VLCB Versatile Layout Control Busm

Minimum Node Service Specification

Service# 1 MNS

Version 1.3, March 2025, for Service version 2

Compatible with CBUS ® 4.0 Rev 8j

0.1 Table of Contents

0.2 Document History

Date	Changed by	Summary of changes	Service version
22nd December 2022	Ian Hogg M.5144	Initial document	1
12 April 2023	Ian Hogg M.5144	Changed name to VLCB	1
25 April 2023	Ian Hogg M.5144	Updated version number to use Patch number and added more clarity.	1
15 May 2023	Ian Hogg M.5144	Allocated bit 6 of module parameter flags to support for service discovery.	1
01 Sept 2023	Ian Hogg M.5144	Changed Parameter 0 to return the number of supported parameters instead of fixed value of 24.	1
26 Nov 2023 Released as v1.1	Martin Da Costa M6223	Updated Section 3 and elsewhere for Mode Commands	1
7 April 2024	Ian Hogg M.5144	Added GRSP(ok) response when MODE to change from Uninitialised to Setup.	1
29 April 2024	Ian Hogg M.5144	EVLRNI no longer deprecated. Heartbeat messages are OFF by default. MODE with an NN=0 can be used to control heartbeat globally	1
15 July 2024	Ian Hogg M.5144	Added missing HEARTB to list of opcodes used. Added further explanation of Service Type and Service Index.	1

9th August 2004	Ian Hogg M.5144	Clarified use of button to transition from Unititialised to Setup mode.	1
15 November 2024	Martin Da Costa	Update button state transitions to more closely follow CBUS	?
10 March 2025	Ian Hogg M.5144	Addition of FCU-compatibility mode	2

1 Introduction / Description / Explanation

This document describes the VLCB Minimum Node Service (MNS). This service is mandatory for all VLCB modules.

All VLCB modules shall conform to this specification.

Introduction describing the goals and solutions developed for this service.

1.1 Dependencies on other services

The Minimum Node Service does not depend on any other services.

2 Module Modes

2.1 MNS Modes

MNS supports the following operation *modes*:

- Uninitialised
- Setup
- Normal

MNS permits valid transitions between these modes. In addition MNS permits additional sub-modes to be specified by other services.

MNS must store the major mode value for Uninitialised, Setup Normal. It also needs to persist data so that it can tell that it has been assigned a node number and is in Normal mode.

MNS is also responsible for storing a flag indicating whether the module should be producing Heartbeat messages.

2.1.1 Uninitialised Mode

After preparing the module's hardware for operation a newly installed module will start in the Uninitialised mode.

The module must indicate that it is in the Uninitialised mode by showing a green LED or other means. Please see the VLCB Technical Introduction document for possible user interface options.

Note Uninitialised mode is conceptually the same as the CBUS SLiM mode. Operation in this mode is not currently supported.

Any previously held node number should be released with a NNREL message as the module enters Uninitialised mode from Normal or Setup modes.

Messages acted upon: QNN.
Messages possibly sent: PNN, NNREL.

2.1.2 Setup Mode

The module is in the process of being assigned a node number. Only a single module on the network can be in Setup mode at a time. Therefore, should a module in Setup mode detect a Mode-message instructing another module to enter Setup-Mode, it must immediately exit Setup mode and revert to its previous mode. The module shall reclaim its previous node number and send NNACK with this node number to signify its new status.

The module will respond to Setup opcodes, which do not require a node number.

The module must indicate that it is in Setup mode by flashing the yellow LED at 1Hz or other means. Please see the VLCB Introduction document for possible user interface options.

The module does not have a node number whilst in Setup mode. The module should internally assign a value of zero for its node number.

Mode change messages acted upon: SNN.
Mode change messages possibly sent: RQNN and NNACK.

2.1.3 Normal Mode

As the name suggests this is the normal operating mode of the module. The module has a node number and will respond to configuration commands containing a NN matching its own node number.

The module will also consume and produce events, if these are supported.

