eMotimo INC 8840 Kenamar Dr. #403 | San Diego, CA 92121 | 415-684-8162 | eMotimo.com

ST4 API Reference
API - V0.108

Contents

Contents

EMOTILINK COMMAND PROTOCOL: Reference Guide

This document is an early release. It is the set of commands to access and control the spectrum ST4. It is intended for developers, and not for regular users. Functionality will be added as requested. Bugs can be reported to help@emotimo.com with the subject line "ST4 API Issue"

ACCESSING ECP IN SPECTRUM ST4

VIA SERIAL PORT

Please only use our cables **- https://emotimo.com/products/usb-to-i-o-port-cable-for-spectrum-st4. Each is tested before it heads out. In the past we have tried to support users with their own custom cable builds and modification, but it has proven to be costly and frustrating for both the customizer and us. If you are comfortable making or sourcing your own cable, please do, but unfortunately, no support from eMotimo can be made available if there are issues.**

Setting up a connection to the spectrum using the following serial protocol:

Baudrate - :57600 bits per second when **eMotimo API** is selected from i/o port
Baudrate - :230400 bits per second when **eMotimoFastAPI** is selected from i/o port
Data Bits: 8
Parity: None
Stop Bits: 1
Flow Control: None
Line Ending Character: /n or <LF> complete each command with this character

TRS (tip, ring, sleeve) connection through spectrum i/o Port.
Tip: spectrum ST4 TX and External Micro RX **
Ring: spectrum ST4 and External Micro TX
Sleeve: spectrum ST4 Ground External Micro TX Ground
Logic Voltage: 0V to 3.3V to 5.0V for High (internal level shifting) **TLL Logic Levels.**
NEVER USE RS232 levels, CAN, or RS485 levels. These will damage the i/o port

DEFINED PARAMETER GROUP

MOTOR GROUP G,WXYZ,T,A

G0 – Go Rapid

goes to a particular position defined by absolute coordinates of all axis. Each motor move independently to position using the currently set max velocities and acceleration for each axis. Use this when coordinated moves are not needed.

Example: Go to absolute pan position 10000, tilt position 20000, M3 position -15000, M4 position 2000

```
G0 X10000 Y20000 Z-15000 W2000
Rapid to:,X10000,Y20000,Z-15000,W2000
```

Notes –

- Virtual Stops are not adhered to when using G0 and G1.
- If no value is given for an axis, no move command is given to that axis.

G1 – Go Coordinated

goes to a particular position defined by absolute coordinates of all axis. Each motor move independently to position using the currently set max velocities and acceleration. Use this when coordinated moves are not needed

Example: Go to absolute pan position 10000, tilt position 20000, M3 position -15000, M4 position 2000 in 1.5 seconds with an acceleration of 0.25seconds on each side.

```
G1 X10000 Y20000 Z-15000 W2000 T1.5 A0.25
Move to:X10000,Y20000,Z-15000,W2000
```

Notes –

- Virtual Stops are not adhered to when using G0 and G1.
- If the move cannot be achieved in the time required, it will move at the fastest speed possible with the current VMAX and AMAX settings.
- If no value is given for an axis, no move command is given to that axis.

G2 – Jog Position – Stops enforced

Jogs motor a particular number of steps.

Example: Jog the following steps Pan 1000, tilt -2000, M3 -1500, M4 2000

```
G2 X1000 Y-2000 Z-1500 W2000
Jog by:X1000,Y-2000,Z-1500,W2000
```

Notes:

- If you are trying to jog over a stop, it will stop at the stop – expected. If you are already over a stop, it will jog back to the limit.

G100 – Sets Motor performance

This is an advanced command that must be used carefully as this controls power and speeds used by the ST4's motors. Do not sets maxes on any axis at a default. It is recommended to use the example parameters below and slowly make changes.

