and white space; multiple data items are separated by commas. The string is the name and revision level of the program.

Record_Information=Value1,Value2,Value3,Value4<CR/LF>

- An optional entry line providing a place for machine-readable text description of the event. Value is an alphanumeric string with printable ASCII characters and white space; multiple data items are separated by commas for which the following values are publicly defined:

Value1: Fault, Unknown, Misoperation, Close, Trip, Reclose, Power Swing, Simulation.

Value2: AG, BG, CG, ABCG, AB, BC, CA, ABC, or any similar series of phase identifier such as 12N, RS, etc.

Value3: Any other text string not being a variation of one of the above that helps describe the event.

Value4: Any other text string being an identifier for a unique device or type of device (e.g., transmission line, transformer).

Location=Value1, Value2<CR/LF>

- An optional entry for information regarding the location of the fault on a transmission line, if it is known. The following entries are publicly defined:

Value1: A real number representing distance to fault in terms of the following parameters.

Value2: Miles, kilometers, percent of line, percent of setting, Ohms.

max_current=Value<CR/LF>

min_current=Value<CR/LF>

max_voltage=Value<CR/LF>

min_voltage=Value<CR/LF>

- Optional entry lines for recorded minimum and maximum values of voltage and current for the record as a whole. The values are either primary or secondary values as specified by the PS variable in the channel definition using the unit specified in the .CFG file. They differ from the variables **min** and **max** in the .CFG file, which are the maximum possible range or physically limited values. Value is a real number corresponding to the highest (max_value) or lowest (min_value) value to be found in the data file after conversion by

the appropriate channel scaling factors $ax+b$; (see 7.4.4). For currents, Value is in amperes. For voltages, Value is in volts.

EventNoteCount=Value<CR/LF>

- An entry line for the number of Public Event Information sections in the .INF file. It is required only if Event Information sections are included. Value is an integer value equal to the total public event information in the information file. If this number is zero or if the EventNoteCount entry line does not exist, it is assumed that there are no public event information sections to be read.

9.11 Public event information definition

9.11.1 General

This public data section defines notes that are related to a specific event, sample, or channel within a COMTRADE record. This allows specific parts of the record to have data and descriptive text attached and later retrieved.

9.11.2 Section heading definition: [Public Event_Information_#n] <CR/LF>

The section heading is the string “Public Event_Information_#n” with the information number “n” directly appended (no interposing space character allowed). The information number is a positive integer, starting at one, consecutive, and limited to the value of EventNoteCount in the Public Record Information section.

9.11.3 Public event information entry line definition

```
Channel_number=Value<CR/LF>
max_value=Value<CR/LF>
min_value=Value<CR/LF>
max_sample_number=Value<CR/LF>
min_sample_number=Value<CR/LF>
Sample_number_Text#=Value1,Value2<CR/LF>
Sample_number_Text#=Value1,Value2<CR/LF>
```

Data definition:

Where the Sample_number string appears in any of the following entries, Value or Value1 is the COMTRADE record sample number to which the information refers. The Sample_number is the ASCII integer number that will be stored in an ASCII data file; binary files sample numbers shall be converted to ASCII integers before the match is made.

Channel_number

An entry line for the COMTRADE record channel number to which the information refers.

max_value and min_value

- Entry lines for recorded minimum and maximum values of voltage and current for the channel to which the information refers. The values are either primary or secondary values as specified by the PS variable in the channel definition using the unit specified in the .CFG file. They differ from the variables **min** and **max** in the .CFG file, which are the maximum possible range or physically limited values. Value is a real number corresponding to the highest (max_value) or lowest (min_value) value in the channel data after conversion by the appropriate channel scaling factors $ax+b$.

max_sample_number and min_sample_number

- Entry lines for the sample number at which the minimum or maximum recorded value occur. Several instances of this entry are possible.

Sample_number_Text#=Value1,Value2

- Entry lines for text notes on events. # is a sequential count of the number of Text entries, beginning at 1 and limited to 99 (2 characters); Value1 is the sample number as described above; Value2 is any alphanumeric string with printable ASCII characters and white spaces. Hard returns (CR and/or LF) are considered terminating characters and are not allowed within the body of the string.

9.12 Public file description section

9.12.1 General

This public data section defines information that describes the record as a whole and is equivalent to data stored in the .CFG configuration file. The .CFG file is mandatory and the .CFG file containing the appropriate information shall be supplied, even if the configuration information is duplicated in the optional .INF file. This optional duplication of data permits users who use the .INF information file to access the data contained in the .CFG file without opening that file.

