Nova mount PID tuning

May 10th, 2021

Table of Contents

1 List of acronyms 3

2 Purpose of this document and prerequisite 3

3 Mount hardware principle and closed loop system 3

4 Using DDR_Astro software to achieve PID tuning 6

4.1 Initial settings 6

4.2 Control panels description 7

4.3 PID tuning in depth 17

4.3.1 Current PI loop 17

4.3.2 Position PID 19

4.3.3 Setting the best PID 25

4.3.4 Slow speed PID control 32

4.3.1 Final checks 33

4.3.1 Final checks 35

1 List of acronyms

PID : Proportional Integral Derivative closed loop algorithm
PI : Proportional Integral closed loop algorithm
RA : Right Ascension
DEC : Declination
AZ : Azimuth
ELV : Elevation
OTA : Optical tube Assembly or telescope tube containing instrumentation and all optics.
STD : Standard Deviation figure computed from significant number of samples

2 Purpose of this document and prerequisite

The purpose of this document is to aim for best mount PID tuning, which means best tracking performance, high speed and reasonable settling time after slew completion.

All described in this document is valid for ALL mount axis kinds (RA, DEC, AZ, ELV).

The NOVA mount is a direct drive mount that uses encoders and brushless frameless motors. The mount controller uses motion generator functions, read encoder values and PID algorithm to stick to the required trajectory with the actual mount motion.

3 Mount hardware principle and closed loop system

Each axis of the mount (RA/DEC/AZ, ELV) has an optical encoder that provides ultra-accurate position angle. The encoder resolution has from 33 million steps to 97 million steps, so be within 40 milli-arcsec down to 10 milli-arcsec. Just remember that one degree is 3600 arcsec. There is also a torque motor, that receives current from the controller.

The encoder information goes to the controller and is compared to the trajectory generator, that computes a theoretical trajectory. There are two trajectories used:

- Constant Velocity contouring: used for tracking mode, where constant speed is required, this will translate into encoders steps per sec.
- Trapezoidal trajectory: used for slewing the telescope from one position to another in the sky. This includes a variable speed with acceleration phase to reach desired slewing speed, then constant slewing speed, then a deceleration phase in order to stop at the required destination position. This also translates into variable encoders steps per sec.

In all cases, the controller produces a variable setpoint angle where the axis should be at any time, reads the encoder position (or axis rotation angle), computes the difference (position error) and finally applies a current to the motor. This current generates a torque which makes the axis to rotate. This is a closed loop system.

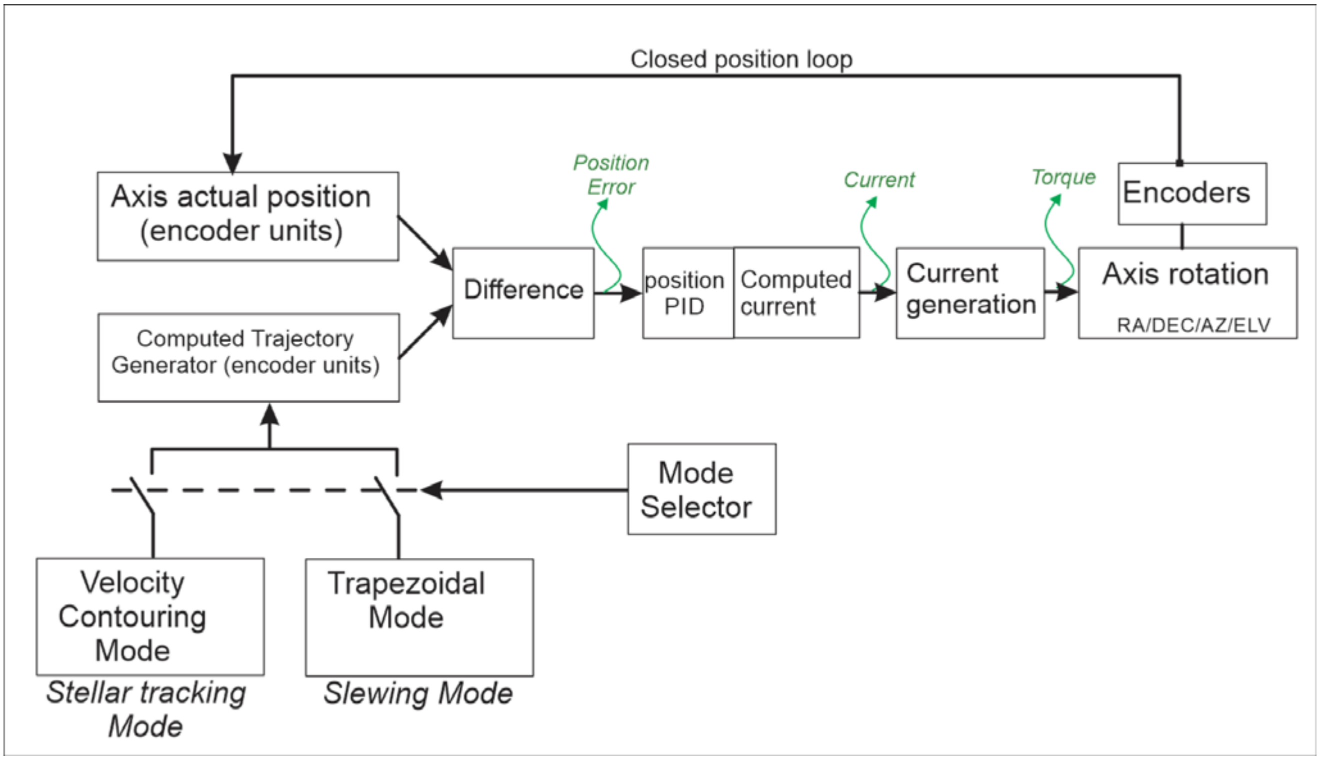


Fig. 1 *Mount closed loop system overview*

The" Position Error" is the difference between the Trajectory Generator and the actual encoder mount position, which the system will tend to bring to a minimum.

This position error goes into a PID algorithm which outputs a current/torque, so that at the next cycle (150µs per cycle) the position error tends to zero. This PID algorithm has many parameters which defines it.

Some basic knowledge about PID is required, please read/watch those documents:

- Understanding PID Control, Part 1: What Is PID Control? - YouTube
- What is a PID Controller and how does it work? - YouTube
- https://en.wikipedia.org/wiki/PID_controller

This document will not describe "what is a PID", and will not cover this topic. There are countless of resources in the web about this topic.

If P, I, D parameters are incorrectly set, either

- The axis will be soft and slow, and "position error" will increase, when the position error is higher to a given figure, for safety reasons the controller will disable the closed loop and stop the motors
- The axis will oscillate, and won't be stable.

So proper PID parameters must be set correctly in order for the axis to behave adequately.

Important: PID parameter are bound to the inertia momentum of the mount (https://en.wikipedia.org/wiki/Moment_of_inertia) and if mount masses distribution around the axis are changed, it might require to change the PID parameters to keep good performances.

The controller has another feature which allows better performance, with current closed loop PI. It ensures that the required current is used to drive the axis. The current error is computed between the PID required output current and measured current at the motor input. This difference goes into a PI (with no D) algorithm and generate the proper current to minimize the current error.

The current can be limited by the user to values from 10% to 100%, this is to ensure safety when tuning the axis. In the end, the controller manages two nested closed loops, the position closed loop and the current closed loop.

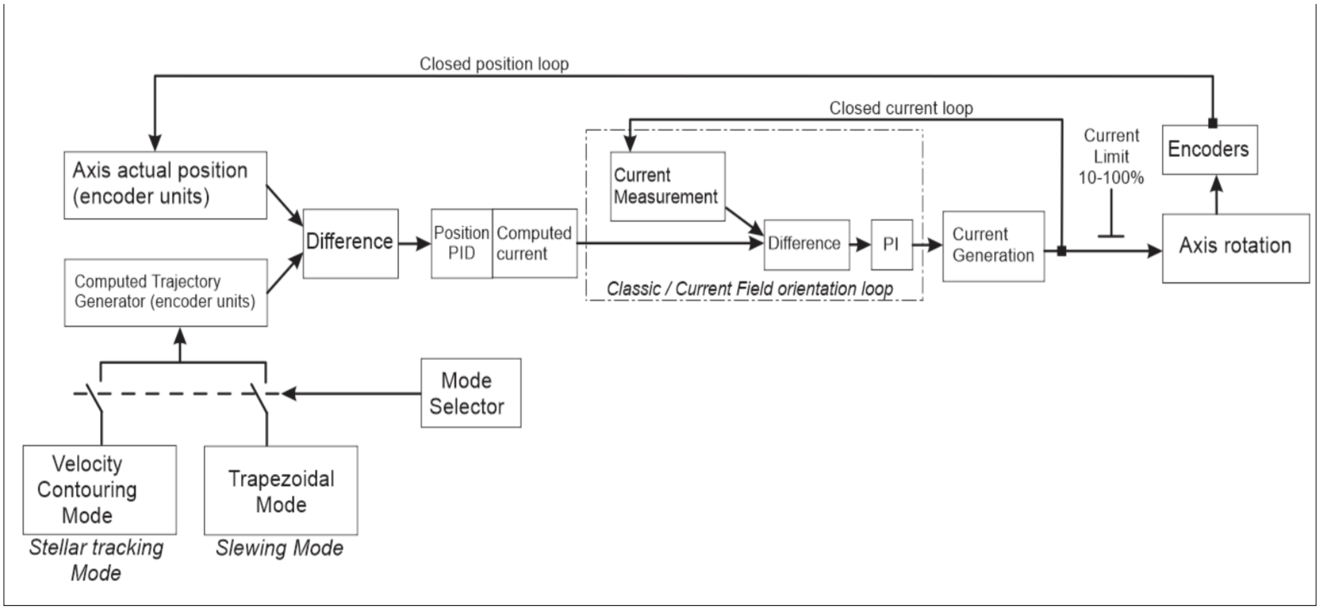


Fig. 2 *Closed loop chart with current loop enabled*

Be aware that this loop operates for one axis, and for dual axis system there's another similar loop that has its own parameters.

Nevertheless, this is possible to disable the closed current loop, if necessary, by software. On some cases this can be useful.

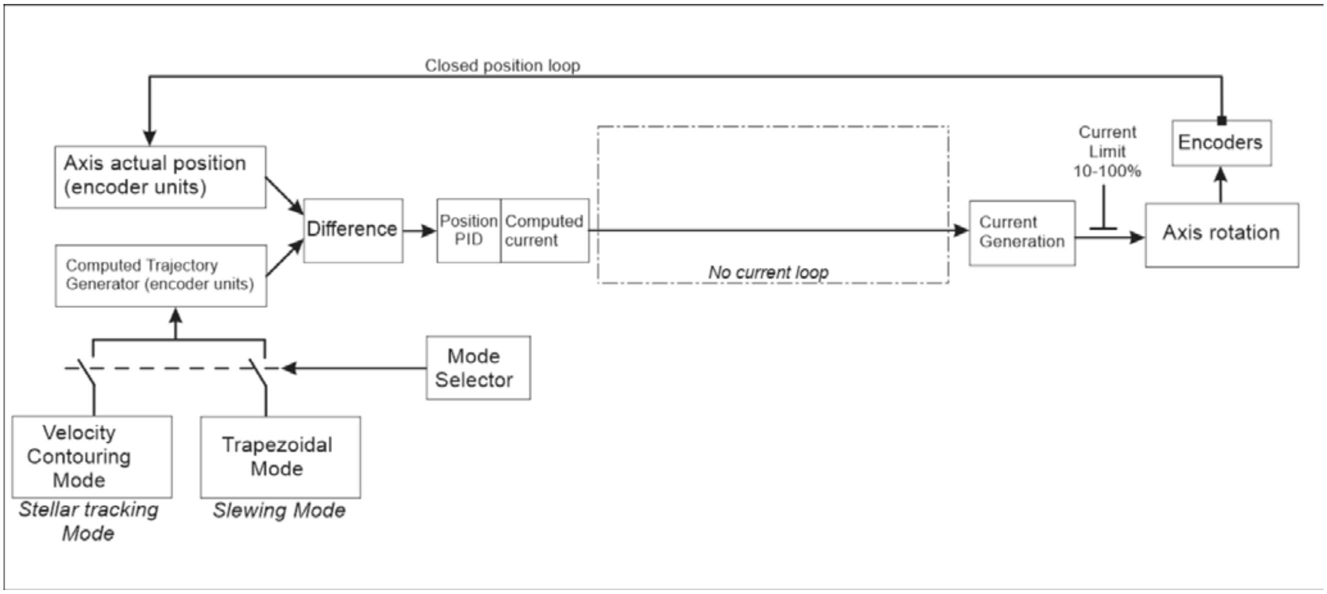


Fig. 3 *Closed loop chart with current loop disabled*

The DDR_Astro software provides full access to the mount PID parameters, for all closed loops (position and current).

4 Using DDR_Astro software to achieve PID tuning

4.1 Initial settings

The hardware controller does not store any parameters, when turned off, all parameters are erased. There is no nonvolatile memory. This is the PC ***DDR_Astro*** software that sends the parameters to the controller upon establishing link. All the mount PID parameters are stored in the window database registry here

HKEY_CURRENT_USER\SOFTWARE\DDR_ASTRO\Axis1

And

HKEY_CURRENT_USER\SOFTWARE\DDR_ASTRO\Axis2

Use "***regedit.exe***" software to access them, if required.

Important use screen PC resolution that are higher and equal to HD resolution, better WUXGA (1920 x 1200) or QXGA (2048x1536) are better, because the tuning process display many floating windows!

Once **DDR_Astro** software is started, go to Menu "***Setup/System setup/***" then click button "***Advanced Setup***"

There, the Current Limit for both axis (see fig 2 and 3) can be set. As well the maximum tracking error (difference between encoder reading and trajectory generator output). Do not set this value too low, otherwise axis can be disabled immediately when error exceed this value, this can happen when slewing starts/stops. This is again a safety action to avoid the mount to send the OTA to undesired speeds and positions.

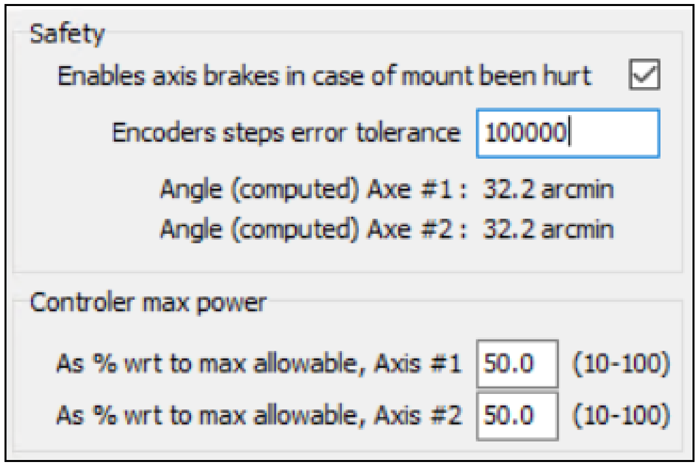


Fig. 4 *Current limit and position error limits*

The maximum power of controllers is 500W per axis (NOVA 120). NOVA 200 uses 1000W controller.

4.2 Control panels description

Then quit this form and connect mount, the advanced axis status can be reach by clicking these small “>>” buttons.

