SynQor MCOTS-C-270-xx-FE Terminal Commands

The SynQor MCOTS-C-270-xx-FE contains a serial port to facilitate system monitoring.
Commands are sent to request individual data, the FE responds to each command individually.
The FE's external serial port IO pins are at 3.3V logic levels, not RS232 levels.
    See the device datasheet for IO pin level & polarity details.
    Commercial data converters are available to translate these levels to RS232 or USB.
The serial port uses 9600 baud, 8 data bits, no parity, and one stop bit. Commands can be sent, and output viewed, with any standard terminal emulator.   The interface is case sensitive. To view user text, set the terminal emulator for local echo.
Some data are writeable; written values are, however, volatile and will revert to default upon power cycling.
A pause of >5 seconds between any two comand characters will timeout/reset the command parser. No response is generated.
    This timeout value is user settable - see Hung Parser Timeout below.
The protocol supports communication to/from individual modules on a shared serial bus via a unique 'network address'.
In this document, the symbol ↙ will be used to represent ascii 13 (cr) OR ascii 10 (lf) OR both (they're all equivalent).

Value Formats

Data supplied to, and returned from, the FE can be communicated in several formats, broadly divided into strings and numeric values. While strings are a unique type, numeric data can be further divided into serveral interchangeable formats.

	Fmt	Numeric Range	Description	
Strings		Up to 64 ascii characters, and "enclosed in double quotes". A null string is "". Strings can contain unprintable and control characters which can, of course, affect the appearance of any terminal emulator receiving them.		
Numerical Formats	char	0 to 127	An ascii character, preceeded by single quote character ' to remove ambiguity. Responses using char format will always contain the ' prefix.	
	hex	0x0 to 0xFFFF	0x and then zero to four ascii hex characters	Hex, U int, and int formats have no internal units conversion
	U int	0 to 65535	Unsigned decimal numeral, optionally preceeded by 'u'	
	int	-32768 to 32767	Signed decimal numeral, specified by '+' or '-' prefix	
	mks	±###.#	Decimal numerals *containing a decimal point* are interpreted as real MKS values and are internally converted to/from machine units. MKS values are naturally signed. MKS format is recommended whenever referring to a real-life datum.	

Command Formats

Both data read and data write commands are supported

Data Write Command Syntax

{Net Addr}	{Access Address}	{Action Specifier}	{Write Value}	{Terminator}
m	0-127	=	"String" or 16-bit numerical	(cr) or (lf) or both, ↙

Net Addr: The network address character, which takes a default value of 'm on power-up. 'A thru 'z is a single unit's address, while '@ is interpreted as all units. To avoid collision, the response to '@ commands is automatically muted.
    See the Enumerate command to automatically assign unique network addresses for multi-unit applications.

Access Addr: The address to write, see table below.
    Addresses can be specified in char, hex, or int formats, see Value Formats below.

Action Specifier: The = character specifies a data write command.
    Only select addresses are writeable.

Write Value: Values to write can be supplied in the formats described below;
    The supplied format will determine the format of the response datum.

Terminator: a carriage return character (cr) = ascii 13, OR a linefeed character (lf) = ascii 10, OR both ↙.

Data Read Command Syntax

{Net Addr}	{Access Address}	{Action Specifier}	[Format]	{Terminator}
m	0-127	?	optional value of 0	(cr) or (lf) or both, ↙

Net Addr: The network address character, which takes a default value of 'm on power-up.
while valid, the '@ address mutes responses to avoid collision; it's not useful for read operations.
See the Enumerate command to automatically assign unique network addresses for multi-unit applications.

Access Addr: The address to read, see table below.
Addresses can be specified in char, hex, or int formats, see Value Formats below.

Action Specifier: The ? character specifies a data read command.

Format: The numerical response datum format of a read command can be controlled by supplying the command
with an argument value of 0. The format of the 0 specifies the desired return datum format (see table below).
In the absence of a 0 argument, the datum is returned in its natural or mks format.

Terminator: a carriage return character (cr) = ascii 13, OR a linefeed character (lf) = ascii 10, OR both ↙.

