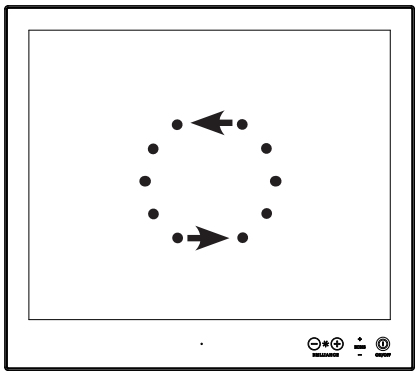
HATTELAND TECHNOLOGY
an EMBRON Company

TECHNICAL MANUAL



Serial Communication Control Interface (SCOM)

Applies for Series X Maritime Multi Computer (MMC) Generation 1 (G1):

HD 12T21 xxC-xxx-Fxxx
HD 15T21 xxC-xxx-Fxxx
HD 17T21 xxC-xxx-Fxxx
HD 19T21 xxC-xxx-Fxxx
HD 24T21 xxC-xxx-Fxxx
HD 26T21 xxC-xxx-Fxxx

Technical Manual SCOM Series X G1	
Updated: 10 May 2022	Doc Id: INB100018-10 (Rev 21)
Created: 6542/6784/363	Approved: 6644

Please visit www.hattelandtechnology.com for the latest electronic version of this manual.

Contents

Serial Communication (SCOM) Interface

Introduction

This document defines the electrical interface, serial data format, and communication protocols of the Communication Control Interface (SCOM). The purpose of this interface is to enable a computer application to control the unit. Unit refer to panel computers (Maritime Multi Computer - MMC, Interface configuration done via VCOM and/or BIOS).

Serial Interface Configuration

The serial interface can have different configurations which are defined as follows:

RS-232	One computer controls one unit, no individual address

Cables

Serial Mode: No cables needed, internal support.

Panel Computer (Maritime Multi Computer - MMC) based units are by factory default manufactured with:

• COM RS-232 : Internal Virtual COM **x** (VCOM), where **x** is OS/configuration dependent.

This Virtual COM port enables you to send commands TO the same unit that you are sending FROM, which means there is no need to differentiate between local control or remote control for communicating through physical ports such as RS-232, RS-485/RS-422 towards other units located externally. VCOM requires a specific Hatteland Technology software driver installed prior to use/communcation attempts.

Installing API/VCOM Drivers

In order to access and communicate with the Panel Computer (MMC) units, it may or may not already have pre-installed drivers depending on factory defaults. If you need to install or re-install drivers, please follow the instructions as described below:

1: Available product range hardware drivers on our website (19 inch used as example):
- https://www.hattelandtechnology.com/drivers?key=X.X.X.X.X.X.301

"HATTELAND® BD82QM57 API Package" requires Visual C++ Redistributable prerequisites from Microsoft®:
- https://www.hattelandtechnology.com/drivers?key=18.10.8.57.X.X.X

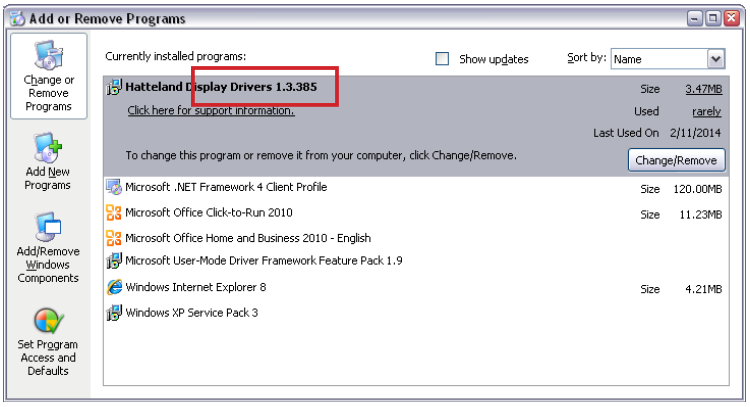
Specific API drivers "HATTELAND® BD82QM57 API Package":
- https://www.hattelandtechnology.com/drivers?key=17.15.32.X.X.X.X

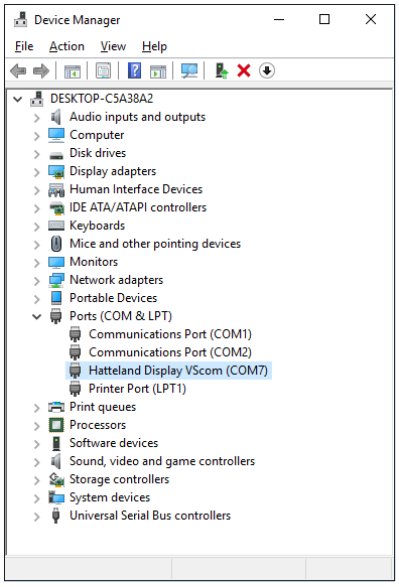
Additional documentation:
https://www.hattelandtechnology.com/hubfs/pdf/misc/doc101163-1_hatteland_display_api_qm57.pdf

How to determine version installed of API/VCOM drivers

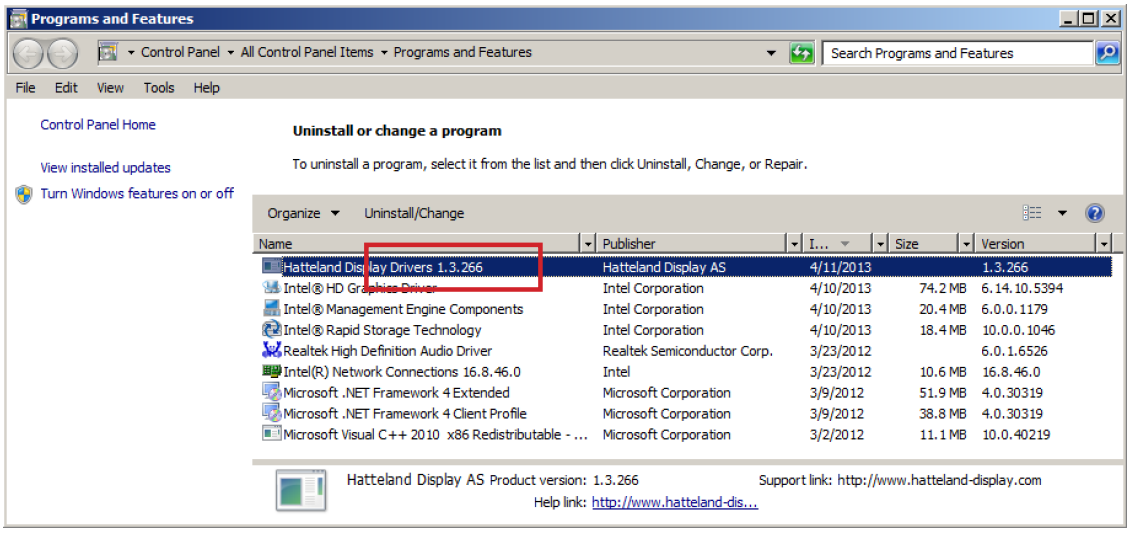
If you have previously installed the API/VCOM drivers on a Panel Computer (MMC), here is how you can find version information via several methods (Microsoft® Windows® Operating systems only):

Via "Control Panel / Add or Remove Programs" or "Control Panel / Programs and Features" (OS dependent) and within Device Manager, showing Port Number:





Example above indicates version "1.3.385".



