Application Note

MPiec and Sigma-5: Recommended Tuning Method for Minimizing Dynamic Position Error with Aggressive Move Profiles

Applicable Product: MPiec, Sigma-5

Yaskawa Electric America
2121 Norman Drive South
Waukegan, IL 60085
1-800-927-5292

Application Overview

The Sigma-5 system has very high performance capabilities. High tuning performances can now be achieved through the advanced autotuning function on the Sigma-5 system. Advanced autotuning can be performed in a matter of minutes to tune the servo system based on various objectives like accurate positioning, overshoot minimization and speed of response. This function in the Sigma-5 is flexible and easy to use for even first time users. 95% of application performance specifications can be met using the advanced autotuning feature of the Sigma-5. Yaskawa's advanced autotuning is proven to be best in class for positioning applications (point to point moves with minimal overshoot). This is achieved because of the high bandwidth available and with the help of vibration suppression and anti resonance algorithms built into the tuning algorithm aided by the high resolution feedback from the encoder. This document describes tuning methods to further reduce dynamic position error by tracking the reference position command when used in conjunction with the MPiec series controller for difficult applications.

Application Highlights:

Industry:	Rigid machines with aggressive move profiles
Major Features:	Very low dynamic position error
Results:	→ Maximum dynamic position error = 0.001 degrees with 5400 rpm move and 1.3:1 inertia mismatch → Maximum dynamic position error = 0.003 degrees with 2400 rpm move and 20:1 inertia mismatch

Application Challenges:

- Sigma-5 advanced autotuning guarantees good performance for positioning applications specifically focusing on overshoot minimization and vibration suppression. For high speed, high inertia mismatch applications demanding low position following error throughout the move profile, manual intervention is required. Moreover, this tuning has to be performed such that a symbiotic relation with gains from the MPiec controller is established.
- Challenge is to bring down position following error on the following two systems
  Tuning challenge 1 → 1.3:1 inertia system running at 5400 rpm
  Tuning challenge 2 → 20:1 inertia system running at 2400 rpm

Products Used:

Component	Product and Model Number
Servopack	SGDVR90F11A
Motor	SGMAV-01A3A61 (directly coupled to load)
Controller	MPiec
Software	MotionWorks IEC and SigmaWin+

Tuning Challenge # 1

Move Profile:

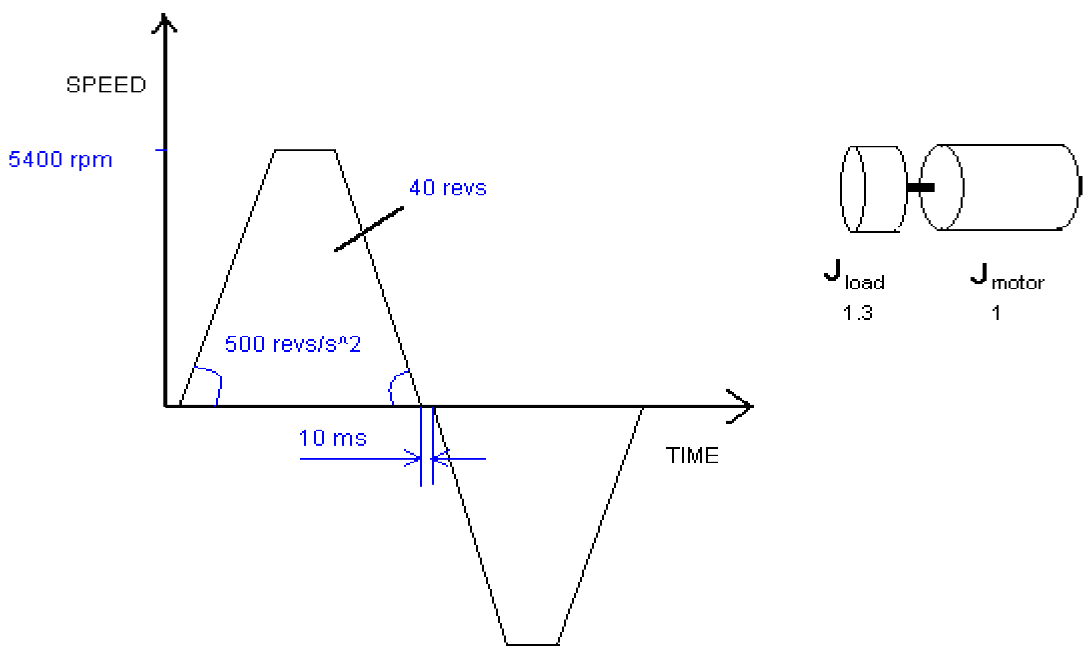


Figure 1: Test 1 Move profile

Inertia Mismatch: 1.3:1

Mechanical coupling: Directly coupled load

Hardware used:

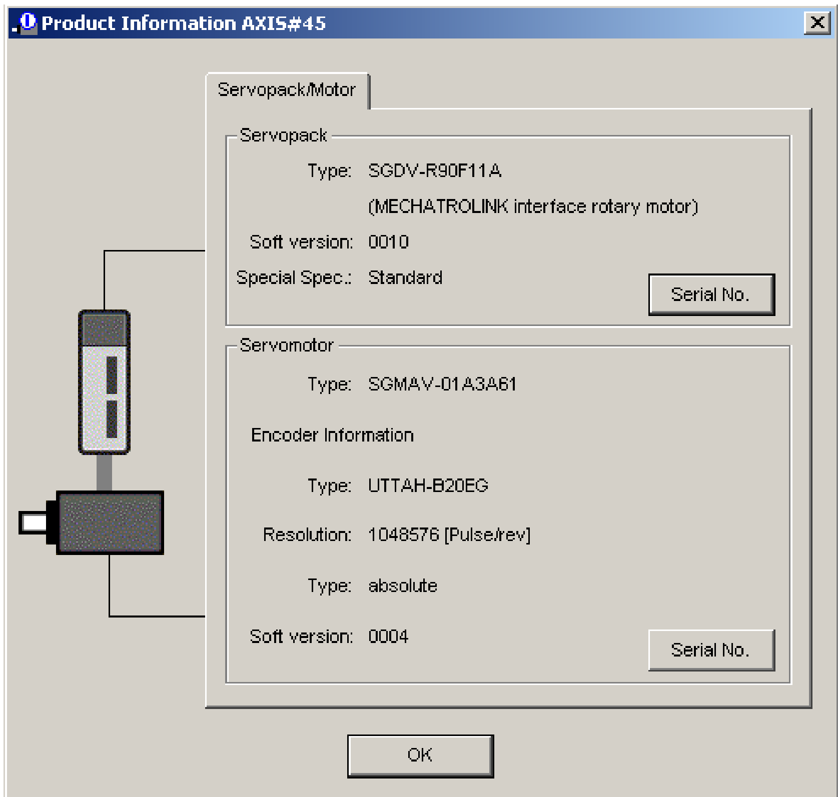


Figure 2: Hardware used

Objective: To bring down maximum position error under 0.01 degrees

Tuning Method:

