NYX-101 Strain Wave Gear Mount

COMMAND PROTOCOL

VERSION 1.0 - 2023-OCT-25

VERSION HISTORY

Version #	Implemented By	Revision Date	Reason
1.0	*Evans Souglakos*	*01/09/2022*	*Initial Document*

TABLE OF CONTENTS

INTRODUCTION **4**

COMMUNICATION **4**
1. Wireless TCP/IP 4
2. USB to Serial 4

COMMANDS **4**
1.1 Limits 5
1.2 Park 5
1.3 Home 6
1.4 Telescope mount time and location 6
1.5 Status 8
1.6 Mount Control 9
1.7 Goto Target 9
1.8 Star Alignment 12
1.9 Guide 12
1.10 Tracking Rates 15
1.11 Motor Tuning 16
1.12 Wi-Fi Network 17
1.13 Mount Information 18
1.14 Sensors 19
1.15 Special 20

INTRODUCTION

Thank you for purchasing the Pegasus Astro NYX-101 Mount.

Strain wave drive technology results in a smooth, backlash-free mount operation with extreme high without the need for counterweights. A fully CNC machined aluminium 6061 body, black and blue anodized, and a selection of branded internal parts, ensure that Nyx-101 will perform flawlessly for many years.

This hybrid combination of a stepper motor & belt system along with sophisticated electronics drives the Right Ascension axis down to 0.10 arcseconds of resolution. Mount has an electronic brake on the RA axis to prevent backsliding when power is off as well as safety limit stops during slew or tracking.

NYX-101 is compatible with ASCOM, INDI, and any software that supports the LX200 command set.

The purpose of this document is to support the user in the operation of the NYX-101. Please read carefully the manual before using the NYX-101 mount.

Any equipment damage or injury caused by improper mount operation is the end user's responsibility.

COMMUNICATION CHANNELS

USB to Serial

The device is announced in the USB Bus as `NYX-101`

The full discovered device instance is: `USB\VID_0403&PID_6015\NYXxxxxxxxx`

(Where `xxxxxxxx` is the mount's unique id)

The default serial port configuration is: *Speed*: 115200, *Data Bits*: 8, *Parity*: None, *Stop Bits*: 1

Wireless TCP/IP

Mount listens to commands in four (4) TCP ports: 9996,9997,9998,9999

- TCP Ports 9996-9998 are persistent channels. Each channel can serve only one client per time.
- The TCP 9999 port is a standard channel and should be used with LX200-compliant clients.

COMMANDS

All commands are case-sensitive. Each command is a string of characters beginning with ":" and ending with "#", with the only exception of the command. A single "#" character clears the receiving buffer.

LX200 AND EXTENDED SET

Commands in red text colour are classic LX200 compatible. Any other command is an extended set to cover all mount's functions.

1.1 Limits

:Gh#	Get Horizon Limit, The minimum elevation of an object above the horizon to which the telescope will slew with reporting a “Below Horizon” error. Returns: sDD*#
:Go#	Get Overhead Limit, The highest elevation above the horizon that the telescope will GoTo. Returns: DD*#
:GXE9#	Get minutes past east meridian. Returns: n#
:GXEA#	Get minutes past west meridian. Returns: n#
:Sh[sDD]#	Set the minimum object elevation limit. Returns: 0 on failure, 1 on success
:So[DD]#	Set highest elevation in degrees to which the telescope will slew. Returns: 0 on failure, 1 on success
:SXE9,[n]#	Set meridian limit east to value [n] in minutes. Returns: 0 on failure, 1 on success
:SXEA,[n]#	Set meridian limit west to value [n] in minutes. Returns: 0 on failure, 1 on success

1.2 Park

:hP#	Moves mount to the park position.. Returns: 0 on failure, 1 on success
:hQ#	Set the mount park position. Returns: 0 on failure, 1 on success

1.3 Home

:hC#	Moves mount to the home position. Returns: Nothing
:hF#	Reset mount at the home position. This position simulates a cold Start. Point to the celestial pole. Returns: Nothing

1.4 Time and location

Gc#	Get the Local Standard Time format. Returns: 24#
:GC#	Get standard calendar date. Returns: MM/DD/YY#
:GG#	Get UTC offset time, hours and minutes to add to local time to convert it to UTC. Returns: Returns: [s]HH:MM#
:GL#	Get Local Standard Time in 24-hour format. Returns: HH:MM:SS#
:GLH#	Returns: HH:MM:SS.SSSS# (high precision)
:Ga#	Get standard time in 12 hour format. Returns: HH:MM:SS#
:Gt#	Get current site Latitude, north is positive. Returns: sDD*MM#
:GtH#	Returns: Returns: sDD*MM:SS.SSS# (high precision)

