BRITISH STANDARD

BS EN 13757-3:2004

Communication systems for and remote reading of meters —

Part 3: Dedicated application layer

The European Standard EN 13757-3:2004 has the status of a British Standard

ICS 33.200; 35.100.70

British Standards

National foreword

This British Standard is the official English language version of EN 13757-3:2004.

The UK participation in its preparation was entrusted to Technical Committee PEL/894, Remote meter reading, which has the responsibility to:

— aid enquirers to understand the text;

— present to the responsible international/European committee any enquiries on the interpretation, or proposals for change, and keep the UK interests informed;

— monitor related international and European developments and promulgate them in the UK.

A list of organizations represented on this committee can be obtained on request to its secretary.

Cross-references

The British Standards which implement international or European publications referred to in this document may be found in the *BSI Catalogue* under the section entitled "International Standards Correspondence Index", or by using the "Search" facility of the *BSI Electronic Catalogue* or of British Standards Online.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, the EN title page, pages 2 to 55 and a back cover.

The BSI copyright notice displayed in this document indicates when the document was last issued.

Amendments issued since publication

Amd. No.	Date	Comments

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 2 December 2004

ISBN 0 580 44960 2

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 13757-3

November 2004

ICS 33.200; 35.100.70

English version

Communication systems for and remote reading of meters - Part 3: Dedicated application layer

Systèmes de communication et de télérelevé de compteurs - Partie 3: Couches d'application spéciale

Kommunikationssysteme für Zähler und deren Fernablesung - Teil 3: Spezieller Application Layer

This European Standard was approved by CEN on 23 September 2004.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the Central Secretariat or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the Central Secretariat has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: rue de Stassart, 36 B-1050 Brussels

Ref. No. EN 13757-3:2004: E

Contents

Foreword

This document (EN 13757-3:2004) has been prepared by Technical Committee CEN/TC 294 “Communication systems for meters and remote reading of meters”, the secretariat of which is held by AFNOR.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by May 2005, and conflicting national standards shall be withdrawn at the latest by May 2005.

This standard consists of the following parts:

EN 13757-1, *Communication system for meters and remote reading of meters - Part 1: Data exchange.*

EN 13757-2, *Communication systems for and remote reading of meters - Part 2: Physical and link layer.*

EN 13757-3, *Communication systems for and remote reading of meters - Part 3: Dedicated application layer.*

prEN 13757-4, *Communication systems for meters and remote reading of meters - Part 4: Wireless meter readout.*

According to the CEN/CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.

Introduction

This document belongs to a series of parts of EN 13757 which covers communication systems for meters and remote reading of meters. Part 1 contains generic descriptions and a communication protocol. Part 2 contains a physical and a link layer for twisted pair base band (M-Bus). Part 4 (currently an enquiry is under preparation) describes wireless communication.

The bus communication system of EN 1434–3 is commonly called M-Bus. Its application layer describes a standard especially for meter readout.

It can be used with various physical layers and with link layers and network layers which support the transmission of variable length binary transparent telegrams. Frequently the physical and link layers of EN 13757-2 (Twisted pair baseband) and prEN 13757-4 (wireless) or the alternatives described in EN 13757-1 are used.

An overview of communication systems for meters is given in EN 13757-1, which also contains further definitions.

This standard is a compatible enhancement of the 6.4 to 6.6 of the original standard EN 1434–3:1997. Besides some clarifications and implementation hints it contains optional enhancements especially for complex meters. Due to technical progress some variants (Fixed format and mode 2=high byte first) are no longer supported in this standard.

It should be noted that this standard contains only directions how data should be coded. It is beyond the task of an application layer standard to define which data will be transmitted under what conditions by which types of slaves or which data transmitted to a slave will have which reactions. Therefore adherence to this standard guarantees the coexistence and common communication and readout capability of slaves via a universal master software (covering all optional features), but not yet functional or communication interchangeabilty of meters following this standard. For several meter types and meter classes a group of remote heating users have provided such application descriptions required for full interchangeability. They are accessible via the www-server of the m-bus users group http://www.m-bus.com/files/default.html (file name: WG4N99R4.EXE (this is a self expanding .doc-file)).

1 Scope

This document applies to communication systems for meters and remote reading of meters.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 13757-2:2004, *Communication systems for and remote reading of meters - Part 2: Physical and link layer.*

NOTE Further information and examples are available in the download area of http://www.m-bus.com.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1 Table of abbreviations

DES	Data Encryption Standard
DRH	Data Record Header
DIB	Data Information Block
DIF	Data Information Field
DIFE	Data Information Field Extensions
VIB	Value Information Block
VIF	Value Information Field
VIFE	Value Information Field Extensions
RSP_UD	Respond User Data
SND_UD	Send User Data to slave
REQ-UD	Request User Data
MDH	Manufacturer Specific Data Block
CI	Control Information Field
E	Extension Bit

3.2
hexadecimal numbers
hexadecimal numbers are designated by a following "h"

Binary numbers

4 General principles : CI-field

4.1 Overview

All application layer telegrams have variable length. The length information is part of the link layer. It shall be known to the application layer in order to properly terminate its decoding of each telegram. Each telegram starts with a one byte CI (control information) field, which distinguishes between various telegram types and application functions. It is also used to distinguish between true application layer communication and management commands for lower layers. The meaning of the remaining bytes of the telegram depends also on the value of the CI-field.

Table 1 — CI-Field codes used by the master or the slave

	Application
00h to 4Fh	reserved for DLMS-based applications
50h	application reset
51h	data send (master to slave)
52h	selection of slaves
53h	reserved
54h to 58h	reserved for DLMS-based applications
55h to 5Bh	reserved
5Ch	synchronize action
60h to 6Fh	reserved
70h	slave to master: report of application errors
71h	slave to master: report of alarms
72h	slave to master: 12 byte header followed by variable format data
73h to 77h	reserved
78h	slave to master: Variable data format response without header
79h	reserved
7Ah	slave to master: 4 byte header followed by Variable data format response
7Bh to 80h	reserved
81h	reserved for a future CEN–TC294– Radio relaying and application Layer
82h	reserved for a future CENELEC–TC205 network/application Layer
82h to 8Fh	reserved
90h to 97h	manufacturer specific (obsolete)
A0h to AFh	manufacturer specific
B0 to B7h	manufacturer specific
B8h	set baud rate to 300 baud
B9h	set baud rate to 600 baud
BAh	set baud rate to 1 200 baud
BBh	set baud rate to 2 400 baud
BCh	set baud rate to 4 800 baud
BDh	set baud rate to 9 600 baud
BEh	set baud rate to 19 200 baud
BFh	set baud rate to 38 400 baud
C0h to FFh	reserved

Note that the CI-codes 50h, 52h, 5Ch, 70h, 71h, 78h, 7Ah, 80h, 81h, A0h – AFh and B8h – BFh are optional compatible enhancements of the EN 1434-3:1997 standard. Note also that even if the functions of these optional CI-codes are not implemented in a slave the link layer protocol requires a proper link layer acknowledge of SND_UD telegrams containing any of these CI-codes.

The EN 1434–3 defined two possible data sequences in multibyte records. This standard supports only the mode where the least significant byte of a multibyte record is transmitted first.

4.2 Application reset (CI = 50h), (optional)

4.2.1 General

With the CI-Code 50h the master can release a reset of the application layer in the slaves. Each slave himself decides which parameters to change – e.g. which data output is default – after it has received such an application reset.

4.2.2 Application reset subcode (optional)

It is allowed to use optional parameters after CI = 50h. If more bytes follow, the first byte is the application reset subcode. Further bytes are ignored. The application reset subcode defines which telegram function and which subtelegram is requested by the master. The datatype of this parameter is 8 bit binary. The upper 4 bits define the telegram type or telegram application and the lower 4 bits define the number of the subtelegram (the meaning of this number is device specific). The lower four bits may be ignored for slaves which provide only a single telegram for each application. The use of the value zero for the number of the subtelegram means that all telegrams are requested.

Slaves with only one type of telegram may ignore application reset and the added parameters. The following codes can be used for the upper 4 bits of the first parameter:

Table 2 — Coding of the upper four bits of the first parameter after CI = 50h

Coding	Description	Examples
0000b	All	
0001b	User data	consumption
0010b	Simple billing	actual and fixed date values + dates
0011b	Enhanced billing	historic values
0100b	Multi tariff billing	
0101b	Instaneous values	for regulation
0110b	Load management values for management	
0111b	Reserved	
1000b	Installation and startup	bus address, fixed dates
1001b	Testing	high resolution values
1010b	Calibration	
1011b	Manufacturing	
1100b	Development	
1101b	Selftest	
1110b	Reserved	
1111b	Reserved	

Note that this table has been expanded with optional elements from the original standard.

4.3 Master to slave data send (51h) (optional)

The CI-Field code 51h is used to indicate the data send from master to slave:

Variable Data Blocks (Records)	MDH(opt)O	Opt.Mfg.specific data Opt)
variable number	1 Byte	variable number

Figure 1 — Variable Data Structure master to slave

Note that this structure is identical to the slave to master direction (see clause 5) with the exception of the fixed header which is omitted in this direction.

4.4 Slave select (52h) (optional)

The CI-Field code 52h is used for the management of the optional secondary addressing (see 11.3).

4.5 Synchronize action (CI = 5 Ch) (optional)

This CI-code can be used for synchronizing functions in slaves and masters (e.g. clock synchronization). Special actions or parameter loads may be prepared but their final execution is delayed until the reception of such a special CI-field command. No data follows this CI-code.

4.6 Report of application errors (slave to master) (CI = 70h) (optional)

For details of the report of general application errors see 8.2. For error reporting of individual data elements see 8.3.

4.7 Report of alarm status (slave to master) (CI = 71h) (optional)

For details of the report of alarm status errors see annex D.

4.8 Variable data respond (slave to master) (CI = 72h, 78h, 7Ah)

For details, see clause 5.

4.9 Baud rate switch commands B8h – BFh (otpional)

These optional commands can be used by a master to switch the baud rate of a slave. For details, see 11.2.

5 Variable Data Respond (CI = 72h, CI = 78h, CI = 7Ah)

5.1 Introduction

Data Header of variable data respond The CI-Field codes 72h, 78h, 7Ah are used to indicate the variable data structure in long frames (RSP_UD) with optional fixed header. Note that the CI-fields 78h and 7Ah are extensions from the EN 1434–3. They are recommended for new master implementations to simplify the integration of radio based communication.

Figure 2 shows the way this data is represented.

Data Header(Req.)	Variable Data Blocks (Records)	MDH(opt)O	Opt.Mfg.specific data Opt)
0 byte (CI = 78h) 4 byte (CI = 7Ah) 12 byte (CI = 72h)	variable number	1 Byte	variable number

Figure 2 — Variable Data Structure in Answer Direction

5.2 Structure of Data Header (CI = 72h)

The first twelve bytes of the user data consist of a block with a fixed length and structure (see Figure 3).

Ident. Nr.	Manufr.	Version	Device type	Access No.	Status	Signature
4 Byte	2 Byte	1 Byte	1 Byte	1 Byte	1 Byte	2 Byte

Figure 3 — Data Header CI = 72h

5.3 Structure of Data Header (CI = 7Ah)

The first four bytes of the user data consist of a block with a fixed length and structure (see Figure 4).

This CI-field is proposed for systems using the future physical and link layer standard for radio communication. In this standard the link layer address contains the information fields of the manufacturer, the device type, the version and the identification number, so that these 8 bytes from the fixed header of the CI = 72h are not required in the application layer part of a telegram.

Access No.	Status	Signature
1 Byte	1 Byte	2 Byte

Figure 4 — Data Header CI=7Ah

5.4 Identification Number

The **Identification Number** is either a fixed fabrication number or a number changeable by the customer, coded with 8 BCD packed digits (4 Byte), and which thus runs from 00000000 to 99999999. It can be preset at fabrication time with a unique number, but could be changeable afterwards, especially if in addition an unique and not changeable fabrication number (DIF = 0Ch, VIF = 78h, see 7.2) is provided.

5.5 Manufacturer identification

The field **manufacturer** is coded unsigned binary with 2 bytes. This manufacturer ID is calculated from the ASCII code of EN 62056-21 manufacturer ID (three uppercase letters) with the following formula:

$$\begin{aligned}\text{Man. ID} = \; & [\text{ASCII(1st letter)} - 64] \bullet 32 \bullet 32 \\ + \; & [\text{ASCII(2nd letter)} - 64] \bullet 32 \\ + \; & [\text{ASCII(3rd letter)} - 64]\end{aligned}$$

Note that the flag association, UK (www.dlms.com/flag) administers these three letter manufacturers ID of EN 62056-21.