Step 1: Turn tuneless mode off (Pn170.0 = 0)

Start autotuning with no reference input. To get started with autotuning, Pn170.0 (Tuneless functionality) will have to be turned off. SigmaWin+ prompts this when the user starts autotuning. Performing one round of autotuning with no reference input (using the reference from the servopack internally) is good practice. This will help set a parameter baseline for more advanced tuning like controller reference tuning, one touch tuning and manual tuning.

Step 2: Start Autotuning with no reference input

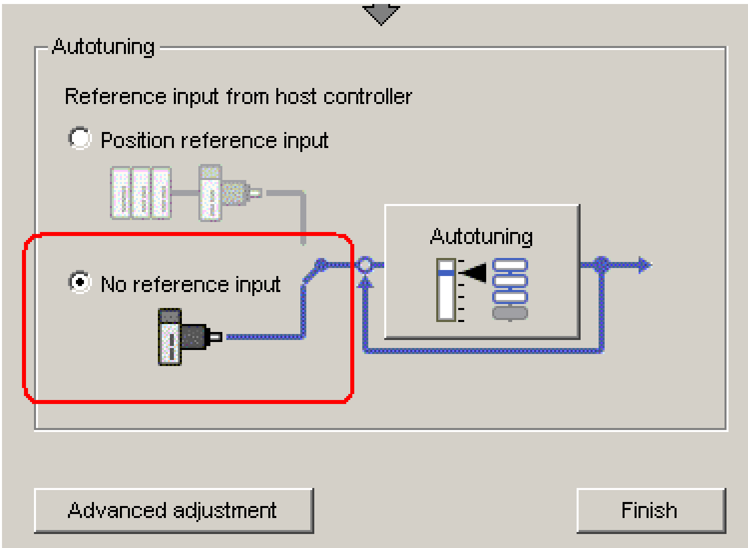


Figure 3: Start tuning with 'No Reference Input'

a) The very first time a system is tuned, it is recommended to choose 0 for switching inertia identification. This selection will set the tuning process to start off by calculating the load inertia automatically. (0 = moment of inertia is presumed). This selection forces the autotuning procedure to calculate the moment of inertia automatically and store it in Pn103.

b) Choose 'for positioning' for mode to tune for position following error.

c) It is recommended to select a distance close to the distance the axis will move while in production mode. The default setting may be short for exhaustive tuning. Once the number of revolutions allowable for tuning is chosen, use the calculation shown below to enter the distance.
(# of allowable rotations x 1048576) / (4 x 1000) = entry for distance
Distance entry that allows 10.6 revolutions for tuning is shown in Figure 4.

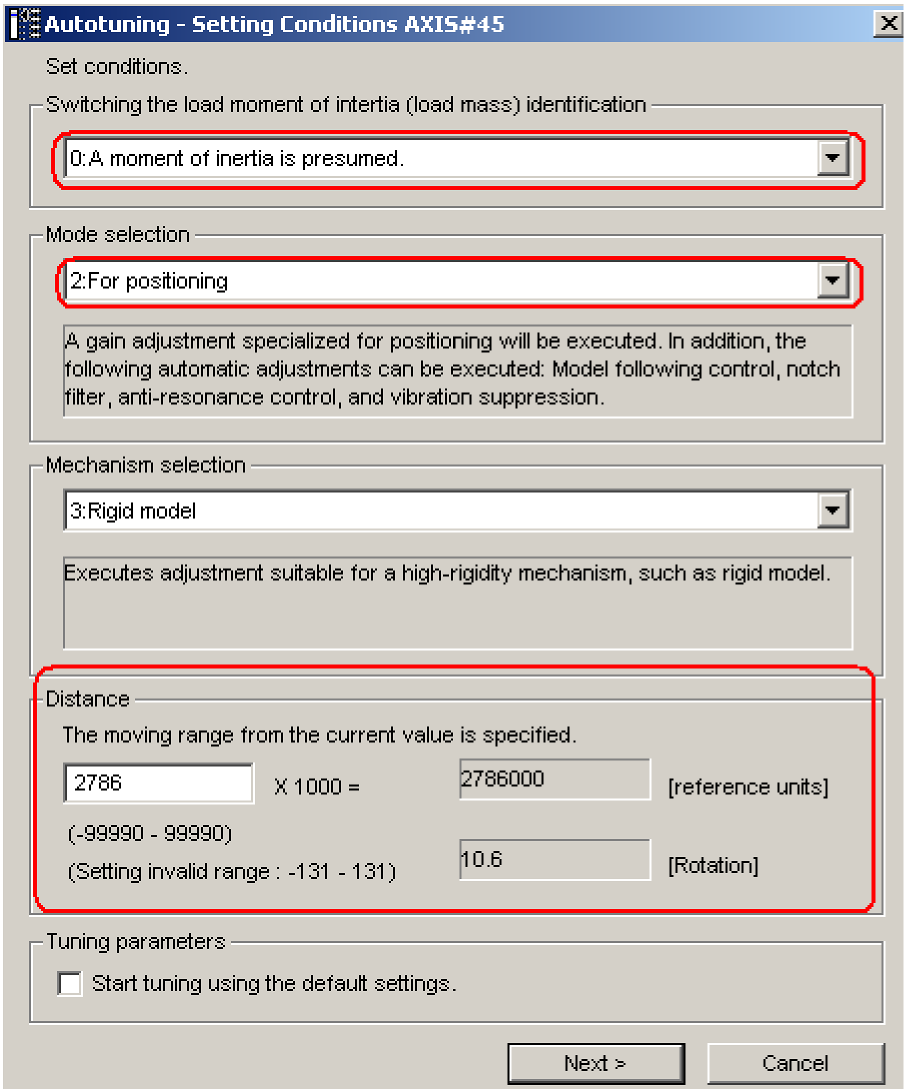


Figure 4: Autotuning setting conditions

Step 3: Start autotuning

'Servo on' and start tuning.