:Gg#	Get Current Site Longitude, east is negative. Returns: sDDD*MM#
:GgH#	Returns: HH:MM:SS.SSSS# (high precision)
:GS#	Get the Sidereal Time as sexagesimal value in 24-hour format. Returns: HH:MM:SS#
:GSH#	Returns: HH:MM:SS.ss# (high precision)
:Gv#	Get the site elevation in meters. Returns: +/-n.n
:GX89#	Date/time ready status Returns: Return: 0 ready, 1 not ready
:SC[MM/DD/YYYY]#	Change local standard date. Returns: 0 on failure, 1 on success
:SG[sHH:MM]#	Set the number of hours added to local time to yield UT1. (where MM is 00, 30, or 45) Returns: 0 on failure, 1 on success
St[sDD*MM:SS]#	Set current site latitude in degrees. Returns: 0 on failure, 1 on success
:Sg[(s)DDD*MM:SS]#	Set current site longitude, east longitudes can be negative or > 180 degrees. Returns: 0 on failure, 1 on success
:SL[HH:MM:SS]#	Set the local Time. Returns: 0 on failure, 1 on success
:Sv[sn.n]#	Sets current site elevation in meters. Returns: 0 on failure, 1 on success

1.5 Status

:Gm# Gets the meridian pier-side.
Returns:
E#, W#, N# (none/parked)

:GU# Get telescope status.
Returns: s#

n	**n**ot tracking
N	**N**o goto
p	Not **p**arked
I	Parking **I**n progress
P	**P**arked
F	Park **F**ailed
H	At **H**ome
h	Slewing **h**ome
G	Pulse **G**uide active
g	**g**uide active
r	**r**efraction comp enabled
(	Lunar rate
O	Solar rate
K	**K**ing rate
w	**w**aiting at home
u	Pa**u**se at home
z	bu**z**zer enabled
E	Equatorial mode
A	Alt-Azimuth mode
O	Pier side none
T	Pier side east
W	Pier side west
Pulse-guide rate	n.n
Guide Rate	n.n

1.6 Mount Control

:GR#	Get Mount Right Ascension. Returns: HH:MM:SS#
:GRH#	Returns: HH:MM:SS.SSSS# (high precision)
:GD#	Get Mount Declination. Returns: sDD*MM:SS#
:GDH#	Returns: sDD*MM:SS.SSS# (high precision)
:GZ#	Get Mount Azimuth. Returns: DDD*MM.SS#
:GZH#	Returns: DDD*MM'SS.SSS# (high precision)
:GA#	Get Mount Altitude. Returns: sDD*MM:SS#
:GAH#	Returns: sDD*MM:SS.SSS# (high precision)

1.7 GoTo Target

:Gr#	Get target Right Ascension. Returns: HH:MM:SS
GrH#	Returns: HH:MM:SS.SSS# (high precision)
Gd#	Get target Declination. Returns: HH:MM:SS
GdH:	Returns: HH:MM:SS.SSS# (high precision)
:Gal#	Get target Altitude. Returns: sDD:MM:SS#

:GaH#	Returns: sDD:MM:SS.SSS# (high precision)
:Gz#	Get target Azimuth. Returns: DDD*MM:SS#
:GzH#	Returns: DDD*MM:SS.SSS# (high precision)
:Sr[HH:MM:SS]#	Set Target Right Ascension. Returns: 0 on failure, 1 on success
:Sd[sDD*MM:SS]#	Set Target Declination. Returns: 0 on failure, 1 on success
:Sa[sDD*MM'SS]#	Set Target Altitude. Returns: 0 on failure, 1 on success
:Sz[DDD*MM'SS]#	Set Target Azimuth. Returns: 0 on failure, 1 on success
:MA#	Slew the target Alt and Az. Returns: 0..9, see :MS#
:MN#	Slew current RA/Dec but opposite Pier side (within meridian limit overlap). Returns: 0..9, see :MS#
:MNe#	Slew current RA/Dec but East of the Pier (within meridian limit overlap). Returns: 0..9, see :MS#
:MNw#	Slew current RA/Dec but West of the Pier (within meridian limit overlap). Returns: 0..9, see :MS#
:MD#	Slew Destination pier side for the Target Object. Returns: 0=destination is East of the pier

1=destination is West of the pier
2=an error occurred

:MS# GoTo the Target Object.
Returns:

0	goto is possible
1	below the horizon limit
2	above overhead limit
3	controller in standby
4	mount is parked
5	slew in progress
6	outside limits
7	hardware fault
8	already in motion
9	unspecified error

:CS# Synchronize the telescope with the current right ascension and declination coordinates.
Returns:
Nothing