Parameters

1. M -**M**otor – M – 1-4, M1 Pan, M2, Tilt, M3, M4
2. D - mo**D**e 1 for normal, 2 for quiet
3. V - **V**MAX: 1 to 600000, positive integer values only
4. A - **A**MAX: 1 to 65535, positive integer values only
5. R - Power During **R**un: 1 to 15 positive integer values only
6. S - Power During **S**top: 0 to 5 positive integer values only

Example: Set the pan axis VMAX to 150000, AMAX 5000, Current During run of 8, Current while stationary of 3.

```
G100 M1 D1 V150000 A4000 R8 S3
Motor performance set for Pan
G100 M2 D1 V300000 A7000 R10 S2
Motor performance set for Tilt
G100 M3 D1  V400000 A6000 R10 S5
Motor performance set for M3
G100 M4 D1  V250000 A5000 R13 S1
Motor performance set for M4
```

Note, *It is not recommended or allowed to set the value of R higher than 15, or S higher than 5. Turning motor to their max power for run or hold uses a high amount of current and can trip overcurrent protection or produce excessive heat in the spectrum st4. Although no indication in the return value is shown, the high limits are enforced.*
***See understanding V and A values for a motor.*

G199 – Returns Motor Performance VALUES – TO BE IMPLEMENTED

G200 - Set Motor Position

Sets the internal motor position to a value. This is good for zeroing a particular axis;

Parameters

1. M - Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4
2. P (Optional) Position: any singed 32 bit integer number between - 2100000000 to 2100000000, If not entered, 0 will be defaulted.

Examples:

```
G200 M1 P10000
Pan position set to:10000
G200 M1 P0
Pan position set to:0
G200 M2 P300
Tilt position set to:300
G200 M2
Tilt position set to:0
```

G201 - Zero All Motors

Sets all motors to zero

Parameters: None

Example:

```
G201
All Motors Zeroed
```

G211 – Set Motor Virtual STOPA to a value or clear

Sets internal Stop A, one side of virtual Stop – use this for the lower value.

Parameters

1. Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4
2. Position: (optional) any integer number -2100000000 to 2100000000, if left clear, this will set the STOPB to the STOPA to the **minimum** value, clearing the stop in practice

Example: Set Tilt Virtual Stop A to -588800, then clear it.

```
G211 M2 P-588800
Tilt Virtual StopA set to:-588800
G211 M2
Tilt Virtual StopA set to:-2100000000
```

Note:

- **Virtual StopA must be lower in value than Virtual Stop B**

G212 – Set Motor Virtual STOPB to a value or clear

Sets internal Stop B, one side of virtual Stop – use this for the higher value.

Parameters

1. Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4
2. Position: (optional) any integer number -2100000000 to 2100000000, if left clear, this will set the STOPB to the **maximum** value, clearing the stop in practice

Example: Set Tilt Virtual Stop B to 120000, then clear it.

```
G212 M2 P120000
Tilt Virtual StopB set to:120000
G212 M2
Tilt Virtual StopB set to:2100000000
```

Note:

- **Virtual StopA must be lower in value than Virtual Stop B**

G213 –Set Motor Virtual STOPA to the current position

This reads the current position and set STOPA to the current position for the requested motor.

Parameters

1. Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4

Example: Move to a known position (stops not adhered to and set virtual stop for M3 Virtual Stop A to that position.

```
G1 X10000 Y20000 Z-15000 W2000 T1.5 A0.25
Move to:X10000,Y20000,Z-15000,W2000
G213 M3
M3 Virtual StopA set to:-15000
```

G214 –Set Motor Virtual STOPB to the current position

This reads the current position and set STOPB to the current position for the requested motor.

Parameters

1. Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4

Example: Move to a known position (stops not adhered to and set virtual stop for M3 Virtual Stop B to that position.

```
G1 X10000 Y20000 Z45000 W2000 T1.5 A0.25
Move to:X10000,Y20000,Z45000,W2000
G214 M3
M3 Virtual StopB set to:45000
```

G215–Query Motor Virtual STOPA

This returns the current STOPA value for the motor. If the stop is not set, the value of the return is the full numerical limit.