Example above indicates version "1.3.266".

Via registry (regedit.exe) - Experienced users only!
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Hatteland Display Drivers]
- "DisplayName"="Hatteland Display Drivers x.y.zzz"
- "DisplayVersion"="x.y.zzz"

Where x.y.zzz is version number, example "1.3.266".

Serial Communication (SCOM) Interface

Data Rates
The unit is configured to transmit and receive data at 9600 bits/second (Serial mode).

Data Format Serial Mode
Data shall be transmitted with no parity, 8 data bits, one start bit and one stop bit. XON/XOFF flow control should be switched off/disabled.

Message Format
The basic message format shall be as follows:

Byte #	0	1	2	3	4	5	6	7..etc	End Byte
	ATTN	ADDR	CMD	CMD	CMD	LEN	IHCHK	DATA	IDCHK

The minimum message size is 7 bytes (0x07). The maximum message size is 82 bytes (0x52), consistent with the EN61162-1 standard. Colors will be used throughout this manual to indicate byte positions. Every byte sent are viewed in this document as HEX values and are based on standard characters in the ASCII table (0 to 255) to send or receive messages in a human readable input/output. No further decoding or decrypting functionality is needed or required. Every command sent and received are always ended with a 0x00 (null byte terminator).

ATTN

Attention (ATTN)
This single byte is used to identify a start of message. 3 values are possible:

ATTN	Description
0x07	Command, also known as ASCII BELL
0x06	Acknowledge, also known as ASCII ACK
0x15	Negative Acknowledge, also known as ASCII NAK

A device shall send a command using the 0x07 Attention Code. The unit will respond to the command with either an ACK if the command completed successfully, or a NAK if the command failed.

NOTE: A complete HEX, ASCII, BIN and Character table overview are available in the APPENDIX chapter.

ADDR

Address (ADDR)
This single byte is used to specify a particular unit to receive a Command and to identify the unit responding (ACK or NAK) to a Command. All units will support the broadcast address. The factory default adress is 0x00, while in this manual illustrated throughout as 0xFF.

The Address field shall have the following values:

ADDR	Description
0xFF	Broadcast - Addressed to single unit

CMD

Message Commands and Queries (CMD) Contents

The command can be one of the following values and consists always of 3 bytes in positions 2,3,4:

Byte 2	Byte 3	Byte 4	ASCII	Description	I/O	Page
0x42	0x52	0x49	"BRI"	Backlight Minimum Value	W	10
0x42	0x52	0x4C	"BRL"	Set LED Glass Display Control™ (GDC) Brightness	R/W	10
0x42	0x52	0x4D	"BRM"	Backlight Maximum Value	W	10
0x42	0x52	0x54	"BRT"	User Brightness Control	W	11
0x42	0x5A	0x5A	"BZZ"	Buzzer Control	W	11
0x44	0x4C	0x3F	"DL?"	Query available ECDIS packages	R	12
0x44	0x4C	0x4E	"DLN"	Download ECDIS package table x	R	12-13
0x45	0x54	0x43	"ETC"	Elapsed Time Counter Query	R	13
0x47	0x4D	0x42	"GMB"	Set LED Glass Display Control™ (GDC) Minimum Brightness	W	14
0x4C	0x49	0x53	"LIS"	Read Ambient Light Sensor	R	14
0x53	0x4E	0x42	"SNB"	Serial Number Query	R	14
0x53	0x57	0x49	"SWI"	Main Software (Video Controller) Version Query	R	15
0x53	0x57	0x4B	"SWK"	Glass Display Control™ (GDC) software version Query	R	15
0x54	0x59	0x50	"TYP"	Type/Model Number Query	R	16
0x51	0x44	0x55	"QDU"	QM57 Chipset Duplex Mode	W	16

I/O = R=Read, W=Write.
Page # = Page number in this manual where command is detailed.

NOTE: Due to firmware revisions, some commands listed above will not be available on earlier units.

LEN

Data Length (LEN)

This single byte defines the length of DATA in the message in bytes. The maximum value for this field is 74 bytes (0x4A in HEX). The minimum value is 0 bytes (0x00 in HEX).

IHCHK

Inverse Header Checksum (IHCHK)

This single byte is a simple 8-bit checksum of the header data, message bytes 0 to 5 on which a bit-wise inversion has been performed. The checksum shall be initialised to 0. The 8-bit sum (without carry) of bytes 0, 1, 2, 3, 4, 5 and 6 shall be 0xFF (255 in value). If the unit receives a message with an incorrect checksum, the unit will reply with the attention code set to NAK and no data field. This does not apply to Broadcast messages in RS-485 mode, in which case there will be no reply.

DATA

Data Field (DATA)

The single byte is the DATA field which shall only be transmitted if LEN is greater than 0. This field depends on the CMD transmitted.

IDCHK

Inverse Data Checksum (IDCHK)

This single byte shall only be transmitted if LEN is greater than 0. This is a simple 8-bit checksum of the data field, message bytes 7 to 7+(LEN-1) on which a bit-wise inversion has been performed. The checksum shall be initialised to 0. The 8-bit sum (without carry) of bytes 7 through 7+LEN inclusive shall be 0xFF. The receiver will reply to any message that the checksum has failed with the attention code set to NAK. This requirement does not reply to broadcast messages in RS-485 (for units that support it) mode, in which case there will be no reply. Basically this byte is located at the very end of a received stream.

NOTE: A complete HEX, ASCII, BIN and Character table overview are available in the APPENDIX chapter.

"BRI" - Backlight Minimum Value

Set the backlight minimum value. Range from 0x00 to 0xFF (0% - 100%).

Example:

Command to set 50% Brightness:

0x07	0xFF	0x42	0x52	0x49	0x01	0x1B	0x80	0x7F

"BRL" - Set LED Glass Display Control™ (GDC) Brightness

Set the backlight intensity for the Glass Display Control™ (GDC) LED's on the front glass. From 0x00 to 0x31 (0% - 100%). The brightness value shall be sent as 1 byte in the DATA field. A setting of 0x00 shall indicate off. A setting of 0x31 shall indicate maximum brightness. You can send "?" to retrieve the current value/status of the LED's.

Example:

Command to set 60% intensity:

0x07	0xFF	0x42	0x52	0x4C	0x01	0x18	0x32	0xCD

"BRM" - Backlight Maximum Value

Set the backlight maximum value. Range from 0x00 to 0xFF (0% - 100%).

Example:

Command to set 60% Brightness:

0x07	0xFF	0x42	0x52	0x4D	0x01	0x17	0x99	0x66

"BRT" - User Brightness Control

This command is sent to the unit to command the backlight brightness control setting.The brightness value shall be sent as one byte in the DATA field. A setting of 0x00 will indicate off. A setting of 0xFF (255 in value) will indicate maximum brightness. Intermediate values will control brightness over the range from minimum to maximum luminance. LEN = one data byte.