The module must indicate that it is in the Normal Mode by showing a yellow LED or other means. Please see the VLCB Introduction document for possible user interface options.

Note: Normal Mode is conceptually the same as the CBUS FLiM Mode.

Messages received: Most.
Messages possibly sent: Most.

2.2 MODE Opcode

The MODE opcode 0x76 is used to request that a module changes its mode. The MODE opcode takes a node number and a mode change request number as parameters.

The MODE opcode should be processed by MNS if the nn matches the node number of the module or if nn is equal to 0 then all modules should process the mode change request according to the table below.

The first 8 mode request values are reserved for transition between the MNS modes, the values 0x0C and 0x0D are used to turn Heartbeat messages on and off, see section 10 Service Specific Modes. The remainder of the values are available for other services to use.

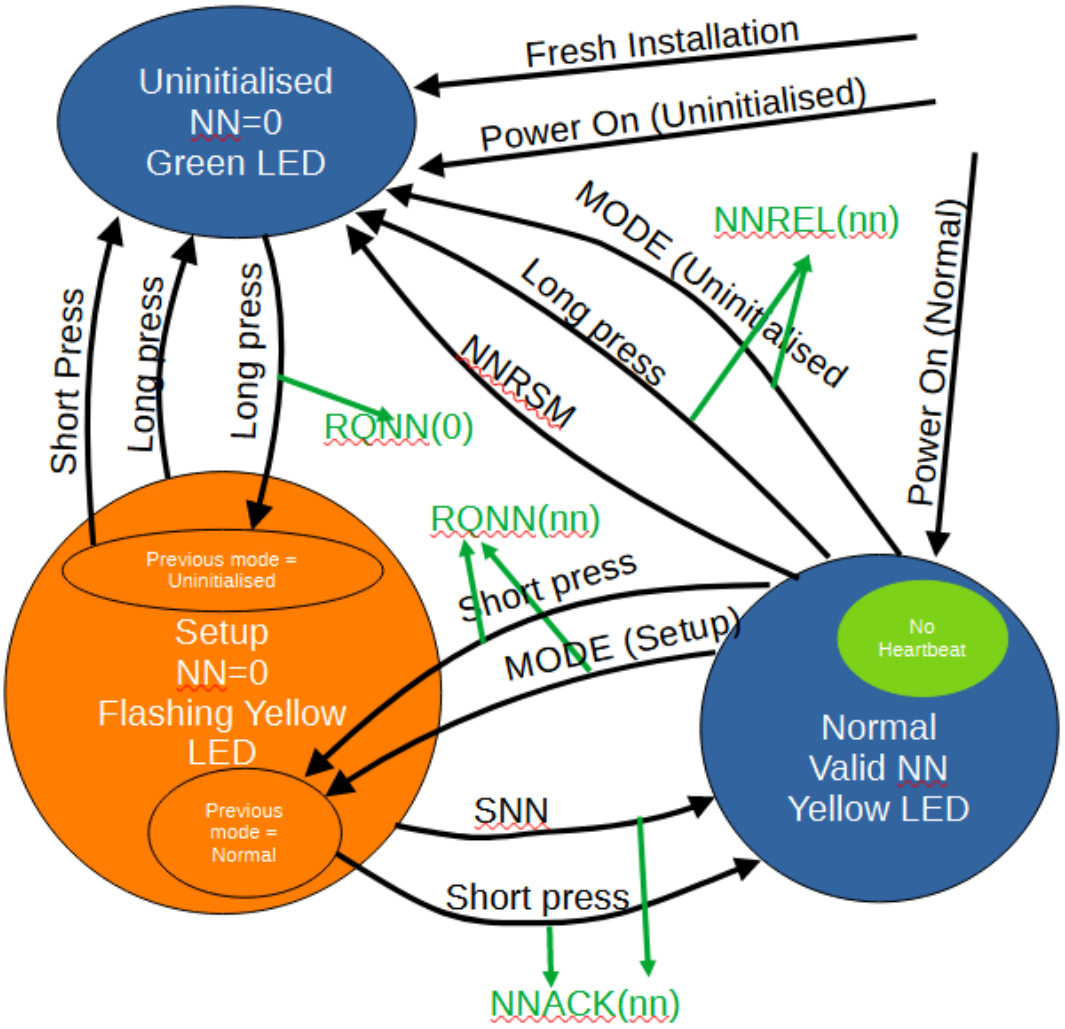
Mode Request Command	Value	Service	Support nn=0 when module node number != 0
change to setup request	0x00	MNS	No
change to normal requested	0x01	MNS	No
Reserved	0x02		
Reserved	0x03		