Examples of controlling the return datum format of read commands

m77?↙	No value supplied, return datum in default (mks or natural) format
m77?0.↙ m77?.↙	Value of 0 supplied in mks, return in mks/natural format
m77?0↙ m77?u0↙ m77?u↙	Unsigned value of 0 supplied, return as unsigned int.
m77?+0↙ m77?+↙	Signed value of 0 supplied, return as signed int.
m77?0x↙ m77?x↙	Value of 0 supplied in hex, return in hex.

Response Format

Immediately after reception of its terminator, each valid command generates a response.
Invalid commands do not generate a response.

Response Syntax

{Net addr}	{Access Address}	{Is Now}	{Value}	{Terminator}
m	0-127	=	"String" or 16-bit numerical	↙

Net Addr: The FEs echoes its network address, 'm in the example above.

Access Address: Echo of the address specifier in the same format as supplied.
Char addresses will be prefixed by '.

Is Now: The character '='

Value: The address's current value.
See above for descriptions of the return datum format to both read and write commands.
Char format will always contain leading ', hex format will always contain 0x and four ascii hex value characters.

Terminator: ↙ concludes the response

Addresses

The following table contains a summary of the addressesthat contain useful data.

Access Address			Name	Read / Write	Natural Format	Natural Units
hex,	dec,	or char				
0x0	0		ID String	R	string	
0xC	12		Build Revision	R	hex	
0xD	13		Code Revision	R	string	
0x23	35	#	Powered Days	R	int	days
0x24	36	$	Powered Hours	R	int	hours
0x4D	77	M	Output Voltage	R	##.###	Volts
0x4E	78	N	Output Current	R	#.###	Amperes
0x53	83	S	Input Voltage	R	###.##	Volts
0x54	84	T	Buck Temperature	R	###.##	°C
0x55	85	U	BQ Temperature	R	###.##	°C
0x56	86	V	Midbus Voltage	R	##.###	Volts
0x57	87	W	Boost Temperature	R	###.##	°C
0x63	99	c	I Share Voltage	R	#.###	Volts
0x6E	110	n	Battle Short Mode Status	R	int	
0x6F	111	o	Battle Short Enable Mode	R/W	int	
0x77	119	w	Battle Short Warnings Status	R	int	

Address Details

Address:	hex	decimal	char
	0x0	0	
Name:	**ID String**		
Description:	The unit's ID string "Synqor MCOTS-C-270-xx-FE"		
Response units:	string		
Read/Write:	Read Only		

Example Cmd:	**Typical Response:**
m0x0?✓	m0x0000= "Synqor MCOTS-C-270-60-FE"✓
m0?✓	m0= "Synqor MCOTS-C-270-60-FE"✓

Address:	hex	decimal	char
	0xC	12	
Name:	**Build Revision**		
Description:	The unit's Build Revision		
Natural Response units:	hex		
Read/Write:	Read Only		

Example Cmd:	**Typical Response:**
m0xC?✓	m0x000C= A✓
m12?✓	m12= A✓

Address:	hex	decimal	char
	0xD	13	
Name:	**Code Revision**		as a number
Description:	The unit's Code Revision		
Response units:	string		
Read/Write:	Read Only		

Example Cmd:	**Typical Response:**
m0xD?✓	m0x000D= "8"✓
m13?✓	m13= "8"✓

SynQor MCOTS-C-270-xx-FE Terminal Commands

	hex	decimal	char	Response units	
Powered Days	0x23	35	#	int 0-65535	Read
Powered Hours	0x24	36	$	int 0-23	Read

Discussion: Together these contain the number of days, hours, minutes and seconds that power has been applied. They can be written to any value (or cleared) at any time, and will count time thereafter.

Example Cmds:	**Typical Responses:**	**Interpretation:**
m35?✓	m35= 329✓	Unit has been powered for 329 days
m36?✓	m36= 3✓	and 3 hours.
m35=0✓	m35= 0✓	Days variable reset to 0.
m36=0✓	m36= 0✓	hours variable reset to 0.

Address:	hex	decimal	char
	0x4D	77	M

Name: **Output Voltage**
Description: The unit's output voltage
Response units: decimal Volts DC
Read/Write: Read Only
Discussion:

Example Cmd:	**Typical Response:**	**Interpretation:**
mM?✓	m'M= 28.000✓	The unit is producing 28.000 VDC.