5.6 Version identification

The field **version** specifies the generation or version of the meter and depends on the manufacturer. It can be used to make sure, that within each version number the identification # is unique.

5.7 Device type identification

The **device byte** is coded as follows:

Table 3 — Device type identification

Device type (previously called medium)	Code bin. Bit 7 … 0	Code hex.
Other	0000 0000	00
Oil	0000 0001	01
Electricity	0000 0010	02
Gas	0000 0011	03
Heat	0000 0100	04
Steam	0000 0101	05
Warm Water (30 °C … 90 °C)	0000 0110	06
Water	0000 0111	07
Heat Cost Allocator	0000 1000	08
Compressed Air	0000 1001	09
Cooling load meter (Volume measured at return temperature: outlet)	0000 1010	0A
Cooling load meter (Volume measured at flow temperature: inlet)	0000 1011	0B
Heat (Volume measured at flow temperature: inlet)	0000 1100	0C
Heat / Cooling load meter	0000 1101	OD
Bus / System component	0000 1110	0E
Unknown Medium	0000 1111	0F
Reserved	…	10 to 14
Hot water (≥ 90 °C)	0001 0101	15
Cold water	0001 0110	16
Dual register (hot/cold) Water meter (see NOTE)	0001 0111	17
Pressure	0001 1000	18
A/D Converter	0001 1001	19
Reserved	…	1Ah to 20h
Reserved for valve	0010 0001	21h
Reserved		22h to FFh

NOTE Such a meter registers water flow above a limit temperature in a separate register with an appropriate tariff ID.

Note that this table has been expanded with optional elements from EN 1434-3.

5.8 Access Number

The **Access Number** has unsigned binary coding, and is incremented (modulo 256) by one before or after each RSP_UD from the slave. Since it can also be used to enable private end users to detect an unwanted overfrequently readout of its consumption meters, it should not be resettable by any bus communication.

5.9 Status byte

Table 4 — Coding of the Status Field

Bit	Meaning with Bit set	Significance with Bit not set
0,1	See Table 5	See Table 5
2	Power low	Not power low
3	Permanent error	No permanent error
4	Temporary error	No temporary error
5	Specific to manufacturer	Specific to manufacturer
6	Specific to manufacturer	Specific to manufacturer
7	Specific to manufacturer	Specific to manufacturer

Table 5 — Application Errors coded with the Status-Field

Status bit 1 bit 0	Application status
0 0	No Error
0 1	Application Busy
1 0	Any Application Error
1 1	Reserved

Note that more detailed error signalling can be provided by application telegrams starting with C **I**= 70h and/or using data records signalling even more detailed error information.

5.10 Signature field

5.10.1 General

The **Signature** is reserved for optional encryption of the application data. Such an encryption might be required for transmit only wireless meter readout. It is assumed, that each meter (or a group of meters) could have an individual encryption key. If no Encryption is used its value shall be 00 00 h.

5.10.2 Functions

— Data privacy for consumption meters values;

— detecting simulated meter transmission;

— preventing later playback of old meter values.

5.10.3 Structure of encrypted telegrams

a) The data header (CI=72h see 5.2 or CI = 7Ah see 5.3) is always unencrypted. The last word of this block is the signature word. If the following data are unencrypted, this signature word contains a zero.

b) If the transmission contains encrypted data, the high byte of this signature word contains a code for the encryption method. The code 0 signals no encryption. Currently only the encryption codes 02xxh or 03xxh (see below) are defined. The other codes are reserved. The number of encrypted bytes is contained in the low byte of the signature word. The content of this signature word had been defined in the EN 1434–3 as zero, corresponding consistently to no encrypted data.

c) The encrypted data follow directly after the signature word, thus forming the beginning of the DIF/VIF-structured part of the telegram.

5.10.4 Partial Encryption

a) If the number of encrypted bytes is less than the remaining data of the telegram, unencrypted data may follow after the encrypted data. They shall start at a record boundary, i.e. the first byte after the encrypted data will be interpreted as a DIF.

b) If a partially encrypted telegram shall contain encrypted manufacturer specific data a record with a suitable length DIF (possibly a variable length string DIF) and a VIF = 7Fh (manufacturer specific data record) shall be used instead of the usual MDH-DIF = 0Fh. This is required to enable after decryption standard DIF/VIF-decoding of a previously partially encrypted telegram containing encrypted manufacturer specific data.

5.10.5 Encryption methods

a) Encryption according to the DES (data encryption standard) as described in ANSI X3.92:1981;

b) Cipher Block Chaining (CBC)-method as described in ANSI X3.106:1983 with an initial initialization vector of zero: (Encryption Method Code = 02xxh). In this case the data records should contain the current date before the meter reading.

   Note that in this case the data after the date record, i.e. especially the encrypted meter reading data change once per day even if their data content itself is constant. This prevents an undetectable later playback of stored encrypted meter readings by a hacker.

c) The "Initialization Vector IV" with length 64 bits of this standard may alternatively be defined by the first 6 bytes of the identification header in mode 1 sequence, i.e. identification number in the lowest 4 bytes followed by the manufacturer ID in the two next higher bytes and finally by the current date coded as in record structure "G" for the two highest bytes.

   In this case the Encryption method is coded as "03xxh". Note that in this case all encrypted data change once per day even if the data content itself is constant. This prevents an undetectable later playback of any stored encrypted data by a hacker.

d) To simplify the verification of correct decoding and to prevent an undetected change in the identification of the not encrypted header, the encrypted part of the telegram shall contain at least together with the appropriate application layer coding (DIF and VIF) again the same identification number as in the unencrypted header;

e) Due to the mathematical nature of the DES-algorithm the encrypted length contained in the low byte of the signature word shall be an integer multiple of 8 if the high byte signals DES-Encryption. Unused bytes in the last 8-byte block shall be filled with appropriatly structured dummy data records to achieve the required record boundary at the end of the encrypted data. One or several bytes containing the filler DIF = 2Fh are suggested to fill such gaps;

f) The application of certain Encryption methods might be prohibited by local laws.

5.11 Address structure if used together with the wireless link layer according to EN 13757-4

This link layer contains an 8 byte address header, which starts with a two byte manufacturer identification according to EN 62056-21 followed by a 6 byte address. If this wireless link layer is used together with the application layer of this standard and the CI-fields 78h or 7Ah this six byte address is structured similarly to the fixed header of the CI-field 72h as follows:

Ident-Nr. (4 byte BCD) according to 5.3 followed by the one byte version identification according to 5.5 and finally the one byte device type according to 5.6.

6 Variable Data Blocks (Records)

6.1 General

The data, together with information regarding coding, length and the type of data is transmitted in data records in arbitrary sequence. As many records can be transferred as there is room for within the maximum total data length of 234 Bytes, taking into account the C, A, and CI fields and the fixed data header. This limits the total telegram length to 255 bytes. This restriction is required to enable gateways to other link- and application layers. The manufacturer data header (MDH) is made up by the character 0Fh or 1Fh and indicates the beginning of the manufacturer specific part of the user data and should be omitted, if there are no manufacturer specific data.

<table>
<tr><td>DIF</td><td>DIFE</td><td>VIF</td><td>VIFE</td><td>Data</td></tr>
<tr><td>1 Byte</td><td>0 … 10 (1 Byte each)</td><td>1 Byte</td><td>0 … 10 (1 Byte each)</td><td>0 … N Byte</td></tr>
<tr><td colspan="2">Data Information Block DIB</td><td colspan="2">Value Information Block VIB</td><td></td></tr>
<tr><td colspan="4">Data Record Header DRH</td><td></td></tr>
</table>

Figure 5 — Structure of a Data Record (transmitted from left to right)

Each data record contains one value (data) with its description (DRH). The DRH in turn consists of the DIB (data information block) to describe the length, type and coding of the data, and the VIB (value information block) to give the value of the unit and the multiplier. Note that an application telegram can contain either just a single data record but also an arbitray number of such data records in arbitrary order, each describing and containing a data element. For examples of such multrecord telegrams see annex E or documents of http://www.m-bus.com.

6.2 Data Information Block (DIB)

The DIB contains at least one byte (DIF, data information field), and can be extended by a maximum of ten DIFE's (data information field extensions).

6.3 Data Information Field (DIF)

The following information is contained in a DIF:

<table>
<tr><td>Bit 7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Extension
Bit
(E)</td><td>LSB of storage number</td><td colspan="2">Function Field</td><td colspan="4">Data Field :
Length and coding of data</td></tr>
</table>

Figure 6 — Coding of the Data Information Field (DIF)

6.4 Data Field

The **data field** shows how the data from the master shall be interpreted in respect of length and coding. The following table contains the possible coding of the data field:

Table 6 — Coding of the data field

Length in Bit	Code	Meaning	Code	Meaning
0	0000	No data	1000	Selection for Readout
8	0001	8 Bit Integer/Binary	1001	2 digit BCD
16	0010	16 Bit Integer/Binary	1010	4 digit BCD
24	0011	24 Bit Integer/Binary	1011	6 digit BCD
32	0100	32 Bit Integer/Binary	1100	8 digit BCD
32 / N	0101	32 Bit Real	1101	variable length
48	0110	48 Bit Integer/Binary	1110	12 digit BCD
64	0111	64 Bit Integer/Binary	1111	Special Functions

Note that this table has been expanded with optional elements from the original standard.

For a detailed description of data types refer to annex A ”Coding of data records” (e.g. BCD = Type A, Integer = Type B, Real = Type H).

Variable Length:

With data field = `1101b´ several data types with variable length can be used. The length of the data is given after the DRH with the first byte of real data, which is here called LVAR (e.g. LVAR = 02h: ASCII string with two characters follows).

LVAR = 00h – BFh : 8-bit text string according to ISO 8859–1 with LVAR (0 to 191) characters. Note that a text string (like all other mutibyte data) is transmitted “Least significant byte first”

LVAR = C0h – C9h : positive BCD number with (LVAR – C0h, i.e. 0 to 9) • 2 digits (0 to 18 digits)

LVAR = D0h – D9h : negative BCD number with (LVAR – D0h) • 2 digits (0 to 18 digits)

LVAR = E0h – Efh : Binary number with (LVAR – E0h) bytes (0 to 15 bytes)

LVAR = F8h : floating point number according to IEEE 754

Others LVAR values : Reserved

Like all multibyte fields the last character is transmitted first.

<u>Special Functions (data field = 1111b):</u>

Table 7 — DIF-coding for special functions

DIF	Function
0Fh	Start of manufacturer specific data structures to end of user data
1Fh	Same meaning as DIF = 0Fh + More records follow in next telegram
2Fh	Idle Filler (not to be interpreted), following byte = DIF of next record
3Fh … 6Fh	Reserved
7Fh	Global readout request (all storage#, units, tariffs, function fields)

Note that this table has been expanded with optional elements from the original standard.

If data follows after DIF = 0Fh or 1Fh these are manufacturer specific unstructured data. The number of bytes in these manufacturer specific data can be calculated from the link layer information on the total length of the application layer telegram. The DIF 1Fh signals a request from the slave to the master to readout the slave once again. The master shall readout the slave until there is no DIF = 1Fh inside the respond telegram (multi telegram readout) or use an application reset.

6.5 Function field

The **function field** gives the type of data as follows:

Table 8 — Function Field

Code	Description	Code	Description
00b	Instantaneous value	01b	Maximum value
10b	Minimum value	11b	Value during error state

6.6 Storage number

The Bit 6 of the DIF serves as the LSB of the **storage number** of the data concerned, and the slave can in this way indicate and transmit various stored metering values or historical values of metering data. This bit is the least significant bit of the storage number and allows therefore the storage numbers 0 and 1 to be coded. If storage numbers higher than “1” are needed, following (optional) DIFE´s contain the higher bits. The storage number 0 signals an actual value. Note that a each storage number is associated with a given time point. So all data records with the same storage number refer to the value of the associated variable at this (common) time point for this storage number. It is recommended, that a time/date record for each storage number used is included somewhere in the telegram to signal this time point associated with this storage number. This date or date/time is coded with a data record with a VIF=E110110n. Normally (but not necessarily) higher storage numbers indicate an older time point. A sequential block of storage numbers can be associated with a sequence of equidistantly spaced time points (profile). Such a block can be described by its starting time, by the time spacing, by the first storage number of such a block and by the length of such a block. The document CBDIPW6.EXE of the download area of http://www.m-bus.com (only available in German) gives detailed examples of configuring meters for profiles and tariffs.

6.7 Extension Bit (E)

The extension bit (MSB) signals that more detailed or extended descriptions (data field extension = DIFE)-bytes follow. E = 1 if other VIFE or DIFE follow.