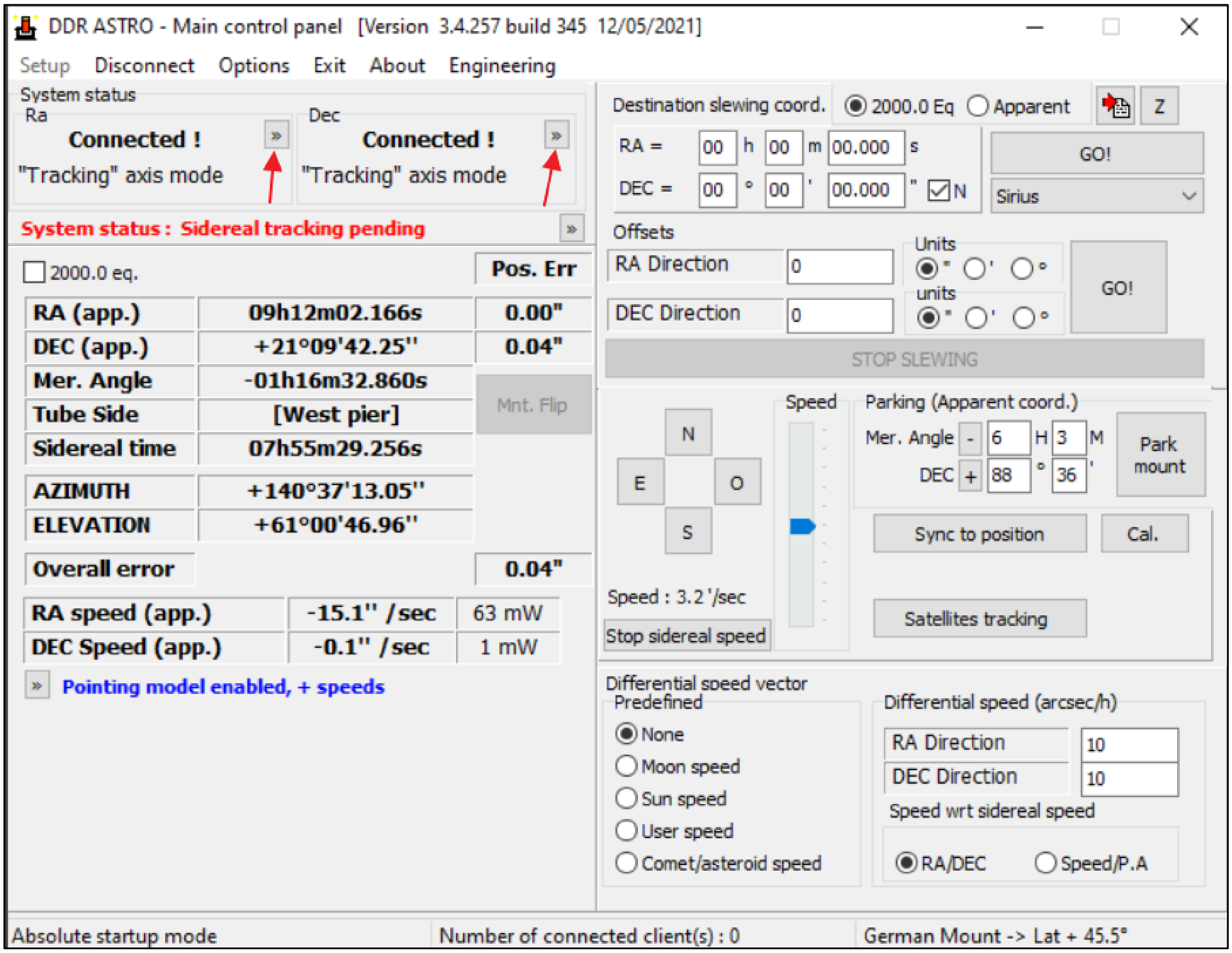


Fig. 5 *Main DDR_Astro control window*

The following form will appear.

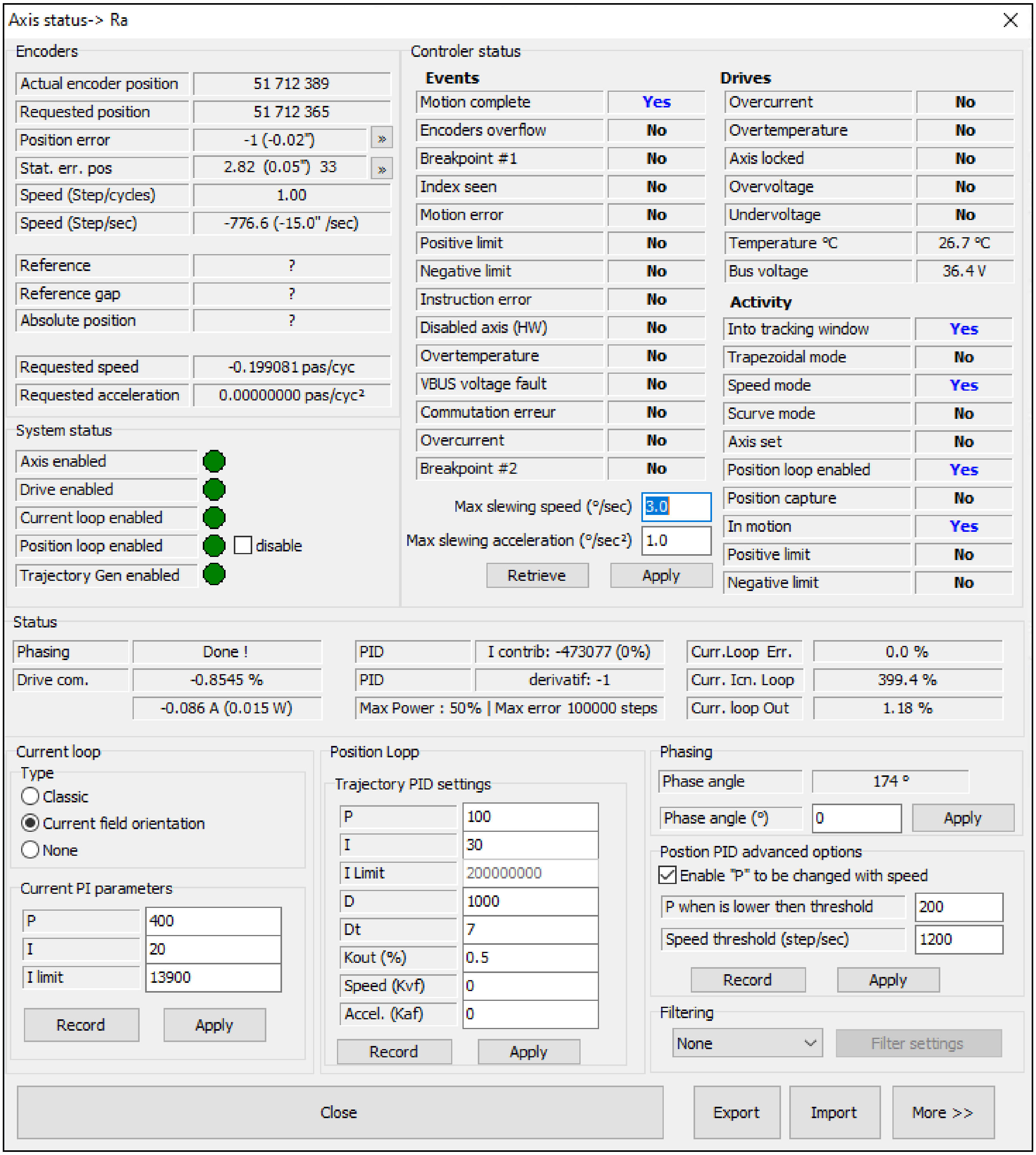


Fig. 6 *Advanced Status window*

The encoder sub-group, shows the position error (that goes into the PID computation box) and by pressing button 2, its resets the computation of the standard deviation error which is performed with the last samples of position error.

In this example, the standard deviation figure has been computed over 1102 samples of position error, and is equal to 44.43 steps or 0.86 arcsec. If this button is hit, the computation restarts from the next sample and waits to get 10 samples to compute this figure.

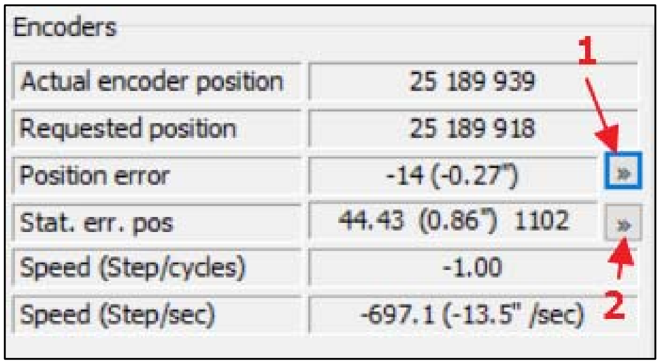


Fig. 7 *Encoder group box*

On a well-tuned system, using sideral or no speed, the STD error must be less than 10 encoders steps.

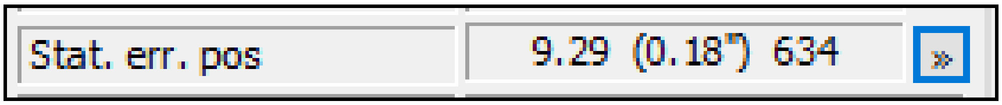
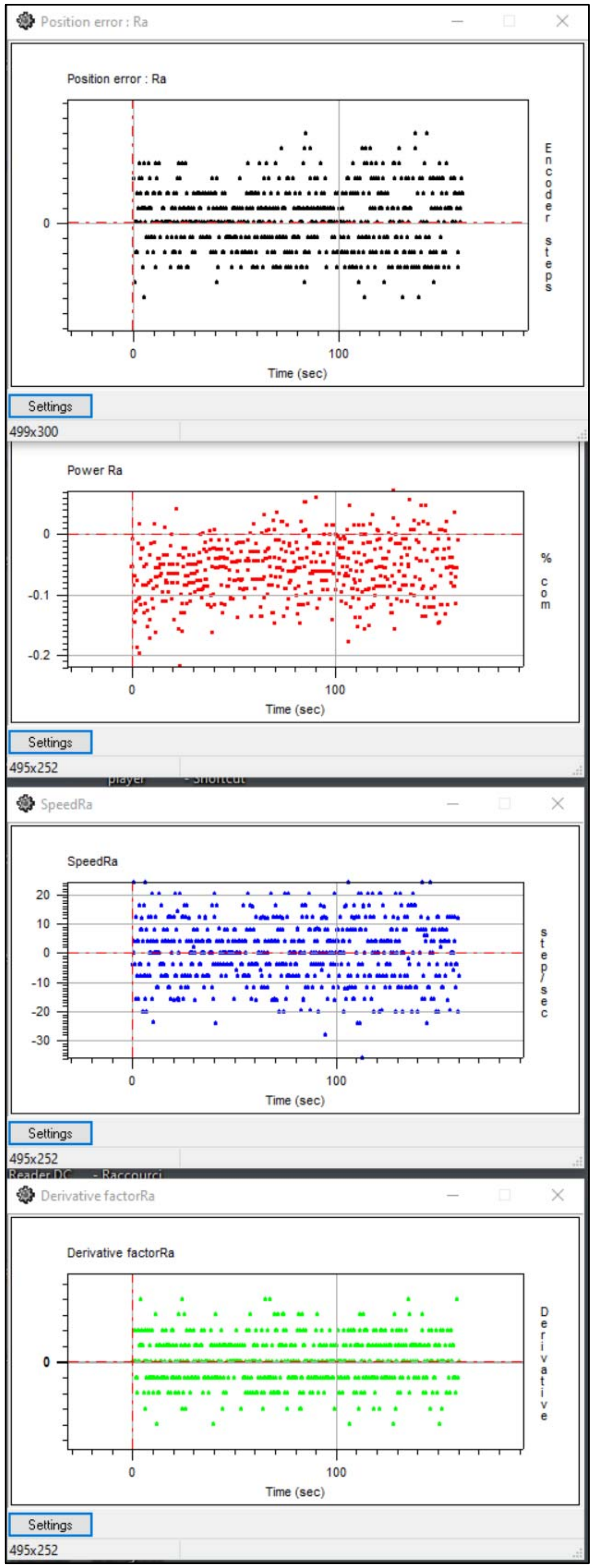


Fig. 8 *Statistical position tracking error and reset button*

The Button #1 will open 5 real time plots, where time is X axis, showing (from top to bottom) the tracking position error, the current in % at the PID output, the measured speed as encoder steps per sec, the derivative factor and the encoder position. In these screen copies the axis does not move (DEC axis).


Position error : Ra
Position error : Ra
0
0
100
Time (sec)
Encoder steps
Settings
499x300
Power Ra
0
-0.1
-0.2
0
100
Time (sec)
% com
Settings
495x252
SpeedRa
SpeedRa
20
10
0
-10
-20
-30
0
100
Time (sec)
step/sec
Settings
495x252
Derivative factorRa
Derivative factorRa
0
0
100
Time (sec)
Derivative
Settings
495x252

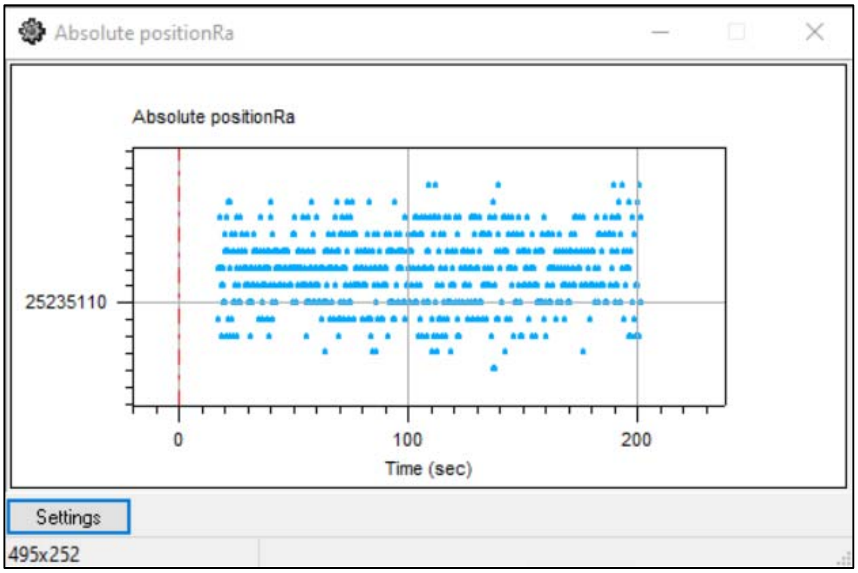


Fig. 9 *Five diagnosis plots*

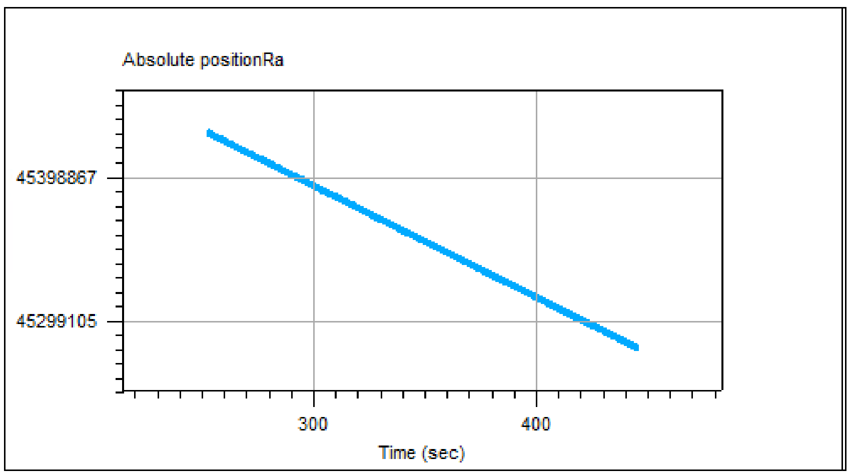


Fig. 10 *Axis position plot with sideral speed*

The max slewing speed and accelerations can be modified hereafter, those applies to **both Axis**

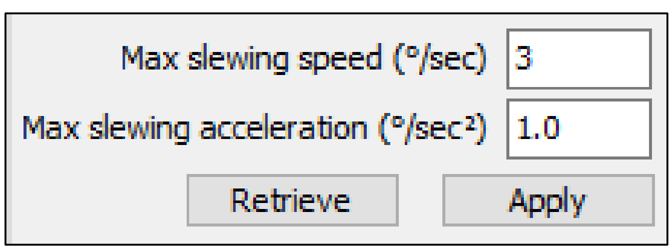


Fig. 11 *Max speed and accelerations parameters*

By clicking the "**>>**" that is at the same line as "***System status***", this simple chart is displayed.

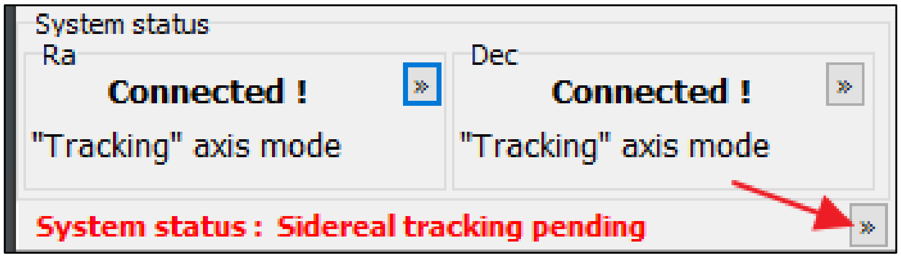


Fig. 12 *Button to show the skymap chart*

This chart allows to move the telescope to any position in the sky by clicking to the right mouse button.