Reserved	0x04		
Reserved	0x05		
Reserved	0x06		
Reserved	0x07		
Available for other service usage	0x08 … 0x0B	Other	
Turn on Heartbeat	0x0C	MNS	Yes
Turn off Heartbeat	0x0D	MNS	Yes
Available for other service usage	0x0E … 0x0F	Other	
Turn on FCU compatibility	0x10	MNS	Yes
Turn off FCU compatibility	0x11	MNS	Yes
Available for other service usage	0x12 … 0xFE	Other	

2.3 Mode transitions

MNS should store a mode variable in RAM which is loaded from non volatile storage upon power up. This variable shall store the possibilities of Uninitialised, Setup, Normal. When the MNS service requires a mode change then this variable is to be updated. If the new mode is either Uninitialised or Normal then the value should also be persisted in the non volatile storage.

The diagram below shows the most commonly used mode transitions.



2.3.1 Factory initial mode

Upon first start of the module the MNS mode should be set to Uninitialised and the NN set to 0. The module should show a green LED or equivalent indication of Uninitialised mode.

Each of the Mode requests and push button operations are detailed below

2.3.2 Current mode is Uninitialised

The MODE command cannot be used to request a transition from Uninitialised to other modes.

2.3.2.1 Uninitialised ⇒ Setup

This mode change is initiated by pressing and holding the button for 4 seconds rather than the MODE command. Note that it is NOT necessary to release the button but instead the transition to Setup mode starts whilst the button is down.

- Start flashing Yellow LED or equivalent Setup indication.

When the button is released:

- The MNS mode variable should be set to indicate Setup mode.
- Send RQNN with 0 as the existing NN.

2.3.3 Current mode is Setup

The MODE command cannot be used to request a transition from Setup to other modes.

A Long button press of at least 4 seconds will transition to Uninitialised mode

- Set LEDs to steady Green or equivalent Uninitialised mode indication
- Set mode to Uninitialised.

2.3.3.1 Setup mode and previous mode was Uninitialised mode ⇒ Uninitialised

If the module was previously in Uninitialised mode then a short button press of between 0.1 to 2 seconds shall move the module back to Uninitialised mode.

- Set LEDs to steady Green or equivalent Uninitialised mode indication
- Set mode to Uninitialised.

No other configurations are changed.

2.3.3.2 Setup and previous mode was Normal mode ⇒ Normal

If the module was previously in Normal mode then a short button press of between 0.1 to 2 seconds on the module may be used to move the module back to Normal mode.

- Restore the previous node number
- Send NNACK using the previous node number
- Set LEDs to steady Yellow or equivalent Normal mode indication

This action cannot be initiated by the MODE command. SNN must be used instead. The module shall respond to a MODE command with GRSP(NN, MNS, MODE, Invalid Command) and continue to show a flashing Yellow LED or equivalent Setup indication.

For completeness the SNN operation should

- SNN should update the mode variable to Normal and persist in non volatile storage.
- The module must then respond by transmitting a NNACK containing its new node number.

2.3.3.3 Setup and previous mode was Uninitialised mode ⇒ Normal

This can only be achieved by sending a SNN to the module. The module shall then:

- Set its node number to that in the SNN message,
- Set its mode to Normal,
- Show a continuous Yellow LED or equivalent indication.
- Send NNACK with the new node number.