6.8 Data field extension byte(s) (DIFE)

Each DIFE (maximum ten) contains again an extension bit to show whether a further DIFE is being sent. Besides giving the next most significant bits of the storage number, DIFE´s allow the transmission of information about the **tariff** and the **subunit** of the device. In this way, exactly as with the storage number, the next most significant bit or bits will be transmitted. The Figure 7 which follows shows the structure of a DIFE:

Bit	7	6	5	4	3	2	1	0
Value	Extension Bit (E)	(Device) Unit	Tariff		Storage Number			

Figure 7 — Coding of the Data Information Field Extension (DIFE)

With the maximum of ten DIFE´s which are provided, there are 41 bits for the storage number, 20 bits for the tariff, and 10 bits for the subunit of the meter. There is no application conceivable in which this immense number of bits could all be used.

6.9 Tariff information

For each (unique) value type designation given by the following value information block (VIB) at each unique time point (given by the storage number) of each unique function (given by the function field) there might exist still various different data, measured or accumulated under different conditions. Such conditions could be time of day, various value ranges of the variable (i.e. separate storage of positive accumulated values and negative accumulated values) itself or of other signals or variables or various averaging durations. Such variables which could not be distinguished otherwise are made different by assigning them different values of the tariff variable in their data information block. Note that this includes but is not necessarily restricted to various tariffs in a monetary sense. It is at the distinction of the manufacturer to describe for each tariff (except 0) what is different for each tariff number. Again as with the storage numbers all variables with the same tariff information share the same tariff associating condition.

6.10 Subunit information

A slave component may consist of several functionally and logically independent subunits of the same or of different functionallity. Such a device may either use several different primary and/or secondary addresses. Such it is from a link layer and an application layer view just several independent devices which share a common physical layer interface. This is recommended for devices which represent a physical collection of several truely independent (often similar or identical) devices. For devices which share common information and values and have logical connections an approach with a common link layer (i.e.a single address) is recommended. The various subunits can include their specific information into a common telegram and have them differentiated by the individual subunit number in the subunit-datafield of their records.

7 Value Information Block (VIB)

7.1 General

After a DIF (with the exception of 0xFh) or a DIFE without a set extension bit there follows the VIB (value information block). This consists at least of the VIF (value information field) and can be expanded with a maximum of 10 extensions (VIFE). The VIF and also the VIFE's show with a set MSB that a VIFE will follow. In the value information field VIF the other seven bits give the unit and the multiplier of the transmitted value.

Bit	7	6	5	4	3	2	1	0
Value	Extension Bit (E)	Unit and multiplier (value)						

Figure 8 — Coding of the Value Information Field (VIF)

There are five types of coding depending on the VIF:

a) **Primary VIF: E000 0000b … E111 1011b**

The unit and multiplier is taken from the table for primary VIF (7.2).

b) **Plain-text VIF: E111 1100b**

In case of VIF = 7Ch / FCh the true VIF is represented by the following ASCII string with the length given in the first byte. Please note that the byte order of the characters after the length byte depends on the used byte sequence. Since only the “LSB first mode” (M = 1) of multibyte data transmission is recommended, the rightmost character is transmitted first. This plain text VIF allows the user to code units that are not included in the VIF tables.

c) **Linear VIF-Extension: FDh and FBh**

In case of VIF = FDh and VIF = FBh the true VIF is given by the next byte (i.e. the first VIFE) and the coding is taken from the Tables 12 respectively Table 13 for secondary VIF (7.4 or 7.5). This extends the available VIF´s by another 256 codes.

d) **Any VIF: 7Eh / Feh**

This VIF-Code can be used in direction master to slave for readout selection of all VIF´s. See special function in 6.4.

e) **Manufacturer specific: 7Fh / FFh**

In this case the remainder of this data record including VIFE´s has manufacturer specific coding.

7.2 Primary VIF's (main table)

The first section of the main table contains integral values, the second typically averaged values, the third typically instantaneous values and the fourth block contains parameters (E: extension bit).

Table 9 — Primary VIF-codes

Coding	Description	Range Coding	Range
E000 0nnn	Energy	$10^{(nnn-3)}$ Wh	0,001 Wh to 10 000 Wh
E000 1nnn	Energy	$10^{(nnn)}$ J	0,001 kJ to 10 000 kJ
E001 0nnn	Volume	$10^{(nnn-6)}$ m^3	0,001 l to 10 000 l
E001 1nnn	Mass	$10^{(nnn-3)}$ kg	0,001 kg to 10 000 kg
E010 00nn	On Time	nn = 00b seconds nn = 01b minutes nn = 10b hours nn = 11b days nn = 11 days	Duration of meter power up
E010 01nn	Operating Time	coded like OnTime	Duration of meter accumulation
E010 1nnn	Power	$10^{(nnn-3)}$ W	0,001 W to 10 000 W
E011 0nnn	Power	$10^{(nnn)}$ J/h	0,001 kJ/h to 10 000 kJ/h
E011 1nnn	Volume Flow	$10^{(nnn-6)}$ m^3/h	0,001 l/h to 10 000 l/h
E100 0nnn	Volume Flow ext.	$10^{(nnn-7)}$ m^3/min	0,000 1l/min to 1 000 l/min
E100 1nnn	Volume Flow ext.	$10^{(nnn-9)}$ m^3/s	0,001 ml/s to 10 000ml/s
E101 0nnn	Mass flow	$10^{(nnn-3)}$ kg/h	0,001 kg/h to 10 000 kg/h
E101 10nn	Flow Temperature	$10^{(nn-3)}$ °C	0,001 °C to 1 °C
E101 11nn	Return Temperature	$10^{(nn-3)}$ °C	0,001 °C to 1 °C
E110 00nn	Temperature Difference	$10^{(nn-3)}$ K	1 mK to 1000 mK
E110 01nn	External Temperature	$10^{(nn-3)}$ °C	0,001 °C to 1 °C
E110 10nn	Pressure	$10^{(nn-3)}$ bar	1 mbar to 1 000 mbar
E110 1100	Date (actual or associated with a storage number/function)		data field =0010b, type G
E110 1101 [b]	Date and time (actual or associated with a storage number/function)		data field= 0100b, type F
E110 1101 [b]	Extendend time point (actual or associated with a storage number/function)	Time to s	data field= 0011b, type J
E110 1101 [b]	Extented Date and Time Point (actual or associated with a storage number/function)	Time and date to sec.	data field= 0110b, type I
E110 1110	Units for H.C.A.		Dimensionless
E110 1111	Reserved for a future third table of VIF-extensions		
E111 00nn	Averaging Duration	nn coded like OnTime	
E111 01nn	Actuality Duration	nn coded like OnTime	
E111 1000	Fabrication No		see E.3
E111 1001	(Enhanced) Identification		
E111 1010	Address		For EN 13757-2: one byte link layer address, data type C (x = 8) For EN 13757-4: data field 110b (6 byte Header-ID) or 111b (Full 8 byte Header)

Note b Meaning depends on data field.

Note that this table has been expanded with optional elements from the original standard.

7.3 VIF-Codes for special purposes

Table 10 — Special VIF-Codes. Note that this table has been expanded with optional elements from the original standard

Coding	Description	Purpose
1111 1011 (FBh)	First Extension of VIF-codes	true VIF is given in the first VIFE and is coded using (Table 11 in 7.5) (128 new VIF-Codes)
E111 1100	VIF in following string (length in first byte)	allows user definable VIF´s (in plain ASCII-String) [a]
1111 1101 (FDh)	Second Extension of VIF-codes	true VIF is given in the first VIFE and is coded using (Table 11 in 7.4) (128 new VIF-Codes)
1110 1111 (EFh)	Reserved for third Extension table of VIF-codes	Reserved for a future table especially for electricity meters
E111 1110	Any VIF	Used for readout selection of all VIF´s (see 6.4)
E111 1111	Manufacturer Specific	VIFE´s and data of this block are manufacturer specific

[a] Coding the VIF in an ASCII-String in combination with the data in an ASCII-String (datafield in DIF = 1101 b) allows the representation of data in a free user defined form.

7.4 Main VIFE-Code Extension table (following VIF = FDh for primary VIF)

Table 11 — Main VIFE-code extension table

Coding	Description	Group
E000 00nn	Credit of 10^{nn-3} of the nominal local legal currency units	Currency Units
E000 01nn	Debit of 10^{nn-3} of the nominal local legal currency units	
E000 1000	Access Number (transmission count)	
E000 1001	Device type	
E000 1010	Manufacturer	
E000 1011	Parameter set identification	Enhanced Identification
E000 1100	Model / Version	
E000 1101	Hardware version #	
E000 1110	Metrology (firmware) version #	
E000 1111	Other software version #	
E001 0000	Customer location	
E001 0001	Customer	
E001 0010	Access Code User	

"to be continued"

Table 11 (*continued*)

Coding	Description	Group
E001 0011	Access Code Operator	Improved Selection
E001 0100	Access Code System Operator	and other user requirements
E001 0101	Access Code Developer	
E001 0110	Password	
E001 0111	Error flags (binary) (Device type specific)	
E001 1000	Error mask	
E001 1001	Reserved	
E001 1010	Digital Output (binary)	
E001 1011	Digital Input (binary)	
E001 1100	Baud rate [Baud]	
E001 1101	Response delay time [bittimes]	
E001 1110	Retry	
E001 1111	Remote control (device specific)	
E010 0000	First storage # for cyclic storage	
E010 0001	Last storage # for cyclic storage	
E010 0010	Size of storage block	
E010 0011	Reserved	Enhanced storage
E010 01nn	Storage interval [sec(s) … day(s)] [a]	management
E010 1000	Storage interval month(s)	
E010 1001	Storage interval year(s)	
E010 1010	Reserved	
E010 1011	Time point second (0 to 59)	
E010 11nn	Duration since last readout [sec(s) … day(s)] [a]	
E011 0000	Start (date/time) of tariff [b]	
E011 00nn	Duration of tariff (nn=01 … 11: min to days)	
E011 01nn	Period of tariff [sec(s) to day(s)] [a]	
E011 1000	Period of tariff months(s)	Enhanced tariff
E011 1001	Period of tariff year(s)	management
E011 1010	Dimensionless / no VIF	
E011 1011	Reserved	
E011 11xx	Reserved	
E100 nnnn	10^{nnnn-9} Volts	electrical units
E101 nnnn	$10^{nnnn-12}$ A	
E110 0000	Reset counter	

"to be continued"

Table 11 (*concluded*)

Coding	Description	Group
E110 0001	Cumulation counter	
E110 0010	Control signal	
E110 0011	Day of week [e]	
E110 0100	Week number	
E110 0101	Time point of day change	
E110 0110	State of parameter activation	
E110 0111	Special supplier information	
E110 10pp	Duration since last cumulation [hour(s) … years(s)][c]	
E110 11pp	Operating time battery [hour(s)..years(s)][c]	
E111 0000	Date and time of battery change	
E111 0001	Reserved	
E111 0010	Day light saving (beginning, ending, deviation) data type K	
E111 0011	Listening window management data type L	
E111 0100	Remaining battery life time (days)	
E111 0101	#times the meter was stopped	
E111 0110 to E111 1111	Reserved	

[a] nn = 00 second(s)
01 minute(s)
10 hour(s)
11 day(s)

[b] The information about usage of data type F (date and time) or data type G (date), I (time to seconds) or J (extendeted date/time) can be derived from the datafield (0010b: type G / 0011b: type J, 0100: type F).

[c] pp = 00 hour(s)
01 day(s)
10 month(s)
11 year(s)

[e] Data type A (1 = Monday; 7= Sunday, 0= all the days)

Note that this optional table has been added to EN 1434-3.

