

Implementation of
ISO 10368:1992

Freight thermal containers — Remote condition monitoring

UDC 621.869.88-758.36:621.56:681.3:621.39

NO COPYING WITHOUT BSI PERMISSION EXCEPT AS PERMITTED BY COPYRIGHT LAW



Document Sharing Hub

Committees responsible for this British Standard

The preparation of this British Standard was entrusted to Technical Committee PKM/18, Freight containers, upon which the following bodies were represented:

- British Industrial Truck Association
- British International Freight Association
- British Railways Board
- Chamber of Shipping
- Department of Transport (Transport Industries)
- European Portable Tank Association (British Section)
- Federation of the Electronics Industry
- Health and Safety Executive
- Institute of Logistics
- Lloyd’s Register of Shipping
- Ports’ Safety Organisation
- Road Haulage Association Ltd.
- Shipowners Refrigerated Cargo Research Association
- Society of Motor Manufacturers and Traders Limited

This British Standard, having been prepared under the direction of the Consumer Products and Services Sector Board, was published under the authority of the Standards Board and comes into effect on 15 June 1995

© BSI 05-2000

The following BSI references relate to the work on this standard:
Committee reference PKM/18
Draft for comment 90/43753 DC

ISBN 0 580 24356 7

Amendments issued since publication

Amd. No.	Date	Comments

Contents

	Page
Committees responsible	Inside front cover
National foreword	ii
Foreword	v
Text of ISO 10368	1

National foreword

This British Standard reproduces verbatim ISO 10368:1992 implements it as the UK national standard.

This British Standard is published under the direction of the Consumer Products and Services Sector Board whose Technical Committee PKM/18 has the responsibility to:

- aid enquirers to understand the text;
- present to the responsible international committee any enquiries on interpretation, or proposals for change, and keep UK interests informed;
- monitor related international and European developments and promulgate them in the UK.

NOTE International and European Standards, as well as overseas standards, are available from Customer Services, BSI, 389 Chiswick High Road, London W4 4AL.

A British Standard does not purport to include all the necessary provisions of a contract. Users of British Standards are responsible for their 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, pages i and ii, the ISO title page, pages ii to vi, pages 1 to 76, an inside back cover and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

INTERNATIONAL STANDARD

ISO
10368

First edition
1992-07-01

Freight thermal containers — Remote condition monitoring

*Conteneurs à caractéristiques thermiques — Système de pilotage à
distance des groupes frigorifiques*



Reference number
ISO 10368:1992(E)

AVA **Documents**
Document Sharing Hub

Contents

	Page
Foreword	v
Section 1. General	
Introduction	1
1.1 Scope	1
1.2 Normative references	1
1.3 Definitions	1
Section 2. Performance requirements	
2.1 Scope	3
2.2 Requirements	3
2.2.1 System components	3
2.2.2 Performance function	3
2.2.3 Performance constraints	5
Section 3. System compatibility requirements	
3.1 Scope	7
3.2 Communications protocol	7
3.2.1 MMU to MDCU communications	7
3.2.2 MDCU to LRCD communications	26
3.2.3 MDCU to HRCD communications	34
3.2.4 RCD to controller communications	41
3.2.5 State diagram	43
3.2.6 Error detection algorithm	47
3.3 Data-logging formats	48
3.3.1 Control words	48
3.3.2 Headers	49
3.3.3 Readings	49
3.3.4 Block 0 definitions	53
3.3.5 Event marks	53
3.4 Message definitions	53
3.4.1 Message packet exchange	53
3.4.2 Extended commands	54
3.4.3 Notation	54
3.4.4 Echo	54
3.4.5 Controller read	55
3.4.6 Controller write	58
3.4.7 Enter untimed process	61
3.4.8 Enter timed process	64
3.4.9 Undefined	65
3.4.10 Reserved	66
3.4.11 Illegal	66
3.5 Low data rate physical requirements — LDCU to LRCD	66
3.5.1 Frequency	66
3.5.2 Modulation method	66
3.5.3 Baud rate	66
3.5.4 Transmission mode	66
3.5.5 Injection mode	67
3.5.6 Receiver sensitivity	67
3.5.7 Non-transmission impedance	67
3.5.8 Bit synchronization	67
3.5.9 Carrier setup time	67

	Page
3.5.10 Out-of-band filtering requirements for “high data rate compatibility”	67
3.6 High data rate physical requirements — HDCU to HRCD	67
3.6.1 Modulation method — Broad band	67
3.6.2 Transmission mode	68
3.6.3 Injection mode	68
3.6.4 Output/input impedance	68
3.6.5 Power density function	68
3.6.6 Synchronization method	68
3.6.7 Demodulation method	69
3.6.8 Receiver sensitivity	69
3.6.9 Data link protocol	69
3.6.10 Out-of-band filtering requirements for “low data rate” compatibility	74
Annex A (informative) Bibliography	Inside back cover
Figure 1 — Remote condition monitoring system components layout	4
Figure 2 — Remote condition monitoring — Communications interfaces	7
Figure 3 — Message format overview	8
Figure 4 — Information transfer format from the MMU to the MDCU	8
Figure 5 — Information transfer format returned to the MMU from the MDCU	8
Figure 6 — Format of RCD map command	14
Figure 7 — Device poll (total 8 bytes or 64 bits)	16
Figure 8 — Block structure	29
Figure 9 — Poll session transfer diagram	31
Figure 10 — Xmit1 session transfer diagram	32
Figure 11 — Xmit2 session transfer diagram	33
Figure 12 — Identified map session transfer diagram	34
Figure 13 — Unidentified map session transfer diagram	34
Figure 14 — Interactive poll session transfer diagram	35
Figure 15 — Routing byte definition	36
Figure 16 — RCD/device poll session transfer diagram	37
Figure 17 — RCD/device xmit1/2 session transfer diagram	38
Figure 18 — RCD/device interactive 1/2 session transfer diagram	39
Figure 19 — RCD/device map session transfer diagram	41
Figure 20 — RCD/controller format	42
Figure 21 — Master message transaction state diagram	44
Figure 22 — Slave (controller) message transaction state diagram	46
Figure 23 — Hardware read message	56
Figure 24 — Hardware write message	59
Figure 25 — Filter specifications — High data rate compatibility	67
Figure 26 — Power density function	68

	Page
Figure 27 — Filter specifications — Low data rate compatibility	74
Table 1 — MDCU interrogation directed command types	9
Table 2 — MDCU power line directed command types	11
Table 3 — Routing byte data rate bit assignments	11
Table 4 — RCD status state bit definitions	12
Table 5 — Map subcommand definitions	14
Table 6 — MDCU reply types	18
Table 7 — MDCU status state bit definitions	18
Table 8 — RCD directed command types	20
Table 9 — RCD directed slave reply types	23
Table 10 — Network control characters	26
Table 11 — Low data rate routing byte definitions	28
Table 12 — Prefix message types	30
Table 13 — Power line timeout variables	31
Table 14 — Message types	36
Table 15 — Master (RCD) state diagram transition conditions	45
Table 16 — Slave (controller) state diagram transition conditions	47
Table 17 — Locations of fields in the first byte of control words	48
Table 18 — Interpretation of message contents	48
Table 19 — Location of fields	50
Table 20 — Definition of field contents	50
Table 21 — Location of additional fields	51
Table 22 — Definition of field contents	51
Table 23 — Definition of additional sensor fields — four sensor system, low resolution	52
Table 24 — Location of fields	52
Table 25 — Definition of field contents	52
Table 26 — Definition of block 0 of the data logger	53
Table 27 — Function cross-reference table	54
Table 28 — Controller sensor channel identification numbers	62
Table 29 — Baud rate definitions	65
Table 30 — Codes for data link protocol	75
Table 31 — (32,8) ECC and EDC codes	76

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

International Standard ISO 10368 was prepared by Technical Committee ISO/TC 104, *Freight containers*, Sub-Committee SC 2, *Specific purpose containers*.