9.12.2 Section heading definition: [Public File_Description] <CR/LF>

The section heading is the string “Public File_Description” (no interposing space character allowed). Only one Public File_Description section is allowed per record. The entry lines duplicate the information in the lines of the .CFG file which define the record as a whole. Channel-specific definitions are contained in separate sections. If used, this section must contain an entry line for each variable in the .CFG file, except for variables in the analog and status channel definition lines. The entries for “Value” shall follow the rules for the equivalent data as specified in Clause 7.

9.12.3 Public file description entry line definition

Station_Name=Value
Recording_Device_ID=Value
Revision_Year=Value
Total_Channel_Count=Value
Analog_Channel_Count=Value
Status_Channel_Count=Value
Line_Frequency=Value
Sample_Rate_Count=Value
Sample_Rate_#1=Value

End_Sample_Rate_#1=Value

.
. .

Sample_Rate_#n=Value
End_Sample_Rate_#n=Value
File_Start_Time=Value
Trigger_Time=Value
File_Type=Value
Time_Multiplier=Value

9.13 Public analog channel section

9.13.1 General

This public section defines entry variables for the analog channels of the record and provides information equivalent to that stored in the .CFG configuration file. The .CFG file is mandatory and a .CFG file containing the appropriate information shall be supplied even if the information is duplicated in the optional .INF file. This optional duplication of data permits users who use the .INF file access to the data contained in the .CFG file without opening that file.

9.13.2 Section heading definition: [Public Analog_Channel_#n]

The section heading is the string “Public Analog_Channel_#n” (no interposing space character allowed), where “n” is a number between 1 and the analog channel count for the record. One public channel description section is required for each analog channel of the record. The entry lines duplicate information in the lines of the .CFG file, which pertain to individual analog channels. If used, this section shall contain an entry line for each variable on the analog channel line in the .CFG file. The entries for “Value” shall follow the rules for the equivalent variables as specified in Clause 7.

9.13.3 Public analog channel entry line definition

Channel_ID=Value
Phase_ID=Value
Monitored_Component=Value
Channel_Units=Value
Channel_Multiplier=Value
Channel_Offset=Value
Channel_Skew=Value
Range_Minimum_Limit_Value=Value
Range_Maximum_Limit_Value=Value
Channel_Ratio_Primary=Value
Channel_Ratio_Secondary=Value
Data_Primary_Secondary=Value

9.14 Public status channel section

9.14.1 General

This public section defines entry variables for the status channels of the record and provides information equivalent to that stored in the .CFG configuration file. The .CFG file is mandatory and a .CFG file containing the appropriate information shall be supplied even if the information is duplicated in the optional .INF file. This optional duplication of data permits users who use the .INF file to access the data contained in the .CFG file without opening that file.

9.14.2 Section heading definition: [Public Status_Channel_#n]

The section heading is the string “Public Status_Channel_#n” (no interposing space character allowed), where “n” is a number between 1 and the status channel count for the record. One public channel section is required for each status channel of the record. The entry lines duplicate information in the lines of the .CFG file, which deal with individual status channels. If used, this section shall contain an entry line for each variable on the status channel line in the .CFG file. The entries for “Value” shall follow the rules for the equivalent variables as specified in Clause 7.