7.5 Alternate VIFE-Code Extension table (following VIF = 0FBh for primary VIF)

Table 12 — Alternate extended VIF-code table [a]

Coding	Description	Range Coding	Range
E000 000n	Energy	$10^{(n-1)}$ MWh	0.1MWh to 1MWh
E000 001n	Reactive energy	$10^{(n)}$ kVARh	1 to 10 kVARh
E000 01nn	Reserved		
E000 100n	Energy	$10^{(n-1)}$ GJ	0.1GJ to 1GJ
E000 101n	Reserved		
E000 11nn	Reserved		
E001 000n	Volume	$10^{(n+2)}$ m^3	
E001 001n	Reserved		
E001 01nn	Reserved		
E001 100n	Mass	$10^{(n+2)}$ t	100 t to 1 000 t
E001 1010 – E010 0000	Reserved		
E010 0001	Volume	0,1 feet^3	
E010 0010	Volume	0,1 Usgallon	Note 1
E010 0011	Volume	1 Usgallon	Note 1
E010 0100	Volume flow	0,001 Usgallon/min	Note 1
E010 0101	Volume flow	1 Usgallon/min	Note 1
E010 0110	Volume flow	1 USgallon/h	Note 1
E010 0111	Reserved		
E010 100n	Power	$10^{(n-1)}$ MW	0,1 MW to 1 MW
E010 101n	Reserved		
E010 11nn	Reserved		
E011 000n	Power	$10^{(n-1)}$ GJ/h	0,1 GJ/h to 1 GJ/h
E011 0010 – E101 0111	Reserved		
E101 10nn	Flow Temperature	$10^{(nn-3)}$ F	0,001°F to 1°F Note 1
E101 11nn	Return Temperature	$10^{(nn-3)}$ ° F	0,001°F to 1°F Note 1
E110 00nn	Temperature Difference	$10^{(nn-3)}$ ° F	0,001°F to 1°F Note 1
E110 01nn	Flow Temperature	$10^{(nn-3)}$ ° F	0,001°F to 1°F Note 1
E110 1nnn	Reserved		
E111 00nn	Cold/Warm Temperature Limit	$10^{(nn-3)}$ °F	0,001°F to 1°F Note 1
E111 01nn	Cold/Warm Temperature Limit	$10^{(nn-3)}$ °C	0,001 °C to 1 °C
E111 1nnn	Cum.Count Max.power	$10^{(nnn-3)}$ W	0,001 W to 10 000 W

[a] These codes shall not be used in new developments. For non metric units in new developments use the corresponding metric unit and append the VIFE 3Dh (alternate unit, see Table 13 in 7.6) and apply the unit translation Table C.1 of annex C.

Note that this optional table has been added to the original standard.

7.6 Combinable (Orthogonal) VIFE-Code Extension table

This code follows immediately the VIF or the VIFE (in case of code extension) and modifies its meaning.

Table 13 — Combinable (orthogonal) VIFE-Table

VIFE-Code	Description
E00x xxxx	reserved for object actions (master to slave): see clause 9 or for error codes (slave to master): see 8.3
E010 0000	per second
E010 0001	per minute
E010 0010	per hour
E010 0011	per day
E010 0100	per week
E010 0101	per month
E010 0110	per year
E010 0111	per revolution / measurement
E010 100p	increment per input pulse on input channel #p
E010 101p	increment per output pulse on output channel #p
E010 1100	per liter
E010 1101	per m^3
E010 1110	per kg
E010 1111	per K (Kelvin)
E011 0000	per kWh
E011 0001	per GJ
E011 0010	per kW
E011 0011	per (K*l) (Kelvin*liter)
E011 0100	per V (Volt)
E011 0101	per A (Ampere)
E011 0110	multiplied by s
E011 0111	multiplied by s / V
E011 1000	multiplied by s / A
E011 1001	start date(/time) of [a, b]
E011 1010	VIF contains uncorrected unit instead of corrected unit
E011 1011	accumulation only if positive contributions (Forward flow contribution)
E011 1100	accumulation of abs value only if negative contributions (Backward flow)
E011 1101	reserved for alternate non-metric unit system (see annex C)
E011 111x	reserved

"to be continued"

Table 13 (*concluded*)

VIFE-Code	Description
E100 u000	U = 1: upper, u = 0: lower limit value
E100 u001	# of exceeds of lower u = 0) / upper (U = 1) limit
E100 uf1b	Date (/time) of:b = 0: begin, b = 1: end of, f = 0: first, f = 1: last, [b] u = 0: lower, u = 1: upper limit exceed
E101 ufnn	Duration of limit exceed (u,f: as above, nn = duration)
E110 0fnn	Duration of [a, b] (f: as above, nn = duration)
E110 1u00	Value during lower (u = 0), upper (u = 1) limit exceed
E110 1001	Leakage values
E110 1101	Overflow values
E110 1f1b	Date (/time) of [a] (f,b: as above)
E111 0nnn	Multiplicative correction factor: 10^{nnn-6}
E111 10nn	Additive correction constant: 10^{nn-3} • unit of VIF (offset)
E111 1100	Reserved
E111 1101	Multiplicative correction factor for value (not unit): 10^3
E111 1110	Future value
E111 1111	Next VIFE's and data of this block are maufacturer specific

[a] ”Date(/time) of” or ”Duration of” relates to the information which the whole data record header contains.

[b] The information about usage of data type F (date and time) or data type G (date) can be derived from the datafield (0010b: type G / 0100: type F).

Note that this optional table has been added to the original standard.

8 Application Layer Status and error reporting

8.1 General

The data link layer reports only communication. It is not allowed to report errors of the application layer (which can occur for example in data writing) via the link layer. There are three different techniques for reporting application errors:

8.2 Status Field

One possible solution is to use the reserved 2 lowest bits of the Status field in the variable data structure for the application layer status (see 5.9).

8.3 General Application Layer Errors

For reporting general application errors a slave can use a RSP_UD telegram with CI = 70h and zero, one or several data bytes, which then describes the type of error:

CI=70h	Optional First Error code byte (See Table 14)

Figure 9 — Application layer telegram for reporting general application errors

The following values for DATA are defined:

Table 14 — First error Code byte for general application errors

0	Unspecified error: also if data field is missing
1	Unimplemented CI-Field
2	Buffer too long, truncated
3	Too many records
4	Premature end of record
5	More than 10 DIFE´s
6	More than 10 VIFE´s
7	Reserved
8	Application too busy for handling readout request
9	Too many readouts (for slaves with limited readouts per time)
10 … 255	Reserved

Note that this optional table has been added to the original standard.

8.4 Record Errors

To report errors belonging just to a special record and not to the full application the slave can add to the record where an error occurred a VIFE containing one of the values of Table 15 to code the type of application error, which has occurred for this record.

Table 15 — Codes for record errors (E = extension bit)

VIFE-Code	Type of Record Error	Error Group
E000 0000	None	DIF Errors
E000 0001	Too many DIFE´s	
E000 0010	Storage number not implemented	
E000 0011	Unit number not implemented	
E000 0100	Tariff number not implemented	
E000 0101	Function not implemented	
E000 0110	Data class not implemented	
E000 0111	Data size not implemented	
E000 1000 to E000 1010	Reserved	
E000 1011	Too many VIFE´s	VIF Errors
E000 1100	Illegal VIF-Group	
E000 1101	Illegal VIF-Exponent	
E000 1110	VIF/DIF mismatch	
E000 1111	Unimplemented action	
E001 0000 to E001 0100	Reserved	
E001 0101	No data available (undefined value)	Data Errors
E001 0110	Data overflow	
E001 0111	Data underflow	
E001 1000	Data error	
E001 1001 to E001 1011	Reserved	
E001 1100	Premature end of record	Other Errors
E001 1101 to E001 1111	Reserved	

Note that this optional table has been added to the original standard.

In case of record errors the data maybe invalid. The slave has some options to transmit the data:

- datafield = 0000b: no data
- datafield = 0000b: no data and idle filler (DIF = 02Fh): fill record up to the normal length
- other datafield: dummy data of correct length
- other datafield: unsafe or estimated data

9 Generalized Object Layer

The fundamental idea of an object is the encapsulation of data and methods or actions for the data. In case of writing data to a slave the master software can pack data and information about the action, which the slave shall do with this data, in one data record. This variable data record with actions is now called an object. Following any VIF including a VIF = FDh or VIF = 0FBh with the true value information in the first VIFE another (usually the last) VIFE can be added which contains a code signalling object actions according to the following table.

Action: (E: extension bit)

Table 16 — Action Codes for the Generalized Object layer (Master to Slave)

VIFE-Code binary	Action	Explanation
E000 0000	Write (Replace)	replace old with new data
E000 0001	Add Value	add data to old data
E000 0010	Subtract Value	subtract data from old data
E000 0011	OR (Set Bits)	data OR old data
E000 0100	AND	data AND old data
E000 0101	XOR (Toggle Bits)	data XOR old data
E000 0110	AND NOT (Clear Bits)	NOT data AND old data
E000 0111	Clear	set data to zero
E000 1000	Add Entry	create a new data record
E000 1001	Delete Entry	delete an existing data record
E000 1010	Delayed Action	A CI = 5Ch will follow and execute the desired action
E000 1011	Freeze Data	freeze data to storage no.
E000 1100	Add to Readout-List	add data record to RSP_UD
E000 1101	Delete from Readout-List	delete data record from RSP_UD
E000 111x	Reserved	
E001 xxxx	Reserved	

Note that this optional table has been added to the original standard.

NOTE The object action "write / replace" (VIFE = E000 0000) is the default and is assumed if there is no VIFE with an object action for this record.

10 Manufacturer Specific unstructured Data Block

The MDH consists of the character 0Fh or 1Fh (DIF = 0Fh or 1Fh) and indicates that all following data are manufacturer specific. When the total number of bytes given from the link/network layers and the number of record-structured bytes and the length of the fixed header is known, the number of remaining unstructured manufacturer specific bytes can be calculated.

Note that structured manufacturer specific data (i.e. those with a known data structure including variable length binary or ASCII but with a manufacturer specific meaning or unit) can be described using normal data records with a value information field of VIF = E1111111b.

In case of MDH = 1Fh the slave signals to the master that it wants to be readout once again (multitelegram readouts). The master shall readout the data until there is no MDH = 1Fh in the respond telegram.

11 Management of lower layers

11.1 General

Because changing of parameters like baud rate and address by higher layers is not allowed in the ISO-OSI-Model, a **Management Layer** beside and above the three layers of the collapsed model is defined:

Management Layer	
Application Layer	
Data Link Layer	Secondary address selection via address 253 and CI = 52h
Physical Layer	Address 254 (255)/251

Figure 10 — Management-Layer of the M-Bus Link Layer according EN 13757-2

So the address 254 and perhaps 255 can be used also for managing the physical layer of the bus and the address 251 is reserved for managing the (primary) M-Bus level converter/bridge and the address 253 (selection) for network layer (see clause 7), which is only used in certain cases. With such a management addresses and or CI-fields we can directly manage each OSI-layer to implement features, which are beyond the elementary OSI-Model.

11.2 Switching Baud rate for M-Bus link layer according to EN 13757-2

All slaves shall be able to communicate with the master using the minimum transmission speed of 300 baud. Split baud rates between transmit and receive are not allowed, but there can be devices with different baud rates on the bus.

In point to point connections the slave is set to another baud rate by a Control Frame (SND_UD with L-Field = 3) with address FEh and one of the following CI-Field codes: Note that for safety reasons a baud rate switch command to the (unacknowledged) broadcast address 255 is not recommended.

Table 17 — CI-Field Codes for Baud rate Switching

CI-Field	B8h	B9h	BAh	BBh	BCh	BDh	BEh	BFh
Baud	300	600	1 200	2 400	4 800	9 600	19 200	38 400
Note	1	2	2	1	2	1	2	2

NOTE 1 Recommended standard baud rates.

NOTE 2 These baud rates are reserved for special operator agreement only and should be avoided.

The slave always confirms the correctly received telegram by transmitting an E5h with the old baud rate and uses the new baud rate from now on, if he is capable of this. Otherwise the slave stays at its previous baud rate after the E5h acknowledge. To make sure that a slave without auto speed detect has properly switched to the new baud rate and that it can communicate properly at the new baud rate in its segment it is required that after a baud rate switch to a baud rate other than 300 Baud the master attempts immediately (< 2min) after the baud rate switch command a communication. If (even after the appropriate number of retries) this is not acknowledged by the slave, the master shall issue a baud rate set command (at the attempted new baud rate) back to the previous baud rate. If a slave without autospeed detect does not receive a valid communication at the new baud rate within (2 … 10) min of the baud rate switch command the slave shall fall back to its previous baud rate. This is required individually and sequentially for each addressable slave. For compatibility with older slaves with fallback to 300 baud the master should also attempt a communication at 300 baud if the slave does not answer at its last baud rate.

11.3 Selection and Secondary Addressing

This technique allows the M-Bus protocol to logically "connect" a slave with a certain (secondary) address and it then associates this selected slave with the primary address of 253 (FDh). So the maximum number of 250 addresses (primary) is extended by this technique to an arbitrary number of possible slaves, effectively increasing the address range of the link layer. This function is only enabled by a SND_UD with CI_Field 52h to address 253.

When addressing in the data link layer with the help of the A-Field, the problem of the address allocation could arise. The addresses are normally set to a value of 0 by the manufacturer of the meters, in order to designate them as unconfigured slaves. A very laborious method of address allocation consists of setting the addresses when installing the slaves, for example with DIP switches. A further method of address allocation is to determine the bus addresses when connecting the equipments to the bus with the master software. This sends a command for address allocation (see E.3) to the address 0. In this case the slaves shall however all be successively connected to the bus, which very much gets in the way of a simple installation procedure.