After any power cycle the BRT will be set to 100%.

If the data checksum is valid and the brightness was set, the unit will reply to this command with an ACK attention code. The DATA field in the reply shall indicate the resulting brightness control setting. If an invalid checksum was received and the message was not Broadcast and RS-485, the unit will reply with an NAK attention code. The DATA field in the reply will indicate the current brightness control setting.

Example:

If BRT is 100%, the user can adjust the brightness from 0-100%. If the BRT is set to 60%, the visual brightness is set to 60%. The user can adjust the brightness from 0-100% within the 60% set by BRT. If the user sets the potensiometer to half, the visual brightness will be 30% (half of 60%). If BRT is set back to 100%, the visual brightness will be 50% (half of 100%).

Command to set 60% Brightness:

0x07	0xFF	0x42	0x52	0x54	0x01	0x10	0x99	0x66

Acknowledge was set to 60% Brightness:

0x06	0xFF	0x42	0x52	0x54	0x01	0x11	0x99	0x66

Negative Acknowledge 40% Brightness:

0x15	0xFF	0x42	0x52	0x54	0x01	0x02	0x66	0x99

"BZZ" - Buzzer Control

This command is sent to the unit to control buzzer on/off if there is a buzzer present. LEN = one data byte.

0x00	Turn the buzzer off
0xFF	Turn the buzzer on

If the data checksum is valid, the unit will reply to this command with an ACK attention code. The DATA field will indicate the buzzer state. If an invalid data checksum was received and the message was not broadcast and RS-485, the unit will reply with a NAK attention code and the current control setting.

Example:

Command to set Buzzer Enable: 0xFF

0x07	0xFF	0x42	0x5A	0x5A	0x01	0x02	0xFF	0x00

Acknowledge Buzzer status was set to Enable: 0xFF

0x06	0xFF	0x42	0x5A	0x5A	0x01	0x03	0xFF	0x00

"DL?" - Query available ECDIS packages

This command will query the unit in order to aquire how many packets (1 packet = 32 bytes) are in the ECDIS memory table (if available) that are available for download. Packet counting starts from 0, so a response of 0x03 is naturally 4 actual packets (4*32 = 128 bytes as ASCII text). If packets end up as example 42.34, it will always be rounded upwards, i.e. 43 packets.

The "DL?" command does not indicate which memory slot was used to store the ECDIS table, so please use SLOT 1 (0x00).

Product Range	Slots Available	OSD ID	OSD Setting
Maritme Multi Computer (MMC)	Internal (DVI/LVDS)	**2**	No Calibration
	Internal (DVI/LVDS)	**0**	No OSD setting available to verify/change

Example:

Command for query num of packets:

0x07	0xFF	0x44	0x4C	0x3F	0x00	0x2A

Acknowledge: DL? indicates 4 available packets:

0x06	0xFF	0x44	0x4C	0x3F	0x01	0x2A	0x03	0xFC

"DLN" - Download ECDIS package table x

Before sending this command, use "DL?" to retrieve how many packets are actually available in the ECDIS table. If you request a download package from a empty slot or above the available packets in memory, you will get a NAK response. The "DLN" command shall be sent to the unit to request a specific data packet stored in the unit's memory. The DATA field shall contain the packet number being requested; the byte in the DATA field represents a hexadecimal word (00 to FF) identifying the block of data to be downloaded.

If the data checksum and packet number is valid, the unit shall reply to this command with an ACK attention code, the hexadecimal packet number, a separator ('-'), and the ASCII packet data. The maximum DATA field size for a packet of data shall be 74 bytes per message; therefore the DATA field in the reply shall be a maximum length of 74 bytes. The DATA field of the message is not required to be of maximum length (it may be smaller than 74 bytes).

Example:

Command to read package number 1,2,3,4 (0x00, 0x01, 0x02, 0x03) from Slot ID 0x00:

0x07	0xFF	0x44	0x4C	0x4E	0x00	0x1B	0x00	0xFF
0x07	0xFF	0x44	0x4C	0x4E	0x00	0x1B	0x01	0xFE
0x07	0xFF	0x44	0x4C	0x4E	0x00	0x1B	0x02	0xFD
0x07	0xFF	0x44	0x4C	0x4E	0x00	0x1B	0x03	0xFC

.... until the end of available packages.

ECDIS table 0 is stored first and read with only packed nr. To read from other table, another byte is added for table nr. The DATA field shall contain the packet number being requested and table nr; the first byte in the DATA field represents a hexadecimal word (00 to FF) identifying the block of data to be downloaded.

The second byte in the DATA represents the table nr from 1-x. If the data checksum and packet number is valid, the unit shall reply to this command with an ACK attention code, the hexadecimal packet number, a separator ('-'), and the ASCII packet data. The maximum DATA field size for a packet of data shall be 74 bytes per message; therefore the DATA field in the reply shall be a maximum length of 74 bytes. The DATA field of the message is not required to be of maximum length (it may be smaller than 74 bytes).

Continued on next page...

Acknowledge Example of a successful DLN query (4 first packages illustrated, actual ASCII text begins at WHITE colored cells):

0x06	0xFF	0x44	0x4C	0x4E	0x20	0xFC	0x00	0x2D	0x56	0x42	0x31	0x30	0x30	0x30	0x31	0x34	0x2D
0x31	0x20	0x42	0x75	0x69	0x6C	0x64	0x20	0x6E	0x75	0x6D	0x62	0x65	0x72	0x3A	0x20	0x56	0x65
0x72	0x73	0x69	0x9A														

0x06	0xFF	0x44	0x4C	0x4E	0x20	0xFC	0x01	0x2D	0x6F	0x6E	0x3A	0x20	0x33	0x2E	0x30	0x2E	0x30
0x2E	0x31	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20	0x20
0x31	0x38	0x2E	0xB5														

0x06	0xFF	0x44	0x4C	0x4E	0x20	0xFC	0x02	0x2D	0x30	0x39	0x2E	0x32	0x30	0x31	0x34	0x0A	0x48
0x44	0x31	0x39	0x54	0x32	0x31	0x4D	0x4D	0x44	0x4D	0x41	0x31	0x46	0x41	0x47	0x41	0x20	0x2D
0x31	0x30	0x32	0x2F														

0x06	0xFF	0x44	0x4C	0x4E	0x20	0xFC	0x03	0x2D	0x32	0x31	0x0A	0x0A	0x5B	0x47	0x72	0x61	0x70
0x68	0x69	0x63	0x61	0x6C	0x20	0x49	0x6E	0x74	0x65	0x72	0x66	0x61	0x63	0x65	0x5D	0x0A	0x4D
0x61	0x74	0x72	0xC6														

ASCII Contents of package 1,2,3,4 (reference example above) received is, 111 bytes:

```
VB100014-1 Build number: Version: 3.0.0.1                18.09.2014
HD19T21MMCMA1FAGA -10221

[Graphical Interface]
Matr
```

"ETC" - Elapsed Time Counter Query

The unit features an elapsed time counter which counts the total number of hours that the unit has been operated. No data shall be sent with this command. The unit will reply to this command with an ACK attention code. The DATA field will be set to a 3 byte string, where the most significant byte is transmitted first.