2.3.4 Current mode is Normal

2.3.4.1 Normal mode ⇒ Uninitialised

A long button press of at least 4 seconds shall cause the Yellow LED to extinguish and the Green LED to light, or equivalent indication. When the button is released, the module shall:

- Send NNREL,
- Set its mode to Uninitialised,
- Set its node number to 0.

No other configurations are changed.

A MODE command to Uninitialised shall initiate the same actions.

Note a NNRSM command also migrates to Uninitialised mode and sets node number to 0. It does not send a NNREL.

2.3.4.2 Normal ⇒ Setup

A short button press of between 0.1 to 2 seconds shall cause the module to re-enter Setup mode so as to allow the node number to be checked and changed if required.

- The MNS mode variable should be set to indicate Setup mode.
- Start flashing Yellow LED or equivalent Setup indication.
- Send RQNN specifying the current NN.

A MODE command to Setup shall initiate the same actions.

2.4 NoHeartbeat Sub-Mode

MNS should also store a heartbeatEnabled flag in RAM which is loaded from non volatile storage upon power up. MNS should send heartbeat messages if it is in Normal mode and the heartbeatEnabled flag is True.

When MNS receives a MODE(Turn on Heartbeat) message the heartbeatEnabled flag should be set and also persisted in non volatile storage.

When MNS receives a MODE(Turn off Heartbeat) message the heartbeatEnabled flag should be cleared and also persisted in non volatile storage.

2.5 FCU compatibility Sub-Mode

MNS should also store a FCU compatibility flag in RAM which is set to default upon power up. The module should not implement any functionality which could disrupt FCU when in FCU compatibility mode.

Examples:

Operation	Opcode	FCU compatibility mode	
		Enabled	Disabled
Read NVs for NV#0	RDNV(nn, 0)	Return number of NVs	Return number of NVs followed by a NVANS message for each NV
Read Parameters for parameter#0	RQNPN(nn, 0)	Return number of parameters	Return number of parameters followed by a PARAN message for each parameter
Read EVs for EV#0	REVAL(nn, EN#, 0)	Return number of EVs	Return number of EVs followed by a NEVAL message for each EV
Read EVs for EV#0	REQEV(enn, en, 0)	Return number of EVs	Return number of EVs followed by a EVANS message for each EV

When MNS receives a MODE(Turn on FCU compatibility) message the FCU compatibility flag should be set.

When MNS receives a MODE(Turn off FCU compatibility) message the FCU compatibility flag should be cleared.

2.5.1 Default value of FCU compatibility

In order to aid the introduction of VLCB modules by default modules must have the FCU compatibility mode enabled. MMC can enable and disable FCU compatibility mode as required. Disabling the FCU compatibility mode is more efficient in terms of the number of messages used when managing the module.

When FCU is able to handle the VLCB improvements it will be beneficial for the default to have FCU compatibility disabled.

It is therefore recommended that two versions of module firmware are produced, one with FCU compatibility enabled and another for FCU compatibility disabled so that the user can select the most appropriate version depending upon whether they use an old version of FCU.

3 Node Number

A Node Number shall be required for MNS modules, The table below shows the allowed range of numbers that can be freely used

Node Number Ranges	
NN Range	**Use**
0	Reserved for special use by system, E.g. NN==0 is uninitialized.
1-65279 (0x0001-0xFEFF)	Available for user use
65280-65535 (0xFF00-0xFFFF)	Reserved for devices with fixed node numbers