When however addressing in the network layer these disadvantages are avoided and the address region is essentially extended beyond the number of 250 with primary addressing (A-Field). The addressing of the slaves takes place with secondary addressing with the help of the following so-called selection:

68h	0Bh	0Bh	68h	53h	FDh	52h	ID1-4	Man 1-2	Gen	Dev	CS	16h

Figure 11 — Application layer structure of a telegram for selecting a slave

The master sends a SND_UD with the control information 52h to the address 253 (FDh) and fills the specific meter secondary address (identification number, manufacturer, version and device type) with the values of the slave which is to be addressed. After the reception of the address FDh the selection mode is entered. If then the proper CI-selection code CI = 52h, is received the internal selection bit is set otherwise it is reset. If further data bytes follow they are compared with the corresponding internal addresses respective values of the meter. If they disagree, the selection bit is cleared otherwise it is left unchanged. Thus “selecting” a meter with only a proper CI-field and no further data will select all meters on the bus capable of secondary addressing. A set selection bit means that this slave can be addressed (e.g. REQ_UD) with the bus address FDh and in this example will reply with RSP_UD. In other words the network layer has associated this slave with the address FDh.

During selection individual positions of the secondary addresses can be occupied with wildcards (Fh). Such a Wildcard means that this position will not be taken account of during selection, and that the selection will be limited to specific positions, in order to address complete groups of slaves (Multicasting). In the identification number each individual digit can be wildcarded by a wildcard nibble Fh while the fields for manufacturer, version and device type can be wildcarded by a wildcard byte FFh.

The state of the selection remains unchanged until the slave is deselected with a selection command (as described above) with non-matching secondary addresses, or a SND_NKE to address 253. The slave, which uses mode 1 for multibyte records, will be selected by a telegram with the CI-Field 52h and the correct secondary address, but it will be deselected by a telegram with any other secondary address.

A slave with implemented primary and secondary addressing should also answer telegrams to his primary address. A slave with only secondary addressing (i.e. internal primary address = 253) should occupy the address field in the RSP_UD telegram with FDh to signal that it will not participate in primary addressing.

11.4 Generalized Selection Procedure

For including new or restructed identification parameters into a selection procedure an enhanced definition of the selection telegram (CI = 52h) can be used:

After the 8 byte of the fixed selection header may also follow standard records with data. In this case only those meters will be selected, where in addition to the fixed header all record data agree. In most but not all cases this means that the DIF and parts of the VIF (not exponent) shall match. Again wildcard rules apply to the record data (digit wildcard for BCD-coded data and byte wildcard for binary or string data).

With this generalized selection it will be possible to select slaves using e.g. additional fabrication number, longer identification numbers, customer, customer location and more information. For inclusion of the fabrication number in the selection process.

After the field “device type” the 8-digit BCD-fabrication number follow. Parts of the fabrication number (Fab1 … Fab4) can be occupied with wildcards (Fh).

If a fabrication number exists the slave should add this data to the variable data blocks in every RSP-UD telegram. If the fabrication number and enhanced selection is not implemented in a slave this device will not confirm the enhanced selection telegram and will be deselected.

Enhanced selection should be used only if the normal kind of selection is not successful.

Enhanced selection with fabrication number

The identification number can be used as a customer number and then can be changed by the operator. Therefore it can be possible that two slaves have the same secondary address. For this reason the selection telegram can be extended by a **fabrication number** to make sure that in any case all slaves are distinguishable. This number is a serial number allocated during manufacture, coded with 8 BCD packed digits (4 Byte) like the identification number, and thus runs from 00000000 to 99999999.

The following figure shows the structure of an enhanced selection telegram released by the master.

CI=52h	ID1 (LSB)	ID2	ID3	ID4	Man1	Man2	Gen	Med	DIF=0Ch	VIF=78h	Fab1	Fab2	Fab3	Fab4

Figure 12 — Application layer structure of a telegram for enhanced selection (mode 1)

After the field medium the new data is given in form of a structured datarecord with DIF = 0Ch and VIF = 78h. Parts of the fabrication number (Fab1 … Fab4) can be occupied with wildcards (Fh).

If a fabrication number exists the slave should add this data to the variable data blocks in every RSP-UD telegram. If the fabrication number and enhanced selection is not implemented in a slave this device will not confirm the enhanced selection telegram and will be deselected.

Enhanced selection should be used only if the normal kind of selection is not successful.

11.5 Searching for Installed Slaves

11.5.1 Primary Addresses

To read out all Installed Slaves the master software shall know all the slaves, which are connected to the bus. Therefore the software searches for slaves with Primary Addressing by sending a REQ_UD2 to all allowed addresses (1 … 250) with all available baud rates. The master notes used primary addresses with the respective baud rates.

11.5.2 Secondary Addresses

The Secondary Addressing described in the preceding section draws attention to the problem of determining the secondary addresses of slaves connected to the bus. The master can after this read out the slaves making use of secondary addresses with previous selection. Testing all possible identification numbers with the master software would take years, since the identification number offers millions of combinations. For this reason, a procedure was developed for the rapid and automatic determination of already installed slaves:

11.5.3 Wildcard searching procedure

The following wildcard searching procedure uses the occupation of individual parts of the secondary address with wildcards (Fh) for selection:

In this case with the identification number (BCD) each individual position, and by manufacturer, version and medium (binary coding), only one complete byte, can be occupied with wildcards. The master begins the selection using a SND_UD with the control information 52h (Mode 1), and occupies all positions in the identification number, except the top one, with wildcards. The top position is run through in ten selections from 0 to 9 (0FFFFFFF to 9FFFFFFF).

If after such a selection the master receives no acknowledgement, it then goes to the next selection. If the master receives an E5h, it then sends a REQ_UD2 and learns the secondary address of the slaves from the reply telegram, as long as no collision occurs. If there is a collision after the selection or the REQ_UD2, the master varies the next positions and holds the existing one. If there is a collision, for example at 5FFFFFFF, the selection is run through from 50FFFFFF to 59FFFFFF. If in this case collisions again occur, then a change is made to a variation of the next position. After running through a complete position, the next higher position is processed up to 9.

With this Wildcard searching procedure, it will be seen that at least the top position shall be run through in order to reach all slaves. Running through further positions may be necessary, depending on the number of the slaves and the distribution of the identification numbers. This procedure allows a statement of the maximum number of selections in relation to the number of slaves, but as disadvantage frequent collisions, which occur, should be mentioned. The wildcard searching procedure shall be performed for all used baud rates and both byte sequences (mode 1 and 2).

The search procedure can be extended with searching for manufacturer, generation and finally device types to find slaves, which have the same identification number. It is also possible to search for all slaves of a certain manufacturer or all slaves of a certain device type by setting the corresponding value. With extended selection meters which differ only in their manufacturer specific fixed fabrication number can be distinguished.

Annex A
(normative)

Coding of Data Records

The following data types are used inside the application layer:

Type A = Unsigned Integer BCD := XUI4 [1 to 4] <0 to 9 BCD>

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
digit 10				digit 1				1UI4 [1 to 4] < 0 to 9 BCD > := digit 10^0
8	4	2	1	8	4	2	1	2UI4 [5 to 8] < 0 to 9 BCD > := digit 10^1
...	...	...	...	...	...	...	...	...
8	4	2	1	8	4	2	1	XUI4 [5 to 8] < 0 to 9 BCD > := digit 10^{X-1}

Digits values of Ah – Eh in any digit position signals invalid.

A hex code Fh in the MSD position signals a negative BCD number in the remaining X-1 digits. For details of this coding see annex B.

Type B = Binary Integer := I[1..X] <(-2^{X-1} -1) to +(2^{X-1}-1)>

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
...							...	1B1 [X] := S = Sign: S <0> := positive S <1> := negative
S	2^{X-2}						2^{X-8}	negative values in two´s complement

The coding “10000000b” signals “invalid”

Type C = Unsigned Integer := UI[1..X] < 0 to 2X-1>

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	UI8 [1 to..8] <(0 to 255)>
...							...	
S	2^{X-2}						2^{X-8}	

Type D = Boolean (1 bit binary information) := XB1 B1[i] < 0 to 1>

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	XB1: B1[i] < 0 to 1>
...							...	B1[i] <0> := false
2^{X-1}							2^{X-8}	B1[i] <1> := true

Type E = Obsolete

Type F = Compound CP32: Date and Time

2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}
2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}
2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}

min: UI6 [1 to 6] < 0 to 59> ; 63 : every minute

hour: UI5 [9 to 13] < 0 to 23> ; 31 : every hour

day : UI5 [17 to 21] < 1 to 31> ; 0 : every day

month: UI4 [25 to 28] < 0 to 12> ; 15 every month

year: UI7 [22 to 24 ; 29 to 32] < 0 to 99> ; 127 every year

hundred year: UI2 [14 to 15] < 0 to 3> this year is 1900+100*hundred year + year

IV B1 [8] IV<0> = valid ; IV <1> = invalid

SU B1 [16] IV<0> = standard time ; IV <1> = summer time

RES1 B1 [7] <0> reserved for future use

For compatibility with old meters with a circular two digit date it is recommended to consider in any master software the years “00” to “80” as the years 2000 to 2080.

Type G: Compound CP16: Date

2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}

day: UI5 [1 to 5] < 1 to 31 >
“0”: every day

month: UI4 [9 to 12] < 1 to 12 >
“15”: every month

year: UI7 [6 to 8,13 to 16] < 0 to 99 > 127: every year

For compatibility with old meters with a circular two digit date it is recommended to consider in any master software the years “00” to “80” as the years 2000 to 2080.

Type H: Floating point according to IEEE-standard

"Short floating Point Number IEEE STD 754" = R32IEEESTD754

R32IEEESTD754 := R32.23 {Fraction, Exponent, Sign}

Fraction	=	F := UI23 [1to 23] < 0 to $1\text{-}2^{-23}$>	
Exponent	=	E := UI8 [24 to 31] < 0 to 255 >	
Sign	=	S := BS1 [32]	S <0> = positive
			S <1> = negative

F <0> and E <0>	:= (-1) S ∗ 0	= ± zero
F <≠0> and E <0>	:= (-1) S ∗ 2E-126(0.F)	= denormalized numbers
E <1 to 254>	:= (-1) S ∗ 2E-127(1.F)	= normalized numbers
F <0> and E <255>	:= (-1) S ∗ ∞	= ± infinite
F <≠0> and E <255>	:= NaN	= not a number, regardless of S

bits	8	7	6	5	4	3	2	1
octet 1	F = Fraction							
	2^{-16}	2^{-17}	2^{-18}	2^{-19}	2^{-20}	2^{-21}	2^{-22}	2^{-23}
octet 2	F = Fraction							
	2^{-8}	2^{-9}	2^{-10}	2^{-11}	2^{-12}	2^{-13}	2^{-14}	2^{-15}
octet 3	E (LSB)	F = Fraction						
	2^{-0}	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}
octet 4	Sign	E = Exponent						
	S	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}

The following ranges are specified by IEE Std 754–1985 for floating point arithmetics:

<u>Range</u>:$(-2^{128} + 2^{104})$ to $(+2^{128} - 2^{104})$, that is $-3{,}4 \ast 10^{38}$ to $+3{,}4 \ast 10^{38}$

smallest negative number:-2^{-149},that is: $-\ 1{,}4 \ast 10^{-45}$

smallest positive number:$+2^{-149}$, that is: $+\ 1{,}4 \ast 10^{-45}$

Type I: Year down to second

Data field = 0110 (48 bits)

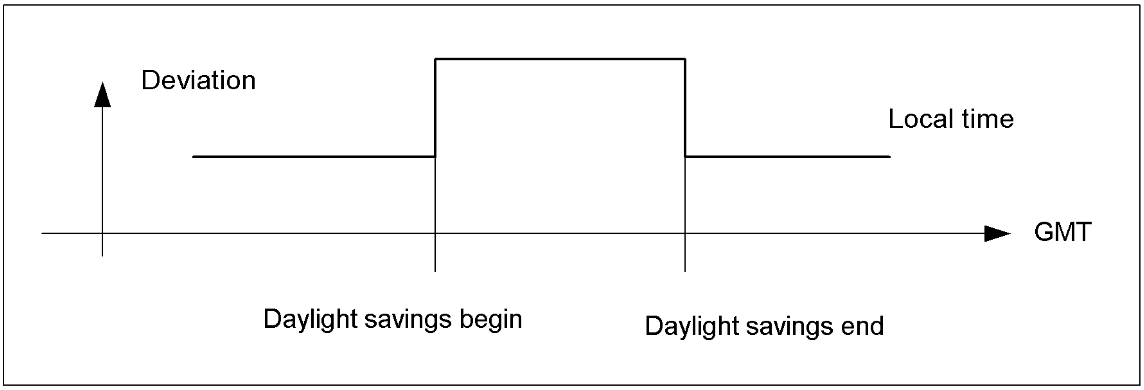
Byte/ bit	msb						lsb
lsB	8	*7*	*6*	*5*	*4*	*3*	*2*	*1*
	16	15	*14*	*13*	*12*	*11*	*10*	*9*
	24	23	22	*21*	*20*	*19*	*18*	*17*
	32	*31*	*30*	*29*	*28*	*27*	*26*	*25*
	40	*39*	*38*	*37*	36	35	34	33
msB	48	47	46	45	44	43	42	41