After tuning is complete, finish the tuning application by clicking 'finish' and close the tuning application windows. On checking the parameter list after autotuning is complete, it may be seen that setting for Pn140 = 0101. Pn140.0 = 1 means that model following control is enabled. Model following algorithm is used by the servopack to track the command reference as close as possible. The model following control sets a feedforward velocity and feedforward torque in the servopack through a model following gain (in Pn141) that is calculated during the tuning process. The positioning of the model following control in the servopack is shown in Figure 5 below.

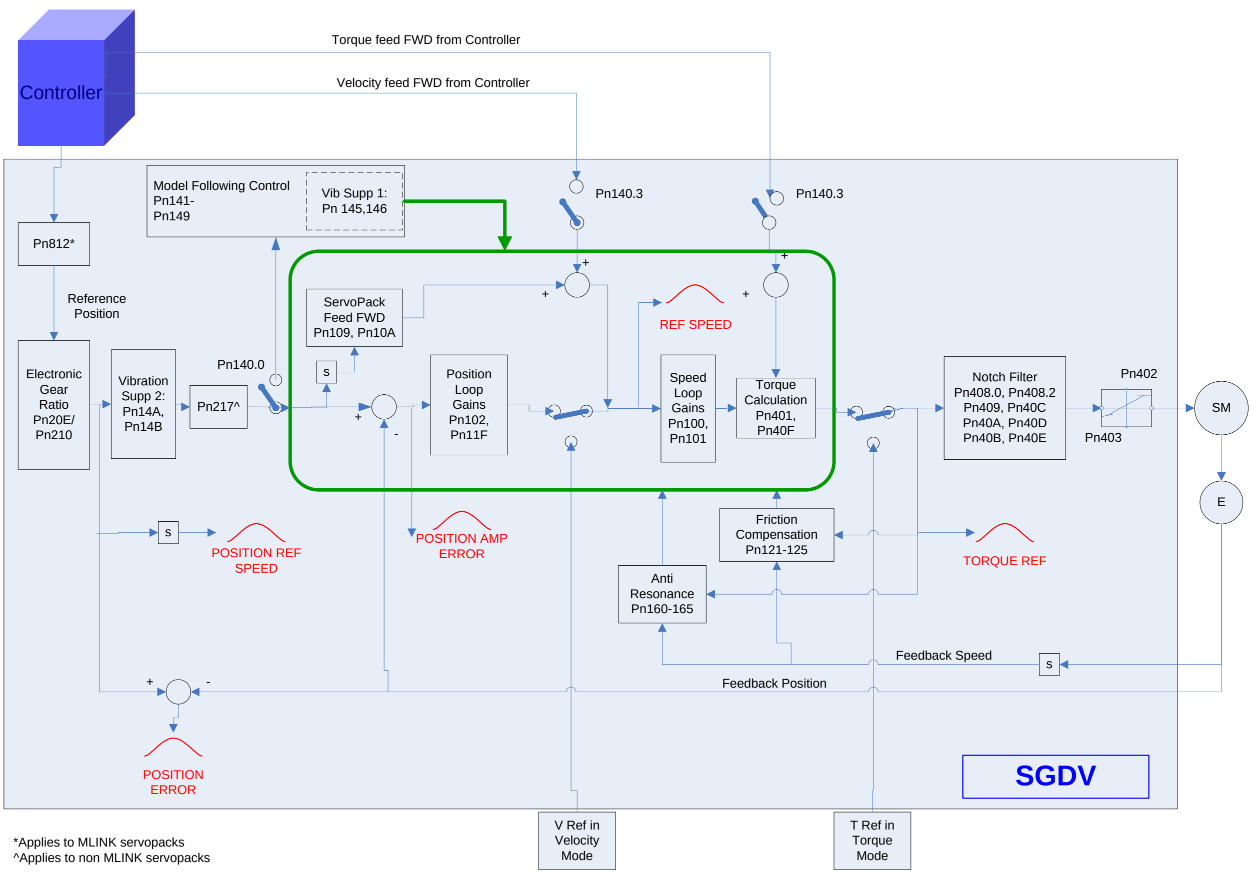


Figure 5: SGDV control loops

The model following gain is calculated based on the test moves performed while autotuning is in progress. Once tuning is complete, it is a static gain. It is not changed adaptively during machine operation. This gain may not be optimal for production moves that are more aggressive than the ones encountered during autotuning without reference. The model following gain for the first tuning cycle is shown in figure 6 (after first trial).

Pn140	Model Following Control Related Switch	0101H
0digit	Model Following Control Selection	1 : Uses n
1digit	Vibration Suppression Selection	0 : Does r
2digit	Vibration Suppression Adjustment Selection	1 : Adjust:
3digit	Selection of Speed Feedforward (VFF) or Torque FF (TFF)	0 : Does r
Pn141	Model Following Control Gain	2431
Pn142	Model Following Control Gain Compensation	1000

Figure 6: Model following gain after tuning with no reference

Figures 7 and 8 display the speed and position loop gains and torque filter time constant.

3digit	Reserved (Do not change.)	0 : Reserved
Pn100	Speed Loop Gain	2399
Pn101	Speed Loop Integral Time Constant	265
Pn102	Position Loop Gain	3598
Pn103	Moment of Inertia Ratio	130
Pn104	2nd Speed Loop Gain	400

Figure 7: Speed and position loop gains

Pn324	Moment of Inertia (Mass) Setting Start Level	300
Pn400	Reserved (Do not use.)	30
Pn401	Torque Reference Filter Time Constant	16
Pn402	Forward Torque Limit	800
Pn403	Reverse Torque Limit	800

Figure 8: Torque filter time constant

On closely examining the commanded speed profile (position reference speed) on SigmaWin+ by sampling data at 125 microseconds, steps in command can be seen with the step update happening at the rate at which the MPiec controller sends commands to the SGDV. In this test, the MECHATROLINK update was set at 2 ms and therefore the step updates are every two ms (Figure 9). It is recommended that this step be eliminated. This is possible by enabling s-curve or exponential functions in the drive to smoothen the command between every MECHATROLINK update.