Annex A of this International Standard is for information only.

Section 1. General

Introduction

During the preparation of this International Standard, information was gathered on patents upon which application of this standard might depend.

For the low data rate system of **3.5**, the relevant patents were identified as belonging to

Thermo King Corporation
314 W. 90th Street
Minneapolis, Minnesota 55420
USA

For the high data rate system of **3.6**, the relevant patents were identified as belonging to

Adaptive Networks Incorporated
1505 Commonwealth Ave.
Suite 30
Brighton, Massachusetts 02135
USA

ISO cannot give authoritative or comprehensive information about the evidence, validity or scope of patent and like rights. The patent holders have stated that licences will be granted under reasonable terms and conditions. Communications on this subject should be addressed to either Thermo King Corporation or Adaptive Networks Incorporated.

1.1 Scope

This International Standard establishes the information and interfaces required to permit complying central monitoring and control systems employed by one carrier or terminal to interface and communicate with complying remote communication devices of differing manufacture and configuration used by other carriers and terminals.

The data-logging formats and message protocols outlined in this International Standard apply to all currently available data rate transmission techniques. These formats and protocols also apply to all future techniques designed to be an ISO standard compatible system.

The performance requirements for the monitoring, communication and control system are given in section 2. The system compatibility requirements are given in section 3. All sections of this International Standard apply to all implementations, except where specified.

1.2 Normative references

The following standards contain provisions which, through reference in this text constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 1496-2:1988, *Series 1 freight containers — Specification and testing — Part 2: Thermal containers*.

ISO 9711-2:1990, *Freight containers — Information related to containers on board vessels — Part 2: Telex data transmission*.

1.3 Definitions

For the purposes of this International Standard, the following definitions apply.

1.3.1

remote communications device (RCD)

device which is physically a part of the refrigeration machinery and which communicates with any complying central monitoring and control systems (CMCSs) using the refrigeration machinery power distribution system as, the data transmission medium. See Figure 1 and Figure 2

there are two distinct types of RCD:

- a stand-alone remote communications device (sRCD) (see **1.3.9**);

— an integrated remote communications device (iRCD) (see 1.3.10).

an sRCD is defined as a device which cannot implement the hardware write messages defined in 3.4.6.1

1.3.2

central monitoring and control system (CMCS)

system consisting of hardware and software which monitors and controls one or more remote communications devices (RCDs). A typical system consists of at least

- a) operator interface devices,
- b) a master monitoring unit (MMU), and
- c) power line data link equipment, such as a multiple data rate central control unit (MDCU).

1.3.3

master monitoring unit (MMU)

central processing unit such as a computer which contains specific hardware and software to control the entire remote condition monitoring system. It is the interface between the human operator and the network

1.3.4

multiple data rate central control unit (MDCU)

device which forms the link between the master monitoring unit (MMU) and the three-phase power line bus which contains the individual remote communications devices (RCDs). An MDCU consists of two components as follows:

- a) a central control unit capable of receiving and transmitting at the data rates which meet the requirements of this International Standard;
- b) a central control interface.

1.3.5

high data rate remote communications device (HRCD)

remote communications device (RCD) which transmits data at a high data rate, e.g. 19 200 baud

1.3.6

low data rate remote communications device (LRCD)

remote communications device (RCD) which communicates data at a low data rate, e.g. 1 200 baud

1.3.7

high data rate central control unit (HDCU)

device which links the master monitoring unit (MMU) and the power line network, communicating with the high data rate remote communications devices (HRCDs)

1.3.8

low data rate central control unit (LDCU)

device which links the master monitoring unit (MMU) and the power line network, communicating with the low data rate remote communications devices (LRCDs)

1.3.9

stand-alone remote communications device (sRCD)

slave remote communications device (RCD) which, with limited capabilities, merely monitors a container refrigeration unit. An sRCD can be either high or low data rate

1.3.10

integrated remote communications device (iRCD)

slave remote communications device (RCD) which interfaces to a refrigeration unit controller via an EIA RS232-C serial interface and can control the refrigeration machinery. An iRCD can be either high or low data rate

1.3.11

controller

device that monitors and controls the refrigeration machinery

Section 2. Performance requirements

2.1 Scope

This section specifies the performance requirements of central monitoring and control systems (CMCSs) necessary for them to interface and communicate with complying remote communications devices (RCDs).

2.2 Requirements

2.2.1 System components

2.2.1.1 *Remote condition monitoring system components*

A single remote condition monitoring system consists of a maximum of one master monitoring unit (MMU) and one multiple data rate central control unit (MDCU). See Figure 1 a).

2.2.1.2 *Multiple data rate central control unit (MDCU)*

An MDCU may include one high data rate central control unit (HDCU) and one low data rate central control unit (LDCU). If an HDCU and an LDCU are both present, the HDCU and LDCU are joined together by a central control unit (CCU) interface, and the three components together form the MDCU.

2.2.1.3 *MMU/MDCU interface*

The preferred method of connecting the MMU to the MDCU complex is through a single port as shown in Figure 1 a). However, certain expansion paths may require multiple connections as shown in Figure 1 b).

2.2.1.4 *High and low data rate remote communications devices (HRCDs and LRCDs)*

HRCDs and LRCDs shall be able to coexist on the same power line network and not interfere with simultaneous communications with either the HDCU or the LDCU.

2.2.1.5 *MDCU components*

An MDCU may consist of either a single HDCU (to communicate with the HRCDs on the network) or a single LDCU (to communicate with LRCDs on the network). However, all signalling protocols, data-logging formats, power levels, insertion rates and other physical requirements shall be identical to that which would be used for a combined system and therefore must be compatible. Refer to 3.2 and 3.3 for the required protocol and data-logging formats.

2.2.2 Performance function

2.2.2.1 *Standard message*

All RCDs shall respond to a minimum list of standardized enquiries (see 2.2.2.4) and commands with a standardized reply or acknowledgement.

2.2.2.2 *Acknowledgement message*

The RCD shall send an acknowledgement message for all commands and enquiries that are received and understood.

2.2.2.3 *“Not able” message*

If the RCD is not capable of executing a command received or of responding to an enquiry because of the configuration of the RCD and the thermal control machinery, it shall respond with a “Not able” message.

2.2.2.4 *Required enquiries*

All RCDs shall respond to the following required enquiries.