Local time :		
Second	UI6 [1 to 6]	<0 to 59> ; 63 : every second (1)
Minute	UI6 [9 to 14]	<0 to 59> ; 63 : every minute (1)
Hour	UI5 [17 to 21]	<0 to 23> >; 31 : every hour (1)
Day	UI5 [25 to 29]	<1 to 31> <0> (0= not specified) (1)
Month	UI4 [33 to 36]	<1 to 12> <0> 0= not specified (1)
Year	UI7 [30 to 32+37 to 40]	<0 to 99> <127> 127= not specified (1)
Day of the week	UI3 [22 to 24]	1 to 7> 1= Monday 7= Sunday 0= not specified (3)
Week	UI6 [41 to 46]	<1 to 53> 0= not specified (1)
Time invalid	UI1 [16]	1= invalid ; 0 = valid
Time during daylight saving	UI1 [7]	1= yes (summer time) ; 0 = no
Leap year	UI1 [8]	1= leap year ; 0 = standard year
daylight saving deviation (hour) (2)	UI1 [15]	<0 to 1> (1= + 0 = -)
	UI2 [47 to 48]	<0 to 3> 0 = no daylight saving

(1) other values reserved for future uses

(2) number of hour by which the local time shall be corrected at daylight savings begin

(3) according to Cosem 13/1208/CDV IEC 62056-62



Type J: Time of day

Data field = 0011 (24 bits)

Byte/ bit	msb							lsb
lsB	8	7	*6*	*5*	*4*	*3*	*2*	*1*
	16	15	*14*	*13*	*12*	*11*	*10*	*9*
msB	24	23	22	*21*	*20*	*19*	*18*	*17*

Local time :		
Second	UI6 [1 to 6]	<0 to 59> ; 63 : every second (1)
Minute	UI6 [9 to 14]	<0 to 59> ; 63 : every minute (1)
Hour	UI5 [17 to 21]	<0 to 23> >; 31 : every hour (1)

(1) other values reserved for future uses

Type K: Day light saving

Data field = 0100 (32 bits)

Byte/bit	msb						lsb
LsB	8	7	6	*5*	*4*	*3*	*2*	*1*
	16	15	14	*13*	*12*	*11*	*10*	*9*
	24	23	22	*21*	*20*	*19*	*18*	*17*
MsB	*32*	*31*	*30*	*29*	*28*	*27*	*26*	*25*

Daylight saving enable	UI1 [16]	<0 to 1> (1) 1 enables daylight savings function
Deviation from local time to the Greenwich Mean Time (hour):	UI5 [6 to 8 + 14 to 15]	<0 to 23 > <31> 31= not specified (1)
Daylight savings begin (given in local time):		
Hour	UI5 [1 to 5]	<0 to 23> (1)
Day	UI5 [9 to 13]	<1 to 31> (1)
Month	UI4 [25 to 28]	<1 to 12> (1)
Daylight savings end: (given in local time):		
Day	UI5 [17 to 21]	<1 to 31> (1)
Month	UI4 [29 to 32]	<1 to 12> (1)
Daylight saving deviation (hour) (2)	UI1[24]	<0 to 1> 1= + 0 = -
	UI2 [22 to 23]	<0 to 3> 0 = no daylight saving

(1) other values reserved for future uses

(2) number of hour by which the local time shall be corrected at daylight savings begin

Type L: Listening window management

Data field = 1101 (variable lengh LVAR=EBh)

Byte/bit	Msb							lsb
LsB	8	7	6	5	4	3	2	1
	16	15	14	13	12	11	10	9
	24	23	22	21	20	19	18	17
	32	31	30	29	28	27	26	25
	40	39	38	37	36	35	34	33
	48	47	46	45	44	43	42	41
	56	*55*	*54*	53	52	51	50	49
	64	63	62	61	*60*	*59*	*58*	*57*
	72	71	70	69	68	67	66	65
	80	79	78	77	76	75	74	73
msB	88	87	86	85	84	83	82	81

Bits 85 to 88 : reserved for future use.

This command is used to initialise the window listening management, which defines when the meter is in "normal mode" or "power saved mode".

We choose the week(s) while the meter could be in normal mode. Set to 1 the matching bit(s) : bit 1 to bit 53. The first week of the year is represented by bit 1, .. , the 52nd by the bit 52.

We choose the day(s) while the meter could be in normal mode. All the weeks are identical for this choice. Set to 1 the matching bit(s) : bit 54 to bit 60. Sunday is represented by bit 54, Monday by bit 55, .. , Saturday by bit 60.

We choose the hour(s) while the meter could be in normal mode. All the days are identical for this choice. Set to 1 the matching bit(s) : bit 61 to bit 83. The first hour is represented by bit 61, .. the 24th hour by bit 84.

At one point, the meter is in "normal mode" if the bits for week, day and hour are set to 1. The meter is in "power saved mode" if one or more of the bits for week, day and hour is set to 0.

For example: if bits 3, 55, 56, 61 and 62 are set to 1 and the others are set to 0. The meter is in normal mode between 0 and 2 hour on Monday and Tuesday of the third week of the year.

Annex B
(normative)

Interpretation of Hex-Codes Ah - Fh in BCD-data fields

B.1 General description

B.1.1 Standard Reference

This standard allows multi-digit BCD-coded datafields. It does however not contain information about what happens if a non-BCD hex code (Ah – Fh) is detected by the master software.

B.1.2 Purpose of this proposal

a) Define the treatment of non BCD-digits in slave to master RSP_UD-telegrams

To fully define a master software including error treatment such a definition would be desirable.

b) Utilize these codes for simplified error treatment by slave.

— Simple visible error signalling

To simplify the design of slaves with integrated displays, the above mentioned non-BCD states of the variables should be both transmittable in the form of suitable (Hex) codes but also be displayable directly from the value codes of a 7-segment (usually LCD) display by extending the normal ten entry BCD to 7-segment decoding a 16-entry decoding table.

B.2 Definition

B.2.1 Hex code meanings

a) Ah – Eh

Such a code in any digit position signals a general error of the complete data field. The display at the meter or a remote readout device should display an appropriate symbol at the appropriate display position.

b) Fh

Such a code in the MSD digit position signals a “minus-sign” in front of the remaining (N–1) digit number. In any other digit position it signals an error.

EXAMPLE A 4-digit BCD code of "F321" should be interpreted by the master software as "– 321" and displayed as - 321 on a 4-digit only display.

B.2.2 LCD-Decoding table

a) Decoding table

	1	2	3	4	5	6	7	8	9	Ah	Bh	Ch	Dh	Eh	Fh
"0"	"1"	"2"	"3"	"4"	"5"	"6"	"7"	"8"	"9"	"A"	"b"	"C"	" "	"E"	"–"

Annex C
(normative)

Non metric units

If the VIF-Extension code 3Dh (Non-metric units) is used, the standard metric units of the VIF table is substituted as follows:

Table C.1 — Metric/Nonmetric units

Standard VIF	Standard unit and range	Nonmetric unit and range	Type
E0000nnn	0,001 Wh to 10 000 Wh	0,001 kBTU to 10 000 kBTU	energy
E0010nnn	0,001 l to 10 000 l	0,001 USgal to 10 000 USgal	volume
E1000nnn	0,001 l/min to 10 000 l/min	0,001 USgal/min to 10 000 USgal/min	Flow ext.
E0101nnn	0,001 W to 10 000 W	0,001 mBTU/s to 10 000 mBtu/s	Power
E10110nn	0,001 °C to 1 °C	0,001°F to 1°F	Temp. forward
E10111nn	0,001 °C to 1 °C	0,001°F to 1°F	Temp. return
E11101nn	0,001 °C to 1 °C	0,001°F to 1°F	Cold/warm temperature limit
E11000nn	0,001°C to 1 °C	0,001°F to 1°F	Temp. difference

Annex D
(informative)

Alarm Protocol

The master software polls the maximum 250 alarm devices by requesting time critical data. A slave can transmit an acknowledgement signalling no alarm or an application telegram with the CI-Field 71h to report an alarm state.

CI=71h	Alarm state byte (Binary)

Figure D.1 — Application layer Telegram for an Alarm-Respond

The alarm state is coded with data type D (boolean, in this case 8 bit). Set bits signal alarm bits or alarm codes. The meaning of these bits is manufacturer specific.

The time out for time critical communication is set to 11 bit … 33 bit periods to ensure a fast poll of all alarm devices. With a baud rate of 9 600 baud and all 250 slaves reporting an alarm just in time before a timeout occurs each slave will be polled in periods of maximum 5,5 s. This seems to be fast enough for alarms in building control systems and other applications. For faster alarm systems the number of alarm sensors could be limited to 63 (reducing the worst case overall signal delay to less than 1,5 s or increase the transmission speed to 38 400 Bd and achieve the same speed for up to 250 devices.

The functionality of the FCB- and FCV-Bit should be fully implemented in this alarm protocol to ensure that one-time alarms are safely transmitted to the master. If the slave has reported an one-time alarm and the next REQ_UD1 has a toggled FCB (with FCV = 1) the slave will answer with an E5h signalling no alarm. Otherwise it will repeat the last alarm frame to avoid that the alarm message gets lost.

Annex E
(informative)

Examples

E.1 General

The Application protocol data unit begins with the CI field and does not includes checksum and stop. The Application Protocol data unit is transmitted by the other layers of the protocol.

The following examples give the whole data telegram for a particular link layer: the twisted pair baseband EN 13757-2. Other physical and link layer could be used (EN 13757-4 for example).

E.2 Example for a RSP_UD with variable data structure answer:

(all values are hex.)

68 1F 1F 68	header of RSP_UD telegram (length 1Fh = 31d bytes)
08 02 72	field = 08 (RSP), address 2, CI field 72H (var.,LSByte first)
78 56 34 12	identification number = 12345678
24 40 01 07	manufacturer ID = 4024h (PAD in EN 62056-21), generation 1, water
55 00 00 00	TC = 55h = 85d, Status = 00h, Signature = 0000h
03 13 15 31 00	**Data block 1:** unit 0, storage No 0, no tariff, instantaneous volume,
	12565 l (24 bit integer)
DA 02 3B 13 01	**Data block 2:** unit 0, storage No 5, no tariff, maximum volume flow,
	113 l/h (4 digit BCD)
8B 60 04 37 18 02	**Data block 3:** unit 1, storage No 0, tariff 2, instantaneous energy,
	218,37 kWh (6 digit BCD)
18 16	checksum and stopsign

E.3 Example baud rate switch:

The master switches the slave (in point to point connection) from now 2 400 baud to 9 600 baud.

(all values are hex.)

Master to slave: 68 03 03 68 53 FE BD 0E 16 with 2 400 baud

Slave to master: E5 with 2 400 baud

From that time on the slave communicates with the transmission speed 9 600 baud, if the slave can handle 9 600 baud, otherwise it remains at 2 400 baud.

In bus mode this is followed within < 2min by an acknowledged communication (i.e. SND_NKE) at 9 600 baud:

Master to slave: 10 40 FE 3E 16

Slave to master: E5

E.4 Example Reset with subcode:

The master releases an enhanced application reset to all slaves. All telegrams of the user data type are requested.

(all values are hex.):

Master to Slave: 68 04 04 68 53 FE 50 10 B1 16

Slave to Master: E5

E.5 Writing Data to a Slave

The master can send data to a slave using a SND_UD with CI-Field 51h for mode 1 (or 55h for old mode 2-meters). It should be noted that the data structure in such a write telegram has been changed in contrast to previous definitions by means of leaving out the fixed data header of 12 byte. The following figure shows the data structure for a write telegram. The order of the first three blocks in the following figure can be turned round, but the write only data record is at the end of the telegram. All records are optional.