The ETC has a limit on maximum 99999 hours (defined into 5 bytes, numbers 0 to 9 each), equivalent to 11 years. If this number is reached, the ETC will stop counting, and the ETC command will always reply with maximum number of hours (99999).

Example:

Command to Query ETC:

0x07	0xFF	0x45	0x54	0x43	0x00	0x1D

Acknowledge "0" + "0" + "0" + "5" + "8" = (combined "00058" or 58 hours):

0x06	0xFF	0x45	0x54	0x43	0x05	0x19	0x30	0x30	0x30	0x35	0x38	0x02

The ETC value has been hardcoded to 5 bytes, prefixed with "0".

"GMB" - Glass Display Control™ (GDC) Minimum Brightness

Sets the minimum value limit for the backlight of the LED's on the GDC, making sure that the LED's and GDC symbols are still visible if they was previously adjusted down to a very low value by the "BRL" command or via the "LED Drive" function from within the OSD menu (or set via the MCC command).
Range from 0x00 to 0x31 (0% - 100%).

Example:

Command to set 50% Minimum Brightness limit:

0x07	0xFF	0x47	0x4D	0x42	0x01	0x22	0x19	0xE6

"LIS" - Read Ambient Light Sensor

Sending this command the light sensor on the Glass Display Control™ (GDC) will return a value about luminance of environment.

Example:

Command to query Light Sensor:

0x07	0xFF	0x4C	0x49	0x53	0x00	0x11

Acknowledge: LIS value is "0"+"0"+"9"+"9"+"9" = "999":

0x06	0xFF	0x4C	0x49	0x53	0x05	0x0D	0x30	0x30	0x39	0x39	0x39	0xF4

"SNB" - Serial Number Query

This query is sent to the unit in order to identify the unit serial number. No data shall be sent with this query.

The unit will reply to this command with an ACK attention code. The DATA field will be set to an ASCII text string to indicate the specified Serial Number, e.g: "12345". Note that the length of Serial Number is not limited to 5 characters. It will decrease or increase in length depending on actual Serial Number stored in the unit.

Example:

Command Display Serial Number

0x07	0xFF	0x53	0x4E	0x42	0x00	0x16

Acknowledge Type/Model Number "12345":

0x06	0xFF	0x53	0x4E	0x42	0x05	0x12	0x31	0x32	0x33	0x34	0x35	0x00

"SWI" - Main Software Version Query

The SWI command is a legacy command which is backward compatible with already existing customer setups (i.e. product ranges released before Series X). For newer systems and implementations, please use "FWV" command when possible. Any future revisions, such as CMD additions or changes to the software will increment the software version. The unit will reply to this command with an ACK attention code. The DATA field will be a ASCII text string indicating the software version, e.g: "120801V1_M".

Byte	Description	Applies to	Example reply as ASCII text after query
0x00	VCOM (Virtual internal COM) firmware version	MMC	SW101010-0181

Example:

Command Software Version query:

0x07	0xFF	0x53	0x57	0x49	0x00	0x06

Acknowledge GEV software query: "120801V1_M"

0x06	0xFF	0x53	0x57	0x49	0x0A	0xFD	0x31	0x32	0x30	0x38	0x30	0x31	0x56	0x31	0x5F	0x4D	0xA0

"SWK" - Glass Display Control™ (GDC) software version Query

This command indicates the firmware (Cypress) version of the Glass Display Control™ (GDC) software.

Example:

Command Software Version query:

0x07	0xFF	0x53	0x57	0x4B	0x00	0x04

"TYP" - Type/Model Number Query

This query is sent to the unit in order to identify the unit type by its model number / part number. No data shall be sent with this query.

The unit will reply to this command with an ACK attention code. The DATA field should be translated to an ASCII text string which indicate the specified Type/Model Number, e.g: "HD17T21MMCMJDOABA".

FYI: Which translates to "HD 17T21 MMC" with a configuration added to it by using the letters "-MJD-OABA". The meaning of "-MJD-OABA" can only be found in the description field of invoice documents and service documents. In this example the exact description for the unit is:
"17.0" MMC IntelQM57 CEL 1.86GHz 2GBRAM 250GBHDD OSNone ACDC Bonded Black GDC Buzzer"

Example:

Command to retrieve Type/Model Number

0x07	0xFF	0x54	0x59	0x50	0x00	0xFC

Acknowledge Type/Model Number "HD17T21MMCMJDOABA":

0x06	0xFF	0x54	0x59	0x50	0x11	0xEC	0x48	0x44	0x31	0x37
0x54	0x32	0x31	0x4D	0x4D	0x43	0x4D	0x4A	0x44	0x4F	0x41
0x42	0x41	0x89								

"QDU" - QM57 Chipset Duplex Mode

This command sets the half or full duplex mode for the external physical RS-422 / RS-485 port on a MMC unit.

0x00	Half Duplex Mode
0xFF	Full Duplex Mode

Operational Requirements

The following sections define the operational requirements.

Serial Message Failure
If serial messages stop being transmitted or are corrupt, the unit will remain at the last commanded brightness.

Periodic Messages
Commands shall be transmitted to the unit at a repetition no faster than 4 Hz.

Sending Multiple Commands / Command Queue
To ensure all commands are transmitted and executed successfully on the unit, a delay between each command in the queue shall be at least 500ms. Some internal commands require slightly longer to process internally in the unit, than others.

Keep-alive Alarm
The "SWI" query can be used for keep-alive alarm logic in the application software on the computer. It is recommended to limit this function to once a second (1000ms).

Individually Addressed Command Response Time
The unit will output the required response within Tr = 2.5 character periods after the last byte of a command message is received (2.6ms at 9600 bit/sec for Serial Mode only), except as specified herein.

Broadcast Command Response Time
In response to Serial mode RS-485 broadcast command messages, after the last byte of the command message is received, all units will reply within the time period defined for Te, below. Further more, any gap between these individual responses will be less than the Intermessage Gap, defined below.

Te = (Tr + Lr) * N, where
Lr = length of the ACK/NAK message response
Tr = response time
N = the total number of units*

*) As the units reply in order to their address, the units must be given subsequent addresses, starting at zero, for N to equal the total number of units. If not, N = the highest unit address + 1.

The maximum Lr for a selected command set are shown in the table below:

Command	BRT	BZZ	ETC	POT	SNB	SWI	SWK	TYP	MCC
Lr	9	9	11	9	13	19	11	28	xx**

**)This command will vary in size, and response time is longer. Make sure ACK is received from all units before sending a new command.