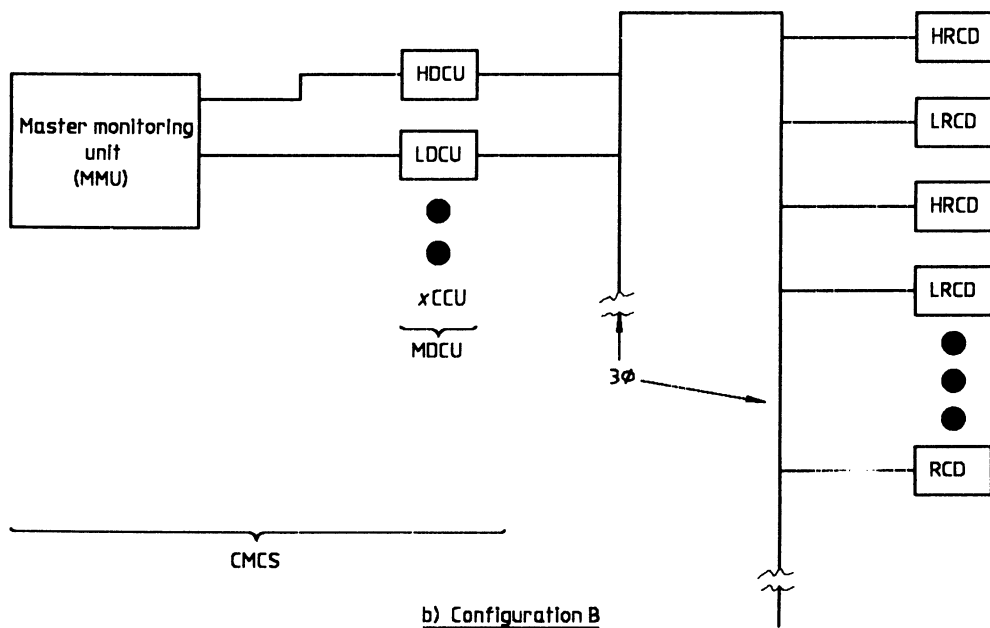
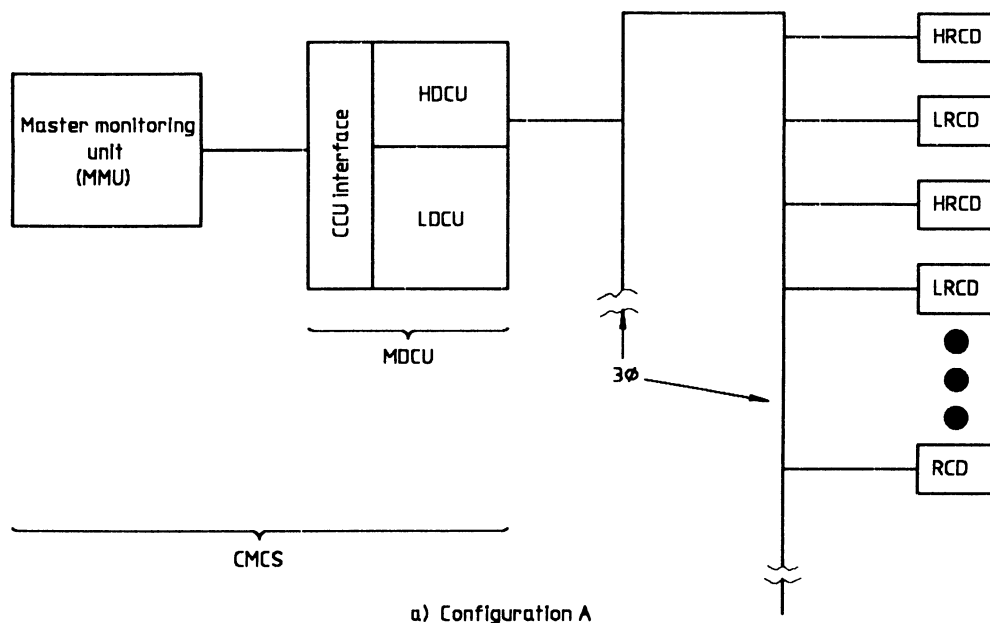
2.2.2.4.1 Identification number: For an integrally refrigerated or thermal container this will be the container ISO number comprising a 4-letter alphabetical prefix and a 7-digit suffix (including the check digit). Where a demountable marine clip-on unit (MCOU) is used, the identification number will be the MCOU number in ISO format.

2.2.2.4.2 Porthole container number: This response will be in addition to the identification number for MCOU systems.

2.2.2.4.3 Porthole number change: This is recorded in the RCD memory in alphanumerical format together with the time of the change.

2.2.2.4.4 Return air temperature: In the form of a positive or negative value, expressed in degrees Celsius to one decimal place, within the range – 30,0 °C to + 38,0 °C.

2.2.2.4.5 Supply air temperature: Expressed in the same format as 2.2.2.4.4.



H = high data rate
L = low data rate
3φ = three-phase power mains

Figure 1 — Remote condition monitoring system components layout

2.2.2.4.6 RCD manufacturer and type: Consisting of a unique identification number registered and controlled by ISO, and for which ISO/TC 104 is the registration authority.

2.2.2.5 Optional standard enquiries

Other optional enquiries are standardized. RCDs arid refrigeration machinery so equipped shall respond to the following enquiries. RCDs not so equipped shall respond "Not able" (see 2.2.2.3).

2.2.2.5.1 Operating mode: Full cool, Partial or Lower capacity cool, Modulated cool, Fans only or Null mode, Defrost, Heat, Off.

2.2.2.5.2 Set-point temperature: Expressed in the same format as 2.2.2.4.4.

2.2.2.5.3 Alarms: High refrigeration pressure, Temperature out of range, Low compressor oil pressure, Defrost/Heat/Overheat, Compressor overload, Controller failure, Sensor failure — Return air, Sensor failure — Supply air, Power off, Amperage draw too high, Amperage draw too low, Defrost (out of time). (Capacity for future development, e.g. controlled atmosphere.)

2.2.2.5.4 All current alarms: In sequence of occurrence.

2.2.2.5.5 Product temperatures: For example, tank, poultry.

2.2.2.5.6 Data-logger interval: One digit in half-hour intervals up to a maximum of 12 h.

2.2.2.5.7 Amperage: 0 to 63,75 A in 0,25 A intervals.

2.2.2.5.8 Destination: Three alphanumerical digits. If the destination changes, both the old and the current destination may be declared.

2.2.2.5.9 Port of discharge: Three alphanumerical digits.

2.2.2.5.10 Origin: Three alphanumerical digits.

2.2.2.5.11 Report results of self-check level 1: One digit, 0 = Fail. 1 = Pass.

2.2.2.5.12 Report results of self-check level n : In the format of up to 256 ASCII characters, where n is a single character between two and nine.

2.2.2.5.13 Vessel and voyage designation: (See ISO 9711-2.)

2.2.2.6 Commands

RCDs and refrigeration machinery if so equipped shall respond to the following commands. RCDs not so equipped shall respond “Not able” (see 2.2.2.3).

2.2.2.6.1 Change set-point temperature: Expressed in the same format as 2.2.2.4.4.

2.2.2.6.2 Initiate self-check level 1

2.2.2.6.3 Initiate self-check level n : In the same format as 2.2.2.5.12, where n is in the range two to nine.

2.2.2.6.4 Change identification number: Expressed in the same format as 2.2.2.4.1.

2.2.2.6.5 Change data-logger interval: Expressed in the same format as 2.2.2.5.6.

2.2.2.6.6 Set data-logger time and date: With the date expressed in the format year/month/day.

2.2.2.6.7 Change operating mode: Expressed in the same format as 2.2.2.5.1.

2.2.2.6.8 Download data-logger record to central monitor

2.2.2.6.9 Change porthole container number: Expressed in the same format as 2.2.2.4.3.

2.2.2.6.10 Change destination: Expressed in the same format as 2.2.2.5.8.

2.2.2.7 Indecipherable or unserviceable messages

Indecipherable or unserviceable messages shall not cause the RCD or CMCS to “crash” or “hang up”. Also, failures of an electronic device in any RCD shall not cause the system to “crash” or “hang up”.

2.2.2.8 Verification of container identification number

The CMCS, if so equipped, shall verify the container identification number, using the check digit (the seventh digit of the numerical suffix) and an algorithm selected.

2.2.3 Performance constraints

2.2.3.1 Power interference

RCDs and CMCSs shall not interfere with the proper functioning of power supply regulating or control devices, such as voltage regulators or protective relaying equipment.

2.2.3.2 Marine device interference

CMCSs and RCDs, individually or as a system, shall not interfere with standard marine navigation and communication devices.

2.2.3.3 System size

All CMCSs shall be suitable to coordinate and report on a system of 1 024 RCDs active at the same time on one CMCS.

2.2.3.4 Status update

The MMU/MDCU system shall generate RCD updated status per **2.2.2.4** at least once per hour per container for a system of up to 1 024 containers active at the same time on one CMCS.