Primary Address Record	Enhanced Identification Record	Normal Data Records	Write Only Data Records

Figure E.1 — Data Structure for Writing Data

- Primary Address Record:

  The primary address record is optional and consists of three bytes:

DIF = 01h	VIF = 7Ah	Data = Address (1 byte binary)

  With this data record a primary address can be assigned to a slave in point to point connections. The master knows all the used addresses on the bus and forbid setting the address of a slave to an already used address. Otherwise both slaves with the same address could not be read out anymore.

- Enhanced Identification Record:

  With this optional data record the identification (secondary address) can be changed. There are two cases to be distinguished:

  1) Data is only the identification number

DIF = 0Ch	VIF = 79h	Data = Identification No. (8 digit BCD)

2) Data is the complete identification

DIF = 07h	VIF = 79h	Data = complete ID (64 bit integer)

The data is packed exactly as in the readout header of a 72 variable protocol with low byte first for mode 1 and high byte first for mode 2:

Identification No.	Manufacturer ID	Generation	Medium
4 byte	2 byte	1 byte	1 byte

— Normal Data Records:

The data records, which can be read out with a REQ_UD2, are sent back to the slave with the received DIF and VIF and the new data contents. Additional features can be implemented using the generalized object layer (see 6.5).

— Write-Only Data:

Data, which cannot be read out of the slave with a normal data block, can be transmitted using the VIF = 7Fh for manufacturer specific coding. The DIF has a value corresponding to the type and length of data.

After receiving the SND_UD correctly without any error in data link layer the slave answers with an acknowledgement (E5h). The slave decides whether to change variables or not after a data write from the master. In case of errors in executing parts of or whole write instructions the slave can decide whether to change no variables or single correct variables. The slave can report the this errors to the master in the next RSP_UD telegram using some of the methods which are described in 6.6.

There are some methods for implementing write protect, for example allowing only one write after a hardware reset of the processor or enabling write if a protect disable jumper is set.

EXAMPLE (all values are hex.)

1) Set the slave to primary address 8 without changing anything else:

68 06 06 68 53 FE 51 01 7A 08 25 16

2) Set the complete identification of the slave (ID = 01020304 (BCD), Man = 4024h (PAD), Gen = 1, Med = 4 (Heat):

68 0D 0D 68 53 FE 51 07 79 04 03 02 01 24 40 01 04 95 16

3) Set identification number of the slave to "12345678 (BCD)" and the 8 digit BCD-Counter (unit 1 kWh) to 107 kWh.

68 0F 0F 68 53 FE 51 0C 79 78 56 34 12 0C 06 07 01 00 00 55 16

E.6 Configuring Data Output (Normative)

For default the slave transmits all his data with a RSP_UD. It could be useful for some applications to read only selected data records out of one or more devices. There are two ways to select data records:

E.6.1 Selection without specified data field

The selection of the wanted data records can be performed with a SND_UD (CI-Field = 51h/55h) and data records containing the data field 1000b, which means "selection for readout request". The following VIF defines the selected data as listed in EN 1434–3 and no data are transmitted. The answer data field is determined by the slave. The master can select several variables by sending more data blocks with this data field in the same telegram.

Special multiple values can be selected with the following methods:

— Any VIF:

  The VIF-Code 7Eh (any VIF) is especially for readout request of "all VIF" from the slave and can be interpreted as a selection wildcard for the value information field.

— Global readout request:

  The DIF-Code 7Fh is defined as "selection of all data for readout request", i.e. all storage numbers, units, tariffs and functions. If this DIF is the last byte of user data or the VIF = 7Eh follows, then all data is requested. So the selection of all data of one slave can de done with a SND_UD and the character 7Fh as the user data. If there follows a DIF unequal to 7Eh, then all data records with this VIF are selected for readout.

— All Tariffs:

  The highest tariff number in the selection record is defined as selection of "all tariffs". For example the tariff 1111b (15) means selection of all tariffs in a record with two DIFE´s.

— All Storage Numbers:

  A selection of all storage numbers can be done with the maximum storage number if there is a minimum of one DIFE. For example the highest storage number is 1Fh (31d) with one DIFE and 1FFh (511d) with two DIFE´s.

— All Units:

  "All units" can be selected by using a data record header with minimum two DIFE´s and the highest unit number.

— High Resolution Readout:

  The master can select the slave to answer with the maximum resolution to a given value / unit by a VIF with "nnn" = 000 (minimum exponent for range coding). The meter may then answer with a resolution of e.g. 1mWh (VIF = 0000000b) or some higher decimal value if required. The unit values have been chosen so that their minimum provides sufficient resolution even for calibration. A readout request for a VIF with "nnn" = max (maximum exponent for range coding) signals a request for the standard resolution of the meter.

After the next REQ_UD2 the slave answers with the selected data in his own format, if the requested data are available. Otherwise the slave transmits his normal data and the master has to find out that the data are not the requested one. If there are more than one variables with the selected VIF, the device should send all these data records.

E.6.2 Selection with specified data field

The master is able to perform a readout request with a specified data field by using the object action ”add to readout list” (VIFE = E000 1100b) from VIFE-table for object actions (see clause 9). The master transmits a SND_UD (CI-Field = 51h/55h) with a data record which consists of the desired DIF (data field), VIF and the VIFE = 0Ch / 8Ch. No data follows this VIFE and the slave should ignore the data field on reception. The slave should transmit this data record with the requested data field from now on, if he is capable of this. If the slave doesn´t support this data field (data coding), it can report a record error using one of the VIFE = E000 011x (data class not implemented or data size not implemented).

E.6.3 Deselection of data records

The master can release a reset of the application layer and especially a fallback to the slaves standard RSP_UD-Telegram by transmitting a SND_UD with the CI-Field 50h.

Single data records can be deselected by transmitting a data record with DIF, VIF and the VIFE for the object action ”Delete from Readout-List” (VIFE = E000 1101b).

If the selected data is supported by the slave but too long for one RSP_UD telegram (especially for readout of all historic values), the slave transmits an additional data record consisting only of the DIF=1Fh, which means that more data records follow in the next respond telegram. In this case the master reads out the slave again until the respond telegram is only an 0E5h (no data) or there is no DIF = 1Fh in the RSP_UD.

To avoid lost of data respond telegrams the slave should in this case support the Frame Count Bit (FCB). If the master wants to premature end such a multi telegram sequential readout of the selected data, it may send an application reset with CI = 50h instead of further REQ_UD2´s.

EXAMPLES

1) A slave with address 7 is to be configured to respond with the data records containing volume (VIF = 13h: volume, unit 1l) and flow temperature (VIF = 5Ah: flow temp., unit 0.1 °C):

   68 07 07 68 | 53 07 51 08 13 08 5A 28 16

2) A slave with address 1 is to be configured to respond with all storage numbers, all tariffs, and all VIF´s from unit 0:

   68 06 06 68 53 01 51 C8 3F 7E 2A 16

3) A slave with address 3 is to be configured to respond with all data for a complete readout of all available. After that the master can poll the slave to get the data:

   68 04 04 68 53 03 51 7F 26 16

With these actions the master can alter the data of the slaves or configure the output data of the slaves (actions 12 and 13). The actions 0 to 6 alter the data of the slave by replacing the old data (action 0, equals to data write without VIFE) or do arithmetical or logical operations with the old and the transmitted data.

Note that this method of configuring the readout list (action 12 and 13) allows not only the adding but also the removal of elements in contrast to the method of using the DIF = 1000b-type of readout request (described before).

All these actions can be used for normal slaves and for intelligent master which are manipulated by a higher order master.

The functions "Add entry" and "Delete entry" are useful to tell an intelligent master to add e.g. a new data record like maximum or minimum values of any slave.

With the action "freeze data to storage #" the master can tell the slave to freeze the actual value corresponding to the transmitted VIF, unit, tariff and function to a certain storage number given in the DIF/DIFE´s. In this case the data field inside the VIF has got the value 0000b (no data). This action allows freeze of selected values or multiple freeze with VIF = 7Eh (all VIF). The date / time should also be freezed to the same storage number.

EXAMPLES

1) Set the 8 digit BCD-Counter (instantaneous, actual value, no tariff, unit 0) with VIF = 06 (1kWh) of the slave with address 1 to 107 kWh:

   68 0A 0A 68 53 01 51 0C 86 00 07 01 00 00 3F 16

2) Same as in example 1) but add 10 kWh to the old data:

   68 0A 0A 68 53 01 51 0C 86 01 10 00 00 00 48 16

3) Add an entry with an 8 digit BCD-Counter (instantaneous, actual value, no tariff, unit 0, 1kWh) with the start value of 511 kWh to the data records of the slave with address 5:

   68 0A 0A 68 53 05 51 0C 86 08 11 05 00 00 59 16

4) Freeze actual flow temperature (0.1 °C: VIF = 5Ah) of the slave with address 1 into the storage number 1:

   68 06 06 68 53 01 51 40 DA 0B CA 16

E.7 FCB-Bit and Selection

FCB-Implementation slave

A slave with implemented secondary addressing and with implemented FCB-administration has an additional set of 0, 1 or 2 separate "Last Received FCB"-memory Bit(s) for all communication via the pseudo primary address 253 (FDh). If it can communicate also alternatively over some other primary address (except the special addresses 254 and 255) an additional set of 0, 1 or 2 "Last received FCB"-memory bit(s) for each of these primary addresses is required. A valid selection telegram will not only set the internal selection bit but will also clear all 0, 1 or 2 internal "Last received FCB"-memory bit(s) associated with secondary addressing via the pseudo primary address 253 (FDh). The master will start the communication (REQ_UD2 or SND_UD) after any selection telegram (CI = 52h) with the FCV-Bit set and the FCB-Bit set. If a slave has more than one alternative secondary identification, only a single set of 0, 1 or 2 "Last received FCB"-memory bit(s) for all secondary addresses is required.

FCB-Implementation master

The master implements a separate pair of "Next FCB image"-Bits for pseudo primary address 253 (FDh) as for each other primary address. Although these "Next FCB image"-bits might be used for many slaves, no confusion exists, since for accessing another slave a selection telegram is required which will define the future FCB sequence both for slave and master.

E.8 Special Slave Features

E.8.1 General

Some optional or recommended features of the slaves will be described in this sub-clause.

E.8.2 Use of the fabrication number

The fabrication number is a serial number allocated during manufacture. It is part of the variable data block (DIF = 0Ch and VIF = 78h) and coded with 8 BCD packed digits (4 Byte).

EXAMPLE

68 15 15 68	header of RSP_UD telegram (length 1Fh = 31d bytes)
08 02 72	C-field = 08 (RSP), address 2, CI field 72H (var.,LSByte first)
78 56 34 12	identification number = 12345678
24 40 01 07	manufacturer ID = 4024h (PAD in EN 61107), generation 1, water
13 00 00 00	TC = 13h = 19d, Status = 00h, Signature = 0000h
0C 78 04 03 02 01	fabrication number = 01020304
9D 16	checksum and stop sign

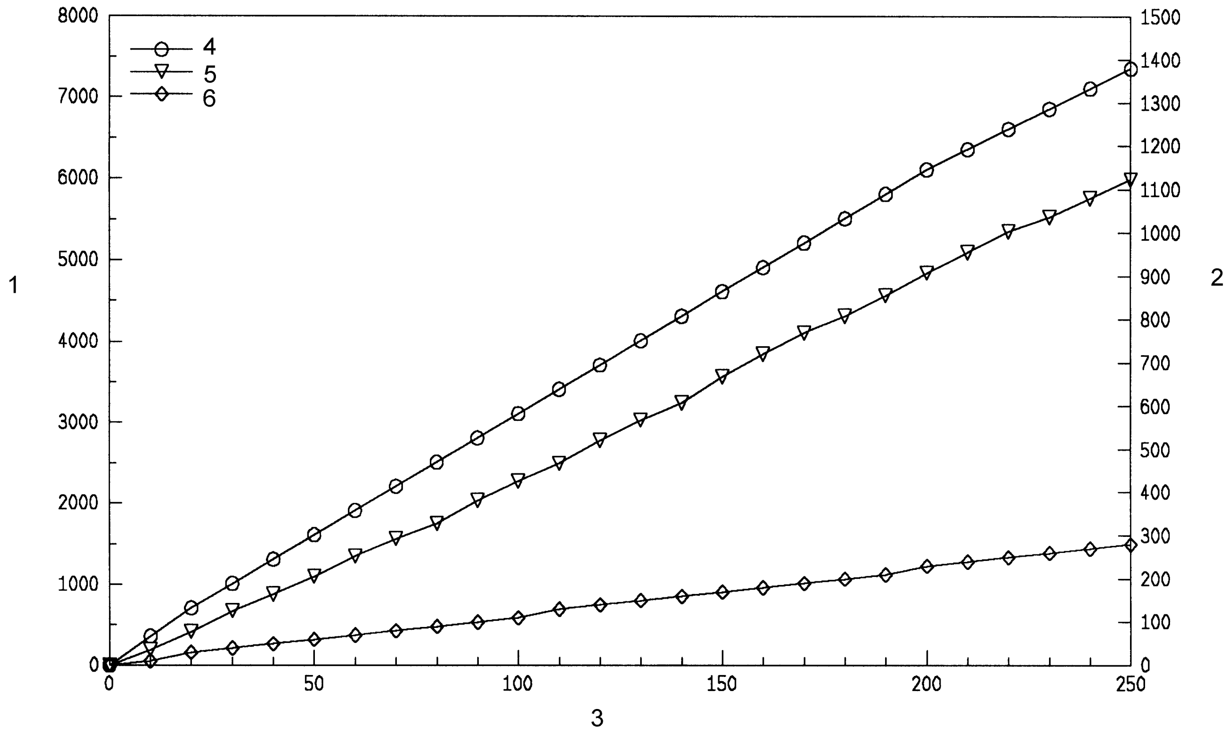
The use of this number is recommended if the identification number is changeable. In this case two or more slaves can get the same secondary address and can not be uniquely selected. The fabrication number together with manufacturer, version and medium field build an unique number instead. Suitable masters use this number for an enhanced selection method if two or more slaves have the same identification number (see 9.3).