9.14.3 Public status channel entry line definition

Channel_ID=Value

Phase_ID=Value
Monitored_Component=Value
Normal_State=Value

9.15 Sample .INF file

```
[Public Record_Information]<CR/LF>
Source=COMwriter, V1.1<CR/LF>
Record_Information=Fault, AG, Trip,Transmission Line<CR/LF>
Location=189.2, miles<CR/LF>
max_current=3405.5<CR/LF>
min_current=-3087.2<CR/LF>
max_voltage=208.6<CR/LF>
min_voltage=-206.4<CR/LF>
EventNoteCount=2<CR/LF>
<CR/LF>
[Public Event_Information_#1] <CR/LF>
Channel_number=2<CR/LF>
max_value=204.5<CR/LF>
min_value=-205.1<CR/LF>
max_sample_number=168<CR/LF>
min_sample_number=15<CR/LF>
Sample_number_Text_#1=168,Transient on reclose<CR/LF>
Sample_number_Text_#2=15,Minimum during normal load <CR/LF>
<CR/LF>
[Public Event_Information_#2] <CR/LF>
Channel_number=1<CR/LF>
max_value=206.5<CR/LF>
min_value=205.1<CR/LF>
max_sample_number=159<CR/LF>
min_sample_number=9<CR/LF>
Sample_number_Text_#1=159,Transient on reclose<CR/LF>
Sample_number_Text_#2=9,Minimum during normal load <CR/LF>
<CR/LF>
[Public File_Description] <CR/LF>
Station_Name=Condie<CR/LF>
Recording_Device_ID=518<CR/LF>
Revision_Year=1999<CR/LF>
Total_Channel_Count=12<CR/LF>
Analog_Channel_Count=6<CR/LF>
Status_Channel_Count=6<CR/LF>
Line_Frequency=60<CR/LF>
Sample_Rate_Count=1<CR/LF>
Sample_Rate_#1=6000.000<CR/LF>
End_Sample_Rate_#1=885<CR/LF>
File_Start_Time=11/07/95,17:38:26.663700<CR/LF>
Trigger_Time=11/07/95,17:38:26.687500 <CR/LF>
File_Type=ASCII <CR/LF>
Time_Multiplier=1<CR/LF>
<CR/LF>
[Public Analog_Channel_#1] <CR/LF>
Channel_ID=Popular Va-g<CR/LF>
Phase_ID=<CR/LF>
Monitored_Component=<CR/LF>
Channel_Units=kV<CR/LF>
Channel_Multiplier=0.14462<CR/LF>
Channel_Offset=0.0000000000<CR/LF>
Channel_Skew=0<CR/LF>
```

```

Range_Minimum_Limit_Value=-2048<CR/LF>
Range_Maximum_Limit_Value=2048<CR/LF>
Channel_Ratio_Primary =2000<CR/LF>
Channel_Ratio_Secondary=1<CR/LF>
Data_Primary_Secondary=P<CR/LF>
<CR/LF>
[Public_Status_Channel_#1] <CR/LF>
Channel_ID=Va over<CR/LF>
Phase_ID=<CR/LF>
Monitored_Component=<CR/LF>
Normal_State=0<CR/LF>
<CR/LF>
[Company1_event_rec] <CR/LF>
recorder_type=1<CR/LF>
trig_set=0,0,0,0,6048,6272,0,0,0,0,0,0,0,0,0,0,0<CR/LF>
ch_type=1,1,1,1,1,1,1,0,0<CR/LF>
<CR/LF>
[Company1_analog_rec_#1] <CR/LF>
op_limit=15<CR/LF>
trg_over_val=f<CR/LF>
trg_under_val=f<CR/LF>
trg_roc=f<CR/LF>
inverted=0<CR/LF>
<CR/LF>

```

10 Single File Format COMTRADE (with CFF extension)

As mentioned in Clause 5, this standard also provides a single file format for COMTRADE. It is strongly recommended to use the single file format described in this clause.

The single file format has many advantages including:

- easier to manage large volumes of COMTRADE records,
- only one file to exchange,
- COMTRADE becoming a standard file for transient records (not just exchange).

The format for the single file (which has the same name as the COMTRADE record but with extension CFF) is merely a collection of the four individual files (.CFG, .INF, .HDR and .DAT as described in Clauses 6 through 9) as separate sections. Each section begins with a separator. The separators are merely used to identify the start of each section. The content of the .CFF file is as follows.

- 1) Line 1 is the first separator indicting the start of the .CFG file contents section.
e.g. --- file type: CFG ---<CR/LF>
- 2) The next lines list the entire contents of the configuration file as per Clause 7.
e.g. SMARTSTATION,IED123,2013<CR/LF>
- 3) The next line is the second separator indicting the start of the .INF file contents section. The end of one section and the beginning of the next section may be separated by multiple <CR/LF> as they need not be continuous.
e.g. --- file type: INF ---<CR/LF>
- 4) The next lines list the entire contents of the information file as per Clause 9. However, there may not be an information section as the information file is optional. In that case, an additional <CR/LF> will be indicated in this section.
e.g. <CR/LF>
- 5) The next line is the third separator indicting the start of the .HDR file contents section.
e.g. --- file type: HDR ---<CR/LF>