Address:	hex	decimal	char
	0x4E	78	N

Name: **Output Current**
Description: The FEs output current
Natural Response units: decimal Amperes DC
Read/Write: Read Only
Discussion:

Example Cmd:	**Typical Response:**	**Interpretation:**
mN?✓	m'N= 30.000✓	The output is delivering 30.000 Adc (out)

Address:	hex	decimal	char
	0x53	83	S
Name:	**Input Voltage**		
Description:	The DC Input Voltage		
Natural Response units:	decimal Volts		
Read/Write:	Read Only		
Discussion:	The value is the DC input voltage.		

Example Cmd:	Typical Response:	Interpretation:
mS?✓	m'S= 270.00✓	The DC input voltage is 270.00 Vdc

Address:	hex	decimal	char
	0x54	84	T
Name:	**Buck Temperature**		
Description:	The unit's internal temperature at Buck Fets.		
Natural Response units:	signed decimal °C		
Read/Write:	Read Only		
Discussion:			

Example Cmd:	Typical Response:	Interpretation:
mT?✓	m'T= 63.00✓	The Buck Fets are at about 63.00 °C.

Address:	hex	decimal	char
	0x55	85	U
Name:	**BQ Temperature**		
Description:	The BusQor internal temperature.		
Natural Response units:	signed decimal °C		
Read/Write:	Read Only		
Discussion:			

Example Cmd:	Typical Response:	Interpretation:
mU?✓	m'U= 52.50✓	The BusQor Fets are at about 52.50 °C.

Address:	hex	decimal	char
	0x56	86	V
Name:	**Midbus Voltage**		
Description:	The BusQor output voltage		
Response units:	decimal Volts DC		
Read/Write:	Read Only		
Discussion:			

Example Cmd:	Typical Response:	Interpretation:
mV?✓	m'V= 36.000✓	The boost stage is producing 36.000 VDC.

	hex	decimal	char
Address:	0x57	87	W
Name:	**Boost Temperature**		
Description:	The unit's internal temperature at Boost Fets.		
Natural Response units:	signed decimal °C		
Read/Write:	Read Only		
Discussion:			

Example Cmd:	Typical Response:	Interpretation:
mW?↙	m'W= 65.00↙	The Boost Fets are at about 65.00 °C.

	hex	decimal	char
Address:	0x63	99	c
Name:	**I Share Voltage**		
Description:	This is the scaled representation of output current. 0.200V = 0A and 2.200V = FL		
Natural Response units:	decimal Amperes DC		
Read/Write:	Read Only		
Discussion:			

Example Cmd:	Typical Response:	Interpretation:
mc?↙	m'c= 1.200↙	The I Share voltage is reading 1.200 Vdc

	hex	decimal	char
Address:	0x6E	110	n
Name:	**Battle Short Mode Status**		
Description:	Status bits associated with Battle Short Mode.		
Natural Response units:	bit 0	Battle Short not enabled.	
	bit 1	Battle Short enabled.	
Read/Write:	Read Only		
Discussion:			

Example Cmd:	Typical Response:	Interpretation:
mn?↙	m'n= 1↙	Battle Short Mode is enabled.

SynQor MCOTS-C-270-xx-FE Terminal Commands

	hex	decimal	char
Address:	0x6F	111	o
Name:	**Battle Short Enable Mode**		
Description:	Allows engaging Battle Short mode via serial interface		
Natural Response units:	hex		
Read/Write:	Read/Write		
Discussion:	value of 0 (0x0) allows Battle Short mode to follow BS Pin [default]		
	writing value of 1 (0x1) turns on Battle short mode regardless of BS Pin value		

Example Cmd:	Typical Response:	Interpretation:
mo?↙	m'o= 0x0000↙	unit m is obeying BS pin
@o=1↙	none (response muted for global addr @)	Everyone: Battle Short Mode = On
mo=0↙	m'o= 0↙	unit m, obey BS pin again

---

	hex	decimal	char
Address:	0x77	119	w
Name:	**Battle Short Warnings Status**		
Description:	Status bits associated with the BATTLE SHORT pin (warnings)		
Responses:	bit 0	NA	
	bit 1	Warn Over Temperature	
Natural Response units:	hex		
Read/Write:	Read Only		
Discussion:	These flags indicate the reason the BATTLE SHORT pin internal pull-down transistor has been turned off		

Example Cmd:	Typical Response:	Interpretation:
mw?↙	m'w= 0x0001↙	Battle Short Warning is due to temperature