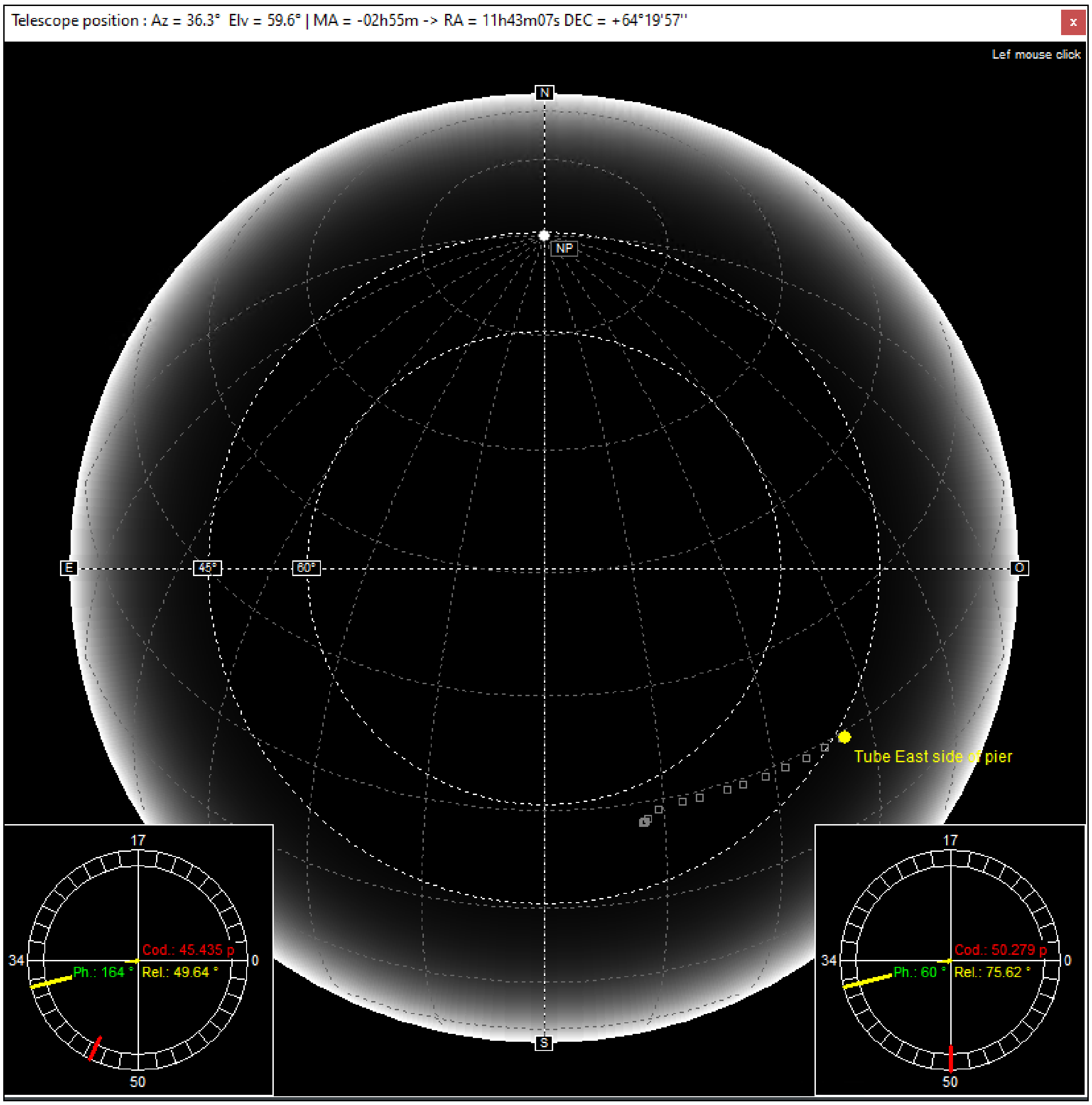


Fig. 13 *Skymap chart, yellow/green dot depicts telescope position*

Encoder positions of both axis are reported on the bottom of the chart (small clock like picture). Understand 34 as 34 000 000 steps. The red line is the current axis position (encoder steps) and the yellow line are the reference encoder steps of the mount's marks, and the green label is the current phase motor angle value. The yellow label "Rel." is the angle of the current position with the reference mark position.

If the user press the right mouse button, the mount destination coordinates are computed and confirmation form is displayed.

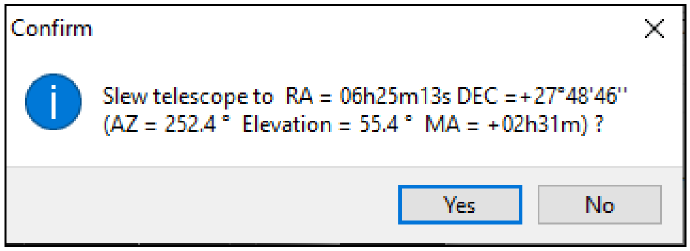


Fig. 14 *Confirmation form*

This is very convenient tool to slew the mount and check that the PID parameters are set properly during a motion test.

Coming back to this panel:

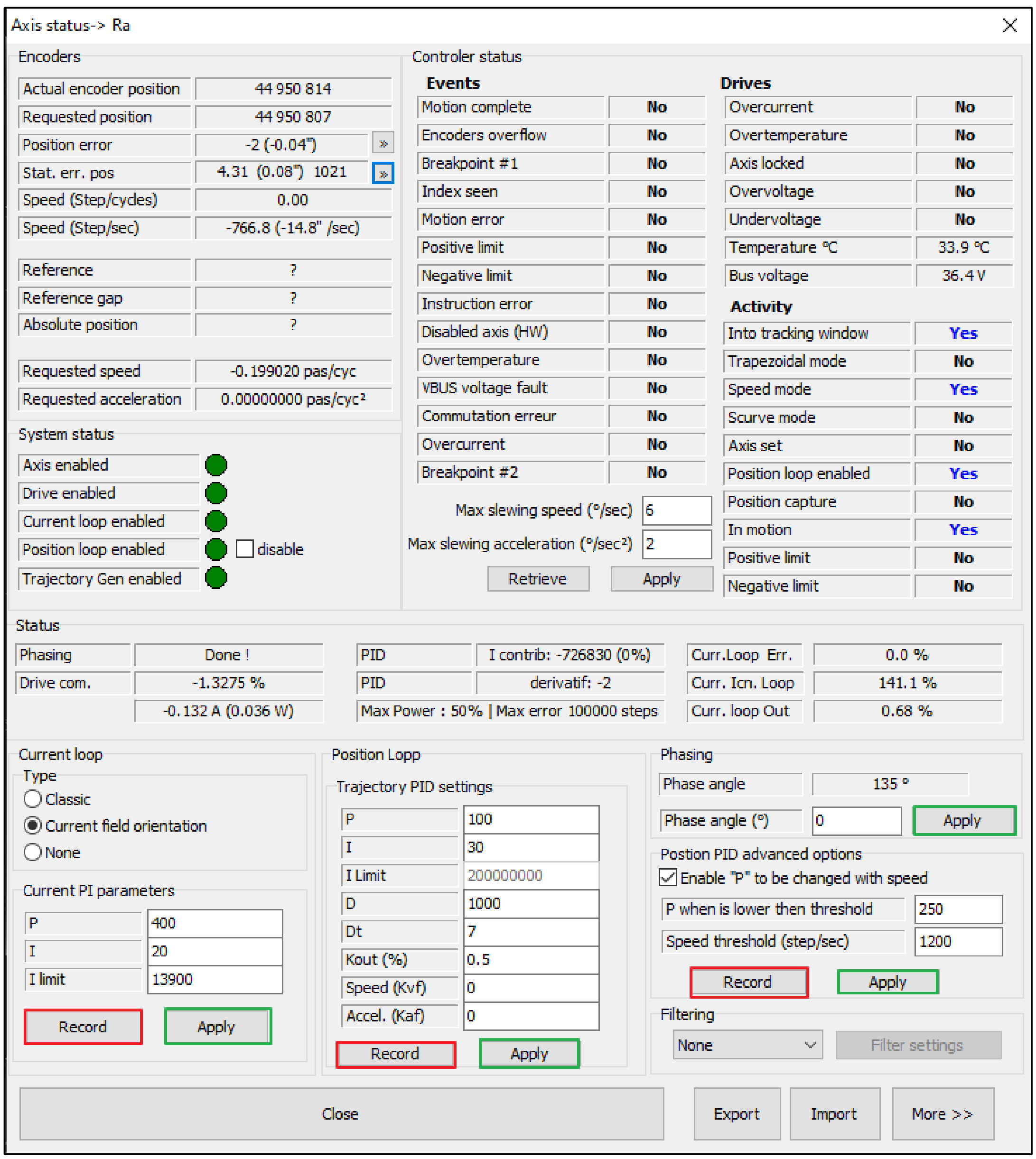


Fig. 15 *Record and Apply buttons*

Buttons "***Apply***": send parameters to the mount controller, DDR ASTRO must be linked to the mount controller otherwise an error message is raised.

Buttons "***Record***": save parameters to the database registry (i.e hard disk) so that when the software closes, those are kept and can be used when the software starts. It does not require the link to the mount controller to operate.

Button "***Export***": save parameters to a text ASCII readable file, this will write all parameters/figures from Current loop, Position loop, Position PID advanced options.

Button "***Import***": loads parameters from text ASCII readable file, this will read all parameters/figures from Current loop, Position loop, Position PID advanced options. A message to apply them to the controller will be asked.

Filtering group box

Fig. 16 *Filter setup*

This allows to put in place the two biquad filters such as notch, or low pass filters, located after the PID box. Please refer to the controller documentation to proceed. Deep knowledge of closed loop systems is required to use this feature. This is not the purpose of this documentation to explore filtering feature. Leave the combo box to "None"

Phasing group box

Motor phase define the phase angle from 0° to 360° that defines the maximum torque force between the rotor and the stator of the motor. At reference marks, this phase is the same and never changes. Details can be read here:

- https://ww1.microchip.com/downloads/en/appnotes/00885a.pdf

Motor phase can be slighted changed or adjusted, where the axis is actually located. This is for test and experiments, and the user can check that the current provided to the motor is minimal, for a given slightly unbalanced axis. The final offset shall be put in the startup panel.

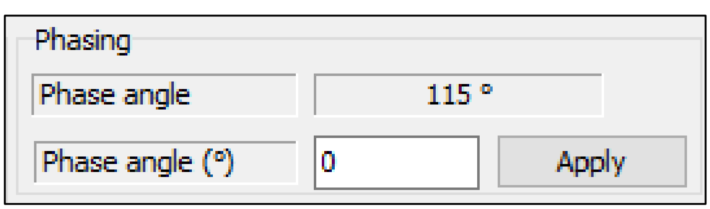


Fig. 17 *Phase angle panel*

Say, that phase has been changed by 5° in RA, and this cause less power consumption, to be taken at the next startup, it should be changed at setup form, and 338 changed to 338° + 5°= 343°
With absolute encoder mount, this is not necessary to change this figure because this is factory tuned once for all. For incremental mounts, this tool can help to refine phase figure when the mount is located at its marks.

Warning, if the phase is changed by a lot; the axis will never work properly whatever the PID parameters.

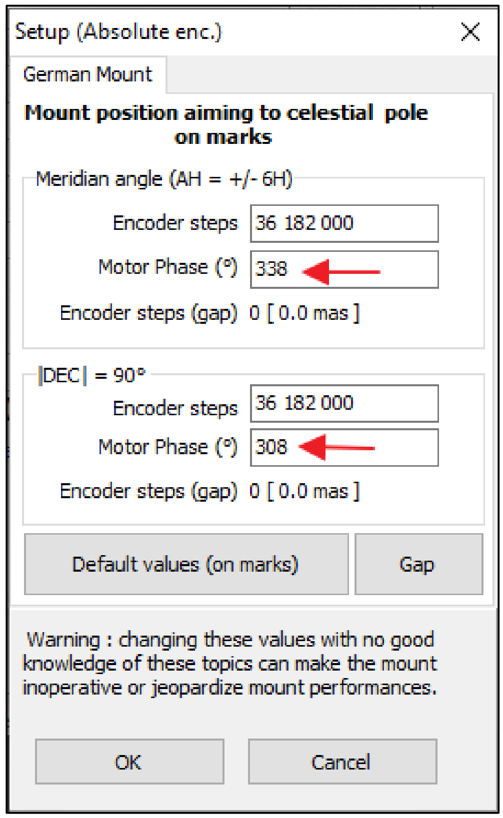


Fig. 18 *Phase startup mount parameters*

Controller Status

This subpanel reports many controller statuses, like motion event, Activity and Drive status (temperature and voltages). This can be useful to debug some errors.

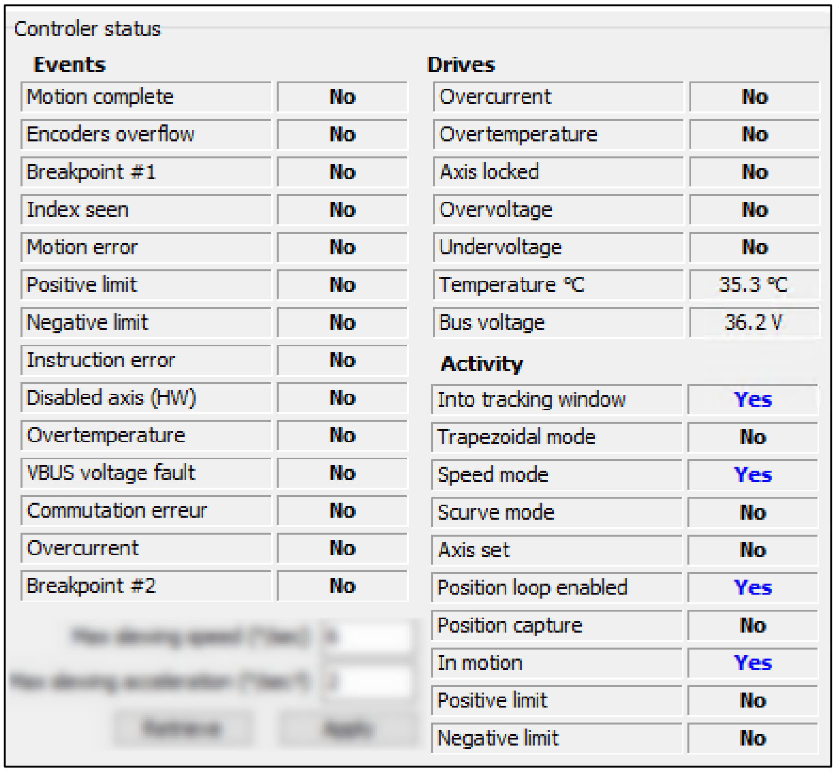


Fig. 19 *Controller status*

System Status group box

It indicates whether the axis is enabled, the driver enabled, the current loop enabled and the trajectory generator is enabled or not.
"***Disable***" checkbox will disable the **position loop**, and will cause the axis to be free. This could be used as an emergency action to cancel all action on the given axis.