- 6) The next lines list the entire contents of the header file as per Clause 6. However, there may not be a header section as the header file is optional. In that case, an additional <CR/LF> will be indicated in this section.
e.g. <CR/LF>
- 7) The next line is the fourth and last separator indicating the start of the .DAT file contents section. This last separator also defines the type of the data file along with the number of bytes in case of BINARY type data.
e.g. --- file type: DAT ASCII ---<CR/LF>, or
e.g. --- file type: DAT BINARY: 702 ---<CR/LF
where, the number 702 indicates the number of bytes in the binary data file.
- 8) The next lines list the entire contents of the data file as per Clause 8.

e.g. 1,72500,-83,68,7,-8,0,0,0,0
 2,73333,-15,5,4,-6,0,0,0,0
 3,74167,55,-53,0,2,0,0,0,0

 40,105000,-169,41,18,-110,1,1,0,1
- 9) The end of the single file information shall be indicated using the end of file marker.
e.g. <EOF>

An example of single format COMTRADE file with CFF file extension is provided in Annex F (with ASCII data) and G (with binary data) respectively. In the case of binary data, actual values are not shown for obvious reasons.

Annex A

(informative)

Sources and exchange media for time sequence data

A.1 General

There are several possible sources of time sequence data that could be converted to the COMTRADE standard for data exchange. Some examples are listed here.

A.2 Digital fault recorders

Digital fault recorders for monitoring power system voltages, currents, and events are supplied by several manufacturers. These devices record analog signals by periodically sampling them and converting the measured signals to digital values. Typical recorders monitor 16 to 128 analog channels and a comparable number of event (contact status) inputs. Sampling rates, analog-to-digital converter resolution, record format, and other parameters have not been standardized.

A.3 Analog tape recorders

Analog tape recorders record analog signals on magnetic tape, usually using frequency modulation techniques. Recorded tapes can be played back to drive oscilloscopes or plotters for visual examination of the recorded waveforms. Typical recorders monitor up to 32 analog signals.

By employing suitable hardware and software, the signals recorded on the analog tapes can be converted to digital records in any desired format. The fidelity of the resultant output is dependent upon the limitations of both the analog recorder and the digital conversion system. The loss in fidelity can be minimized by a proper choice of the sampling system.

A.4 Digital protective relays

New relay designs using microprocessors are currently being developed and marketed. Some of these relays have the ability to capture and store relay input signals in digital form and transmit this data to another device. In performing this function, they are similar to digital fault recorders, except that the nature of the recorded data may be influenced by the needs of the relaying algorithm. As with the digital fault recorders, record format and other parameters have not been standardized.

A.5 Phasor measurement units

Phasor measurement units (PMUs) convert voltage and current waveforms into a phasor equivalent that includes both magnitude and phase angle. These measurements are precisely time synchronized, usually by GPS, for universal comparability. PMUs can also record time synchronized digital status and sampled analog values with the phasor data. Data can be sampled many times in a second; 30 Hz is a typical sampling rate that is used. IEEE Std C37.118™, the Synchrophasor Standard, describes a real-time output format for this data but no format for recording as a file. The IEEE working group report "Schema for Phasor Data using the COMTRADE File Standard" provides a guide for recording synchrophasor data as a file in the COMTRADE file format which is based on IEEE Std C37.111™ -1999. This schema maps data directly from the real-time transmission format to the file format. It can be used for

data from a single PMU or from multiple PMUs through a data concentrator. The following subclauses provide a description of the schema which will be updated in future based on this standard.

A.6 Transient simulation programs

Unlike the above devices that record actual power system events, transient simulation programs produce time sequence data by analyzing mathematical models of the power system. Because this analysis is carried out by a digital computer, the results are inherently in digital form suitable for digital data dissemination. While originally developed for the evaluation of transient overvoltage in power systems, these programs are finding increased usage in other types of studies, including test cases for digital relaying algorithms. Because of the ease with which the input conditions of the study can be changed, transient simulation programs can provide many different test cases for a relay.

A.7 Analog/digital simulators

Analog simulators model power system operations and transient phenomena with scaled values of resistance, inductance, and capacitance while operating at greatly reduced values of voltage and current. The components usually are organized with similar line segments that can be connected to form longer lines. The frequency response of the analog simulator primarily is limited by the equivalent length of the model segment and typically ranges from 1 kHz to 5 kHz. As with the output of analog tape recorders, the analog output of the simulator could be converted to digital records with appropriate filtering and sampling.