Step 4: On Mechatrolink systems, set controller parameter 1311 to 1 or 2. Set servopack parameters 811 or 812 to the Mechatrolink update. On an MP2600iec system set Pn216 or Pn217. There is no 1311 parameter that needs to be set in the case of MP2600iec.

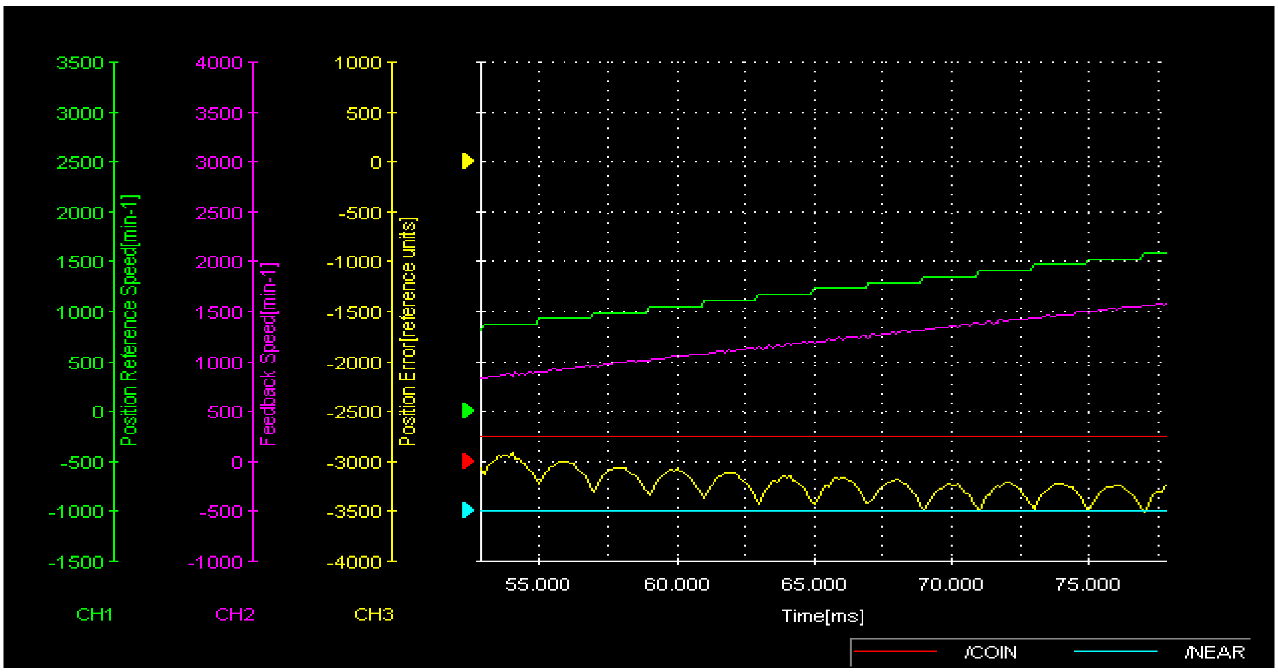


Figure 9: Step command at MECHATROLINK rate

The s-curve or exponential functionality in the drive can be set up from the Hardware Configuration of MotionWorks IEC. Select the servopack and use the 'All Parameters' tab in the CT.

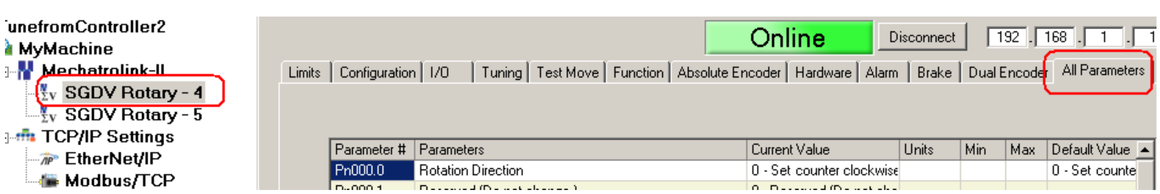


Figure 10: All Parameters tab

Edit parameter 1311 as shown in figure 11. Depending on which smoothening technique is selected through parameter 1311, select an appropriate time constant in parameter Pn811 or Pn812. It is recommended to select the time constant to be equal to the MECHATROLINK update selected for the application.

1310	Velocity Feed Forward Enabled/Disabled	Enabled				1
1311	**Drive Motion Command OPTION ACCFIL Value**	**2: S-curve accelera**				**0**

Figure 11: Parameter 1311

Pn810	Exponential adjustment speed bias	0	ref units/s	0	65535	0
Pn811	Exponential Function Accel/Decel Time Constant	2.0	ms	0.1	510.0	0.0
Pn812	Movement Average Time	2.0	ms	0.1	510.0	0.0
Pn813	Reserved (Do not change.)	0		-10737	10737	0
Pn814	Latch target default move distance	100	ref units	-10737	10737	100

Figure 12: Parameter 811, 812

Figure 13 shows how the command smoothing technique described above made the position reference speed smooth.