:CM# Synchronize the telescope with the current right ascension and declination coordinates.
Returns:
Returns: "N/A#" on success, "En#" on failure where n is the error code per the :MS# command

---

1.8 Star Alignment

:A?#	Align status. Returns: mno# where: m is the maximum number of alignment stars n is the current alignment star (0 otherwise) o is the last required alignment star when an alignment is in progress (0 otherwise)
:A[n]#	Start Telescope Manual Alignment Sequence. This is to initiate a n-star alignment for 1..MAX_NUM_ALIGN_STARS: 1) Before calling this function, the telescope should be in the polar-home position 2) Call this function with the # of align stars you'd like to use 3) Set the target location (RA/Dec) to a bright star, etc. (not too near the NCP/SCP) 4) Issue a goto command 5) Center the star/object using the guide commands (as needed) 6) Call :A+# command to accept the correction i. (for two+ star alignment) 7) Back to #3 above until done, except where possible choose at least one star on both meridian sides Returns: 0 on failure, 1 on success
:A+#	Align accept target location. Returns: 0 on failure, 1 on success
:AW#	Align Write to memory. Returns: 1 on success

1.9 Move and Guide

:GX90#	Get pulse guide rate [0.25, 0.50, 1.00] Returns: n.nn#
:GX91#	Get both axis rate in degrees (this rate is applied to control arrows). Returns: n.nn#
:GX94#	Get motor position in steps (zero is the start-up position where the mount point to NCP or zero position) Returns: n.n#

:GX97#	Get max slew rate in degrees Returns: n.n#
:Mg[d][n]#	Pulse guide command where d = direction (n,s,e,w) and where n = guide time in milliseconds. Returns: Nothing
:MG[d][n]#	Pulse guide command where d = direction (n,s,e,w) and where n = guide time in milliseconds. Returns: 0 on failure, 1 on success
:Mw#	Move Telescope West at current guide rate. Returns: Nothing
:Me#	Move Telescope East at current guide rate. Returns: Nothing
:Mn#	Move Telescope North at current guide rate. Returns: Nothing
:Ms#	Move Telescope South at current guide rate. Returns: Nothing
:Mp#	Move Telescope Spiral Search at current guide rate. Returns: Nothing
:Q#	Halt all current slewing. Returns: Nothing
:Qw#	Halt Westward slews. Returns: Nothing
:Qe#	Halt Eastward slews. Returns: Nothing

:Qn#	Halt Northward slews. Returns: Nothing
:Qs#	Halt Southward slews. Returns: Nothing
:RA[n.n]#	Set RA/Azimuth Slew rate to DD.D degrees per second. Returns: Nothing (Note: Tracking, if active, is suspended in this mode of operation)
:RE[n.n]#	Set Dec/Elevation Slew rate to DD.D degrees per second. Returns: Nothing (Note: Tracking, if active, is suspended in this mode of operation)
:RC#	Set axis rate: Centering [8X] Returns: Nothing
:RM#	Set axis rate: Find [20X] Returns: Nothing
:RF#	Set axis rate: Fast [64X] Returns: Nothing
:R[n]#	Set axis rate preset. Returns: Nothing

N	**speed**
0	x0.25 pulse guide rate
1	x0.5 pulse guide rate
2	x1.0 pulse guide rate
3	x2
4	x8
5	x20
6	x64
7	x128
8	½ Max speed
9	Max speed

Tracking Rates

:GT#	Get tracking rate. Returns: n.n
:ST[H.H]#	Sets tracking rate to Hz, where a 60.0 Hertz will cause the RA axis 1rev/24h. Returns: 0 on failure, 1 on success
:TS#	Set Tracking Rate Solar [60Hz] Returns: 0 on failure, 1 on success
:TK#	Set Tracking Rate King [60.136Hz] Returns: 0 on failure, 1 on success
:TL#	Set Tracking Rate Lunar [57.9Hz] Returns: 0 on failure, 1 on success
:TQ#	Set Tracking Rate Sidereal [60.164Hz] Returns: 0 on failure, 1 on success
:T+#	Master sidereal clock faster by 0.02 Hz (stored in memory) Returns: Nothing
:T-#	Master sidereal clock slower by 0.02 Hz (stored in memory) Returns: Nothing
:TR#	Master sidereal clock reset (to calculated sidereal rate, stored in memory) Returns: Nothing
:Te#	Tracking enable. Returns: 0 on failure, 1 on success
:Td#	Tracking disable. Returns:

	0 on failure, 1 on success
:Tr#	Track refraction compensation on. Returns: 0 on failure, 1 on success