Example:
For the BRT command, and 8 units, this corresponds to Te = (2.5 * 10 + 9 * 10) * 8 / 9600 = 95.8 ms

Intermessage Gap - Serial Mode
Following an individually addressed command, the next command shall not be issued until at least Tg = 5 character periods after the ACK or NAK message received. At 9600, that is 5 * 10 / 9600 = 5.2ms.

Following the issue of a broadcast command message, the next command shall not be issued until at least Tc = Te + Tg, where Te is as defined for Broadcast Command response and Tg is defined above.

Unit Response and Addresses

When individual unit addressing is supported by an installed configuration of units in a RS-485 (for units that support it) system, a separate ACK or NAK message for each unit will be transmitted providing each unit's individual address in response to any broadcast addressed Command.

NAK messages will not be generated when an error in a Broadcast message is detected. When individual unit addressing is not supported, the unit will only respond to the broadcast address and will include the broadcast address in the ACK and NAK messages. NAK messages will not be generated when an error in a Broadcast message is detected.

When a unit receives an incomplete message and the next byte is not received until after a time equal to the Intermessage Gap, the next bytes received shall be processed to check for the start of a new command (0x07, ASCII Bell).

If the header checksum is valid, but the first byte of the command message is not 0x07, as specified, the unit may wait until after the next inter-message gap to resume checking. A NAK message shall not be generated.

If the header checksum is valid, but the value of the CMD field does not equal one of the defined commands, the unit shall reply by generating a NAK message as though a VER command had been received.

If the header checksum is valid, but the value of the LEN field is greater than the maximum allowed, the unit shall ignore the message. A NAK message shall not be generated.

If the data checksum is valid, but the value in the DATA field associated with a command is invalid (out of range, undefined, etc.), the unit shall generate a NAK message indicating the current data value in the DATA field.

Additional Commands

In time, additional commands and corresponding data fields may be defined. These additions will not conflict with the operation of the interface as defined herein in this document.

Appendixes

Calculating Checksums (IDCHK, IHCHK)

Here is a simplified method to calculating checksum. The example is using decimal numbers, for explanation purposes only. The actual values are in hexadecimal throughout the user manual.

Visual representation of the byte:

Attention	Address	CMD	CMD	CMD	Length	IHCHK	DATA	IDCHK

Header

Number Base Systems mentioned in this section.

Binary	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal	128	64	16	16	8	4	2	1
Hex	80	40	20	10	8	4	2	1

The total value of a byte is represented by 8 bits, all bits have the value of either 0 or 1. One byte can represent a decimal number between 0 and 255 (256 different combinations).

For example: let us convert 55 from decimal to binary. We place in the table below (marked in green) 1 or 0 for the highest available decimal number in that cell and subtract until we reach 0.

So in the case of 55, it is: 55 - 32 - 16 - 4 - 2 - 1 = 0

1 or 0	0	0	1	1	0	1	1	1
Binary	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal	128	64	32	16	8	4	2	1

This results in binary value 0011 0111 based on the input in the table, which in decimal are 55, and HEX 0x37.

IDCHK

Calculating IDCHK require us to find the inverted value of the sum of all data bits. We send a data package with the data value as 55, which we know is 0011 0111. The inverted data value is the inverted of 0011 0111 (55 converted to binary), which is 1100 1000.

If we then convert 1100 1000 to decimal using the conversion table above, we get 200
In simpler terms, you could say we are doing 55 - 255 = -200. Ignore the fact that it is a negative number.

Attention	Address	CMD	CMD	CMD	Length	IHCHK	55	200

Calculating Checksums (IDCHK, IHCHK)

IDCHK with two data fields (2 bytes). DATA1 DATA2 IDCHK

Calculating IDCHK while having two data fields (or more) is almost the same as single data field. Sum the two data fields, subtract 1 for every time you exceed 255 and start from 0.

Example of 2 data bytes:

Data field 1: 55 or as HEX 0x37
Data field 2: 230 or as HEX 0xE6

Add all the data fields together.
230 + 55 = 285

Subtract 255 until the summed data value is below **255**.
285 - **255** = 30

Subtract 1 for every time subtracted **255** above.
30 - 1 = 29

Subtract 29 based on decimal values from the binary table on previous page until you reach 0:
29 - 16 - 8 - 4 - 1 = 0

Which gives us binary number (by using explanations on previous page).
0001 1101
inverted is (by using explanations on previous page).
1110 0010

Convert 1110 0010 to decimal is 226 or as HEX 0xE2

Attention	Address	CMD	CMD	CMD	Length	IHCHK	55	230	226

IDCHK with three data fields (3 bytes). DATA1 DATA2 DATA3 IDCHK

Example using 3 data bytes:

Data field 1: 233 or as HEX 0xE9
Data field 2: 229 or as HEX 0xE5
Data field 3: 228 or as HEX 0xE4

Add all the data fields together.
233 + 229 + 228 = 690

Subtract **255** until the summed data value is below **255**.
690 - 255 = 435
435 - 255 = 180

Subtract 2 for every time subtracted **255** above.
180 - 2 = 178

Subtract 178 based on decimal values from the binary table on previous page until you reach 0:
178 - 128 - 32 - 16 - 2 = 0

Which gives us binary number (by using explanations on previous page).
1011 0010
inverted is (by using explanations on previous page).
0100 1101

Convert 0100 1101 to decimal is 77 or as HEX 0x4D

Attention	Address	CMD	CMD	CMD	Length	IHCHK	233	229	228	77

Calculating Checksums (IDCHK, IHCHK)

IHCHK

Let us calculate IHCHK. We send the same data package, the data value does not matter for this calculation. Instead, we focus on the following formula that sums all the header values and inverts them. Note that DATA of 55 and IDCHK of 200 is present in table below as described on previous page. Length is set to 1 as there is only 1 byte in the DATA field.

Attention	Address	CMD	CMD	CMD	Length	IHCHK	Data	IDCHK
7	255	66	82	73	1	IHCHK	55	200

Add together as indicated below.

Attention field:	7	or as HEX	0x07
Address field:	255	or as HEX	0xFF
CMD 1 field:	66	or as HEX	0x42
CMD 2 field:	82	or as HEX	0x52
CMD 3 field:	73	or as HEX	0x49
Length field:	1	or as HEX	0x01

(Length of 1 indicates only 1 byte 55 is present in DATA field).

Add all the fields together.

7 + 255 + 66 + 82 + 73 + 1 = 484

Subtract 255 until the summed data value is below **255**.

484 - **255** = 229

Subtract 1 for every time subtracted **255** above.

229 - 1 = 228

Subtract 228 based on decimal values from the binary table (marked in green) until you reach 0:

228 - 128 - 64 - 32 - 4 = 0

1 or 0	1	1	1	0	0	1	0	0
Binary	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal	128	64	32	16	8	4	2	1

Which gives us binary number (by using explanations on previous page).

1110 0100

inverted is (by using explanations on previous page).