2.2.3.5 Automatic RCD system list

The population or database of RCDs on the CMCS shall be self-generating. No input to the MMU, whether from an operator or from another computer, shall be necessary to determine the RCDs connected to that system.

2.2.3.6 Identification of new RCDs

The MMU/MDCU system shall be designed to identify an average of at least one new container every 10 s, or 6 per minute.

2.2.3.7 Voltage and frequency requirements

RCDs shall be suitable for operating on the voltage systems specified in ISO 1496-2.

2.2.3.8 Dual voltage requirements

No special handling shall be required for RCDs to operate on the three voltage types of refrigeration machinery. (See ISO 1496-2:1988, subclause **7.2.**)

2.2.3.9 RCD connection

The RCD shall be connected on the line side of the refrigeration machinery disconnect Or circuit breaker, if any, so that communication is possible when the refrigeration machinery is switched off. The RCD may have its own disconnect switch for servicing.

2.2.3.10 Error rates

2.2.3.10.1 All CMCSs and RCDs shall be designed to meet the following error rate criteria.

The RCD/MDCU communication system may have two different types of “undetected and uncorrected” communication errors. An “undetected and uncorrected” communication error is one which is not detected and corrected within 5 min after occurrence.

2.2.3.10.2 An error whereby an RCD executes a command which was not commanded by the MMU shall not occur more often than one time in 25×10^6 messages (i.e. any power line disturbance which the receiver interprets as a message), or more often than once in 10 years for each CMCS, whichever is greater.

2.2.3.10.3 An error whereby a CMCS misinterprets a message (i.e. any power line disturbance which the receiver interprets as a message) shall not occur more often than one time in 25×10^5 messages.

Section 3. System compatibility requirements

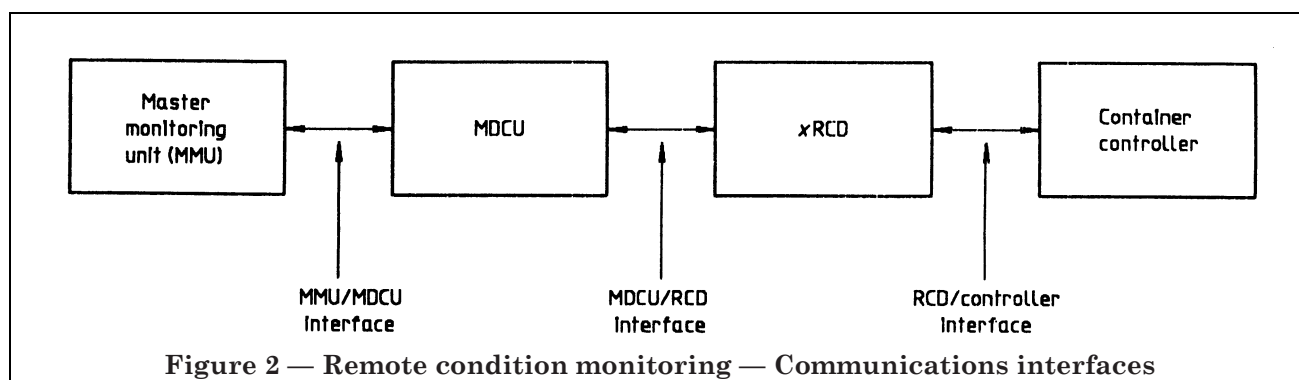
3.1 Scope

This section specifies the interface requirements for communications protocol, data-logging formats, message definitions, and physical requirements for low data rate and high data rate (CU and CD).

3.2 Communications protocol

Each remote condition monitoring system has three interface areas as follows (see Figure 2):

- MMU to MDCU interface;
- MDCU to RCD interface;
- RCD to refrigeration machinery controller interface.



3.2.1 MMU to MDCU communications

This subclause, in part, defines the communications protocol to be used when the MDCU is implemented as a discrete system component which is separate from the MMU architecture. The requirements given in this subclause do not preclude the use of bus-based open architecture MDCU applications where the EIA RS232-C is not appropriate.

The MMU communicates with the MDCU via a full duplex EIA RS232-C serial interface. The baud rate shall be at least two times the baud rate of the fastest RCD in the system. A typical communications baud rate is 4 800 baud. Each character transferred requires 1 start bit (low logic level), 8 data bits, and 1 stop bit (high logic level). The minimum time delay required between packets is one character time. This, therefore, restricts deadtime between any 2 bytes in a packet to less than one character delay.

The messages for succeeding interfaces are embedded in the formats of earlier stages (see Figure 3). These messages may be intended for action by the MDCU only. These messages are described fully in 3.2.1.1. If the message contains embedded data intended for an RCD, the format is as described in 3.2.1.3. Similarly, embedded data intended for the refrigeration machinery is described in 3.2.1.5 and 3.2.4. Note that the field length, in bytes, is also defined in Figure 3.

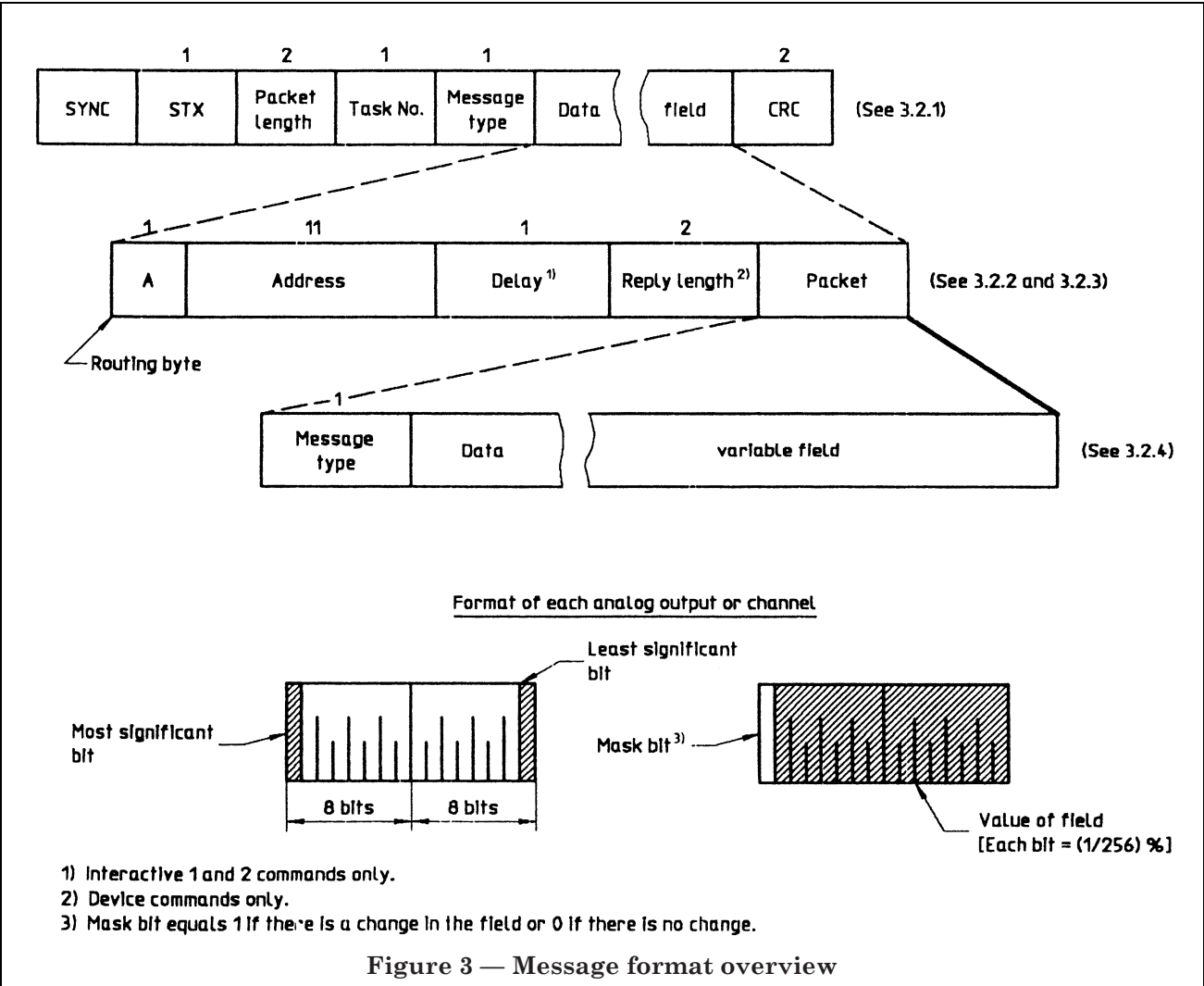


Figure 3 — Message format overview

The information transfer format from the MMU to the MDCU is as shown in Figure 4.