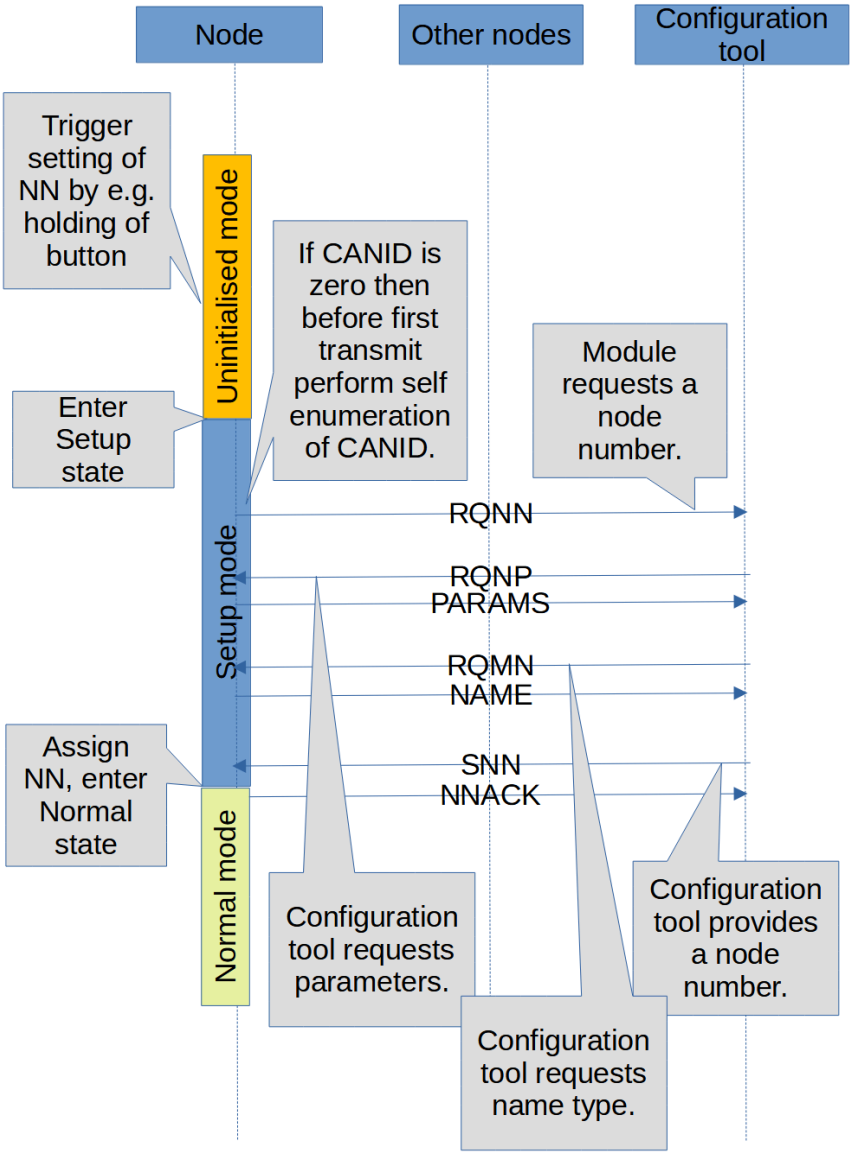
Node number Range

3.1 Duplicate Node Number

The VLCB does not under normal circumstances permit multiple nodes to have the same Node Number. Nodes should avoid this situation, and must implement a detection and reporting (via diagnostics) facility when it sees a response message with its own NodeNumber that it did not send.

3.2 Node Number Assignment

A node must facilitate the assignment of a node number by one or more mandatory methods, namely PushButton, local communication interface or software setup.



Note that, only one node **will** execute a RQNN and SNN, NNREL, NNACK sequence at one time. At the end of this sequence, Parameter 8 Bit 3 shall be set to indicate “Normal” (FLiM) mode.

4 OPCODE support

The list of all MNS mandatory opcodes is recorded here as commands, their parameters, the required action and the required response opcode and action. For the full description, parameter formats and possible errors please see the VLCB MNS opcode specification.

