C-Bus Application Messages & Behaviour

Chapter 6 – Metering

Document Number: CBUS-APP/06
Issue: 1.3
Date: 13 February 2007

Comments on this document should be addressed to:

Engineering Manager
Clipsal Integrated Systems
PO Box 103 Hindmarsh
South Australia 5007

TABLE OF CONTENTS

6 METERING APPLICATION

6.1 Application ID

$D1

6.2 Description

The Metering Application is used to obtain information about consumption from utilities such as electricity, water, gas, oil, etc.

Meters are normally proprietary, and use some kind of interface device to convert data from the meter into a form where it can be transported using C-Bus.

6.3 Document Convention

Numbers are shown in decimal (base ten) with no other special prefixes or indications.

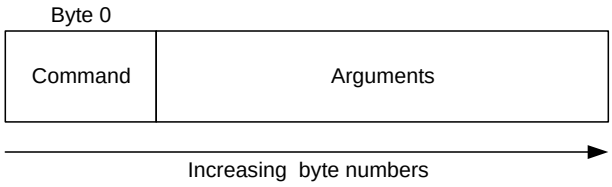
Binary numbers (base 2) are shown with the prefix %.

Hexadecimal numbers (base 16) are shown with the prefix $.

Example: 157 = %10011101 = $9D

6.4 Message Structure

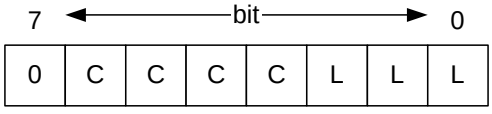
C-Bus messages can be up to 64 bytes long[1], though in practice Metering Application are no more than 6 bytes long. Metering Application messages have the form:



The number of arguments is variable, and is dependent on the command.

The command byte is broken into bit-fields to support encoding of a command and the number of bytes following as parameters.

The command byte form is:



[1] Due to a limitation in the C-Bus PC interface, a single message cannot be longer than 14 bytes.

6.4.1 Commands

The following commands are supported:

%0001 = EVENT[2]

All others reserved.

The length field reflects the number of arguments.

6.4.2 Event Codes

The byte immediately following the command byte is an event type code. It is used to indicate the type of information in the message.

The following convention is used:

Event Codes:

Size:	8-bit byte
Range:	$00 .. $FF
Special Cases:	$00 and $FF are reserved for future expansion
Usage:	$01 .. $05 used for meter measurement requests
	$06 .. $80 reserved for future expansion
	$81 .. $85 used for meter measurement responses
	$86 .. $FE reserved for future expansion

6.5 Defined Messages

All messages provide a degree of compatibility with C-Bus lighting application commands[3]. The command byte is followed by an event code, and any additional arguments. The length field encodes the number of argument bytes which follow and apply to that command.

All messages listed are mandatory for C-Bus metering systems, unless explicitly stated otherwise. C-Bus meter interface systems which receive a meter measurement request for a meter type not supported must always respond with a measurement of zero.

6.5.1 Meter System Measurement Request Messages

Meter System Measurement Requests are sent to a meter measurement device to request a measurement be performed.

The meter measurement device shall respond with the corresponding Meter System Measurement Response.

[2] This message form is compatible with a C-Bus Lighting Application TERMINATE RAMP command.

[3] The event code can be thought of as similar to a lighting application group address.

6.5.1.1 Measure Electricity

Command: $09

Arguments: $01

Meaning: Request a measurement of electricity consumption be returned

Originator: Anywhere

Notes:

6.5.1.2 Measure Gas

Command: $09

Arguments: $02

Meaning: Request a measurement of electricity consumption be returned

Originator: Anywhere

Notes:

6.5.1.3 Measure Drinking Water

Command: $09

Arguments: $03

Meaning: Request a measurement of drinking water consumption be returned

Originator: Anywhere

Notes:

6.5.1.4 Measure Other Water

Command: $09

Arguments: $04

Meaning: Request a measurement of water consumption be returned

Originator: Anywhere

Notes:

6.5.1.5 Measure Oil

Command: $09

Arguments: $05

Meaning: Request a measurement of oil consumption be returned

Originator: Anywhere

Notes:

6.5.2 Meter System Measurement Response Messages

Meter System Measurement Responses are sent by a meter measurement device to indicate the consumption. These messages are sent only in response to a previous request message.