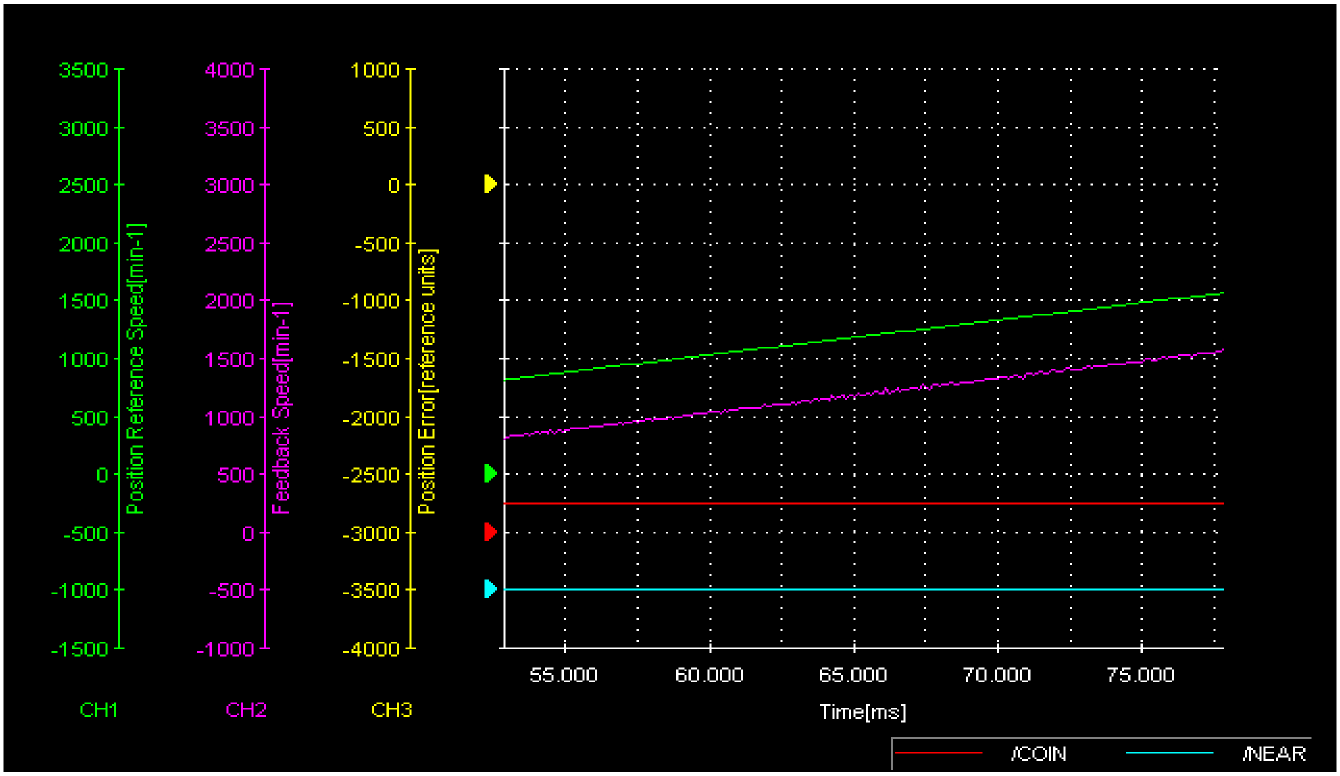


Figure 13: Smooth position reference speed

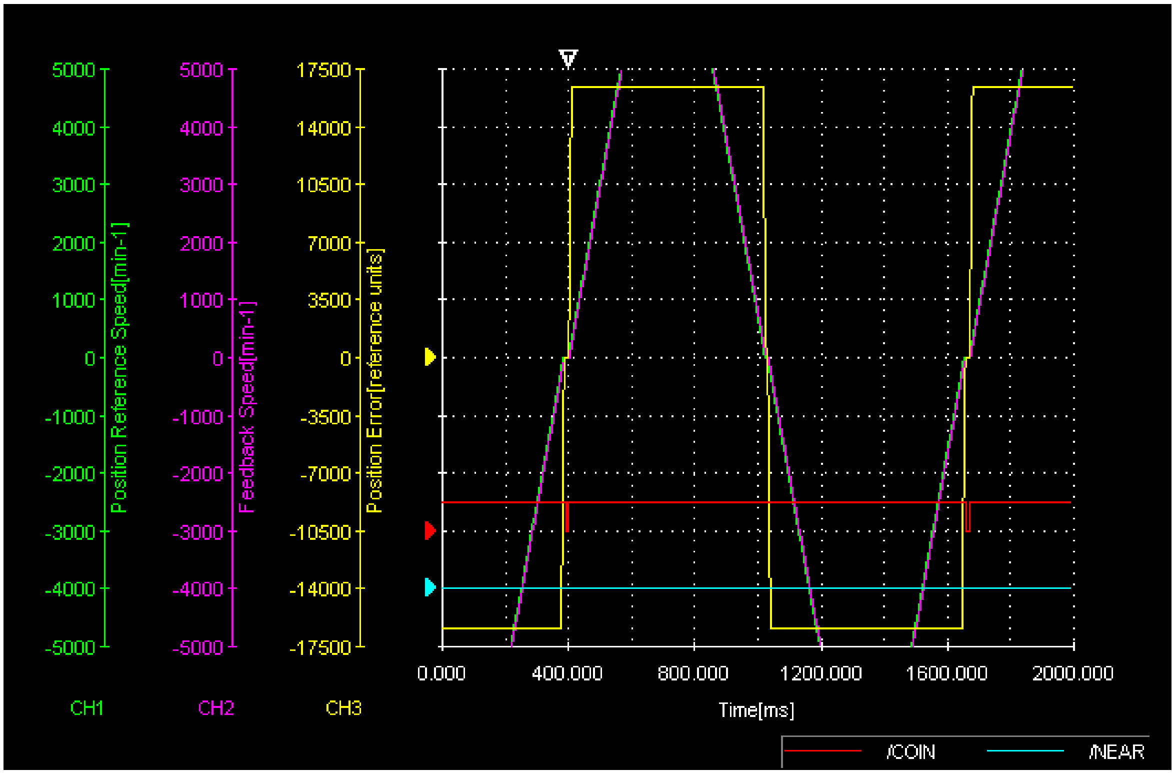


Figure 14: Position following error after no reference autotuning

In SigmaWin+ revert back to the sampling rate set before the 125 microsecond setting. Run the axis with the desired move profile. In this test the move profile was as shown in Figure 1. Position Reference Speed, Feedback Speed and Position Error are plotted. The high position error (>16500 pulses) can be attributed to the aggressive moves that were not seen during the autotuning process. Since the reference signals used for autotuning without reference can be different from the application commands during production, it is recommended to tune the servopack using reference commands from the controller. (Using the same profile that will be used for production)

On an MP2600iec system, set Pn216 or Pn217 to the DPRAM update since the motion network is not over MECHATROLINK but over the DPRAM interface. Pn216 and Pn217 are analogous to Pn811 and Pn812. Make sure only one of the two (Pn216 , Pn217) is set to the DPRAM update. The other Pn should be kept at 0.

Step 5: Start Autotuning with position reference input from host controller.

Choose 'Position reference Input' from the autotuning option (Figure 15). Choose positioning mode and the mechanism that needs tuning as shown in figure 16.