	SYNC (optional)	STX	Packet length	Task No.	Xmit type	Data	CRC
No. of bytes		1	2	1	1		2

Figure 4 — Information transfer format from the MMU to the MDCU

The information transfer format returned to the MMU from the MDCU is as shown in Figure 5.

	SYNC (optional)	STX	Packet length	Task No.	Reply type	Data	CRC
No. of bytes		1	2	1	1		2

Figure 5 — Information transfer format returned to the MMU from the MDCU

The fields in Figure 4 and Figure 5 are defined as follows.

SYNC	An optional synchronization field. It can be any number of bytes, the contents of which is 16H. The MDCU strips all SYNC characters from the start of a message.
STX	Delimits the start of a valid message. It indicates that the next 2 bytes are the packet length.
Packet length	A 2 byte unsigned integer which gives the packet length in bytes not including the SYNC, STX or CRC fields. It is transmitted high byte followed by low byte.
Task No.	A 1 byte number assigned to the job before transmission to the MDCU. It is not used by the MDCU and is defined by the application. The MDCU reply will contain the same value in this field. Task numbers shall be assigned sequentially by the MMU from 0 to 254 and then resume at 0. Task number 255 is reserved for the MDCU reset command.
Xmit type	A 1 byte command directing the MDCU as to the type of processing which is to be performed on the data. The assignments are discussed in 3.2.1.1.
Reply type	A 1 byte command response from the MDCU indicating the success or failure status of the corresponding job received by the MDCU. The assignments are discussed in 3.2.1.2.
Data	Variable length data field in which the contents are dependent on the type of message to be processed. The contents are defined in 3.2.1.1 and 3.2.1.2 for each appropriate command.
CRC	A 16 bit error check field generated by using the CRC-16 polynomial: $x^{16} + x^{15} + x^2 + 1$ The optional SYNC characters are not included in the CRC calculation.

3.2.1.1 MMU to MDCU transmit command types

There are two groups of commands which are sent to the MDCU: those directed to the MDCU for internal processing only and those to be transferred over the power line. The former group is given in Table 1, while the latter group is given in Table 2. An explanation of each command follows.

3.2.1.1.1 MDCU directed command types

This group of commands requires processing within the MDCU, i.e. the power line communications network is not accessed. The commands currently supported by the MDCU are given in Table 1 and explained in 3.2.1.1.1.1 to 3.2.1.1.1.3.

Table 1 — MDCU interrogation directed command types

Command	Value
MDCU reset	20H
MDCU parameters	22H
MDCU status	23H

3.2.1.1.1.1 MDCU reset command

Upon request, the MDCU will perform its internal diagnostics and initialize with default parameters. Any data associated with this command will be ignored. Since the MDCU is reset, the initial status condition with default parameters will be returned to the MMU as defined in 3.2.1.1.1.2. Task number 255 will be used in the reply.

3.2.1.1.1.2 MDCU parameters command

The MDCU parameters command loads the MDCU with information passed in the data field portion of the command. A data field containing more than 15 bytes will cause the MDCU to ignore the command. The parameters to be loaded are defined below. Note that the parameters used for the low data rate system only are prefixed with an L while the parameters used for the high data rate system only are prefixed with an H. First byte transmitted:

Parameter0 — Test status byte is a read-only location and, therefore, this parameter is ignored by the MDCU.

Parameter1-2 — MDCU software version/subversion is a read-only location and, therefore, this parameter is ignored by the MDCU.

Parameter3 — Change the MMU handshaking flag byte between the MDCU and MMU. Each bit is discussed below.

Bit7 — MDCU external UART CTS status (MMU communications). This bit will always be received by the MMU in the active, or logic one, state since the UART is in the CTS mode (i.e. communicating with the MMU). This bit is a read-only bit; the MMU cannot set/reset it.

Bit6-0 — Although accessible to the MMU, at present not used by the MDCU.

Parameter4-5 — Change available buffer size high (4) and low (5) bytes. Decreases/expands power line buffers. If the new size is below/above the MDCU's minimum/maximum value, its minimum/maximum value is used. A value of zero, or a value received which is equivalent to the previous setting, will retain the previous setting. If a change in the buffer size is performed, all jobs within queue will be lost.

LParameter6 — Change the maximum size of the packets (or blocks) to be transferred on the power line. If received larger than the available buffer size setting (described above), that value will be substituted. A value of zero will retain the previous setting (unless that setting is greater than the new available buffer size). It is recommended that the packet size not be set greater than its default value (128 or 80H) unless the power line message timeout settings within the MDCU are changed accordingly.

LParameter7 — Change the number of retransmissions to be attempted for a given job on the power line.

LParameter8 — Change the counter indicating the number of retries which have been performed on the power line.

LParameter9 — Change the counter indicating the number of successful normal polls which required one retry on the power line.

LParameter10 — Change the counter indicating the number of successful normal polls which required two retries on the power line.

Parameter11 — Change the time delay base value used within an interactive command (**3.2.1.1.2.3** and **3.2.1.1.2.8**) that is required between the MDCU transmission of data to an RCD and the interactive poll used to obtain a response from the RCD. Each count provides a 0,1 s incremental delay.

Parameter12-14 — At present undefined. The information will be copied into the status buffer as received. Since the first 3 bytes of the status buffer are read-only locations, any data field containing less than 3 bytes will not change any parameters in the MDCU status buffer. Also, Parameter4-5 is 2 bytes in length and a byte count which transfers only one of these bytes will not update this parameter. Regardless of whether any parameters were changed, the MDCU reply will return the current status value to the MMU in a general valid type reply, as defined in **3.2.1.2.1**.

3.2.1.1.1.3 MDCU status command

The MDCU status command simply requests the MDCU to reply with its present status value. Any data associated with this command will be ignored. The definition of this status buffer corresponds to the assignments discussed in **3.2.1.1.1.2** (MDCU parameters command), or as defined in **3.2.1.2.1**.

3.2.1.1.2 MDCU power line command types

This group of commands requires communication with RCDs over the power line communications network. The commands currently supported by the MDCU are given in Table 2. The format of the power line data field for the RCD directed commands to be deciphered by the RCD (RCD message) is given in **3.2.1.2**, while the format of the power line data field for the device-related commands to be deciphered by the controller device for an iRCD, or a device input/output processing for an sRCD (device message), is given in **3.2.4**.

In all power line communication commands, the first byte of the data field is the routing byte. The routing byte's identification is derived from the original RCD map response (see **3.2.1.1.2.6**) and is used by the MDCU to route the message to the proper data rate modem. Bits 7-6 define routing information for the low data rate RCDs and high data rate RCDs as described in Table 3. The remaining bits (5-0) are defined in the appropriate low or high data rate subclauses, **3.2.2** and **3.2.3** respectively.