0001 1011

Convert 0001 1011 to decimal is 27 or as HEX 0x1B

Attention	Address	CMD	CMD	CMD	Length	IHCHK	Data	IDCHK
7	255	66	82	73	1	27	55	200

HEX, ASCII, BIN and Character table

HEX	DEC	BIN	Character/Symbol	Description
0x00	0	00000000	NUL	Null terminator / character / End of string
0x01	1	00000001	SOH	Start of Heading
0x02	2	00000010	STX	Start of Text
0x03	3	00000011	ETX	End of Text
0x04	4	00000100	EOT	End of Transmission
0x05	5	00000101	ENQ	Enquiry
0x06	6	00000110	ACK	Acknowledgment
0x07	7	00000111	BEL	Bell
0x08	8	00001000	BS	Back Space
0x09	9	00001001	HT	Horizontal Tab
0x0A	10	00001010	LF	Line Feed
0x0B	11	00001011	VT	Vertical Tab
0x0C	12	00001100	FF	Form Feed
0x0D	13	00001101	CR	Carriage Return
0x0E	14	00001110	SO	Shift Out / X-On
0x0F	15	00001111	SI	Shift In / X-Off
0x10	16	00010000	DLE	Data Line Escape
0x11	17	00010001	DC1	Device Control 1 (oft. XON)
0x12	18	00010010	DC2	Device Control 2
0x13	19	00010011	DC3	Device Control 3 (oft. XOFF)
0x14	20	00010100	DC4	Device Control 4
0x15	21	00010101	NAK	Negative Acknowledgement
0x16	22	00010110	SYN	Synchronous Idle
0x17	23	00010111	ETB	End of Transmit Block
0x18	24	00011000	CAN	Cancel
0x19	25	00011001	EM	End of Medium
0x1A	26	00011010	SUB	Substitute
0x1B	27	00011011	ESC	Escape
0x1C	28	00011100	FS	File Separator
0x1D	29	00011101	GS	Group Separator
0x1E	30	00011110	RS	Record Separator
0x1F	31	00011111	US	Unit Separator
0x20	32	00100000		Space " "
0x21	33	00100001	!	Exclamation mark
0x22	34	00100010	"	Double quotes
0x23	35	00100011	#	Number
0x24	36	00100100	$	Dollar
0x25	37	00100101	%	Percentage
0x26	38	00100110	&	Ampersand
0x27	39	00100111	'	Single quote
0x28	40	00101000	(	Open parenthesis (or open bracket)
0x29	41	00101001	)	Close parenthesis (or close bracket)
0x2A	42	00101010	*	Asterisk
0x2B	43	00101011	+	Plus
0x2C	44	00101100	,	Comma
0x2D	45	00101101	-	Minus / Hyphen
0x2E	46	00101110	.	Period, dot or full stop
0x2F	47	00101111	/	Slash or divide
0x30	48	00110000	0	Zero
0x31	49	00110001	1	One
0x32	50	00110010	2	Two

HEX, ASCII, BIN and Character table

HEX	DEC	BIN	Character/Symbol	Description
0x33	51	00110011	3	Three
0x34	52	00110100	4	Four
0x35	53	00110101	5	Five
0x36	54	00110110	6	Six
0x37	55	00110111	7	Seven
0x38	56	00111000	8	Eight
0x39	57	00111001	9	Nine
0x3A	58	00111010	:	Colon
0x3B	59	00111011	;	Semicolon
0x3C	60	00111100	<	Less than (or open angled bracket)
0x3D	61	00111101	=	Equals
0x3E	62	00111110	>	Greater than (or close angled bracket)
0x3F	63	00111111	?	Question mark
0x40	64	01000000	@	At symbol
0x41	65	01000001	A	Uppercase A
0x42	66	01000010	B	Uppercase B
0x43	67	01000011	C	Uppercase C
0x44	68	01000100	D	Uppercase D
0x45	69	01000101	E	Uppercase E
0x46	70	01000110	F	Uppercase F
0x47	71	01000111	G	Uppercase G
0x48	72	01001000	H	Uppercase H
0x49	73	01001001	I	Uppercase I
0x4A	74	01001010	J	Uppercase J
0x4B	75	01001011	K	Uppercase K
0x4C	76	01001100	L	Uppercase L
0x4D	77	01001101	M	Uppercase M
0x4E	78	01001110	N	Uppercase N
0x4F	79	01001111	O	Uppercase O
0x50	80	01010000	P	Uppercase P
0x51	81	01010001	Q	Uppercase Q
0x52	82	01010010	R	Uppercase R
0x53	83	01010011	S	Uppercase S
0x54	84	01010100	T	Uppercase T
0x55	85	01010101	U	Uppercase U
0x56	86	01010110	V	Uppercase V
0x57	87	01010111	W	Uppercase W
0x58	88	01011000	X	Uppercase X
0x59	89	01011001	Y	Uppercase Y
0x5A	90	01011010	Z	Uppercase Z
0x5B	91	01011011	[	Opening bracket
0x5C	92	01011100	\	Backslash
0x5D	93	01011101	]	Closing bracket
0x5E	94	01011110	^	Caret - circumflex
0x5F	95	01011111	_	Underscore
0x60	96	01100000	`	Grave accent
0x61	97	01100001	a	Lowercase a
0x62	98	01100010	b	Lowercase b
0x63	99	01100011	c	Lowercase c
0x64	100	01100100	d	Lowercase d
0x65	101	01100101	e	Lowercase e

HEX, ASCII, BIN and Character table

HEX	DEC	BIN	Character/Symbol	Description
0x66	102	01100110	f	Lowercase f
0x67	103	01100111	g	Lowercase g
0x68	104	01101000	h	Lowercase h
0x69	105	01101001	i	Lowercase i
0x6A	106	01101010	j	Lowercase j
0x6B	107	01101011	k	Lowercase k
0x6C	108	01101100	l	Lowercase l
0x6D	109	01101101	m	Lowercase m
0x6E	110	01101110	n	Lowercase n
0x6F	111	01101111	o	Lowercase o
0x70	112	01110000	p	Lowercase p
0x71	113	01110001	q	Lowercase q
0x72	114	01110010	r	Lowercase r
0x73	115	01110011	s	Lowercase s
0x74	116	01110100	t	Lowercase t
0x75	117	01110101	u	Lowercase u
0x76	118	01110110	v	Lowercase v
0x77	119	01110111	w	Lowercase w
0x78	120	01111000	x	Lowercase x
0x79	121	01111001	y	Lowercase y
0x7A	122	01111010	z	Lowercase z
0x7B	123	01111011	{	Opening brace
0x7C	124	01111100	|	Vertical bar
0x7D	125	01111101	}	Closing brace
0x7E	126	01111110	~	Equivalency sign - tilde
0x7F	127	01111111		Delete (no visible character)
0x80	128	10000000	€	Euro sign
0x81	129	10000001		(no visible character)
0x82	130	10000010	‚	Single low-9 quotation mark
0x83	131	10000011	*ƒ*	Latin small letter f with hook
0x84	132	10000100	„	Double low-9 quotation mark
0x85	133	10000101	…	Horizontal ellipsis
0x86	134	10000110	†	Dagger
0x87	135	10000111	‡	Double dagger
0x88	136	10001000	ˆ	Modifier letter circumflex accent
0x89	137	10001001	‰	Per mille sign
0x8A	138	10001010	Š	Latin capital letter S with caron
0x8B	139	10001011	‹	Single left-pointing angle quotation
0x8C	140	10001100	Œ	Latin capital ligature OE
0x8D	141	10001101		(no visible character)
0x8E	142	10001110	Ž	Latin captial letter Z with caron
0x8F	143	10001111		(no visible character)
0x90	144	10010000		(no visible character)
0x91	145	10010001	‘	Left single quotation mark
0x92	146	10010010	’	Right single quotation mark
0x93	147	10010011	“	Left double quotation mark
0x94	148	10010100	”	Right double quotation mark
0x95	149	10010101	•	Bullet
0x96	150	10010110	–	En dash
0x97	151	10010111	—	Em dash
0x98	152	10011000	˜	Small tilde