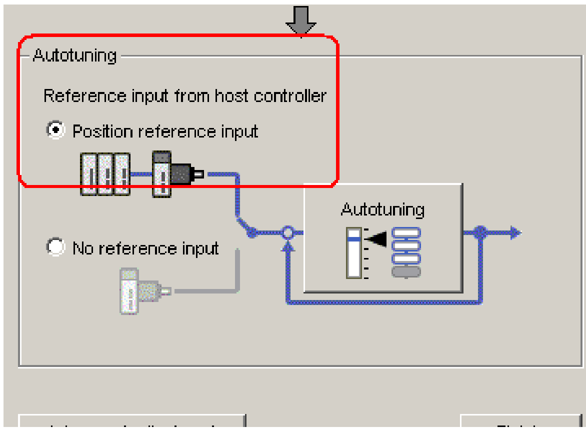


Figure 15: Autotuning using position reference input

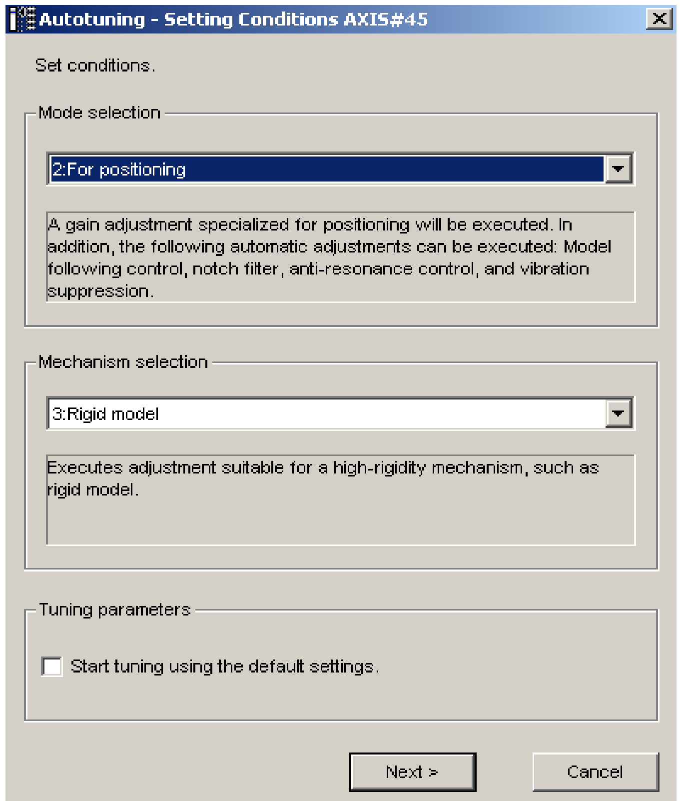


Figure 16: Model selection

Confirm the Moment of Inertia ratio from the previous autotuning cycle. Start the motion profile commanded from the controller and start tuning in SigmaWin+.

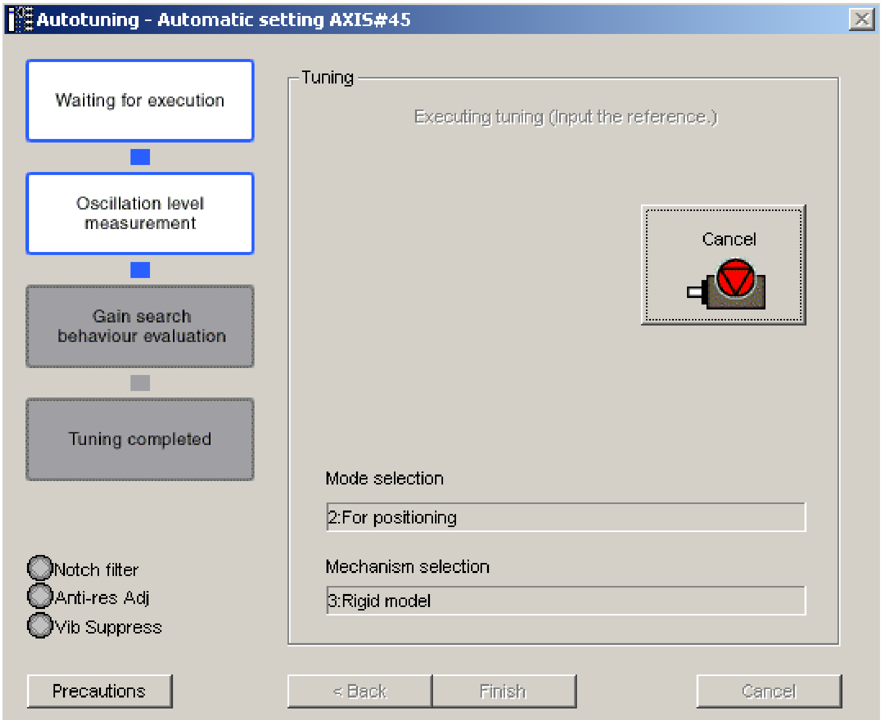


Figure 17: Autotuning with

Finish the tuning application. Upon checking the tuning parameters, it can be seen that the speed and position loop gains and the torque filter have not changed from the previous tuning cycle. This can be seen in figures 18 and 19.

Pn100	Speed Loop Gain	2399
Pn101	Speed Loop Integral Time Constant	265
Pn102	Position Loop Gain	3598
Pn103	Moment of Inertia Ratio	130
Pn104	2nd Speed Loop Gain	400

Figure 18: Speed and position loop gains after autotuning with controller reference

Pn324	Moment of Inertia (Mass) Setting Start Level	300
Pn400	Reserved (Do not use.)	30
Pn401	Torque Reference Filter Time Constant	16
Pn402	Forward Torque Limit	800

Figure 19: Torque loop filter after autotuning with controller reference

It can be seen that the regular tuning gains have not changed after the controller based tuning process was completed. However, the model following gain can be seen to have increased from 2431 to 4527.

Pn13D	Current Gain Level	2000
Pn140	Model Following Control Related Switch	0101H
0digit	Model Following Control Selection	1 : Uses model following control.
1digit	Vibration Suppression Selection	0 : Does not perform vibration supp
2digit	Vibration Suppression Adjustment Selection	1 : Adjusts vibration suppression a
3digit	Selection of Speed Feedforward (VFF) or Torque FF (TFF)	0 : Does not use model following c
Pn141	Model Following Control Gain	4527
Pn142	Model Following Control Gain Compensation	1000

Figure 20: Model Following Gain after autotuning with controller reference

Run the axis using the controller based command profile. Plot the position error to verify if the tuning has improved performance.