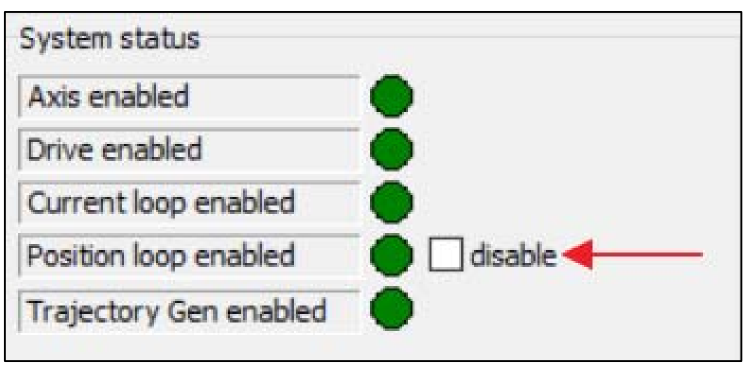


Fig. 20 *Loop status and controller status*

4.3 PID tuning in depth

4.3.1 Current PI loop

There is three different current loop that can be used, ***"Classic"***, "***Current Field Orientation***" and "***None***"

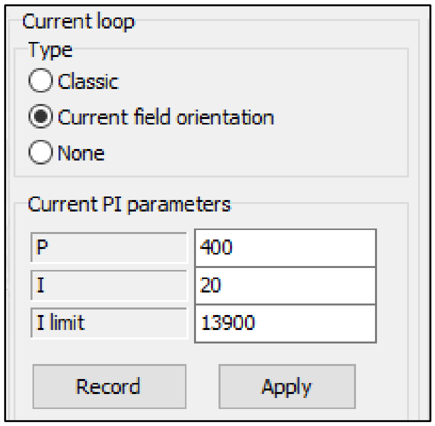


Fig. 21 *Current loop setup and PI setup*

Classic: use classic PI loop see below

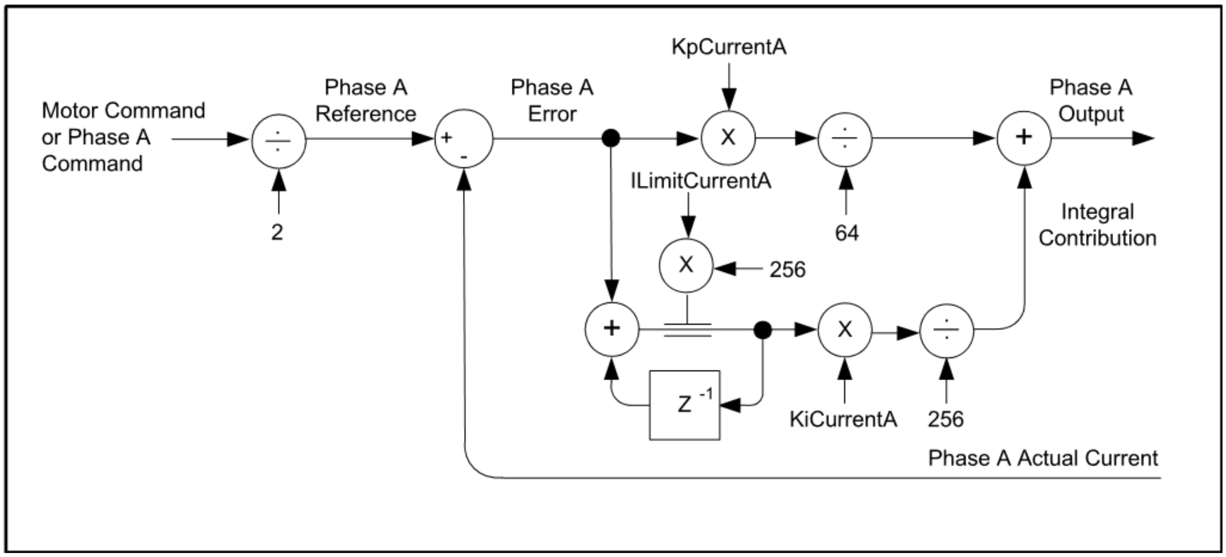


Fig. 22 *Current loop PI flowchart*

Keep in mind that current loop cycle is 47 Khz or 21µs, so this is pretty fast

Current Field Orientation: the uses a very advanced current loop control using Parks/Clark transform, where phase A,B,C are changed to D, Q output. For more information, please visit these places:

- https://www.youtube.com/watch?v=YPD1_rcXBIE
- FIELD ORIENTED CONTROL 3-PHASE AC-MOTORS
- brushless dc motor - Why are two PID controllers needed for FOC (Field Oriented Control) ?
- https://www.youtube.com/watch?v=URJwuU1_w6w

This is a pure torque mode for driving brushless motors, and is more accurate than classical current PI.

None: simply disable the current loop, the mount can work in this mode also.

The most efficient strategy is to set once, the current loop P and I figures (if any), and tune the position loop PID without changing the current loop PI.

Keep in mind that there are 10 parameters to set, and a proper methodology must be use to find them. Otherwise, this can be impossible.

Important:

If current loop type is changed, PID position loop parameters must be changed also.
There is no dramatic difference on PID position parameters if "***Classic***" or "***Current field orientation***" are used, but if current loop type is set to "***None***", the PID position parameter changes are quite important.

ALCOR SYSTEM did a lot of trials with "***P***" and "***I***" for current loop. "***I***" parameter removes the current static error and "***P***" behave like a closed loop gain. "***I limit***" shall be set to a high value.

"***P***" should not exceed 5000, and "***I***" not 120. Figure like P=400 and I=20 can be used and provides good results.

If P is too high, the motor will make a white acoustic noise, that reflect white noise of the current measurement system, it will also impact the D parameter of the PID loop parameter, and D cannot be increased as it should.

This sub panel provides hints about the current loop performance

Curr.Loop Err.	0.0 %
Curr. Icn. Loop	-107.1 %
Curr. loop Out	-0.42 %

Fig. 23 *Current loop PI information*

Current loop error is 0.0%, meaning that proper PI parameters settings have been used.

4.3.2 Position PID

The heart of this document is to find the best P, I, D and Dt to ensure reliable tracking and slewing.

As said in this document, those parameters can change accordingly to the mount payload, position of the counterweights. This is involving mass distribution around the axis, and thus inertia momentum changes.

If these parameters are not set properly the mount will simply stop working as it should, it will produce large tracking errors, or behave like a "soft" axis, and safety measures will be applied to open the loop, and stop any slews or tracking.

As a rule of thumb, the higher the load on the mount (including counterweights), the higher the P figure. There's need of more gain/force when the momentum of inertia is higher.

Nevertheless, if P is too large the mount will also oscillate.

So PID setting parameters is a matter of trade-off.

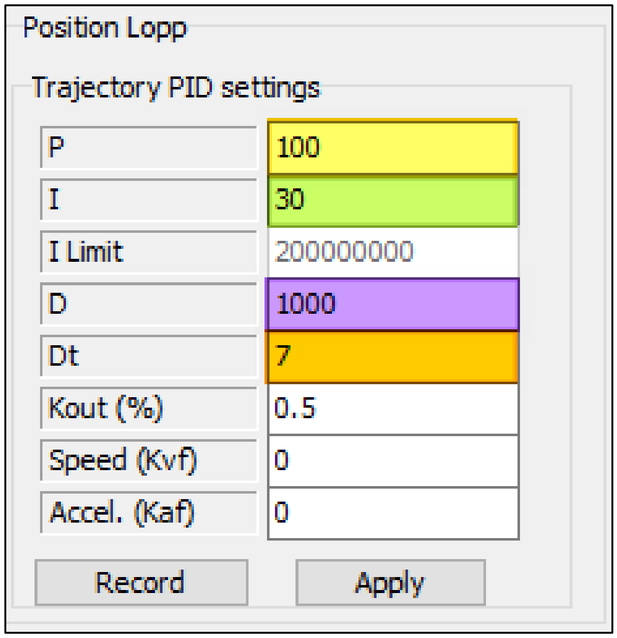


Fig. 24 *PID RA figure for empty NOVA 120 RA mount axis*

Many trial and test must be performed, this is not a one-minute task. All the plot panels must be opened (figure 9) to allow to diagnose the different output parameters from the PID closed loop.

First let's see what a motion does, below is a 45 ° span RA axis motion

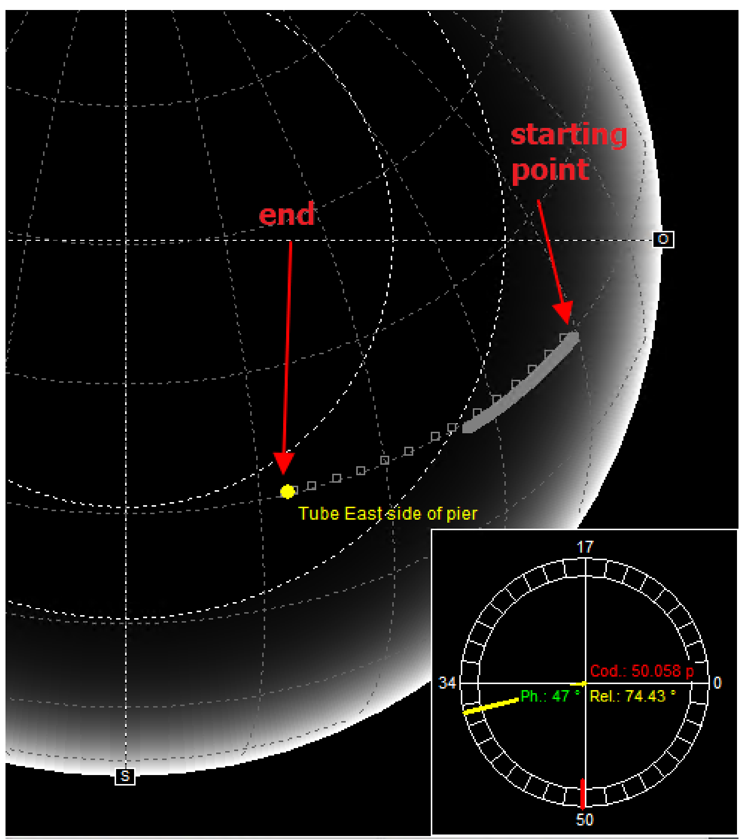


Fig. 25 *Small RA motion*

The axis speed increases to a given value (here as encoders steps /sec -> 6°/sec) there's also an acceleration and deceleration phases.

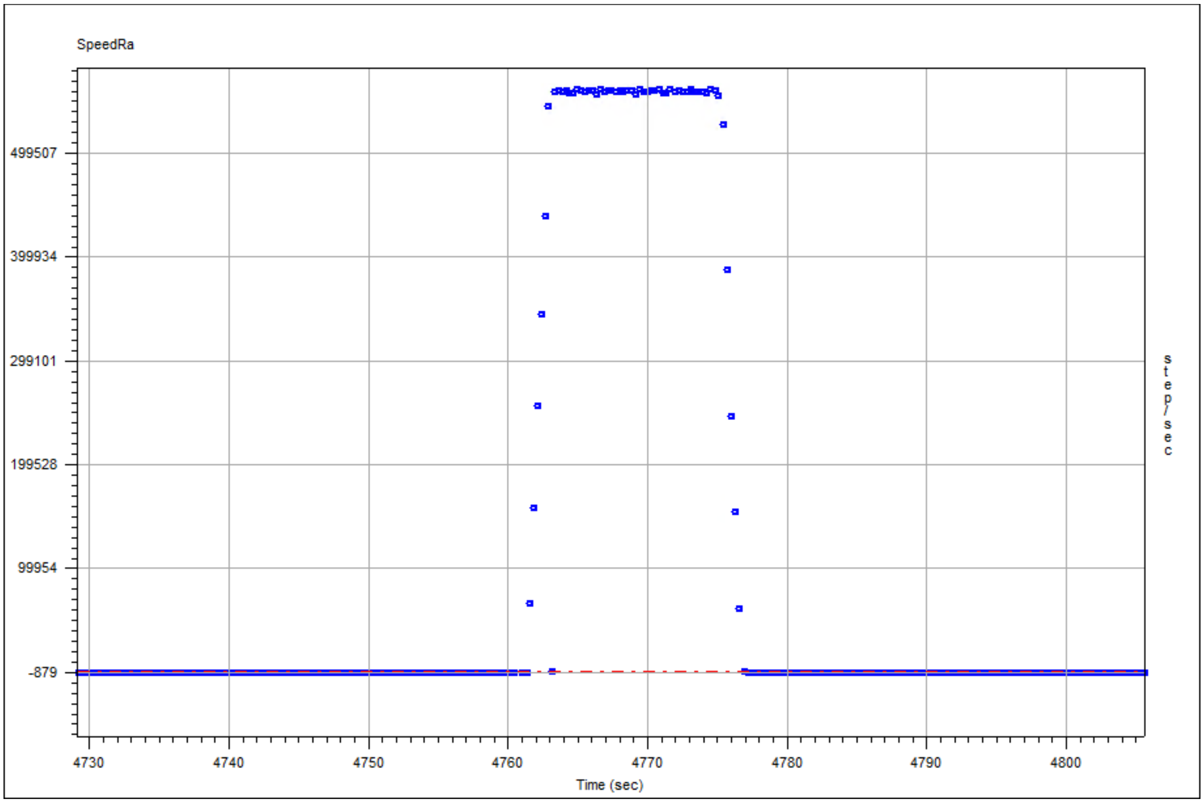


Fig. 26 *Speed plot during a small span RA motion (encoder/sec units)*

The axis position changes from 41.9 million steps to 49 million steps

.

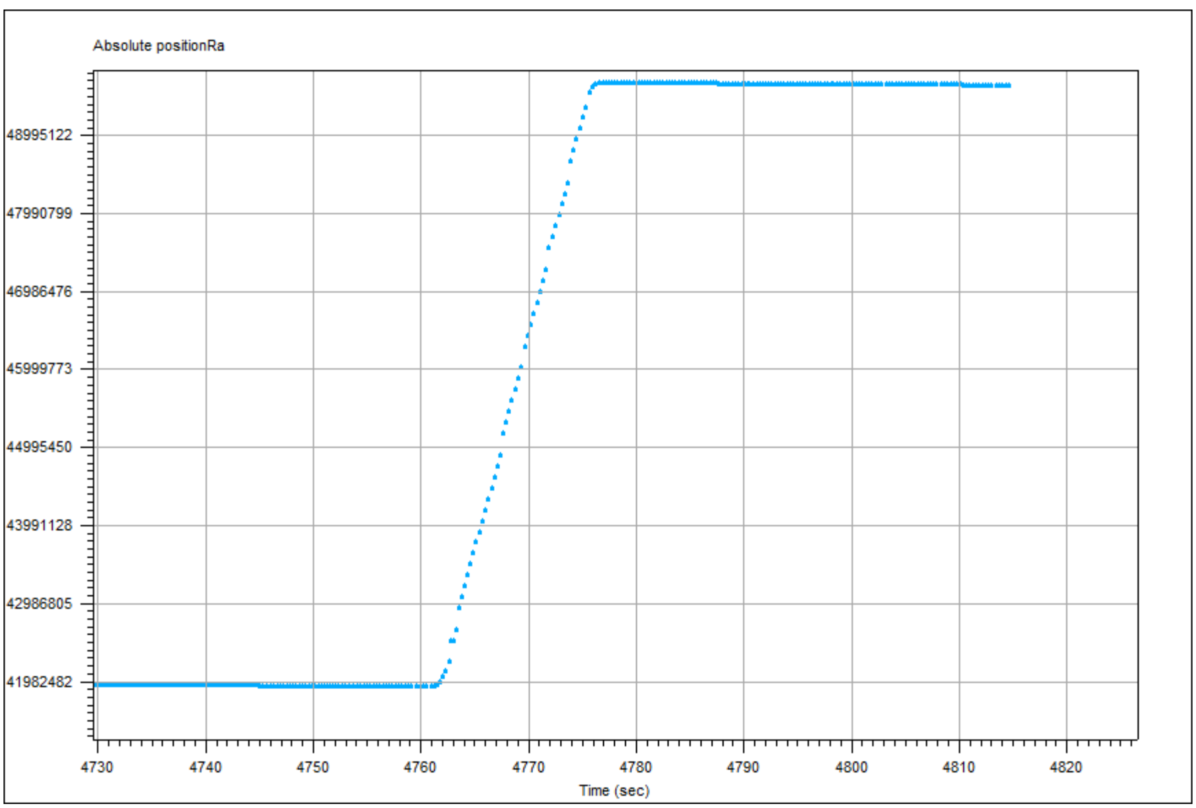


Fig. 27 *Position plot during a small span RA motion (encoder units)*

The tracking error plot (below) shows the difference between the axis actual position and the computed position. The error increase to 450 steps (450 x 0.019 arcsec = 8.7 arcsec) during the acceleration phase and is between +/- 100 steps (+/- 1.9 arcsec) when the speed is set to 6°/sec.

when the motion completes the errors goes down to +/- 10 steps in 3 secs. This a good performance and PID parameters are set properly. The tracking error is larger when speed is 500 000 steps/sec, sideral speed (before and after RA motion) is 1700 steps per second.