1.10 Motor Tuning

:%BR#	Get RA/Azm backlash in arc-seconds. Returns: Return: n#
:%BD#	Get Dec/Alt backlash in arc-seconds. Returns: Return: n#
:$BR[n]#	Set RA/Azm backlash in arc-seconds. Returns: 0 on failure, 1 on success
:$BD[n]#	Set Dec/Alt backlash in arc-seconds. Returns: 0 on failure, 1 on success

1.11 Wireless Network

:WSxxxxx..#	Set Wi-Fi Client SSID [1-25 chars] Returns: 0 on failure, 1 on success
:WPxxxxx..#	Set Wi-Fi Client Password [8-30 chars] Returns: 0 on failure, 1 on success
:WAxxxxx..#	Set Access Point SSID (1-25 chars) Returns: 0 on failure, 1 on success
:WBxxxxx#	Set Access Point Password [8-30 chars] Returns: 0 on failure, 1 on success
:WLC#	Reload Wi-Fi Subsystem (applies all changes) Returns: 0 on failure, 1 on success
:WLZ#	Reset Wi-Fi Subsystem to defaults (removes all settings about Wi-Fi hotspot and client) Returns: 0 on failure, 1 on success
:WL?#	Wi-Fi Status Returns: 0 on failure or “string” on success (check below) N = No SSID available C:n = Connected: signal strength % F = Connect failed L: Connection lost D: Disconnected U: Unknown
:WL>#	Prints configured Wi-Fi hotspot SSID and Password. *The ‘**NYX_**’ prefix is not printed but is included in the SSID name* Returns: 0 on failure or “string” on success (check below) [ssid_name],[password]
:WLI#	Prints connected to the Wi-Fi network name. Returns: [ssid],[signal strength],[encryption] or 0 , if no network is connected

:WLD#	Prints connected SSID with Wi-Fi Client IP. Returns: [ssid]:[ip]
:WLS#	Scan and list available networks (blocking – requires ~5 seconds) Returns: (up to 15 networks ordered by high to low signal strength): [ssid],[signal_strength],[encryption] : [ssid],[signal_strength],[encryption]: …

1.12 Mount Information

:GVD#	Get firmware build date Returns: MTH DD YYYY#
:GVM#	Get product name along with firmware number Returns: NYX-101 n.nn
:GVN#	Get firmware number Returns: n.nn
:GVP#	Get product name Returns: NYX-101#
:GVT#	Get firmware build time Returns: HH:MM:SS#
:GVU#	Get mount unique identification (8 chars) Returns: PEG_NYX-101:xxxxxxxx
:GVY#	Get uptime in seconds Returns: nn

1.13 Sensors

:GX9A# — Temperature (ambient) in °C
Returns:
+/-n.n

:GX9B# — Barometric Pressure in mb.
Returns:
+/-n.n

:GX9C# — Barometric Altitude in meters.
Returns:
+/-n.n

:GX9D# — Leveler Pitch and Roll in degrees.
Returns:
Pitch : Roll
n.n : n.n

:GX9E# — Compass (orientation in degrees).
Returns:
n.n

:GX9F — Temperature of MCU in °C
Returns:
n.n

:GX9V — Input Voltage in Volts.
Returns:
n.n

:GXU1# — Motor Controller RA Axis Status.
Returns: (check below table for string)
,,,,,,,#

:GXU2# — Motor Controller DEC Axis Status.
Returns: (check below table for string
,,,,,,,#

ST	At Standstill
OA	Output A Open Load
OB	Output B Open Load
GA	Output A Short to Ground
GB	Output B Short to Ground
OT	Over Temperature > 150C

PW	Over Temperature > 120C
GF	Fault
[nnnn]	Motor Load. A higher value signals a lower motor load and more torque headroom. Value is valid only during slewing. During tracking, the value is always zero.

1.14 Special

:ERESET# — Reload the mount's firmware.
Returns:
Nothing

:ENVCLEAR# — Clear memory contents – Reset to factory defaults (after restart).
Mount boot will stall for 2-3 minutes as memory requires some time for re-initialization. Please be patient.
Returns:
Nothing:

:GXEM# — Get mount type
Returns:
1 = GEM
3 = ALT/AZ

:SXEM,[n]# — Set mount type (for next restart) where n=0 for default.
1 = GEM
3 = ALT/AZ
Returns:
0 on failure, 1 on success

:SX91,[s]# — Control the electric power-off brake on the right ascension axis
U = Unlock brake (disengage)
L = Lock brake
Returns:
0 on failure, 1 on success

:SBn# — Set Baud Rate n, where n is an ASCII digit (1..9) with the following interpretation:
0=115.2K, 1=56.7K, 2=38.4K, 3=28.8K, 4=19.2K, 5=14.4K, 6=9600, 7=4800, 8=2400, 9=1200
Returns: 1
(at the current baud rate and then changes to the new rate for further communication)