HEX, ASCII, BIN and Character table

HEX	DEC	BIN	Character/Symbol	Description
0x99	153	10011001	™	Trade mark sign
0x9A	154	10011010	š	Latin small letter S with caron
0x9B	155	10011011	›	Single right-pointing angle quotation mark
0x9C	156	10011100	œ	Latin small ligature oe
0x9D	157	10011101		(no visible character)
0x9E	158	10011110	ž	Latin small letter z with caron
0x9F	159	10011111	Ÿ	Latin capital letter Y with diaeresis
0xA0	160	10100000		Non-breaking space (no visible character)
0xA1	161	10100001	¡	Inverted exclamation mark
0xA2	162	10100010	¢	Cent sign
0xA3	163	10100011	£	Pound sign
0xA4	164	10100100	¤	Currency sign
0xA5	165	10100101	¥	Yen sign
0xA6	166	10100110	¦	Pipe, Broken vertical bar
0xA7	167	10100111	§	Section sign
0xA8	168	10101000	¨	Spacing diaeresis - umlaut
0xA9	169	10101001	©	Copyright sign
0xAA	170	10101010	ª	Feminine ordinal indicator
0xAB	171	10101011	«	Left double angle quotes
0xAC	172	10101100	¬	Not sign
0xAD	173	10101101		Soft hyphen
0xAE	174	10101110	®	Registered trade mark sign
0xAF	175	10101111	¯	Spacing macron - overline
0xB0	176	10110000	°	Degree sign
0xB1	177	10110001	±	Plus-or-minus sign
0xB2	178	10110010	²	Superscript two - squared
0xB3	179	10110011	³	Superscript three - cubed
0xB4	180	10110100	´	Acute accent - spacing acute
0xB5	181	10110101	µ	Micro sign
0xB6	182	10110110	¶	Pilcrow sign - paragraph sign
0xB7	183	10110111	·	Middle dot - Georgian comma
0xB8	184	10111000	¸	Spacing cedilla
0xB9	185	10111001	¹	Superscript one
0xBA	186	10111010	º	Masculine ordinal indicator
0xBB	187	10111011	»	Right double angle quotes
0xBC	188	10111100	¼	Fraction one quarter
0xBD	189	10111101	½	Fraction one half
0xBE	190	10111110	¾	Fraction three quarters
0xBF	191	10111111	¿	Inverted question mark
0xC0	192	11000000	À	Latin capital letter A with grave
0xC1	193	11000001	Á	Latin capital letter A with acute
0xC2	194	11000010	Â	Latin capital letter A with circumflex
0xC3	195	11000011	Ã	Latin capital letter A with tilde
0xC4	196	11000100	Ä	Latin capital letter A with diaeresis
0xC5	197	11000101	Å	Latin capital letter A with ring above
0xC6	198	11000110	Æ	Latin capital letter AE
0xC7	199	11000111	Ç	Latin capital letter C with cedilla
0xC8	200	11001000	È	Latin capital letter E with grave
0xC9	201	11001001	É	Latin capital letter E with acute
0xCA	202	11001010	Ê	Latin capital letter E with circumflex
0xCB	203	11001011	Ë	Latin capital letter E with diaeresis

HEX, ASCII, BIN and Character table

HEX	DEC	BIN	Character/Symbol	Description
0xCC	204	11001100	Ì	Latin capital letter I with grave
0xCD	205	11001101	Í	Latin capital letter I with acute
0xCE	206	11001110	Î	Latin capital letter I with circumflex
0xCF	207	11001111	Ï	Latin capital letter I with diaeresis
0xD0	208	11010000	Ð	Latin capital letter ETH
0xD1	209	11010001	Ñ	Latin capital letter N with tilde
0xD2	210	11010010	Ò	Latin capital letter O with grave
0xD3	211	11010011	Ó	Latin capital letter O with acute
0xD4	212	11010100	Ô	Latin capital letter O with circumflex
0xD5	213	11010101	Õ	Latin capital letter O with tilde
0xD6	214	11010110	Ö	Latin capital letter O with diaeresis
0xD7	215	11010111	×	Multiplication sign
0xD8	216	11011000	Ø	Latin capital letter O with slash
0xD9	217	11011001	Ù	Latin capital letter U with grave
0xDA	218	11011010	Ú	Latin capital letter U with acute
0xDB	219	11011011	Û	Latin capital letter U with circumflex
0xDC	220	11011100	Ü	Latin capital letter U with diaeresis
0xDD	221	11011101	Ý	Latin capital letter Y with acute
0xDE	222	11011110	Þ	Latin capital letter THORN
0xDF	223	11011111	ß	Latin small letter sharp s - ess-zed
0xE0	224	11100000	à	Latin small letter a with grave
0xE1	225	11100001	á	Latin small letter a with acute
0xE2	226	11100010	â	Latin small letter a with circumflex
0xE3	227	11100011	ã	Latin small letter a with tilde
0xE4	228	11100100	ä	Latin small letter a with diaeresis
0xE5	229	11100101	å	Latin small letter a with ring above
0xE6	230	11100110	æ	Latin small letter ae
0xE7	231	11100111	ç	Latin small letter c with cedilla
0xE8	232	11101000	è	Latin small letter e with grave
0xE9	233	11101001	é	Latin small letter e with acute
0xEA	234	11101010	ê	Latin small letter e with circumflex
0xEB	235	11101011	ë	Latin small letter e with diaeresis
0xEC	236	11101100	ì	Latin small letter i with grave
0xED	237	11101101	í	Latin small letter i with acute
0xEE	238	11101110	î	Latin small letter i with circumflex
0xEF	239	11101111	ï	Latin small letter i with diaeresis
0xF0	240	11110000	ð	Latin small letter eth
0xF1	241	11110001	ñ	Latin small letter n with tilde
0xF2	242	11110010	ò	Latin small letter o with grave
0xF3	243	11110011	ó	Latin small letter o with acute
0xF4	244	11110100	ô	Latin small letter o with circumflex
0xF5	245	11110101	õ	Latin small letter o with tilde
0xF6	246	11110110	ö	Latin small letter o with diaeresis
0xF7	247	11110111	÷	Division sign
0xF8	248	11111000	ø	Latin small letter o with slash
0xF9	249	11111001	ù	Latin small letter u with grave
0xFA	250	11111010	ú	Latin small letter u with acute
0xFB	251	11111011	û	Latin small letter u with circumflex
0xFC	252	11111100	ü	Latin small letter u with diaeresis
0xFD	253	11111101	ý	Latin small letter y with acute
0xFE	254	11111110	þ	Latin small letter thorn
0xFF	255	11111111	ÿ	Latin small letter y with diaeresis

Notes

Glass Display Control™ (GDC) LED & Button operations:

Prior to the procedure below, it is required you understand the terms and have the nesseccary knowledge how to interpret the functions in order to successfully use them as described below.