List of Opcodes that MNS must support:

Request to Module	Module's Response	Use/meaning
	RQNN	Request NN (node number). Used to obtain a (new) NN. Enter setup mode.
SNN	NNACK	Set a new NN for the module in setup mode. Exit setup mode
	NNREL	Release previous node number
QNN	PNN	Query NNs
RQNP	PARAMS	Request first 8 parameters
RQMN	NAME	Request module type name
RQNPN	PARAN	Request a single parameter
	CMDERR	Error response
	GRSP	Error response
RDGN	DGN	Request diagnostics
RQSD	SD	Request service discovery
	ESD	Request service discovery
MODE	GRSP	Sets a node's Mode
NNRST		Reset module, just perform a processor restart
NNRSM	NNREL	Restore module to manufacturer's defaults
	HEARTB	Heartbeat message produced when enabled by MODE

5 Service Discovery support

The module **will** implement the VLCB Service Discovery protocol.

A service is any feature which a specific module implements **irrespective** of whether that feature is mandatory or not. The Service Discovery lists the current feature list and its feature identification, called the Service#.

5.1 Service Discovery Request

The service Discovery is a command response system, carried out at normal priority.

On receipt of opcode RQSD (request service discovery) directed at a specific node and with a service equal to zero, that node will respond with a SD message indicating the number of services followed by a SD message for each service supported by the module.

The node will respond within 2 seconds with SD for ServiceIndex zero (Minimum Node Specification). All other supported services within 5 seconds.

On receipt of a message with opcode RQSD directed at a specific node and with a non-zero ServiceIndex, that node will respond with a ESD message for that specific service if supported or GRSP(Invalid service) if the node does not support that requested service.

5.2 Service Discovery Response

There will be one SD response for each service supported by the module. The format and data of the command and response is detailed in the VLCB opcode specification.

The version of the service definition implemented by the module is also returned. This is not the version of the software implementation.

The SD response contains a ServiceType byte to indicate the type of the service. ServiceTypes may be introduced at any time but examples of some ServiceTypes are:

0	None, not a true service
1	MNS
2	NV
3	CAN
4	Event Teaching
5	Event Producer
6	Event Consumer

7	Indexed Event Teaching
8	Event Consume own events
9	EventAck

The SD message also includes a ServiceIndex which is the index into the module's list of services. This allows specific instances of a service to be addressed when a module implements multiple instances of a type of service.
The ServiceIndex values in the SD responses may not be contiguous increasing integers as the module may implement its own internal services which are not exposed in the SD responses.

5.3 ESD Extended Discovery

In addition the module will implement the Service Discovery extended information option. This uses RQSD with a Service#, the node then responds with ESD which contains additional specific information about that feature.

The service must document the meaning of the data bytes within the ESD message.

6 Diagnostics support

A MNS module **will** implement VLCB diagnostics. Each Service can specify its own set of diagnostics, please consult individual Service specifications for the details.

6.1 Heartbeat

6.1.1 Heartbeat message

When enabled by the MODE command the module shall transmit a heartbeat message with the HEARTB opcode. This message shall be broadcast every 5 secs at the lowest priority. This heartbeat message will contain the node number, a sequence counter and a number of global flags indicating module status.

The payload will be

- NodeNumberHi , The high byte of the designated Node.
- NodeNumberLo, The low byte of the designated Node.
- SeqenceCnt, This is a count from 0 incrementing on each message transmitted and wrapping around to zero, It facilitates detection of missing frames.
- StatusByte1: This is a binary representation of the module error states as outlined in Section 8.3 Module Status, 0x00 Shall always represent error free, normal operation.
- StatusByte2: Reserved for future expansion, set to 0x00.

A module will NOT transmit any HEARTB message until it has been assigned a NodeNumber by any specified means., as soon thereafter as practical the HEARTB low priority diagnostic should begin.

The heartbeat message is disabled by manufacturer defaults but may be enabled using the MODE command to turn heartbeat messages on.Similarly heartbeat messages can be disabled using the MODE command to turn heartbeat messages off. Whether the Heartbeat is enabled or disabled must be saved in non volatile memory and restored upon power up.