Table 2 — MDCU power line directed command types

Command	Value
RCD poll	30H
RCD xmit1	31H
RCD xmit2	32H
RCD interactive1	33H
RCD interactive2	34H
RCD map	35H
RCD interactive poll	36H
Device poll	40H
Device xmit1	41H
Device xmit2	42H
Device interactive1	43H
Device interactive2	44H
Device map	45H
Device interactive poll	46H

Table 3 — Routing byte data rate bit assignments

Bits 7-6	Definition
0 0	Illegal
0 1	ISO LRCD
1 0	ISO HRCD
1 1	Reserved

The routing byte is always followed by an 11 byte ASCII RCD network address. Two address fields are reserved for special use in the RCDs. A universal address consisting of all ASCII 0s, or all 30H, is recognized by all RCDs (i.e. a broadcast address) while a field containing nine ASCII 0s followed by two ASCII 1s or nine 30Hs plus two 31Hs, is an address substituted in an RCD containing an invalid personal address.

The command or message type, routing byte and RCD network address are called the power line prefix and shall comprise ASCII alphanumeric characters. This provides certainty for distinguishing these prefix characters from control characters.

3.2.1.1.2.1 RCD poll command

The RCD poll command requests the status buffer from the selected RCD. The data field sent to the MDCU from the MMU contains the routing byte followed by the 11 byte ASCII RCD network address, i.e.

Data field to MDCU = Routing byte/address

A valid reply from the MDCU contains the routing byte/address and the RCD status buffer in the data field of a general valid type reply (see **3.2.1.2.1**), i.e.

Data field from MDCU valid = Routing byte/address + RCD status values

The RCD status buffer contains the information defined below. Note that the status bytes used for the low data rate system only are prefixed with an L while the status bytes used for the high data rate system only are prefixed with an H.

First byte transmitted:

Status0 — RCD status is written during initialization of the RCD.

A bit which is set indicates that an error condition occurred during the particular test, as shown in Table 4. This is a read-only location.

Table 4 — RCD status state bit definitions

Bit	RCD state
7	RCD type status: set — Integrated RCD reset — Stand-alone RCD
6	Datalog initialization incomplete
5	Reserved
4	RCD tests had an error:
3	Internal RAM error
2	External RAM error
1	Non-volatile memory error
0	Program check sum error

Status1-2 — RCD software version (1) and subversion (2) status is a read-only location.

Status3 — Handshake byte, used by the host to reset internal RCD buffers and also to prevent data tearing when multiple transfers of a particular buffer are obtained. Each bit is discussed below.

Bit 7 — RCD external UART CTS status (controller communications for the iRCD and external portable data collection computer for the sRCD). Set active to a logic one when the UART is in the CTS mode, i.e. currently capable of transfers. This bit is a read-only bit, the MMU cannot set/reset it.

Bit 6 — RCD device status buffer status. Set active indicates that the buffer contains information available to the MMU. The MMU may reset this status by transmitting logic zero in this bit position, while a logic one will not change the buffer's state.

Bit 5 — RCD device response buffer status. Set active indicates that the buffer contains information available to the MMU. The MMU may reset this status by transmitting logic zero in this bit position, while a logic one will not change the buffer's state. Note that a buffer in the process of being loaded by the controller (iRCD) will not abort the transmission.

Bit 4 — RCD device datalog buffer status. Set active indicates that the buffer contains information available to the MMU. The MMU may reset this status by transmitting logic zero in this bit position, while a logic one will not change the buffer's state. Note that a buffer in the process of being loaded by the controller (iRCD) will not abort the transmission.

Bit 3 — RCD response buffer status. Set active indicates that the buffer contains information available to the MMU.

Bit 2 — For an sRCD only, RCD external portable data collection computer buffer status, if applicable. Set active to a logic one when either the external serial channel UART receive or the transmit buffer contains valid information. This information is not exchanged on the power line communications network. This bit can be reset by the MMU, but the serial channel communications are not affected.

Bit 1 — At present undefined.

Bit 0 — RCD data-logger buffer update handshake bit. The RCD sets this bit whenever the RCD has updated its internal data-logger buffer. The MMU resets the bit prior to transfer and checks it after the transfer is complete, thus providing a means to check possible data tearing of the buffer.

Status4-5 — Available power line buffer size high (4) and low (5) bytes.

LStatus6 — Indicates the maximum size for any data packet (or block) transmitted by the RCD on the power line network. This number will never exceed the available buffer size. Its default is set at 128 or 80H.

LStatus7 — Indicates the number of attempts the RCD will perform for a particular job in which it received an invalid response on the power line from the MDCU. The default is set at 02.

LStatus8 — Indicates the number of retries required on the power line. Rollover 0FFH to 00H will not occur. It can be reset using the RCD parameters command, as described in 3.2.1.3.2.

Status9 — Indicates the current baud rate the iRCD is communicating with the controller. For the sRCD, this indicates the current baud rate the device is communicating with an external device, if applicable. The corresponding baud rates are given in Table 29.

Status10 — Indicates the maximum baud rate the iRCD can communicate with the controller. For the sRCD, this indicates the maximum baud rate the device can communicate with an external device, if applicable. The corresponding baud rates are given in Table 29.

Status11-14 — At present undefined.

3.2.1.1.2.2 RCD xmit1 and xmit2 commands

The RCD xmit1 and xmit2 command transfers text data to a selected RCD slave. The xmit1 versus xmit2 selection defines the method of transfer on the power line and is further described for the appropriate low data rate or high data rate modem (3.2.2 and 3.2.3 respectively). From the point of view of the MMU, these process differences are invisible, except for the time of response. The data field sent to the MDCU from the MMU contains the routing byte followed by the 11 byte ASCII RCD network address. The information following this address is the text data, or message, to be given to the selected RCD, i.e.

Data field to MDCU = Routing byte/address + RCD message

The RCD message field is discussed in 3.2.1.3. Inscribed in it is a subcommand which the RCD evaluates and processes accordingly. If this subcommand requires a response from the RCD, the RCD buffers it in an RCD response buffer. An RCD interactive poll command (as described in 3.2.1.1.2.5) is then used to obtain this buffered response. These xmit commands, however, only return a general valid type reply (see 3.2.1.2.1) from the MDCU to the MMU after the MDCU receives the proper acknowledgement of transfer from the slave RCD. The routing byte/address are returned in the data field of this reply, i.e.

Data field from MDCU valid = Routing byte/address

3.2.1.1.2.3 RCD interactive1 and interactive2 commands

The RCD interactive1 and interactive2 commands transfer text data to a selected RCD slave in exactly the same way as the RCD xmit commands described in 3.2.1.1.2.2. As with the xmit commands, the interactive1 versus interactive2 selection defines the method of transfer on the power line and is further described for the appropriate low data rate or high data rate modem (3.2.2 and 3.2.3 respectively). From the point of view of the MMU, these process differences are invisible, except for the time of response. The interactive commands provide the RCD with a subcommand inscribed in the text data, which the RCD evaluates and processes accordingly. (The RCD processes the interactive commands in the same way as the xmit commands.) If this subcommand requires a response from the RCD, the RCD buffers it in an RCD response buffer. The difference between the RCD xmit and interactive commands occurs here in the MDCU processing. Instead of returning a valid reply to the MMU when the MDCU receives a proper acknowledgement from the slave RCD for the transfer of text data, the MDCU queues an RCD interactive poll command, as described in 3.2.1.1.2.5, without intervention from the MMU. A time delay later, as specified by a byte in the interactive command's data field, this interactive poll is sent to the RCD. Thus the data field sent to the MDCU is

Data field to MDCU = Routing byte/address + Response delay + RCD message

where the response delay is a 1 byte value. Each count of this value causes an approximate incremental delay, calculated as this number times the base value established in the MDCU Parameter11 byte described in 3.2.1.1.1.2, between the completion of the interactive command and the transmission of the interactive poll. The response of the RCD to this poll is then returned to the MMU as the reply for the interactive command and, therefore, its format is as given in 3.2.1.1.2.5. This reply will be dependent on the amount of RCD processing required for the RCD message subcommand. Note that subcommands not requiring RCD responses should not use an interactive command since the interactive poll would receive a negative acknowledgement from the RCD (i.e. no response buffer will be available).