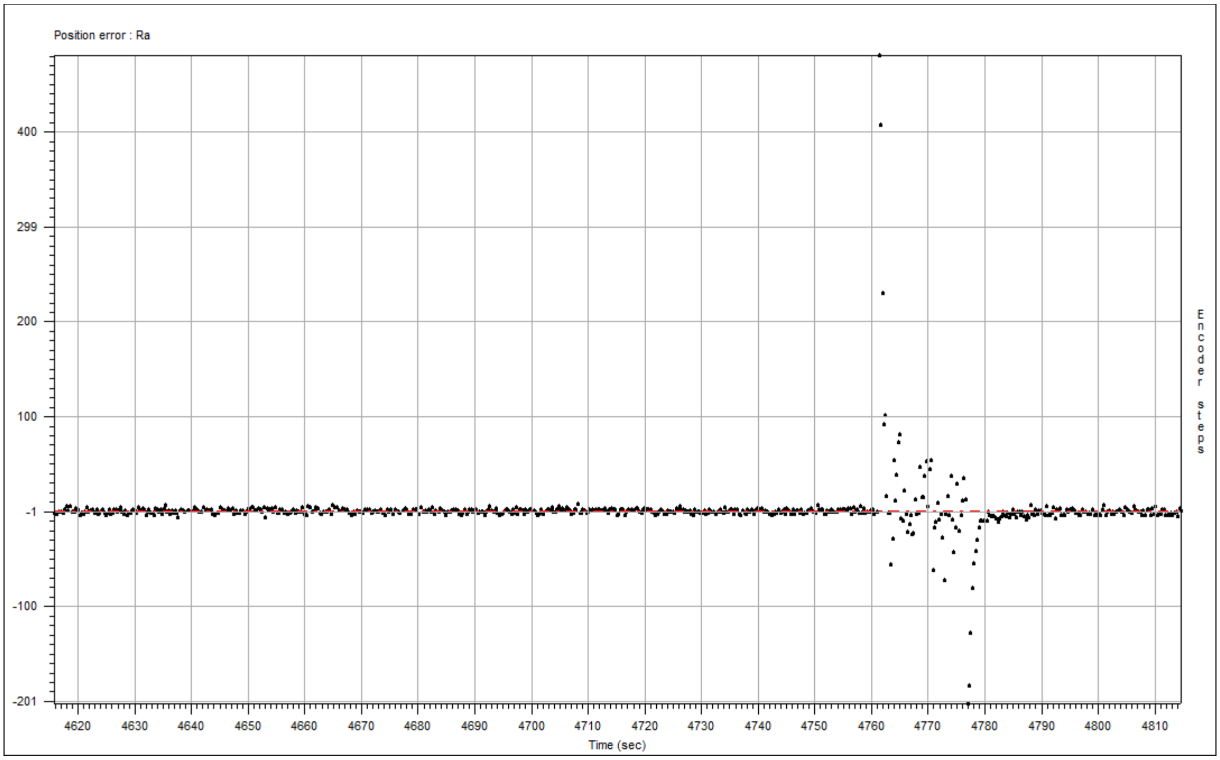


Fig. 28 *Tracking error position plot during a small span RA motion (encoder units)*

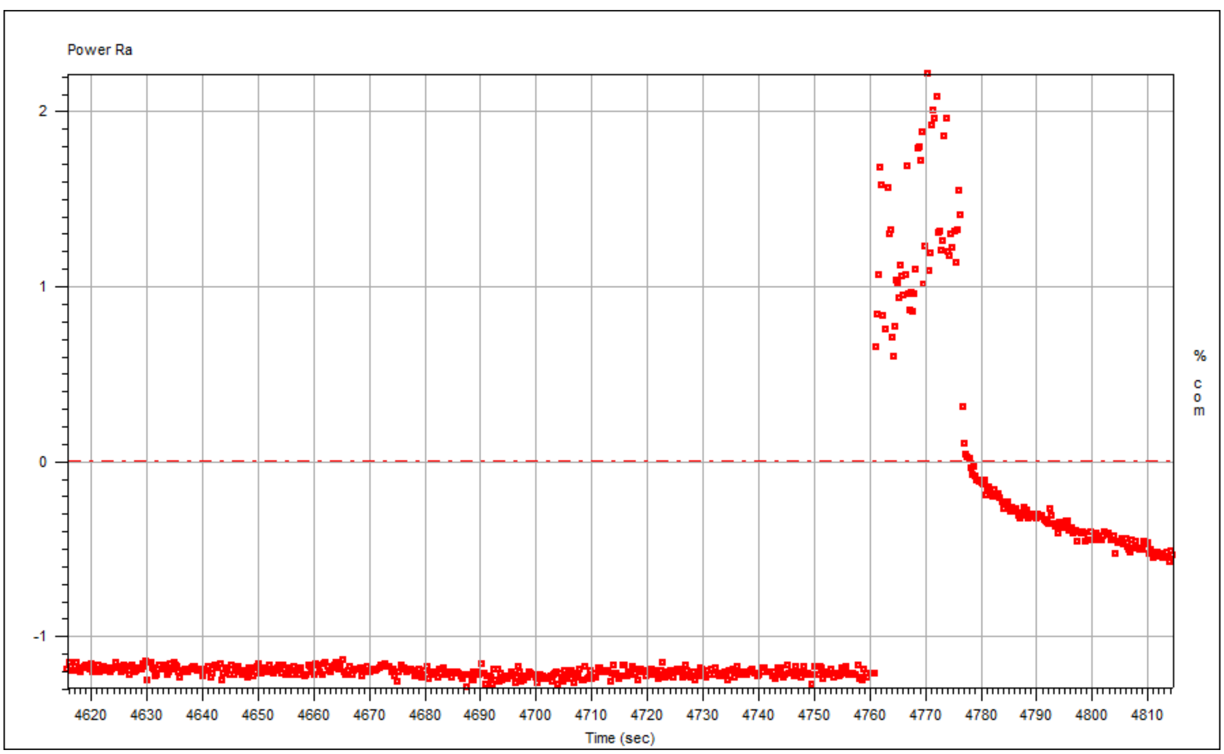


Fig. 29 *Electric Current sent to the motor (% units)*

The current level used to perform this motion, started at -1.2% then was raised to 2% and went down to -0.5%
The axis as no load, except the inertia momentum of the DEC axis. The RA axis is well balanced also.

Note that the current is expressed in %, 100 % of the current mean that the controller draws the maximum current (Amperes) that the motor with given winding resistance R, and given VBUS can withstand.
VBUS is the DC voltage which is applied to the controller. The controller has an AC mains to DC converter producing 24V, 36V, 48V or 52V, depending on the mount type.

For instance, if the motor winding resistance is 4 Ohms and VBUS is 36 V the maximum current the controller can supply at 100% is 36 V / 4 Ohms that is 9 Amps. So 1% is 90 mA.
Actual current values are not so relevant for PID tuning.

Here is the flow chart of position PID in the controller.

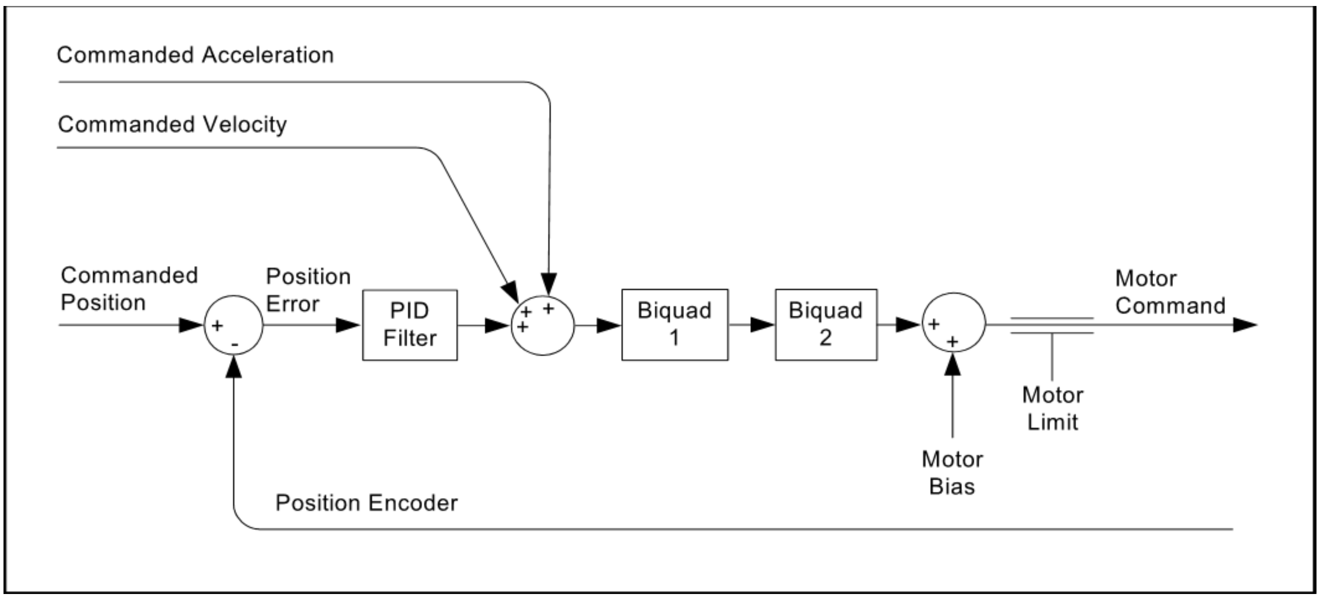


Fig. 30 *Position PID flow chart*

And more accurately the detailed “PID Filter” box flow chart. It looks complex, but is quite typical for PID control.

The information below is given to understand fully how it was implemented, but is not required for tuning the mount.

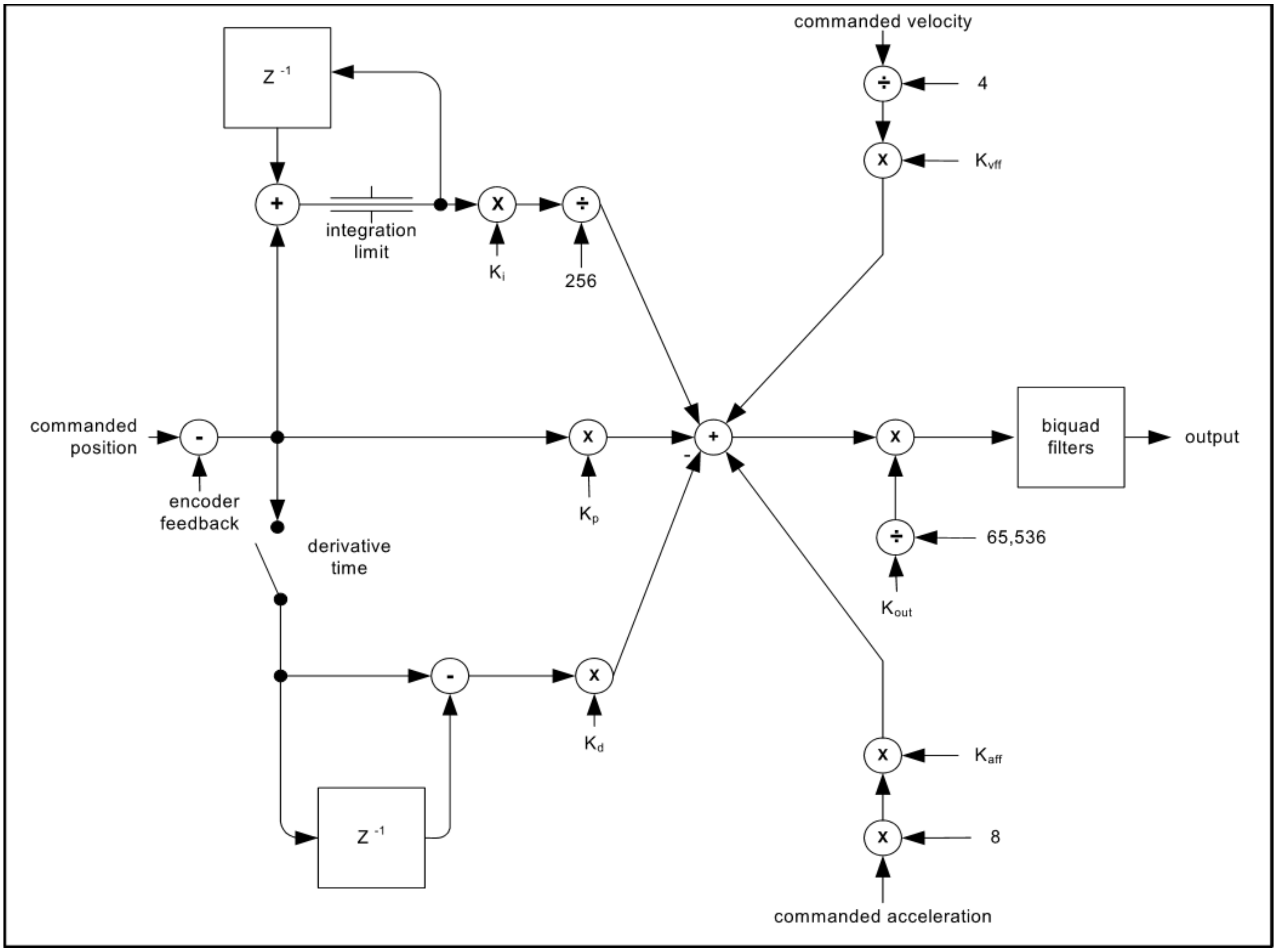


Fig. 31 *Detailed PID filter*

Please note that P is K_p, D is K_d, I is K_i, Derivative time is Dt.

The formula that provides the output level versus the position error E_j yield as follow

$$Output_n = \left[K_p E_n + K_d (E_k - E_{k-1}) + \sum_{j=0}^{n} E_j \times \frac{K_i}{256} + K_{vff}\left(\frac{CmdVel}{4}\right) + K_{aff}(CmdAccel \times 8) \right] \times \frac{K_{out}}{65{,}536}$$

k	= n – modulus (n/derivative time)
E_n	= position loop error at the derivative sampling interval (Commanded Position – Actual Position)
E_k	= position loop error at the derivative sampling interval
K_i	= Integral Gain
K_d	= Derivative Gain
K_p	= Proportional Gain
K_{aff}	= Acceleration feed-forward
K_{vff}	= Velocity feed-forward
K_{out}	= scale factor for the output command

The PID parameters are normalized this way. DDR_ASTRO deals with unitless PID parameters.

Term	Name	Representation & Range
Ilimit	Integration Limit	unsigned 32 bits (0 to 2,147,483,647)
K_i	Integral Gain	unsigned 16 bits (0 to 32,767)
K_d	Derivative Gain	unsigned 16 bits (0 to 32,767)
K_p	Proportional Gain	unsigned 16 bits (0 to 32,767)
K_{aff}	Acceleration feed-forward	unsigned 16 bits (0 to 32,767)
K_{vff}	Velocity feed-forward	unsigned 16 bits (0 to 32,767)
K_{out}	Output scale factor	unsigned 16 bits (0 to 32,767)
DerivativeTime	Derivative Sampling Time	unsigned 16 bits (1 to 32,767)

4.3.3 Setting the best PID

The Ziegler Nichols method can be used like described below

https://en.wikipedia.org/wiki/Ziegler-Nichols_method

But this can be tedious to use. A simpler method will be described hereafter, called manual PID setting.

First, ensure this option is left unchecked.

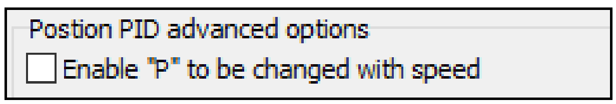


Fig. 32 *No P for slow speed*

Regarding current loop, NOVA 120 mount with no payload has been used, and current loop PI was set this way

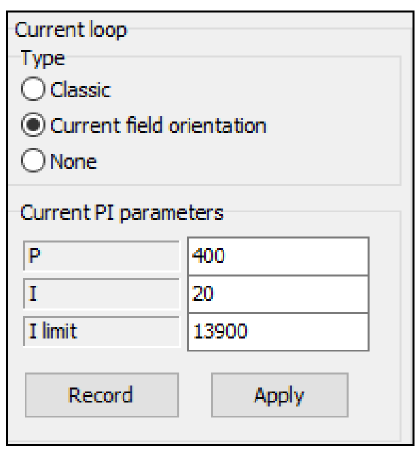


Fig. 33 *PI typical figures for NOVA 120 motors*

A simple step-by-step method can be used. Starting with a very low P and D (that go from 0 to 32767).

Let's fix Dt to 7, and I to 5 at first place. Then a small motion of 15° will be used to check that the axis can rotate smoothly, and check what it does at 6°/sec. K_{out} must be always set to less than 1%. K_{vf} and K_{af} are set to zero.