To be able to change GDC buttons and LED functions for the Panel Computers (Maritime Multi Computer - MMC) Series X product range, the VCOM (Virtual COM port visible in the Operating System (OS) device list as "COM **x**:", where **x** is OS/configuration dependent number; such as 1, 3 or other) needs to be installed and accessible from within your system.

The following commands referenced below are only available from GDC Firmware* version **"130225R1"** and up. To determine your unit's GDC Firmware version, send the command **"SWK"** via VCOM.

Excerpt from "Message Commands and Queries (CMD) Contents" section in this manual:

0x53	0x57	0x50	"SWK"	Glass Display Control™ (GDC) software version Query	R

*Reference Engineering Change Notification (ECN):
https://www.hattelandtechnology.com/product-notifications/series-x-firmware-updates

Note: It is important to read the GDC LED register (step 1) before you change the bit for your function (Step 3)

Step 1 - Read register GDC LEDs:
- Send Query command: "RBY"
- Data (HEX): 0x93, 0x00

Step 2 - Read register GDC buttons:
- Send Query Command: "RBY"
- Data (HEX): 0x93, 0x07

Step 3 Address only the bit for the function you need to change as illustrated in tables below.

Step 3A - Disable / Enable GDC LEDs:
- Command: "WBY"
- Data (HEX): 0x92, 0x00, 0x**YY**

Where "**YY**" is one of the folllowing bits (in a byte):

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Reserved	ON / OFF Text	Brilliance Text	Reserved	Reserved	Brilliance < * > LED symbols	Power ON/OFF LED Symbol	All off
0	0 = Off 1 = On	0 = Off 1 = On	0	1 = Normal (reserved) 0 = Reserved	0 = Off 1 = On	0 = Off 1 = On	1 = active 0 = Reserved

Step 3B - Disable / Enable GDC Buttons:
- Command: "WBY"
- Data (HEX): 0x92,0x07,0x**ZZ**

Where "**ZZ**" is one of the folllowing bits (in a byte):

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
Reserved	Reserved	Reserved	Reserved	Power ON / OFF Button	Outdoor Mode*	Reserved	Brilliance Button
0	0	0	0	0 = Enable 1 = Disable	0 = Enable 1 = Disable	0	0 = Enable 1 = Disable

*Outdoor mode (Command "MCC - OSD Outdoor Mode") adds 5 seconds to the brilliance buttons before they react.

Revision History

Rev.	By	Date	Notes
00_01	AK SE	14 Sep 2012	Internal draft
00_02	AK BU SE	12 Oct 2012	Revised and improved after input. Added Ethernet specification.
01	BU AK SE	06 Nov 2012	Release for internet
02	BU SE	07 Jan 2013	Added new command “GMB”, page 14 Added new command “OSD Lock Mode (full)” to MCC command, page 21 Added new command “OSD Key Outdoor” to MCC command, page 25 - Reference to ECN: http://www.hatteland-display.com/mails/01_2013_ecn.html Added note “not all command available...”, page 8,19
03	BU SE	12 Jun 2013	Revised wrong example in MCC Contrast Control, ref: QAR/118774, page 19 Revised “periodic messages” and added note to table “xx”, page 28 Added note/procedure for GDC LED/Buttons, page 38 - Reference to ECN: http://www.hatteland-display.com/mails/05_2013_ecn.html Revised BRL,GMB max range values, page 10,14 Revised Glass Display Control™ (GDC) LED & Button operations, bit table, page 38
04	BU SE	23 Sep 2013	Added command “MAN”, page 8,15
05	BU AK SE	19 Feb 2014	Added “API/VCOM Drivers” installation / version information, page 6,7 Revised DL?, DLN commands to differentiate between Calibrated DVI/VGA slots, page 14,15 Revised FWV, table overview for MMD/STD/MMC units, page 16 Revised SWI, table overview for MMD/STD/MMC units, page 19 - Reference QAR/120762
06	BU SE	06 Mar 2014	Revised DL? text description with reference to using MCC command, page 14-15
07	ME WJ SE	30 Oct 2014	Removed “Note: DC power only” notice for “Power Down / Up” command. Works with AC & DC, page 26
08	MS SE	11 Dec 2014	Added note for XON/XOFF flow control, page 9 Improved description of DLN command, page 16
09	AK SE	03 Jun 2015	Added “Sending Multiple Commands / Command Queue” delay information, ref: QAR/126502, page 33
10	ME SE	04 Sep 2015	Removed “?” from BRU commmand, not supported (ref: QAR/130439), page 13
11	VM WJ SE	28 Jan 2016	BRU command also available for certain Customized Industrial Standard Displays (STD) units, page 10
12	ME SE	13 May 2016	BRU example malformed, corrected now (from 09 to 0F), page 13
13	JE MJ SE	21 Dec 2016	Added details for Buzzer pins, page 8
14	VM WJ SE	18 Jan 2017	Removed “Set Runtime Counter” in the MCC list (command reserved for service partners only), page 27 Reference: QAR/135613
15	VM SE	28 Mar 2019	Revised text for GDC button definitions, page 31 Removed “max 16” limit for Ethernet, page 5
16	VM JE WJ SE	04 Jun 2019	General updated throughout the manual based on latest company profile Added link to Support Document (Linux OS drivers etc. DOC101163-1), page 6 Removed limitation of 16 units, page 6, 9 ref: https://www.hatteland-display.com/mails/12_2019_ecn.html
17	WJ WM SE	27 Aug 2019	Revised max 16 units to 255, page 9 Added new SCOM command “RS Address (extended), page 31
18	SE	05 May 2020	General maintenance performed throughout the entire manual
19	YR WJ VM SE	20 Aug 2021	Performed maintentance throughout entire manual regarding text, commands, company profile/layout etc.

Revision History

20	YR VM SE	02 Nov 2021	This INB100018-10 manual is based on rev 19 of INB100018-4, but extracted to new version to only cover MMC models due to EOL, ref: https://www.hattelandtechnology.com/product-notifications/panel-computers-series-x-hd-xxt21-mmc-with-qm57-chipset-eol-12_2021_eol Revised “Calculating Checksums (IDCHK, IHCHK)” chapter, page 20-22
21	VM SE	10 May 2022	Replaced drivers links and removed reference to obsolete “menu_run.exe” software, page 5

Revision History

HATTELAND TECHNOLOGY

an EMBRON Company