3.2.1.1.2.4 RCD map command

The RCD map command broadcasts a polling message from the MDCU requesting, with varying probabilities, an RCD device to return its network address. The purpose of this command is to log in, or map, any newly acquired or unpolled RCDs on the power line network. Since the mapping technique uses a polling-type process on the network, the format of the data field from the MMU to the MDCU is similar to the poll commands described in 3.2.1.1.2.1 and 3.2.1.1.2.6, i.e. the routing byte followed by the 11 byte ASCII RCD network address. However, being a broadcast type of message transfer, it cannot simply select a particular RCD by using a unique individual network address. Instead, the 11 byte ASCII field is broken into ASCII subfields supplying all RCDs on the power line network with a mapping strategy. The format of this field is shown in Figure 6.

	Alpha network address	Type repeat	Map subcommand	Unused HRCD destination address	Shift variable	Destination address individual
No. of bytes	4	3	1	2	1	2

Figure 6 — Format of RCD map command

The fields in Figure 6 are defined as follows.

Alpha network address — 4 byte ASCII characters defining an RCD alpha portion (usually denoting the container operator) of the RCD network address on the power line network. All RCDs recognize the map command by using the universal address, or four ASCII 0s, or 30H 30H 30H 30H.

Type repeat — A 3 byte check field filled with a map command (045H). The device map command value is used instead of the RCD map command to allow an alphanumeric value to reside in the normally numeric polling field.

Map subcommand — A 1 byte ASCII character which provides an RCD with a processing procedure in the mapping strategy. These procedures, and their associated values, are given below.

Unused HRCD destination address — A 2 byte hexadecimal field which provides an HRCD with a network address. Only the HRCD(s) responding to the map command use this address. This field is used only when the alpha network address specifies the universal address, i.e. four ASCII 0s, or 30H 30H 30H 30H.

Destination address individual — A 2 byte hexadecimal field which provides an HRCD with a network address. Only the HRCD responding to the map command uses this address. This field is present only when the map address specifies an individual 11 byte address.

Shift variable — The upper nibble of the byte and shall be 1111 B. The lower nibble contains a number from 0 to 15 which indicates the number of bits to mask from an RCD's 16 bit random number (0 will provide a 1111111111111111 B bit mask, while 15 will provide 0000000000000001 B). An RCD which obtains a value of zero by performing the logical AND of this mask with the 16 bit random number will attempt to return its network address provided that its internal respond to map flag is set (see below). Thus, this shift variable provides the mapping strategy with the capabilities of adjusting the range, or size, of the expected number of RCD responding.

Any field described above which does not contain a value as required will ignore the command. Thus, the data field sent to the MDCU from the MMU is

Data field to MDCU = Network address tag/map information

An RCD will not accept a map command unless the alpha network address characters match those of the particular RCD or the universal address. This provides for selection of different classes of RCDs using the same data rate and thus the same routing byte. When accepted, the RCD decodes the map subcommand to perform different strategies of the mapping sequence. These subcommands are listed in Table 5. Any other value received by the RCD in the subcommand will force the RCD to ignore the command. Each RCD addressed will perform the map function received regardless of the respond to map state. Only after execution of the map function does the RCD check the respond to map state. If a map response is indicated, the RCD performs the logical AND of the shift variable mask with its 16 bit random number. A value of zero will allow the RCD to return its network address.

Table 5 — Map subcommand definitions

Subcommand	ASCII value	HEX value	RCD process
Initiate new map	I	49	Sets its respond to map flag, generate a new 16 bit random, and set up a new shift variable.
General new map	G	47	Generate a new 16 bit random and set up a new shift variable.
Change shift map	S	53	Set up a new shift variable.
Decrement map	D	44	Decrement its random (variable rollover will occur) and set up a new shift.

An RCD will set its internal respond to map flag in one of three ways:

- a) the RCD starts a power-up sequence, either by initially being turned on, or by an internal deadman reset occurrence by an RCD directed reset command from the MMU device (see 3.2.1.3);
- b) the RCD has not received an individually addressed command within 1 h;
- c) the RCD receives an initialize map subcommand (defined above).

The flag is reset each time the RCD receives an individually addressed command. At this time the RCD initiates the 1 h timer discussed in b) above.

When the MDCU receives a network address from the RCD it returns a general valid type reply (see 3.2.1.2.1) to the MMU with the data field containing this RCD network address following the map command message it used, i.e.

Data field from MDCU value = Routing byte/map information + RCD network address

3.2.1.1.2.5 RCD interactive poll command

The RCD interactive poll command is used to retrieve information from an RCD after the RCD has been sent an RCD xmit1 or xmit2 command (see 3.2.1.1.2.2) in which its subcommand in the RCD message field caused the RCD to buffer a response (see 3.2.2.1). The format of the data field from the MMU to the MDCU for the interactive poll is exactly the same as that of a normal poll command, i.e.

Data field to MDCU = Routing byte/address

The time between the xmit and interactive poll is the responsibility of the MMU, and shall be enough to allow the RCD to process and buffer its response. If an RCD response buffer does not contain information, a negative acknowledge (see 3.2.1.2.4) is returned from the RCD to the MDCU, and consequently to the MMU. If, however, the RCD has buffered information, this information is returned to the MMU via a general valid type reply (see 3.2.1.2.1) with this information preceded in the data field by the network tag and the RCD's address, i.e.

Data field from MDCU valid = Routing byte/address + RCD message

Once the RCD transfers this buffered data, it resets (and thus empties) its RCD response buffer, therefore allowing another RCD xmit or interactive command to be accepted by the RCD.

3.2.1.1.2.6 Device poll command

The device poll command requests the quick status buffer of the controller which is attached to the selected iRCD or the internal quick status of the sRCD. The data field sent to the MDCU from the MMU contains the routing byte followed by the 11 byte ASCII RCD network address and the expected reply length, i.e.

Data field to MDCU = Routing byte/address

A valid reply from the MDCU contains the routing byte/address and the controller (iRCD) or internal (sRCD) quick status buffer in the data field of a general valid type reply (see 3.2.1.2.1), i.e.