	P	D	I
Step #1	20	200	5

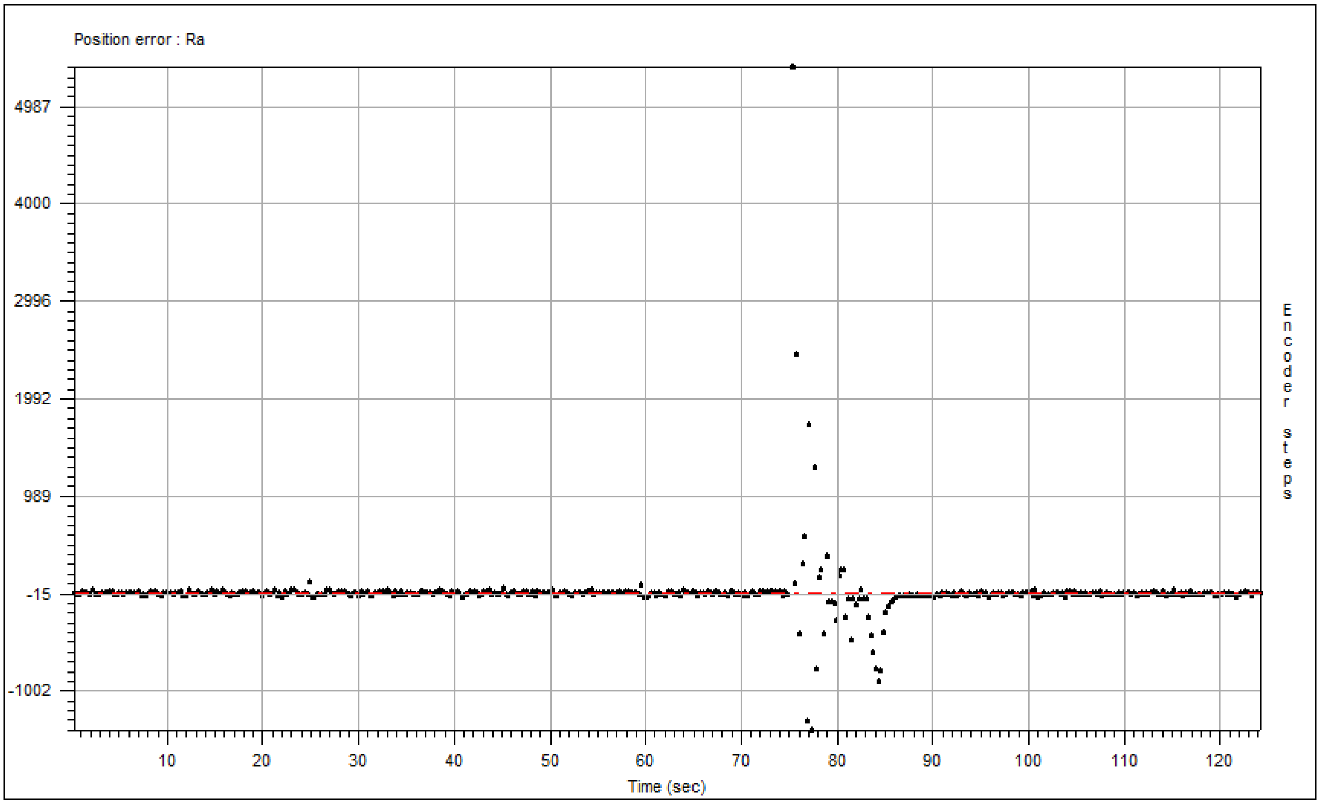


Fig. 34 *Tracking error for 45° span RA motion P=20 and D=200*

The error is quite large (more than 5000 encoder steps) meaning that P is not aggressive enough, especially when the axis starts to accelerate to reach cruise speed. Let's change P from 20 to 40 and see what is happening with a small axis movement.

	P	D	I
Step #2	40	200	5

When the axis moves, the position error oscillate strongly and this oscillation exceeds the 100 000 steps.

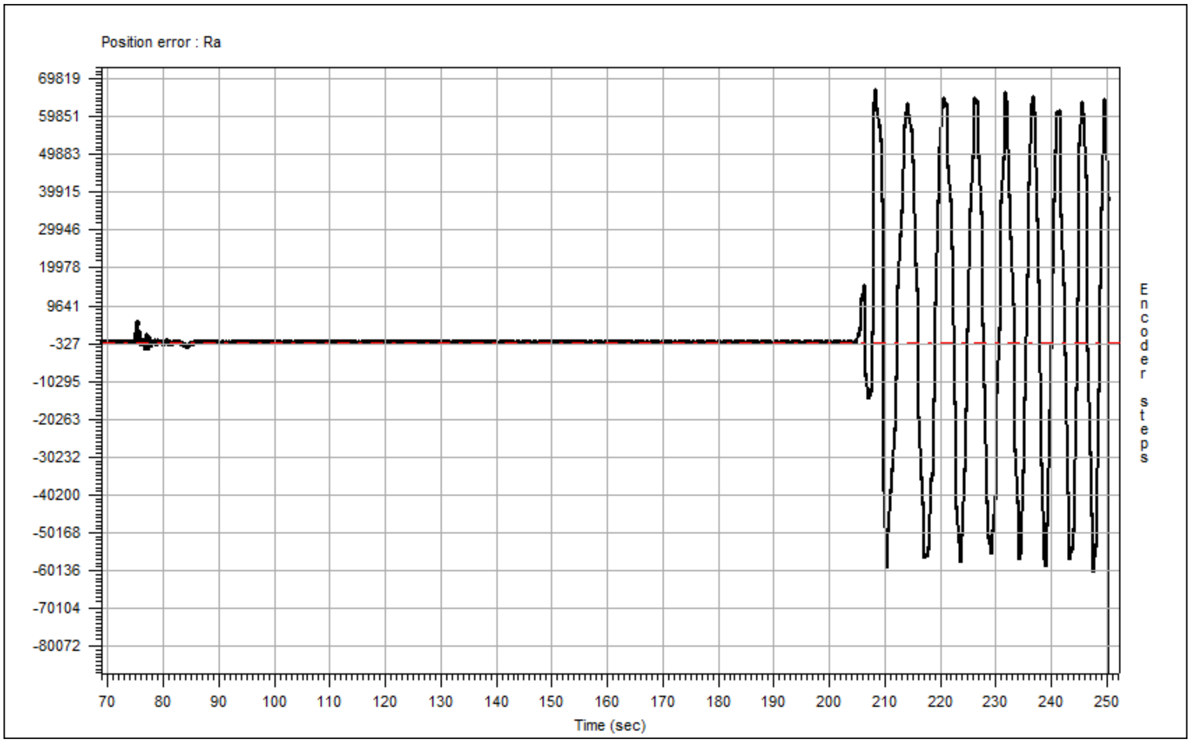


Fig. 35 *Tracking error for 45° span RA motion P=40 and D=200*

The controller automatically disables the position loop thanks to this setting, because it exceeds 100 000 encoder steps.

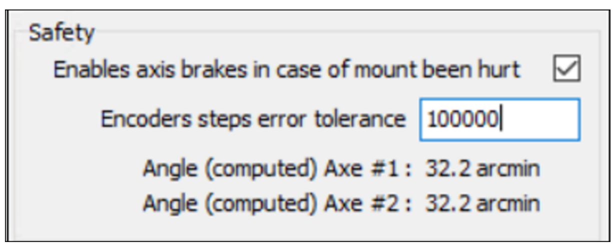


Fig. 36 *Maximum tracking allowable error*

This message will appear when the axis position loop is disabled.

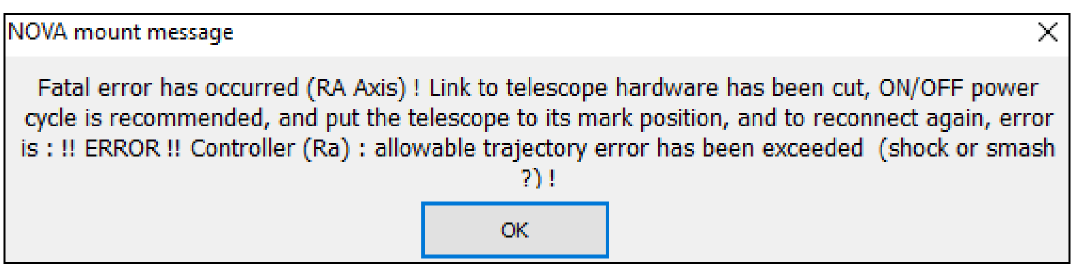


Fig. 37 *Error message when tracking error is too large*

To overcome this, D must be increased, the role of “D”, in PID loop is to dampen the effects of P, so that it does not oscillate. Let’s increase D to 400, and press “***Record***”, and “***Connect***”.

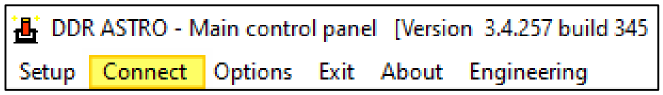


Fig. 38

Execute a small motion (30°)

	P	D	I
Step #3	40	400	5

The next plot show what track error position does during this motion, it still oscillates. If D is changed from 400 to 600 during the motion, it kills immediately the oscillations.

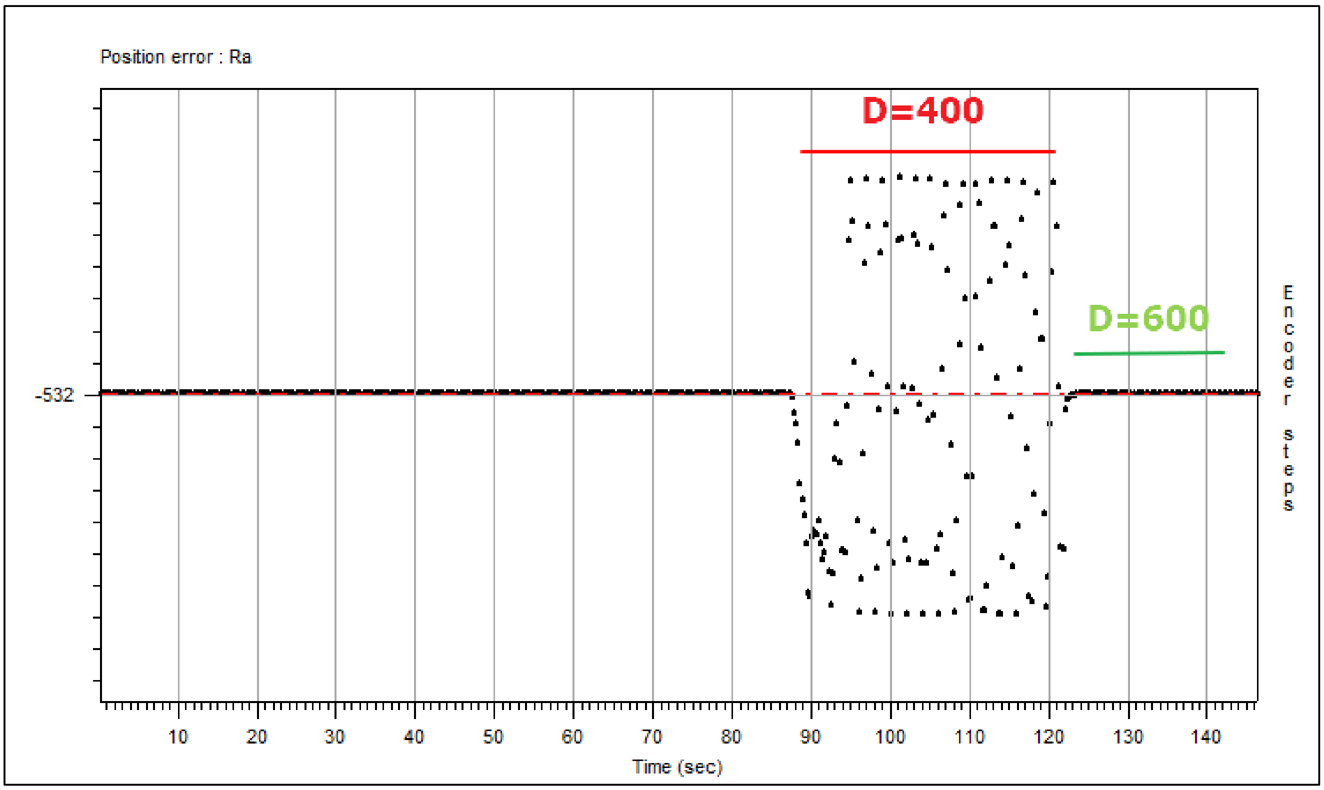


Fig. 39 *Tracking error for 45° span RA motion P=40 and D=400, and then D=600*

	P	D	I
Step #4	40	600	5

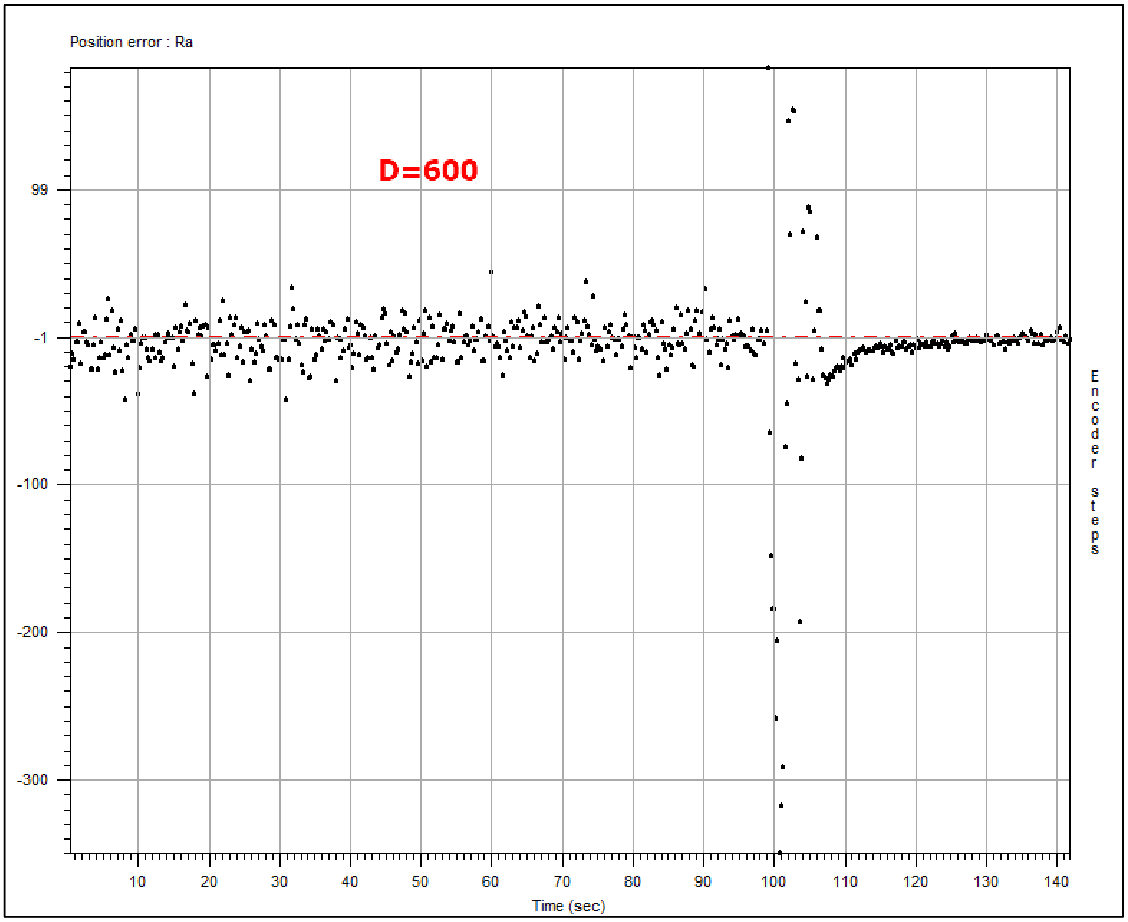


Fig. 40 *Tracking error for 45° span RA motion P=40 and D=600*

Then P can be increased (40 to 90) as the same time D is increased (600 to 1000)

	P	D	I
Step #5	90	1000	5

Do not forget to achieve RA motion to see what effect it does. **NEVER TRY TO TUNE PID** parameters with no speed applied to the axis. Always a small motion must be applied to see the PID changes effects.