Digital simulators model power systems with mathematical equations which are solved either in real-time or in non-real-time to generate transient signals. These transient signals are played to any device connected to the real-time digital simulator in real-time and the data is saved for further analysis. COMTRADE is a preferred format for such storage. In the case of non-real-time digital simulators, transient data are usually saved in COMTRADE format for playing back to devices at a later time. Frequency response of both types of digital simulators can be significantly higher depending on the mathematical model used. In case of real-time digital simulators, frequency response also depends on the available hardware and size of the network modeled.

A.8 Data exchange medium

A.8.1 General

Electric power utilities record fault data for post-fault analysis to determine the nature and location of the fault and to store a record for future use. The data are generally stored as oscillograms on magnetic tapes or paper or in computer data files. An oscillogram contains voltage and current waveforms that can be examined and analyzed. Digital computers cannot record voltage and current waveforms directly. The waveforms are quantified for storage in computer files. More recently, personal computers have been used to record fault data on diskettes and cassettes.

It is not convenient to transport magnetic tapes that are used with mainframe computers in the form of reel-to-reel or cassettes between utilities and individual users. This is especially true if the users are separated by long distances or are located in different countries. Also, the recipient of a magnetic tape must have a computer system compatible with the system on which the tape was prepared. It is more convenient to transport cassettes than to transport magnetic tapes. However, transferring data to and from cassettes is a slow process.

A.8.2 Recommended medium

The most commonly used computer systems today are personal computers equipped with CD, DVD, and USB drives. One of these mediums can be effectively used for exchanging data. However, some other devices may be available in the future which may be more advanced both in terms of amount of data storage capability and the size of the device. Users should adopt the latest available technology that is popular without waiting for the next revision of the standard.

Annex B (informative)

Data exchange sampling rates

B.1 General

This annex is concerned with issues of sampling rates, filters, and sample rate conversions for time sequence data being exchanged. Of special concern is the case in which data are captured at a high sample rate but a lower sample rate is required by the device or software using the data. The simple expedient of dropping every n^{th} sample is **not** the correct way of making the conversion. This section discusses the correct way to perform this common function, as well as other related topics.

Since it is difficult to anticipate all future uses of such standard test cases (e.g., future algorithms, architectures, microprocessors), it seems clear that high accuracy and high sampling rates are desirable in the test cases. Although many existing digital relays use 12 bit accuracy, 16 bit or higher resolution A/D converters may be used in the near future.

The sampling rate issue is similar. Samples obtained at a sampling frequency of 240 Hz, for example, must be obtained using a filter with a cutoff frequency of 120 Hz to avoid aliasing. It is straightforward to convert these samples to samples at higher sampling frequencies, but the effect of the anti-aliasing filter cannot be removed. It is possible to obtain samples at 960 Hz equivalent to the output of the 120 Hz anti-aliasing filter, but it is not possible to obtain samples at 960 Hz of the original (unfiltered) signal.

B.2 Sampling process structure

It is recommended that the original samples be obtained (after a proper anti-aliasing filter is used, if necessary) at as high an accuracy and as high a sampling rate as possible in a given installation. However, specific choices of sampling rates (see sampling rates in Tables B.1 and B.2) could make further use of the data much easier. Consider data obtained at a sampling rate of f_s Hz. It would be most convenient if there were a standard technique to convert from the data at f_s Hz to data that would have been obtained by the user's proposed system shown in Figure B.1.

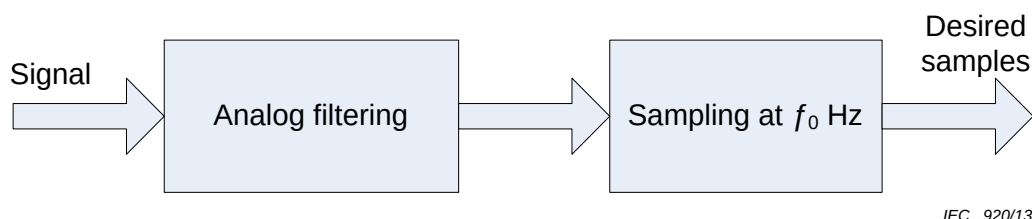


Figure B.1 – Typical signal processing

Developments in digital signal processing present an efficient solution to the problem if there are integers L and M such that

$$Lf_s = Mf_0 = f_{\text{LCM}} \quad (\text{B.1})$$

where

f_{LCM} is the least common multiple. The solution is shown in Figure B.2.