Parameters

1. *Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4*

Example: Move to a known position (stops not adhered to) and set virtual stop for M3 Virtual Stop A to that position. Query Stop A, clear the STOPA and then query STOPA again.

```
G1 X10000 Y20000 Z-15000 W2000 T1.5 A0.25
Move to:X10000,Y20000,Z-15000,W2000
G213 M3
M3 Virtual StopA set to:-15000
G215 M3
M3 Virtual StopA set to:-15000
G211 M3
M3 Virtual StopA set to:-2100000000
G215 M3
M3 Virtual StopA set to:-2100000000
```

G216–Query Motor Virtual STOPB

This returns the current STOPA value for the motor. If the stop is not set, the value of the return is the full numerical limit.

Parameters

1. *Motor: 1 for Pan, 2 for Tilt, 3 for M3, 4 for M4*

Example: Move to a known position (stops not adhered to) and set virtual stop for M3 Virtual Stop B to that position. Query Stop B, clear the STOPB and then query STOPB again.

```
G1 X10000 Y20000 Z45000 W2000 T1.5 A0.25
Move to:X10000,Y20000,Z45000,W2000
```

```
G214 M3
M3 Virtual StopB set to:45000
G216 M3
M3 Virtual StopB set to:45000
G212 M3
M3 Virtual StopB set to:2100000000
G216 M3
M3 Virtual StopB set to:2100000000
```

G300 – Sets Motor Velocity

This is inherently a dangerous command as by setting a motor velocity, it will continue to run until it hits it virtual stops unless another command is given to stop it.   Use with care. Velocities are limited to 600000.

Parameters

1. M -**M**otor – M – 1-4, M1 Pan, M2, Tilt, M3, M4
2. V - **V**elocity: -600000 to 600000

Example:

```
G212 M3 P1
M3 Virtual StopB set to:1
G300 M3 V100000
Velocity Move: M3 100000
G211 M3 P-100000
M3 Virtual StopA set to:-100000
G300 M3 V-100000
Velocity Move: M3 -100000
```

Note;

If a value of velocity is passed in that is greater than 100000, it will be defaulted to zero

G500 – Status - What's moving and location

Returns has 5 parameters:

4digit set – Is moving flag for Pan,Tilt,M3,M4. 1 indicates moving, 0 indicates still

The next 4 parameters are the current location of each of the Pan, Tilt, M3, and M4 Values. In the example below, no motor is moving, and the pan value is 11297, tilt value is 132545, M3 4530 and M4 -26249.

```
G500
0000 11297,132545,4540,-26249
```

G911 – Stop All Motors

Initiates a hard stop of all motors. This is not depowering, but decelerating quickly. This should not be used as a general stop, but as an emergency stop.

Example:

```
G911
```

CAMERA CONTROL GROUP

Series of commands to interact with the camera triggering port. This could be fire a shot, focus, or fire a shot with a specific amount of time.

G400 – Trigger Shutter/Focus NOW

Trigger Focus and Shutter for a set period of MS

Parameters

1. S -Shutter: - time in ms of Shutter Trigger

Example:

```
G400 S2000
Shutter/Focus 2000ms
G400 S150
Shutter/Focus 150ms
```

G410/G411 –Focus Off/On

Turn on and off Focus

Parameters - none

Example:

```
G411
Focus On
G410
Focus Off
```

G420/G421 –Shutter Off/On

Turn on and off Shutter

Parameters - none

Example:

```
G421 S2000
Shutter On
G420 S150
Shutter Off
```

PROGRAMMED SHOTS GROUP

G600 – Export Move from RAM

Returns the move parameters and the line by line export of the move.

Example:

```
G600
<Placeholder, response very large>
```

SYSTEM GROUP - S

G700 – Returns current firmware Version

Example:

```
G700
Version: ST4_RC007_36
```

G710 – Turn on/off Pre/Post amble.