6.2 MNS Service Specific Diagnostics

A module **must** implement the OPCODE command RDGN and its response DGN opcode. On receipt of the command the module will respond as quickly as practical, with the response packets. The format of the data etc is detailed in the Diagnostics Specification.

Type 2 Command responses should be sent at normal priority, and cannot be disabled.

The following Diagnostics are available:

6.2.1 RDGN Request Diagnostics

This RDGN message may be issued to the module by any node. RDGN takes the following parameters:

- Node NumberHi ; The high byte of the designated node thats is to respond,
- NodeNumberLo : The low byte of the designated node that is to respond,
- Service# : The service identifier,
- DiagnosticCode: see below.

The diagnostic data associated with the specified service and DiagnosticCode is returned in a DGN message. The target node will respond as soon as practical (within 15 secs) with DGN messages.

If DiagnosticCode is specified as zero then a sequence of messages for each DiagnosticCode associated with the service is returned. If the Service# is specified as zero then a sequence of DGM messages is produced for each DiagnosticCode for each Service. This may generate a large number of messages, these must be sent at a rate so that other modules are not stressed. A suggested inter-message time of 10 ms is suggested.

6.2.2 DiagnosticCode for MNS

The DiagnosticCode will indicate the type of information requested. The following DiagnosticCodes are defined by MNS. Other services may define their own diagnostics.

0x00: return a series of DGN messages for each supported DiagnosticCode data.

0x01: return module status code.

0x02: return uptime upper word.

0x03: return uptime lower word.

0x04: return memory error count.

0x05: return number of Node Number changes.

0x06: return number of received messages acted upon.

6.2.3 Diagnostics Data

On receipt the Node will respond with DGN and valid Data or GRSP (see opcodes).

The DGN response message contains two data bytes: DiagnosticVal1 and DiagnosticVal2.

Two bytes of data are always returned, 0x00 being used for no data.

The DiagnosticVal bytes may be combined to form a 16 bit counter value with DiagnosticVal1 being the higher byte and DiagnosticVal2 the lower byte. -

6.2.4 MNS Diagnostics payload data return

The DiagnosticByte1 and DianosticsByte2 are defined for the MNS service and each DiagnosticCode thus:

DiagnosticCode	DiagnosticByte1	DiagnosticByte2	Description
0x01	STATUS	0x00	Facsimile of the global module status byte, see section 6.3 Module Status
0x02	UPTIME Upper Hi	UPTIME Upper Lo	Upper Word of 32bit uptime measured in seconds.
0x03	UPTIME Lower Hi	UPTIME Lower Lo	Lower Word of 32bit uptime measured in seconds.
0x04	MEMFLT	0x00	Memory fault indicator. 0x00: no fault , 0x01: Flash write fault, 0x02: Flash read Fault , 0x04: EEPROM read fault , 0x08: EEPROM Write fault , 0x10: Stack overflow, 0x20: RAM fault , 0x80: general unspecified memory error. Indicator values may be OR'ed together.
0x05	NNCNT Hi	NNCNT Lo	Number of nodeID changes. The node number was assigned or reassigned since power up by any node assignment method.
0x06	MESSACTED Hi	MESSACTED Lo	Count of messages processed by the module. The principle is this count is messages the module "acted on" in some way.

6.3 Module Status

The Heartbeat message and DiagnosticCode 1 contain a module status byte providing an overall view of the health of the module. This is represented by a count of recent errors.

The module status shall be calculated by having an 8-bit counter which is incremented whenever an error diagnostic is incremented by any of the module's services. This increment should max at 255 and not rollover to 0. The counter shall be decremented if it is above 0 every 5 seconds.

7 MNS Service Specific GRSP response codes

Codes 1~12 match those of CMDERR. Additional error codes have been added with error codes beyond those used with CMDERR, starting at 0xFF, and descending.