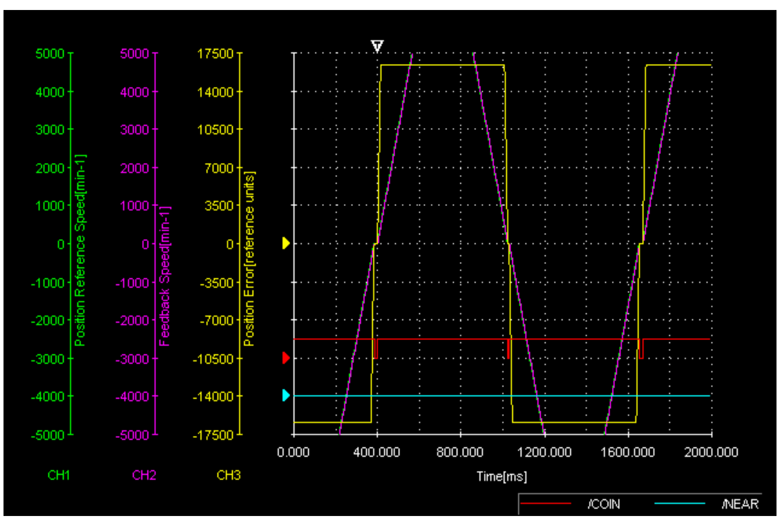
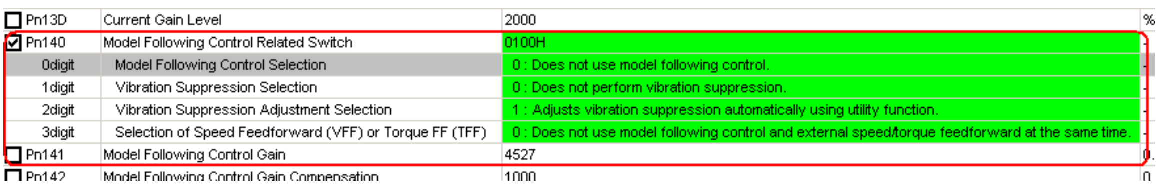


Figure 21: Error after autotuning with controller reference

Autotuning with the controller reference input however has not reduced the position error. It can be seen that the position error is positive, which indicates that the feedback velocity is lagging behind the reference velocity.

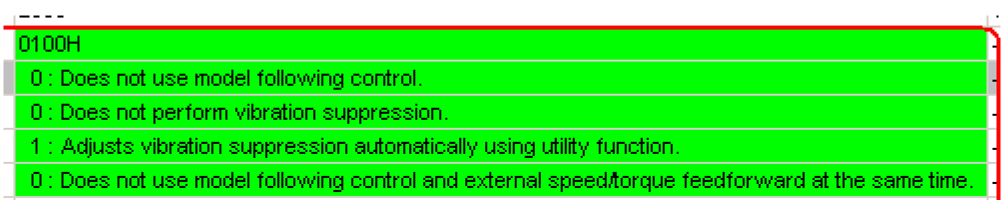
Step 6: Once reference based autotuning is done, set Pn 140.0 = 0.

The MP2300Siec controller has a feedforward velocity term that can be incorporated into the velocity loop. After autotuning is completed, when model following control is in effect, the setting is to ignore any other feed forward component from an external source. This can be seen by Pn140.3 =1. In order to study the effect of the controller's feedforward component alone (without the servopack model following gain), disable the model following gain by setting Pn140 = 0100. The controller feedforward term which is enabled by default in the controller will act on the axis alone.



Pn13D	Current Gain Level	2000	%
Pn140	Model Following Control Related Switch	0100H	
0digit	Model Following Control Selection	0 : Does not use model following control.	
1digit	Vibration Suppression Selection	0 : Does not perform vibration suppression.	
2digit	Vibration Suppression Adjustment Selection	1 : Adjusts vibration suppression automatically using utility function.	
3digit	Selection of Speed Feedforward (VFF) or Torque FF (TFF)	0 : Does not use model following control and external speed/torque feedforward at the same time.	
Pn141	Model Following Control Gain	4527	
Pn142	Model Following Control Gain Compensation	1000	

Figure 22: Pn140



0100H
0 : Does not use model following control.
0 : Does not perform vibration suppression.
1 : Adjusts vibration suppression automatically using utility function.
0 : Does not use model following control and external speed/torque feedforward at the same time.