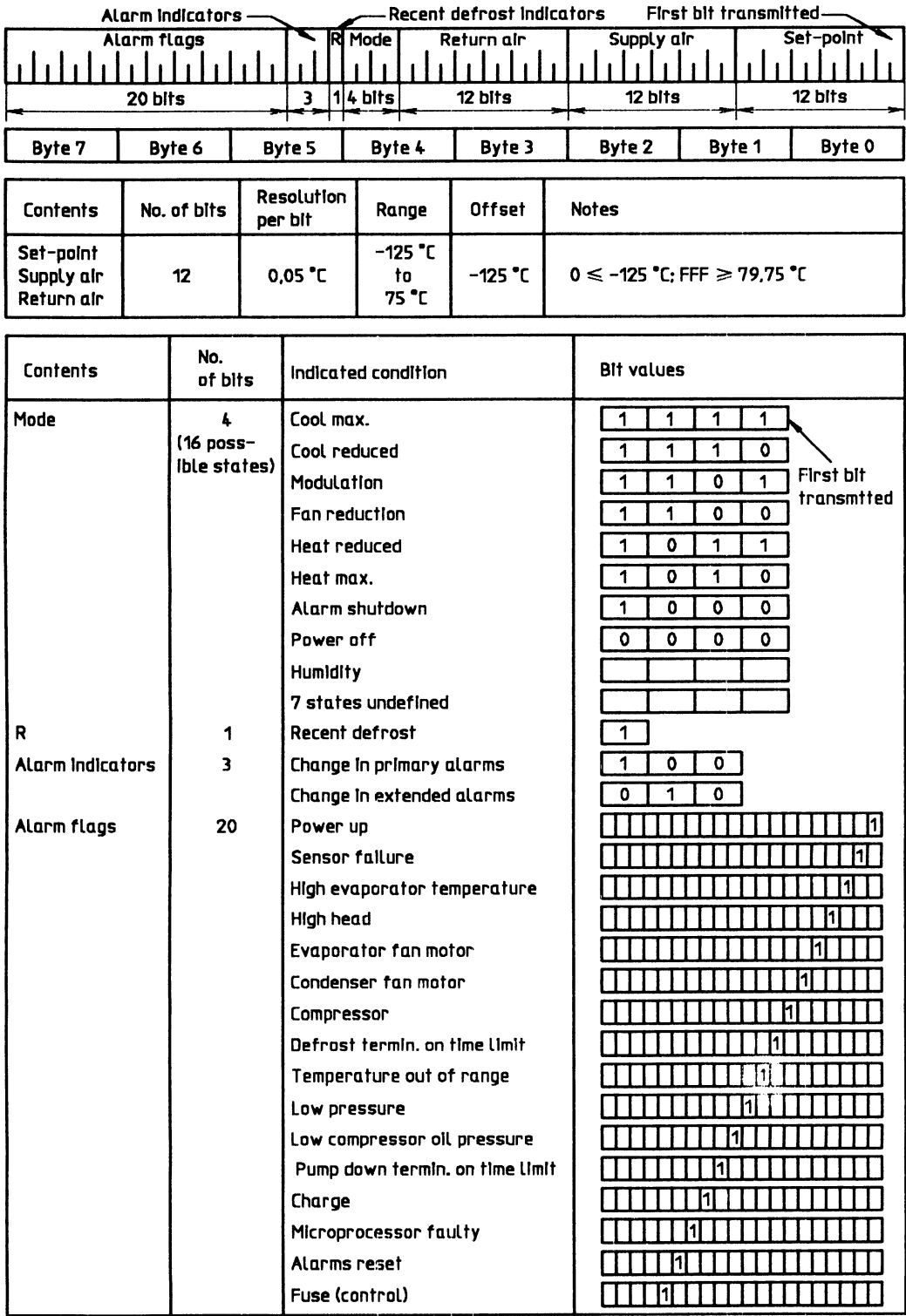
Data field to MDCU valid = Routing byte/address + Quick status

For an iRCD, an image of the controller quick status buffer is contained within the iRCD and updated approximately every 30 s. For both the iRCD and the sRCD, the reply format of the data field of this message is presented in Figure 7.

3.2.1.1.2.7 Device xmit1 and xmit2 commands

The device xmit1 and xmit2 commands transfer text data to a selected RCD slave. The xmit1 versus xmit2 selection defines the method of transfer on the power line and is further described for the appropriate low data rate or high data rate modem (3.2.2 and 3.2.3 respectively). From the point of view of the MMU, these process differences are invisible, except for the time of response. The data field sent to the MDCU from the MMU contains the routing byte followed by the 11 byte ASCII RCD network address. The information following this address is the text data, or message, to be given to the selected iRCD for transfer to the controller, or the selected sRCD for device input/output processing, i.e.

Data field to MDCU = Routing byte/address + Device message



NOTE All multiple byte fields shall be transmitted low-order byte first.

Figure 7 — Device poll (total 8 bytes or 64 bits)

This device message field is discussed in **3.2.1.5**. Inscribed in it is a subcommand which the RCD evaluates and processes accordingly. If this subcommand requires a response from the controller (iRCD) or the input/output function (sRCD), the RCD buffers it in a device response buffer. A device interactive poll command, as described in **3.2.1.1.2.10**, is then used to obtain this buffered response. These xmit commands, however, only return a general valid type reply (see **3.2.1.2.1**) from the MDCU to the MMU after the MDCU receives the proper acknowledgement of transfer from the RCD (this acknowledgement is an indication of the MDCU/RCD transfer and has no bearing on, for instance, the iRCD/controller transfer status). The routing byte/address are returned in the data field of this reply, i.e.

Data field to MDCU = Routing byte/address

3.2.1.1.2.8 *Device interactive1 and interactive2 commands*

The device interactive1 and interactive2 commands transfer text data to a selected RCD slave in exactly the same way as the device xmit commands described in **3.2.1.1.2.7**. As with the xmit commands, the interactive1 versus interactive2 selection defines the method of transfer on the power line and is further described for the appropriate low data rate or high data rate modem (**3.2.2** and **3.2.3** respectively). From the point of view of the MMU, these process differences are invisible, except for the time of response. The interactive commands provide the RCD with a subcommand inscribed in the text data, which the RCD evaluates and processes accordingly. (The RCD processes the interactive commands in the same way as the xmit commands.) If this subcommand requires a response from the controller (iRCD) or the input/output function (sRCD), the RCD buffers it in an RCD response buffer. The difference between the device xmit and interactive commands, like the difference between the RCD commands described in **3.2.1.1.2.2** and **3.2.1.1.2.3**, occurs here in the MDCU processing. Instead of returning a valid reply to the MMU when the MDCU receives a proper acknowledgement from the RCD for the transfer of text data, the MDCU queues a device interactive poll command, as described in **3.2.1.1.2.10**, without intervention from the MMU. A time delay later, as specified by a byte in the interactive command's data field, this interactive poll is sent to the RCD. Thus the data field sent to the MDCU is

Data field from MDCU valid = Routing byte/address + Response delay + Device message

where the response delay is a 1 byte value. Each count of this value causes an approximate incremental delay, calculated as this number times the base value established in the MDCU Parameter11 byte described in **3.2.1.1.1.2**, between the completion of the interactive command and the transmission of the interactive poll. The response of the RCD to this poll is then returned to the MMU as the reply for the interactive command and, therefore, its format is as given in **3.2.1.1.2.5**. This reply will be dependent on the RCD processing required for the device message subcommand. Note that subcommands not requiring device responses should not use an interactive command since the interactive poll would receive a negative acknowledgement from the RCD (i.e. no response buffer will be available).

3.2.1.1.2.9 *Device map command*

The device map command is indistinguishable from the RCD map command (see **3.2.1.1.2.4**).

3.2.1.1.2.10 *Device interactive poll command*

The device interactive poll command is used to retrieve information from an RCD after the RCD has been sent a device xmit1 or xmit2 command (see **3.2.1.1.2.7**) in which its subcommand in the data field required the iRCD to buffer a controller response or the sRCD to buffer a device input/output process (see **3.2.1.5**). The format of the data field from the MMU to the MDCU for the interactive poll is exactly the same as that of a normal poll command, i.e.

Data field to MDCU = Routing byte/address

The time between the xmit and interactive poll is the responsibility of the MMU, and shall be enough to allow the RCD to process and buffer its response. If the RCD's device response buffer does not contain information, a negative acknowledge (see **3.2.1.2.4**) is returned from the RCD to the MDCU and consequently to the MMU. If, however, the RCD has buffered information, this information is returned to the MMU via a general valid type reply (see **3.2.1.2.1**) with this information preceded in the data field by the routing byte/address, i.e.

Data field to MDCU valid = Routing byte/address + Message

Once the RCD transfers this buffered data, it resets (and thus empties) its device response buffer, therefore allowing another device xmit or interactive to be accepted by the slave RCD. (Note that the RCD's device response buffer is independent of the RCD's RCD response buffer discussed in **3.2.1.1.2.5**.)