By default the spectrum doesn't add any tags to delimit the start and end of a response. Most terminal programs will parse this just fine, but if you need to systematically and buffer multiple commands and response, delimiters are helpful.

This command either sets an internal flag (volatile) in the spectrum and then returns the version. If the pre/post amble is set, then the spectrum prepends all returns with "<STX>" and appends all returns with "<ETX>".
Note - "<STX>" and "<ETX>" are control characters, not 5 byte strings.
"\x02", //hex 02, dec 2, <STX>
"\x03" //hex 03, dec 3, <ETX>

Parameters

1. S -**Set** – 1 adds pre/postamble to all returns of <STX> and <ETX>. All other values including omission, returns the spectrum to the default state of no prepends/appends.

Example:

```
G700
Version: ST4_RC007_36
G710
Version: ST4_RC007_36
G710 S1
<STX>Version: ST4_RC007_36<ETX>
G710 S0
Version: ST4_RC007_36
```

G720 – Turn on/off Data Streaming of telemetry feed

The spectrum can output real-time telemetry of its motor position over the serial port. The commands will set up streaming that will report only when there is a change at a target frequency of up to 200Hz.

Parameters

1. S -**Set** – 0 turns off the streaming. Any positive value helps to set a divisor to manage target frequency of updates. The base frequency is 200Hz for updates. For example 200Hz/100= 2 Hz. Or 200Hz/10=20Hz

```
G720 S0
Position Stream Off
G720 S100
Position Stream 2Hz
G720 S10
Position Stream 20Hz
```

The format of the streamed data is a 25byte string in Hex Format:

```
txString[0] = 'S';
txString[1] = 'T';
txString[2] = '_';
txString[3] = currentFrame & 0xFF;
txString[4] = currentFrame >> 8  & 0xFF;
txString[5] = currentFrame >> 16  & 0xFF;
txString[6] = currentFrame >> 24  & 0xFF;
txString[7] = mPan.XACTUAL & 0xFF;
txString[8] = mPan.XACTUAL >> 8  & 0xFF;
txString[9] = mPan.XACTUAL >> 16  & 0xFF;
txString[10] =mPan.XACTUAL >> 24  & 0xFF;
txString[11] =mTilt.XACTUAL & 0xFF;
txString[12] =mTilt.XACTUAL >> 8  & 0xFF;
txString[13] =mTilt.XACTUAL >> 16  & 0xFF;
txString[14] =mTilt.XACTUAL >> 24  & 0xFF;
txString[15] =M3.XACTUAL & 0xFF;
txString[16] =M3.XACTUAL >> 8  & 0xFF;
txString[17] =M3.XACTUAL >> 16  & 0xFF;
txString[18] =M3.XACTUAL >> 24  & 0xFF;
txString[19] =M4.XACTUAL & 0xFF;
txString[20] =M4.XACTUAL >> 8  & 0xFF;
txString[21] =M4.XACTUAL >> 16  & 0xFF;
txString[22] =M4.XACTUAL >> 24  & 0xFF;
txString[23] ='\r';
txString[24] ='\n';
```

There are

- 3 characters “ST_” to start, then:
- 4 Byte current frame if running a programmed move.
- 4 Byte Pan Position – signed 32 bit, LSByte first
- 4 Byte Tilt Position – signed 32 bit, LSByte first
- 4 Byte M3 Position – signed 32 bit, LSByte first
- 4 Byte M4 Position – signed 32 bit, LSByte first
- Then \r\n to close out a line.

Placeholder State – Read/Write Limited

Returns current numerical state of the firmware. A State Reference Chart will be added. This is useful for bring up a particular display screen or function. Care must be taken moving from state to state as not all states are valid so only a few states will be available to write to. May consider opening this up if there’s no risk.