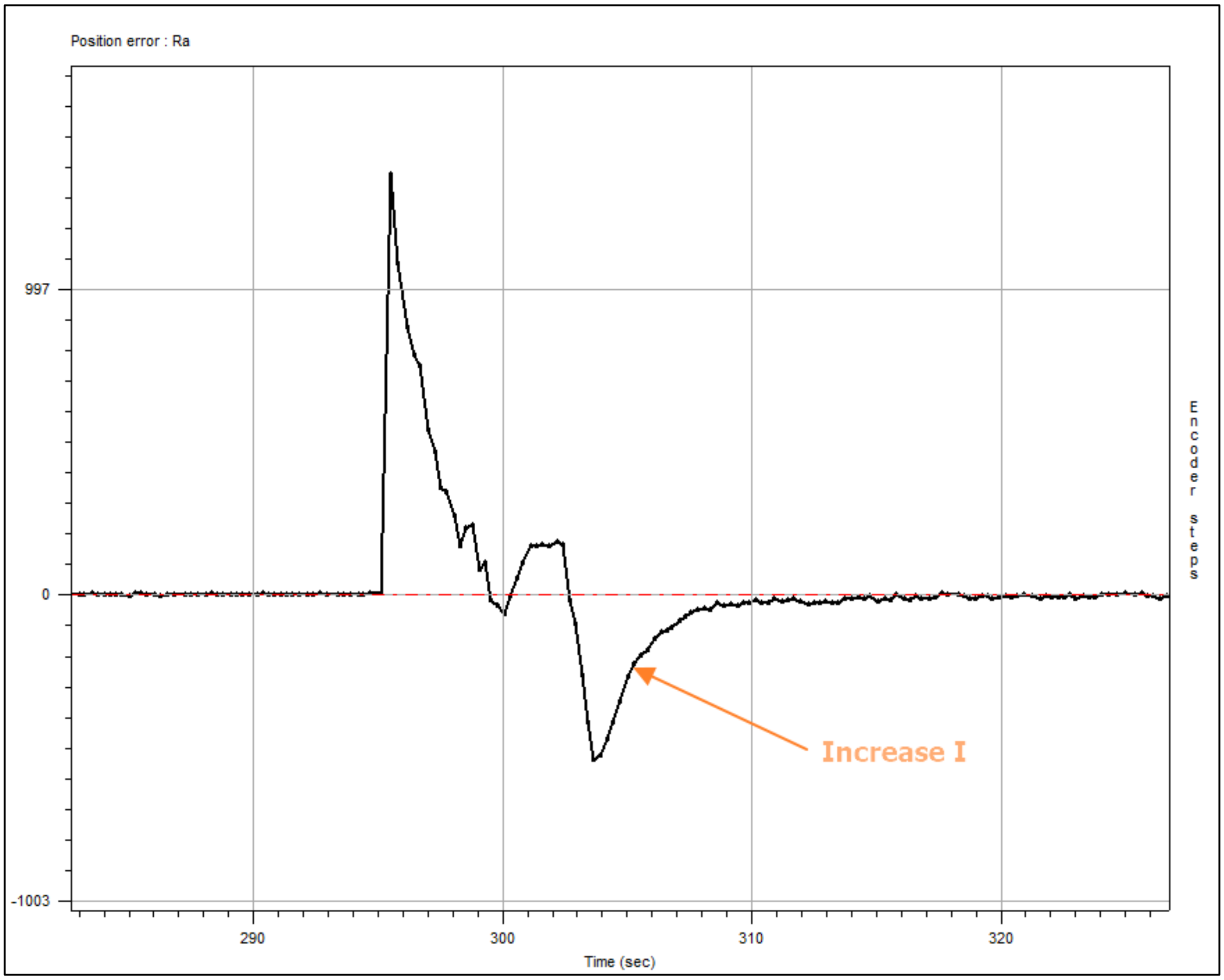


Fig. 41 *Tracking error for 45° span RA motion P=90 and D=1000*

With those parameters, the axis seems to work properly. The strategy was to start with a very small P and D and to increase them slowly.
Nevertheless, a long lag can be seen and when slew is completed, to reduce this lag increase I from 5 to 30

	P	D	I
Step #4	90	1000	30

The improvement of I increase is really significant and can be seen in the next plot.

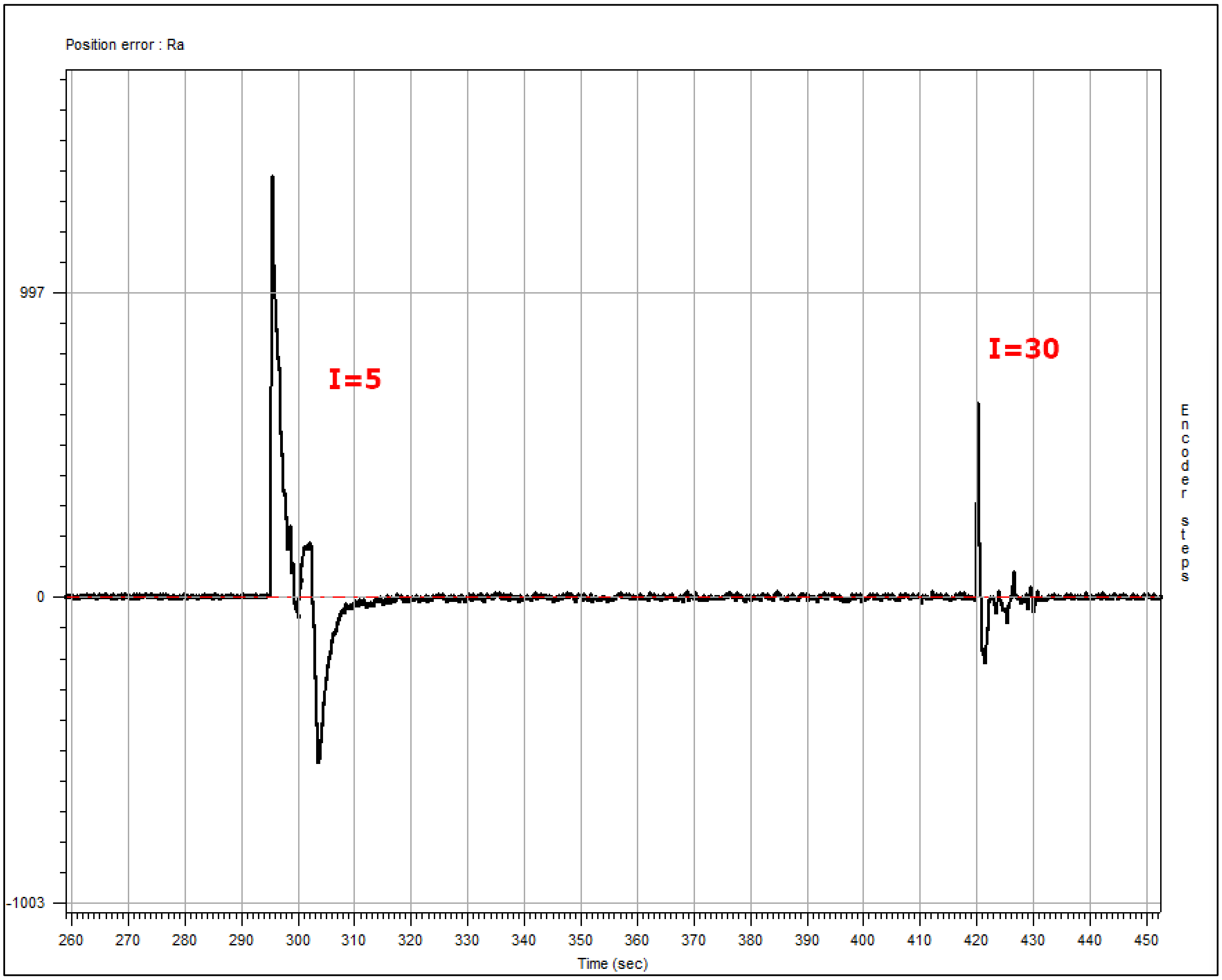


Fig. 42 *Tracking error for 45° span RA motion P=90 and D=1000 and I=5 then I=30*

After some trials, good parameters have been found. This might take an hour to do so, this is normal.

	P	D	I
Step #N	120	950	30

At the end of the process, the mount should move with no noise (no rumble, no whistle).

If P would be increased from 120 to say 200, D must be increased and a limit will show: the axis will oscillate, whistle and tracking error position will be large and immediate just after slew starts.

But try to put the P and D parameters as high as possible.
Nevertheless, if the mechanics (Mount and OTA) is not stiff enough and/or that the first eigenfrequency of the telescope structure is less than 10Hz, the PID will be low and soft, and high speed cannot be used (oscillations issues). For such structures, the maximum speed an acceleration must be reduced.

Then apply the same procedure to the DEC axis, since the DEC axis “sees” less inertia momentum, the P and D parameter might be lower.

A simple method can be used to check that the axis is stable, is to use a Teflon hammer, and hit one gently of the two axes.
This cause a perturbation in the system that should be recovered as fast as possible. If the axis starts to oscillate, the D parameter is too low.

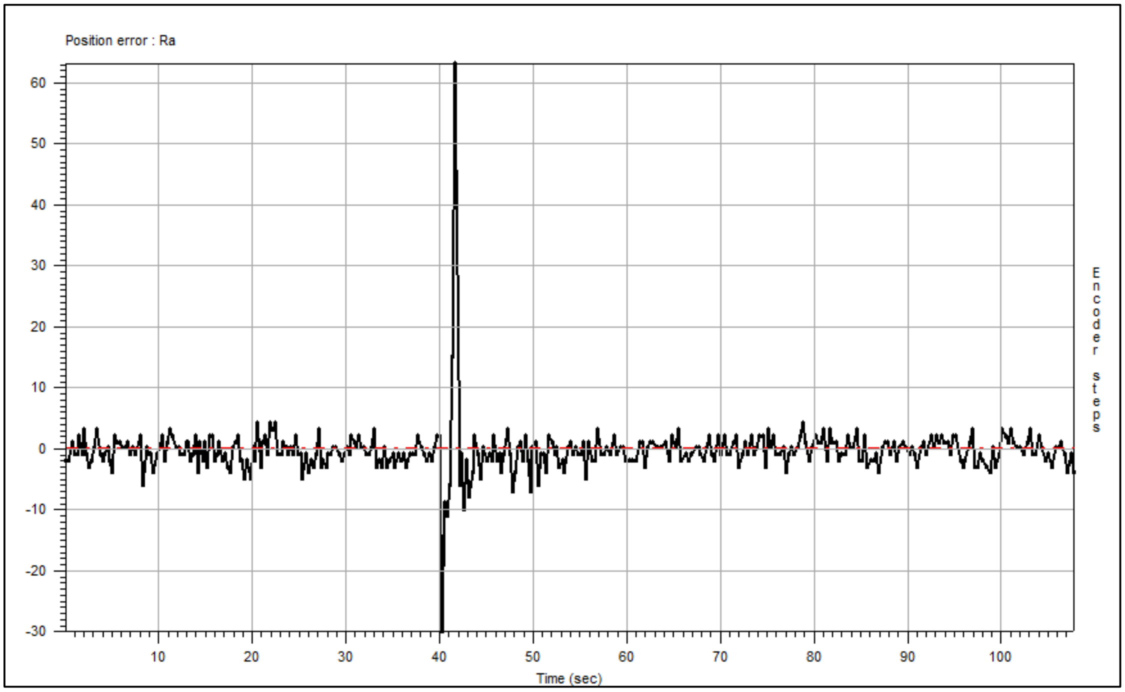


Fig. 43 *Tracking error when RA axis got a small hit with Teflon hammer*

4.3.4 Slow speed PID control

At low speeds (i.e tracking speeds, or fixed position), the tracking error might be too still high (more than +/- 15 encoders steps).
When the speed is below a user input threshold (say twice the sideral speed), the software will apply the new P and D parameter, in this case 500 and 3000. Normal slew P and D are set to 120 and 950 respectively.
Do not input too high P and D figures, because this can cause errors at the end of slewing, when transition between regular P and D figures and slow speed P and D figure occurs. Do not forget to check the enable “P & D” box.

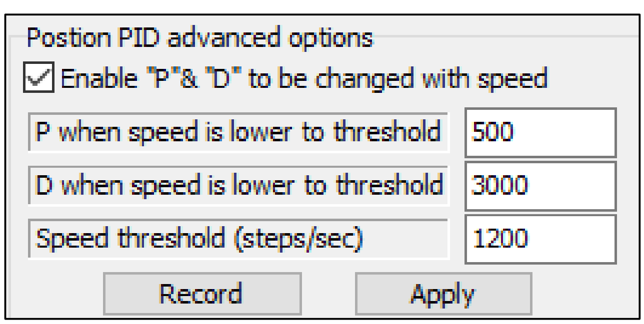


Fig. 44 *PID advanced options*

The next plot show at t=1510sec, when the "***Enable "P" & "D" to be changed with speed***" has been unchecked, the STD tracking error changed from 2.5 encoder steps to 4.8 encoder steps.

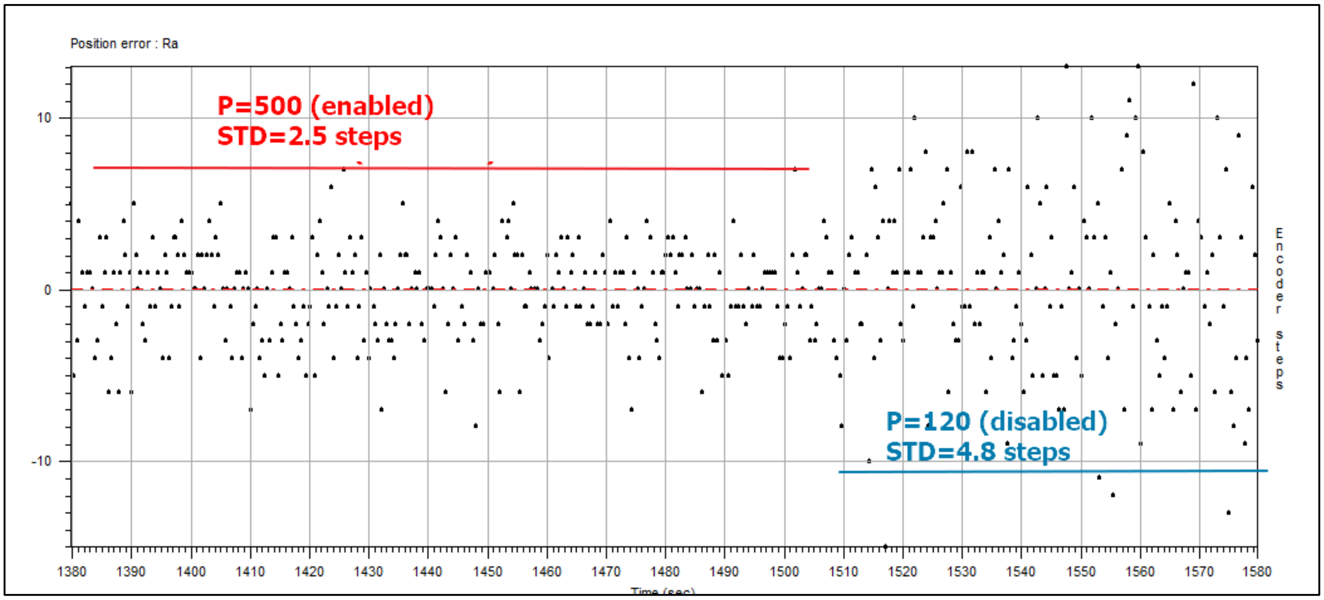


Fig. 45 *PID advanced options effects on sideral tracking error*

4.3.5 Typical PID figures for different mounts

4.3.5.1 NOVA 120 mounts incremental encoders (last sold 2018)

Power supply is 24V

Load: Newton 400 mm OTA

Current PI for RA axis			Current PI for DEC Axis		
P	I	Ilimit	P	I	Ilimit
7500	130	13900	8000	130	13900

Position PID for RA Axis					Position PID for DEC Axis				
P	I	D	Dt	Kout	P	I	D	Dt	Kout
800	9	175	50	1.8	900	9	175	50	1.5

4.3.5.2 NOVA 200 mounts incremental encoders (last sold 2018)

Power supply is 48V - 3000W controller

Load: Officina Stellare 600mm RC OTA

Current PI for RA axis			Current PI for DEC Axis		
P	I	Ilimit	P	I	Ilimit
3000	200	13900	1000	200	13900

Position PID for RA Axis					Position PID for DEC Axis				
P	I	D	Dt	Kout	P	I	D	Dt	Kout
700	4	700	20	1.0	600	4	300	20	1.0

4.3.5.3 NOVA 120 mounts, absolute encoders

Power supply is 36V - 500W controller

Load: None

Current PI for RA axis			Current PI for DEC Axis		
P	I	Ilimit	P	I	Ilimit
400	20	13900	400	20	13900

Position PID for RA Axis					Position PID for DEC Axis				
P	I	D	Dt	Kout	P	I	D	Dt	Kout
120	30	1000	7	0.5	20	20	150	7	0.5