GRSP Generic Response		
Basic responses, matching CMDER		
Result Code	Description	Comment
0	ok	
1	Command Not Supported.	Used for Invalid Command
2	Not In Learn Mode.	
3	Not in Setup Mode.	
4	Too Many Events.	
5	No Event.	
6	Invalid Event variable index.	
7	Invalid Event.	
8	Reserved.	
9	Invalid Parameter Index.	
10	Invalid Node Variable Index.	
11	Invalid Event Variable Value.	
12	Invalid Node Variable Value.	

Additional MNS specific GRSP codes:

Code	**Service #**	**Error**
0	1	OK
252	1	Invalid Service
253	1	Invalid diagnostic code.
254	1	Unknown non-volatile memory type.

8 Service Specific Automatic Power on Tests

MNS does not support any automatic power on tests.

9 Service Data

9.1 Module Parameters

See https://merg.org.uk/merg_wiki/lib/exe/fetch.php?media=cbus:cbus_-_new_parameter_structure-6.pdf for details for parameter structure.

The module parameters provide information about the module's capability, information for the bootloader and information about the physical hardware. Entries in red are deprecated and replaced by service discovery but included for backwards compatibility. Entries in blue are only required for the CBUS PIC bootloader.

The following table describes the parameters supported by the Minimum Node Service. Other parameters should be set to 0 unless other services included by the module specify other values.

Param#	Name	Usage	VLCB should set these values
0	Number of parameters		The number of supported parameters. E.g. 20 or 24
1	Manufacturer	module manufacturer	See ModuleId
2	Minor version		The minor version ascii character x.Y.z.
3	ModuleId	Module type identifier.	Combined with Manufacturer for a 16 bit code unique to the type of module. The ID can be obtained by the module developer using the VLCB web portal.
7	Major version		The major number X.y.z
8	flags	Module's capabilities and settings	See below for each bit.
8.2	Normal (FLiM)	Indicates if the module is in Normal mode (1) or Uninitialised (0).	Set to 1. VLCB does not support an equivalent of CBUS SLiM.

8.6	VLCB compliant	Indicates if the module supports VLCB capabilities	Set to 1. CBUS modules should have this set to 0.
20	Beta version		The Patch number x.y.Z

9.2 ESD data bytes

The Minimum node service currently does not return any data within the ESD response, all data bytes are set to 0.

10 Service Specific Modes

MNS requires the following MODE Commands:

Mode Command	Request
0x00	Change to Setup mode
0x01	Change to Normal mode
0x0C	Turn on Heartbeat
0x0D	Turn off Heartbeat
0x10	Turn on FCU compatibility
0x11	Turn off FCU compatibility

Note that MNS should handle mode commands less than 8 according to the processes defined in section 2.1 MNS Modes.

The MODE opcode for enabling and disabling Heartbeat messages shall be obeyed when the specified NN matches the module's Node Number to control heartbeat for a single module and also when NN=0 to enable or disable heartbeat messages across all modules on the network.

11 Service Documentation

None.

12 Glossary

CAN	Controller Area Network. A standard communications bus originally defined by Bosch. Widely used in cars, industry and other electrically noisy environments.
CANID	CAN identifier
CBUS	A set of messages for model railway control. The CBUS system was developed over 4 years by Mike Bolton and Gil Fuchs and introduced with specifications and an initial range of kits in 2007. Since then the system has been further developed by many MERG members into a very comprehensive Layout Control Bus.
EN	Event Number
EN#	Event Number index
EV#	Event Variable index
FLiM	Full Layout implementation Mode
GridConnect	A means of encoding CAN frames for transmission over an ASCII serial link. See https://www.google.com/url?q=https://www.gridconnect.com/products/can-usb-adapter-pcan-usb&sa=D&source=docs&ust=1665387273449617&usg=AOvVaw0fepa9tJq-858M5sbX66fV
MNS	Minimum Node Specification
NN	Node Number
Parameter	Describe the capabilities of a module. Parameters are read-only and set by the module's firmware. Some parameters are dynamic and can change during module operation.

PCB	Printed Circuit Board
SLiM	Simple Layout implementation Mode