Placeholder - KEYS GROUP – Read/Write

Emulates control inputs from the PS4 Remote Control and 9 Way switch.

Key Monitor

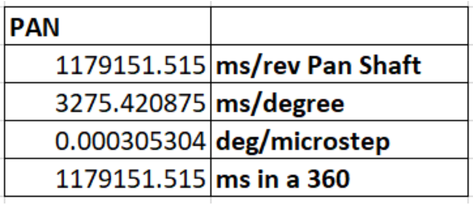
As keys are pressed and inputs given ouput the codes as they are read. This function will be disabled until UART DMA in place and throughput and blocking time is too high.

Understanding Velocity and Acceleration Values for each axis

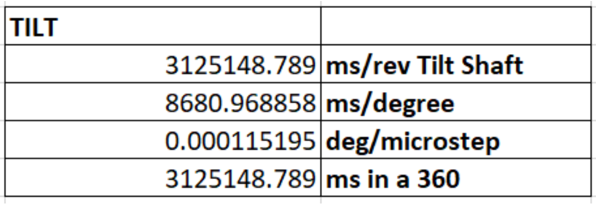
To help programmers plan their motion better, detailed information on gearing by axis and constants to relate motor move parameters to real world units is needed.

Understanding Velocity (V). This is a constant that relates to the maximum **microsteps per second** that our motor drivers will deliver. Based on the motor / gearing, it relates to angular and linear velocities with real-world units. Since the spectrum ST4 only has Pan and Tilt motors / gearing, real world units can only be related for these axis. The M3 and M4 axis are customizable so real world units for angular and linear velocity cannot be fully determined since only part of the equation is known.

For Positions – note – ms refers to microstep, not milliseconds – timing will always be related as seconds, or sec.



PAN	
1179151.515	**ms/rev Pan Shaft**
3275.420875	**ms/degree**
0.000305304	**deg/microstep**
1179151.515	**ms in a 360**



TILT	
3125148.789	**ms/rev Tilt Shaft**
8680.968858	**ms/degree**
0.000115195	**deg/microstep**
3125148.789	**ms in a 360**

For Velocities, the value entered into move commands and motor setup is not steps per second, rather, it is a factor of that based on other parameters that are hardware dependent. For the ST4, **the translation the constant is 0.953674316**

Example calculation:

You want the Pan to spin at 10 degrees per second. This is (10 deg/sec) * (3275.420875 ms/deg)= 32754.20875 ms/sec

To get to a programmed velocity you take 32754.20875 / **0.953674316=** 34345. 34345 is what you would put into G300 command to set the VMAX value for the Pan Axis.

Open Software Requests

Implement G199 – return motor performance.

Add WatchDog Timer command to reset after a period of time to enable RX buffer resets if command is not received in that time. This will alleviate issues with pulled cables that can leave our RX buffer in an unrecoverable state that can't process new commands.

Document Versioning and Changelog

ST4 API Reference API - V0.109 **March 3, 2020**–

ST4_RC007_80 and above

- Added information on Line Ending Character
- <Placeholder, need to add checksum and only process commands that begin with G>

ST4 API Reference API - V0.108 **Jan 29, 2020**–

ST4_RC007_70 and above

- Fixed typo

ST4 API Reference API - V0.107 **Jan 6, 2020**–

ST4_RC007_70 and above

- Added position streaming G720 function

ST4 API Reference API - V0.106 **Feb 5, 2019** –

ST4_RC007_42 and above

- Addition of details in Understanding Velocity and Acceleration Values

ST4 API Reference API - V0.105 **July 5, 2018** – Versioning implemented initial version created.

This works with

ST4_RC007_42 and above

- Addition of G215 and G216

ST4 API Reference API - V0.104 **May 30, 2018** – Versioning implemented initial version created.

This works with

ST4_RC007_36 and above

- Bugfix – for Min Max default values for stops resolved. Affects G211, G212, G213 and G214
- Addition of G700, G710