6.5.2.1 Electricity Consumption

Command: $0D

Arguments: $81, <4 bytes>

Meaning: Electricity consumption is returned

Originator: Meter measurement interface device

Notes: The 4 byte argument has the following format:

. byte order: Most .. Least significant bytes

. scaling: 1 least significant bit = 1000 Whr (Watt hour)

. min value: 0

. max value: $2^{32} - 1$

6.5.2.2 Gas Consumption

Command: $0D

Arguments: $82, <4 bytes>

Meaning: Gas consumption is returned

Originator: Meter measurement interface device

Notes: The 4 byte argument has the following format:

. byte order: Most .. Least significant bytes

. scaling: 1 least significant bit = 1,000,000 Joules

. min value: 0

. max value: $2^{32} - 1$

6.5.2.3 Drinking Water Consumption

Command: $0D

Arguments: $83, <4 bytes>

Meaning: Drinking water consumption is returned

Originator: Meter measurement interface device

Notes: The 4 byte argument has the following format:

. byte order: Most .. Least significant bytes

. scaling: 1 least significant bit = 1000 litres

. min value: 0

. max value: $2^{32} - 1$

6.5.2.4 Other Water Consumption

Command: $0D

Arguments: $84, <4 bytes>

Meaning: Other water consumption is returned

Originator: Meter measurement interface device

Notes: The 4 byte argument has the following format:

. byte order: Most .. Least significant bytes

. scaling: 1 least significant bit = 1000 litres

. min value: 0

. max value: $2^{32} - 1$

6.5.2.5 Oil Consumption

Command: $0D

Arguments: $85, <4 bytes>

Meaning: Oil consumption is returned

Originator: Meter measurement interface device

Notes: The 4 byte argument has the following format:

. byte order: Most .. Least significant bytes

. scaling: 1 least significant bit = 1 litre

. min value: 0

. max value: $2^{32} - 1$

6.6 Message Priority

C-Bus metering application messages shall always be transmitted at Class 4 (lowest) priority. The header byte of a local point-multipoint message will therefore be $05.

6.7 Internetwork Routing

C-Bus meter measurement devices may receive messages that have been routed via one or more C-Bus bridges or gateway devices. Such messages will be received with a message type indicating point-multipoint, but will have a non-zero Network routing.

To ensure the response is directed back to the correct network, via the same bridges and message path, any request messages received with internetwork routing information must always have their response generated as a point-point-multipoint message with a routing stack derived from the request message.

6.8 Application Behaviour

6.8.1 Concatenated Commands

A meter measurement device may receive and transmit messages containing more bytes than a single command. This permits a single C-Bus transmission to contain multiple commands for a single application.

Devices using C-Bus Metering messages must process all received bytes. This is achieved by placing the received bytes in a buffer, and using the following simple algorithm:

```
WHILE the buffer contains bytes LOOP
    The first byte defines the command type and argument
    count (refer section 6.4).
    Process the first (command) byte and its arguments
    Once processed, remove the command and argument bytes
    from the buffer
END LOOP
```

6.8.2 State

C-Bus meter measurement devices are expected to maintain consumption records through failures of power to the meter measurement device.

6.9 Status Reporting

C-Bus metering applications shall not respond to C-Bus status requests (MMI) issued against the Metering Application Address.

6.10 Limitations

A single C-Bus network should not contain more than 1 meter measurement device.

6.11 Examples

These examples assume a metering measurement device and an interrogating device both interface to C-Bus using the C-Bus Serial Interface, which is described in more detail in CBUS-SIUG.

The examples assume the Serial Interface SRCHK option is set, so data transfer both to and from the Serial Interface uses a checksum.

6.11.1 Device Requests Metering Application to Measure Electricity

Refer to section 6.5.1.1 (Page 5). To request a meter measurement device on the local network make and forward a measurement of electricity use, a device could issue:

To PCI: \05D100090120

To perform the same operation on a remote network (through a single bridge with unit address $56 on the side of the sending device, and unit address $37 on the side of the meter measurement device), a device could issue:

To PCI: \03**5609**D10901C3

The internetwork routing bytes ($5609) would be modified by the bridge as the message passed through. to construct the reverse route.

6.11.2 Meter Measurement Device sends Electricity Use

Refer to section 6.5.2.1 (Page 6). If a request was received from a local network for a measurement of electricity consumption, and the consumption is 56312 kWhr ($0000DBF8) the meter measurement device would issue:

To PCI: \05D1000D810000DBF8C9

To continue the above example for a measurement requested from a remote network, the meter measurement device extracts the received reverse routing information, and can issue:

To PCI: \03**3709**D10D810000DBF88B

This would ensure that the reply is routed back to the same network that sent the original request.