Annex F
(informative)

Secondary Search

F.1 General

The search procedure has been simulated to find the minimum, the average and the maximum number of selections as a function of the number of slaves. For the minimum number of attempts the optimum distribution of the identification numbers was chosen, for the maximum number the most unfavourable, and for the average number of attempts a random distribution. The following diagram shows the result of these calculations:



Key

1 Number of selections (worst case)

2 Number of selections (best case/random case)

3 Number of slaves

Figure F.1 — Number of Selections with Wildcard Searching Procedure

F.2 Instructions for implementation of Wildcard Search

The following program flow diagram shows the realisation of the Wildcard Searching procedure, whereby the search is made only with the identification number. The codes for manufacturer, version and medium are in general specified with wildcards, but can be changed by the user in order (for example) to locate all meters from a particular manufacturer. In order to avoid the categorization by a factor of eight of the "For-To" loops for the eight positions, the array "Value" is defined with 8 byte numbers, which are intended to define the contents of the positions. The digit number of the identification number which is presently running is noted in the variable "Pos" of type byte.

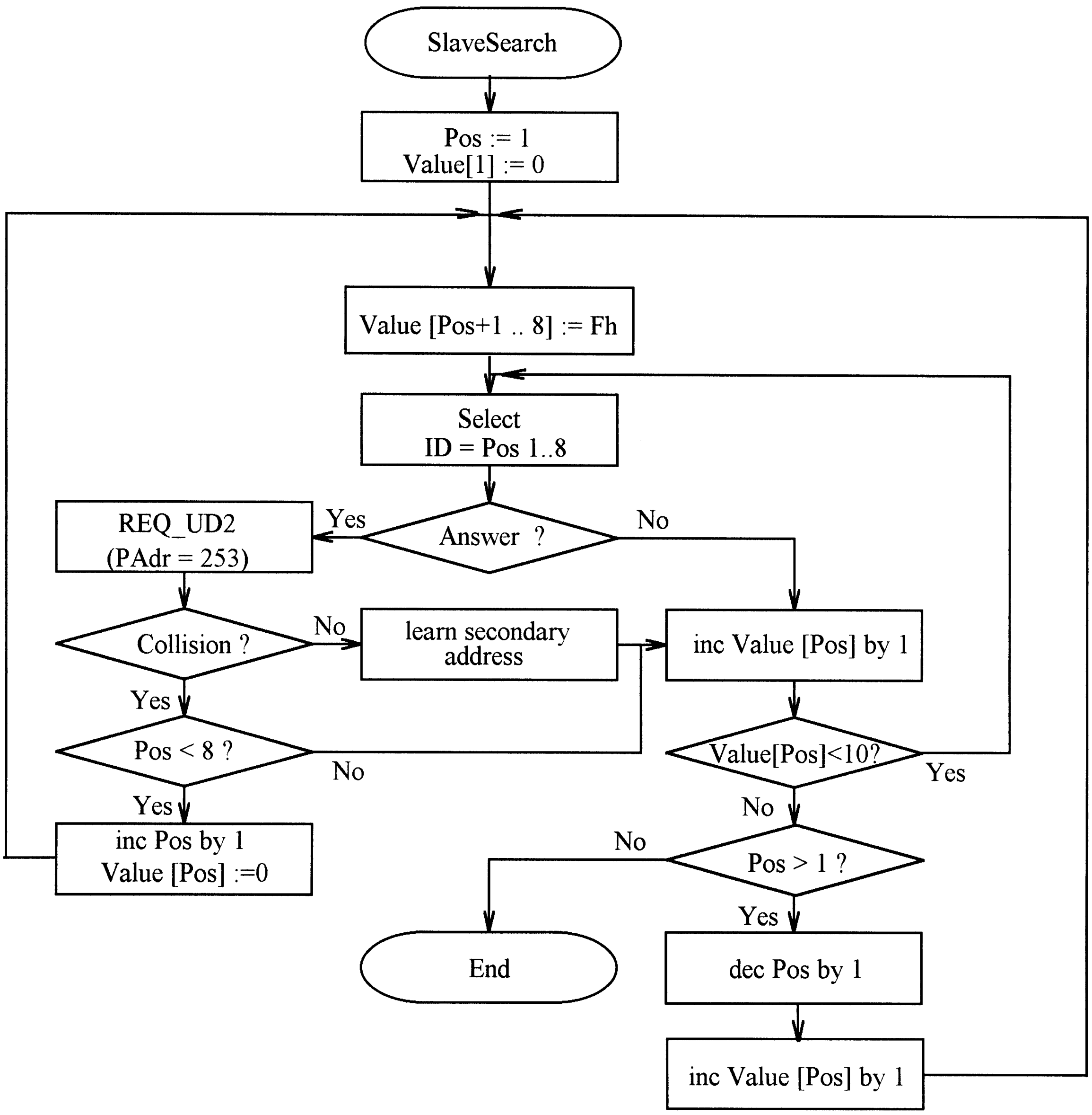


Figure F.2 — Flow Diagram for Slave Search with Wildcards

The routine begins at the first position, and implements the following actions for the value of this position from 0 to 9:

- selection with the ID-Nr. Pos 1, Pos 2, …, Pos 8;
- if no reply, Value [Pos] is raised by 1;
- if there is a reply, a REQ_UD2 is sent to address 253, and if the telegram is correctly received the secondary address is learnt and the Value [Pos] raised by 1;
- if there is a collision a jump is made to the next position (Pos increased by 1), as long as the last position has not yet been reached;
- after going through a complete position from 0 to 9 the subroutine proceeds to the next lower position, or ends the search if the position Nr. 1 has already been processed.

EXAMPLE

The next figure shows an example for secondary addresses in order from top to bottom, as they will be found by the master software:

No.	Identification-Nr.	Manufacturer. (hex.)	Version (hex.)	Device type(hex.)
1	14 491 001	1057	01	06
2	14 491 008	4567	01	06
3	32 104 833	2010	01	02
4	76 543 210	2010	01	03

Figure F.3 — Secondary Addresses found with a Wildcard Search of Four Slaves

Search Process:

1) Start with ID = 0FFFFFFF: no reply
2) ID = 1FFFFFFF: collision between Nr.1 and Nr.2
3) ID = 10FFFFFF, 11FFFFFF, 12FFFFFF, 13FFFFFF: no reply
4) ID = 14FFFFFF: collision between Nr.1 and Nr.2
5) Repeated steps 3 to 4 up to the ID = 1449100F
6) Learn ID = 14491001 and 14491008
7) Go backwards to 19999999
8) ID = 2FFFFFFF: no reply
9) ID = 3FFFFFFF: learn ID = 32104833
10) .ID = 4FFFFFFF, 5FFFFFFF, 6FFFFFFF: no reply
11) ID = 7FFFFFFF: learn ID = 76543210
12) ID = 8FFFFFFF, 9FFFFFFF: no reply
13) End of the Search

Annex G
(informative)

International reference works

WWW-server operated by the m-bus-user group at “http://www.m-bus.com” provides a forum for up to date information on the M-bus.

Annex H
(informative)

Meaning of "device type specific" parameters of Mbus for Radio products

H.1 VIF VIFE = FDh 17h (error flag)

For a Radio product, the first byte of error flag has the following meaning:

B7	b6	b5	b4	B3	b2	b1	b0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

b0 : 1 = tamper

b1 : 1 = battery low

b2 : 1 = external alarm

b3 : 1 = battery cut

b4 - b7 = RSSI coded as :

B4	b5	b6	B7	dBm	Comment
0	0	0	0		RSSI not available
1	0	0	0	-100 or less	
0	1	0	0	-90	
1	1	0	0	-80	
0	0	1	0	-60	
1	0	1	0	-50	
0	1	1	0	-40	
1	1	1	0	-30	
0	0	0	1	-20	
1	0	0	1	-10	
0	1	0	1	0	
1	1	0	1	10	
0	0	1	1	20 or more	
1	0	1	1		reserved for future use
0	1	1	1		reserved for future use
1	1	1	1		reserved for future use

H.2 VIFE = FDh 1 Fh for passing remote control on a node

For a Radio product, the bits of the first byte of remote control have the following meaning:

b7	b6	b5	b4	b3	b2	b1	b0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

b0	b1	**Power adjust** For a Radio product, the bits of the first byte of remote control have the following meaning:
0	0	do nothing
0	1	reserved for future use
1	0	decrease power (one step)
1	1	increase power (one step)

b3	**b4**	**b5**	**Test mode**
0	0	0	do nothing
0	0	1	test mode : temporary emission of permanent "0"
0	1	0	test mode : temporary emission of permanent "0101"
0	1	1	test mode : temporary emission of permanent carrier , no modulation
1	0	0	test mode : temporary emission of permanent "1"
1	0	1	test mode : temporary reception
1	1	0	reserved for future use
1	1	1	reserved for future use

b6	**Mode select**
0	power saved mode
1	normal mode

b7	b8	reserved for future use

Bibliography

[1] EN 1434–3:1997, *Heat meters — Part 3: Data exchange and interfaces*.

[2] EN 13757-1:2002, *Communication systems for meters and remote reading of meters - Part 1: Data exchange.*

[3] EN 62056-21, *Electricity metering — Data exchange for meter reading, tariff and load control — Part 21: Direct local data exchange (IEC 62056-21:2002).*

BSI — British Standards Institution

BSI is the independent national body responsible for preparing British Standards. It presents the UK view on standards in Europe and at the international level. It is incorporated by Royal Charter.

Revisions

British Standards are updated by amendment or revision. Users of British Standards should make sure that they possess the latest amendments or editions.

It is the constant aim of BSI to improve the quality of our products and services. We would be grateful if anyone finding an inaccuracy or ambiguity while using this British Standard would inform the Secretary of the technical committee responsible, the identity of which can be found on the inside front cover. Tel: +44 (0)20 8996 9000. Fax: +44 (0)20 8996 7400.

BSI offers members an individual updating service called PLUS which ensures that subscribers automatically receive the latest editions of standards.

Buying standards

Orders for all BSI, international and foreign standards publications should be addressed to Customer Services. Tel: +44 (0)20 8996 9001. Fax: +44 (0)20 8996 7001. Email: orders@bsi-global.com. Standards are also available from the BSI website at http://www.bsi-global.com.

In response to orders for international standards, it is BSI policy to supply the BSI implementation of those that have been published as British Standards, unless otherwise requested.

Information on standards

BSI provides a wide range of information on national, European and international standards through its Library and its Technical Help to Exporters Service. Various BSI electronic information services are also available which give details on all its products and services. Contact the Information Centre. Tel: +44 (0)20 8996 7111. Fax: +44 (0)20 8996 7048. Email: info@bsi-global.com.

Subscribing members of BSI are kept up to date with standards developments and receive substantial discounts on the purchase price of standards. For details of these and other benefits contact Membership Administration. Tel: +44 (0)20 8996 7002. Fax: +44 (0)20 8996 7001. Email: membership@bsi-global.com.

Information regarding online access to British Standards via British Standards Online can be found at http://www.bsi-global.com/bsonline.

Further information about BSI is available on the BSI website at http://www.bsi-global.com.

Copyright

BSI
389 Chiswick High Road
London
W4 4AL