Power supply is 36V - 500W controller

Load: RASA 14 (and misc. instruments) -> 36 cm OTA

Current PI for RA axis			Current PI for DEC Axis		
P	I	Ilimit	P	I	Ilimit
2500	100	32767	3500	100	32767

Position PID for RA Axis					Position PID for DEC Axis				
P	I	D	Dt	Kout	P	I	D	Dt	Kout
900	50	150	100	0.2	90	15	250	7	0.5

Power supply is 48V - 500W controller

Load: RC 400 mm OTA (and misc. instruments)

Current PI for RA axis			Current PI for DEC Axis		
P	I	Ilimit	P	I	Ilimit

Position PID for RA Axis					Position PID for DEC Axis				
P	I	D	Dt	Kout	P	I	D	Dt	Kout

4.3.6 Final checks

When PID mount parameters are found to be suitable, <u>on BOTH axis</u>, a “stress test” must be performed. It will slew the mount throughout the sky, in random places forever. This will ensure that all motions are carried out till the end, and no error occurs. In the “***Options/Random slewing (stress test)***”

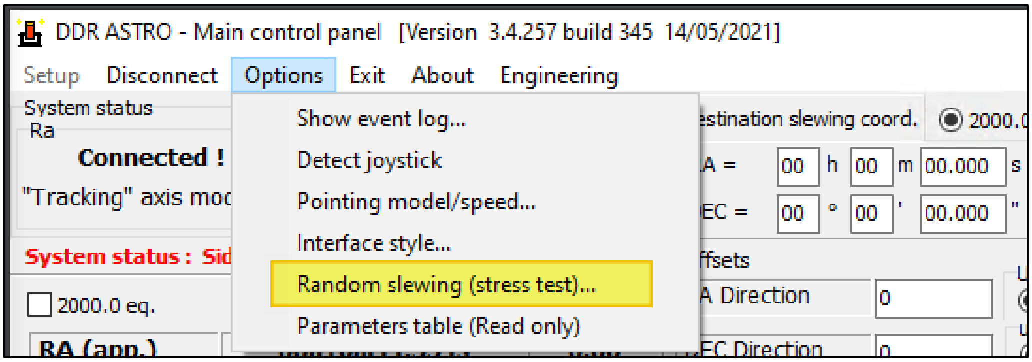


Fig. 46

This form shows up and allow the user to input these parameters

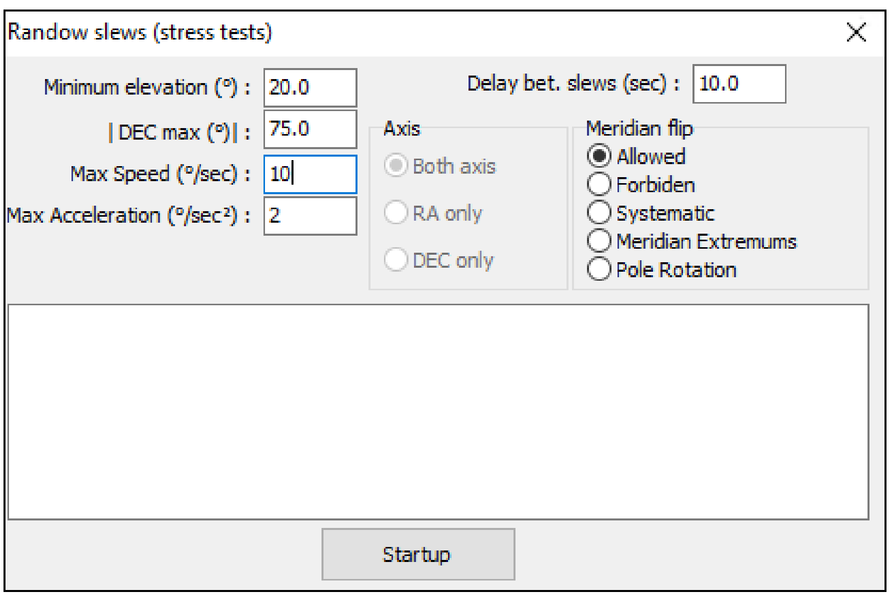


Fig. 47 *Stress test input panel*

The mount will move toward all places above the horizon, meridian flip is allowed, destinations must have elevations more than 20°, and DEC must be lower than absolute value of DEC=70° (to be compliant with southern hemisphere mounts). Speed will be 10°/sec of speed and acceleration will be 2°/sec². Be sure the OTA is free to move all around this space, and cannot hit something.

Then the mount will start to slew, and wait for 10 sec, till the next destination will be used. This runs forever and the user need to hit the "***Stop***" button to cancel this process.

When the number of slews exceed more than one hundred, and that no error occurred, the mount PID for both axes can be regarded as suitable. If an error occurred, the PID parameters needs to be re-worked to ensure more reliability.

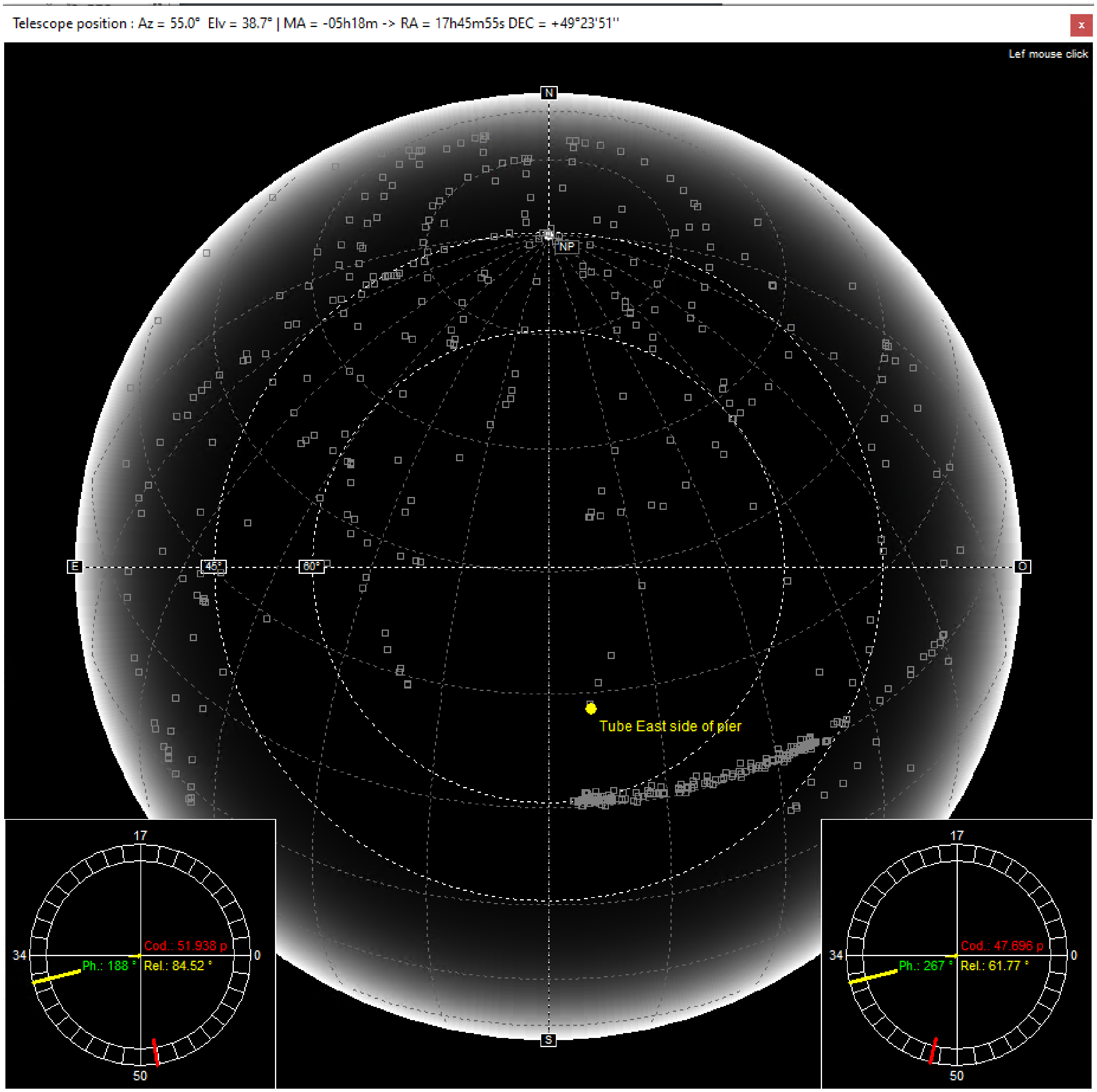


Fig. 48 *All motions plotted in sky map*

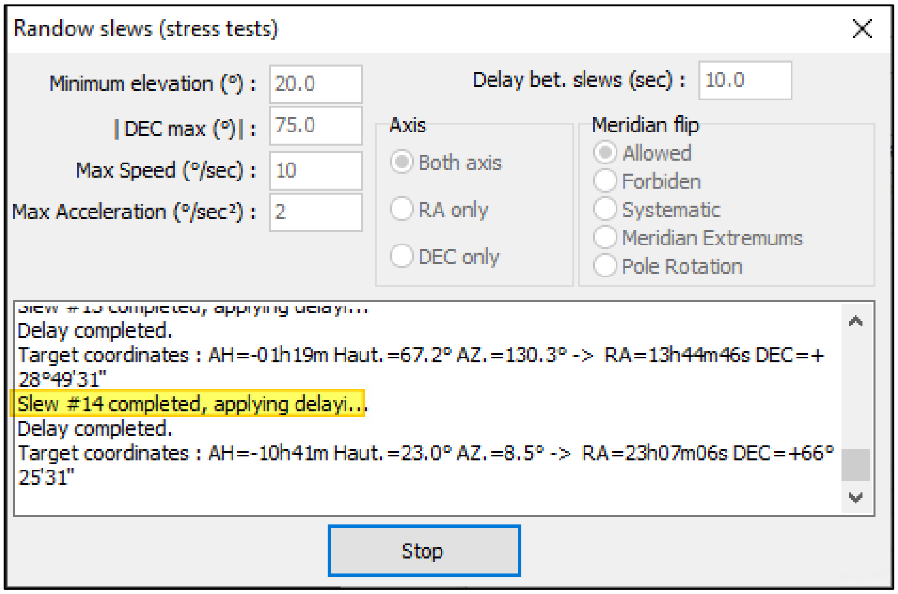


Fig. 49 *Stress test output panel*

4.4 ALTAZ MOUNT and position PID

4.4.1 Principle

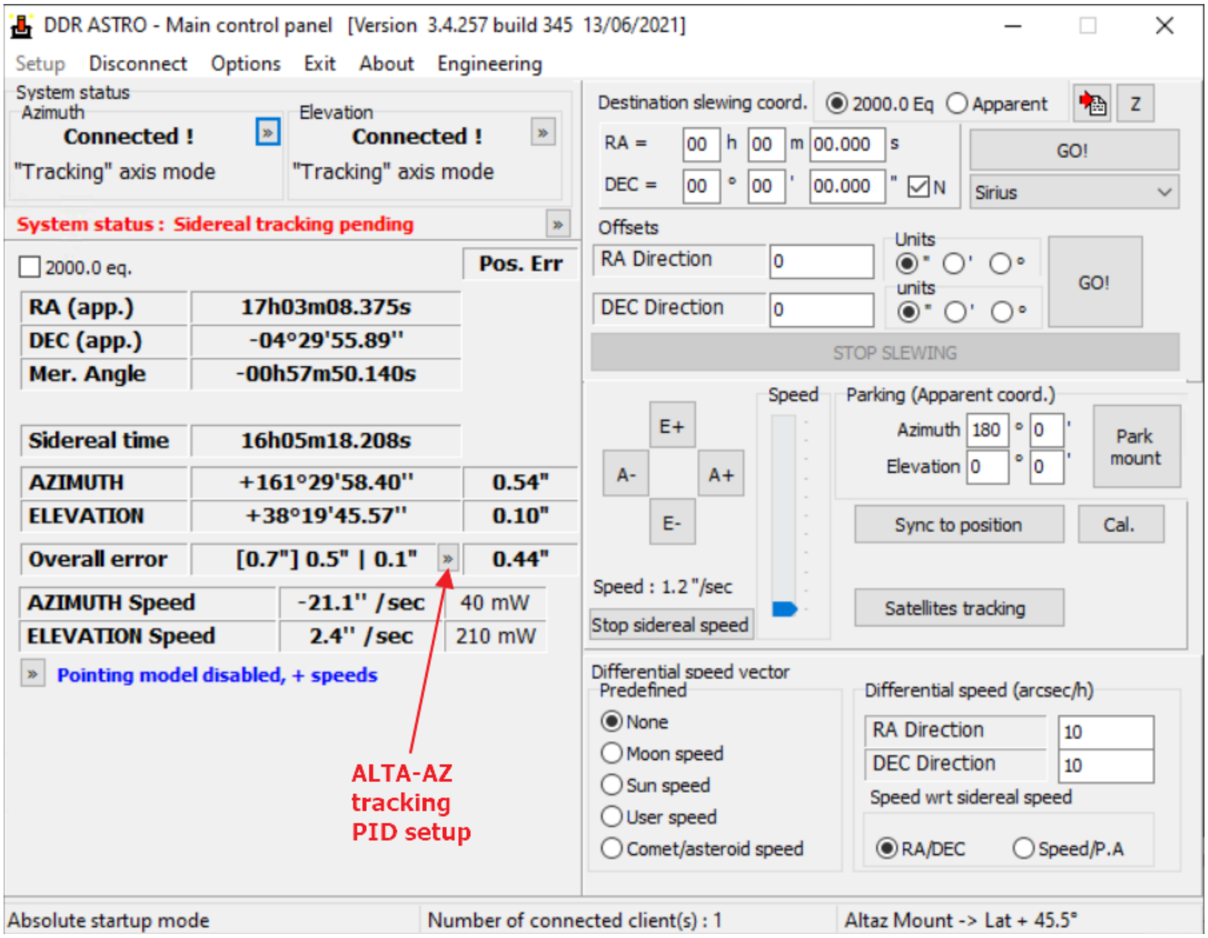
ALT-AZ tracking uses variables speeds from both axis at the same time. Those speeds are variable from zero to a large value for ALT-AZ axis (depending of the zenithal distance when passing the meridian) and ELV speed are also variable.

In order to get the best tracking, the DDR_Astro software performs these changes of speeds at a rate of 10Hz and correct the speeds according to the computed position of the object at any time.

This correction uses, again, a PID algorithm, and it determines the time to reach stability just after slew completion. This PID parameters can be set by the user.

4.4.2 Tuning

This correction uses, again, a PID algorithm, and it determines the time to reach stability just after


DDR ASTRO - Main control panel [Version 3.4.257 build 345 13/06/2021]
Setup Disconnect Options Exit About Engineering
System status
Azimuth
Connected !
"Tracking" axis mode
Elevation
Connected !
"Tracking" axis mode
System status : Sidereal tracking pending
2000.0 eq.
Pos. Err
RA (app.) 17h03m08.375s
DEC (app.) -04°29'55.89"
Mer. Angle -00h57m50.140s
Sidereal time 16h05m18.208s
AZIMUTH +161°29'58.40" 0.54"
ELEVATION +38°19'45.57" 0.10"
Overall error [0.7"] 0.5" | 0.1" 0.44"
AZIMUTH Speed -21.1" /sec 40 mW
ELEVATION Speed 2.4" /sec 210 mW
Pointing model disabled, + speeds
ALTA-AZ tracking PID setup
Destination slewing coord. 2000.0 Eq Apparent
RA = 00 h 00 m 00.000 s
DEC = 00 ° 00 ' 00.000 " N
GO!
Sirius
Offsets
RA Direction 0
DEC Direction 0
Units
units
GO!
STOP SLEWING
Speed
E+
A-
A+
E-
Speed : 1.2 "/sec
Stop sidereal speed
Parking (Apparent coord.)
Azimuth 180 ° 0 '
Elevation 0 ° 0 '
Park mount
Sync to position
Cal.
Satellites tracking
Differential speed vector
Predefined
None
Moon speed
Sun speed
User speed
Comet/asteroid speed
Differential speed (arcsec/h)
RA Direction 10
DEC Direction 10
Speed wrt sidereal speed
RA/DEC
Speed/P.A
Absolute startup mode
Number of connected client(s) : 1
Altaz Mount -> Lat + 45.5°

---oOo---