Figure 23: Pn140 setting that turns model following control off

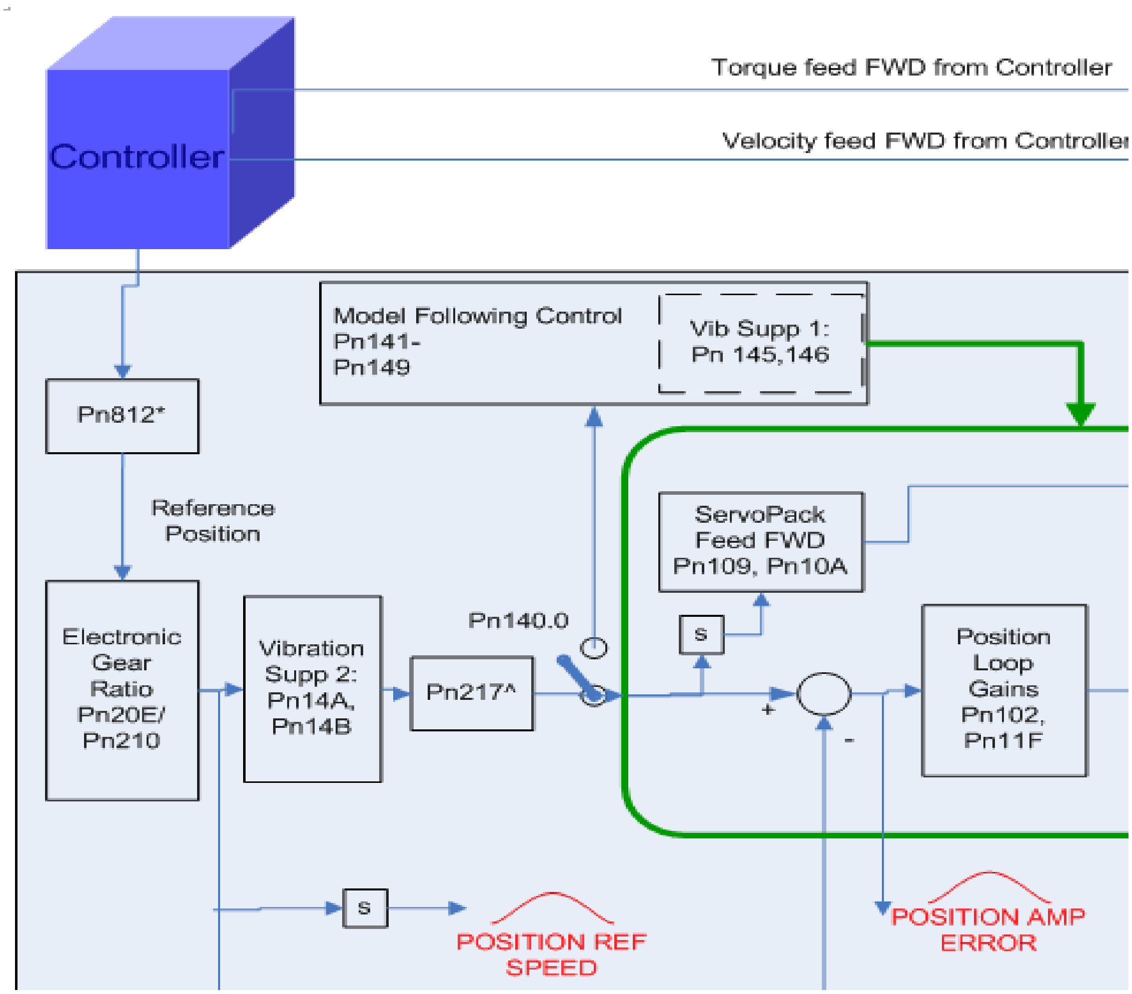


Figure 24: Pn140 switching effect

When disabling Model Following Control (Pn140.0 = 0), and using only the feed forward component of the controller, the position following error plot for the axis is vastly improved as can be seen from Figure 25. The position following error maximum value can be seen is very much reduced when compared to plots where only the model following control was acting. The maximum following error is 3000/1048576 = 0.0028 degrees.

Step 7: Verify position error. Check to verify if position error is positive or negative (actual position lagging or leading)

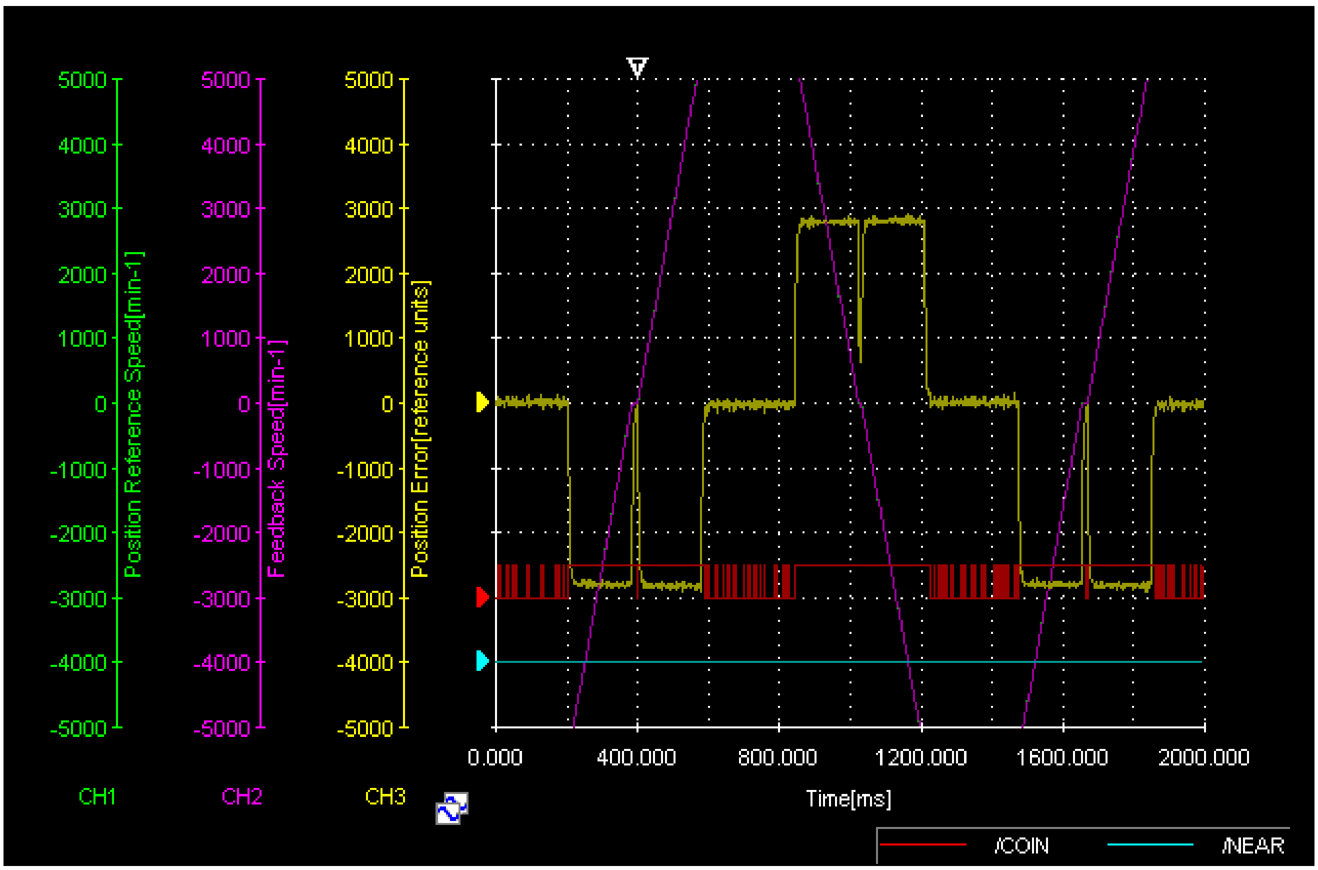


Figure 25: Model Following Control off and Controller feed forward on

Step 8: Set Pn 140.0 = 1, Pn 140.3 = 1

In an effort to further reduce the following error, change Pn140.3 = 1 to allow controller external feed forward to act. This change in Pn140.3 now lets model following gain and controller side external feed forward work together. When plotting position error for this case, it can be seen that the following error is negative (actual position leads command) as seen from figure 26. This is an indication that the feed forward component is very high.

Step 9: Adjust Pn 141 to reduce the absolute value of dynamic position error as close to zero as possible.

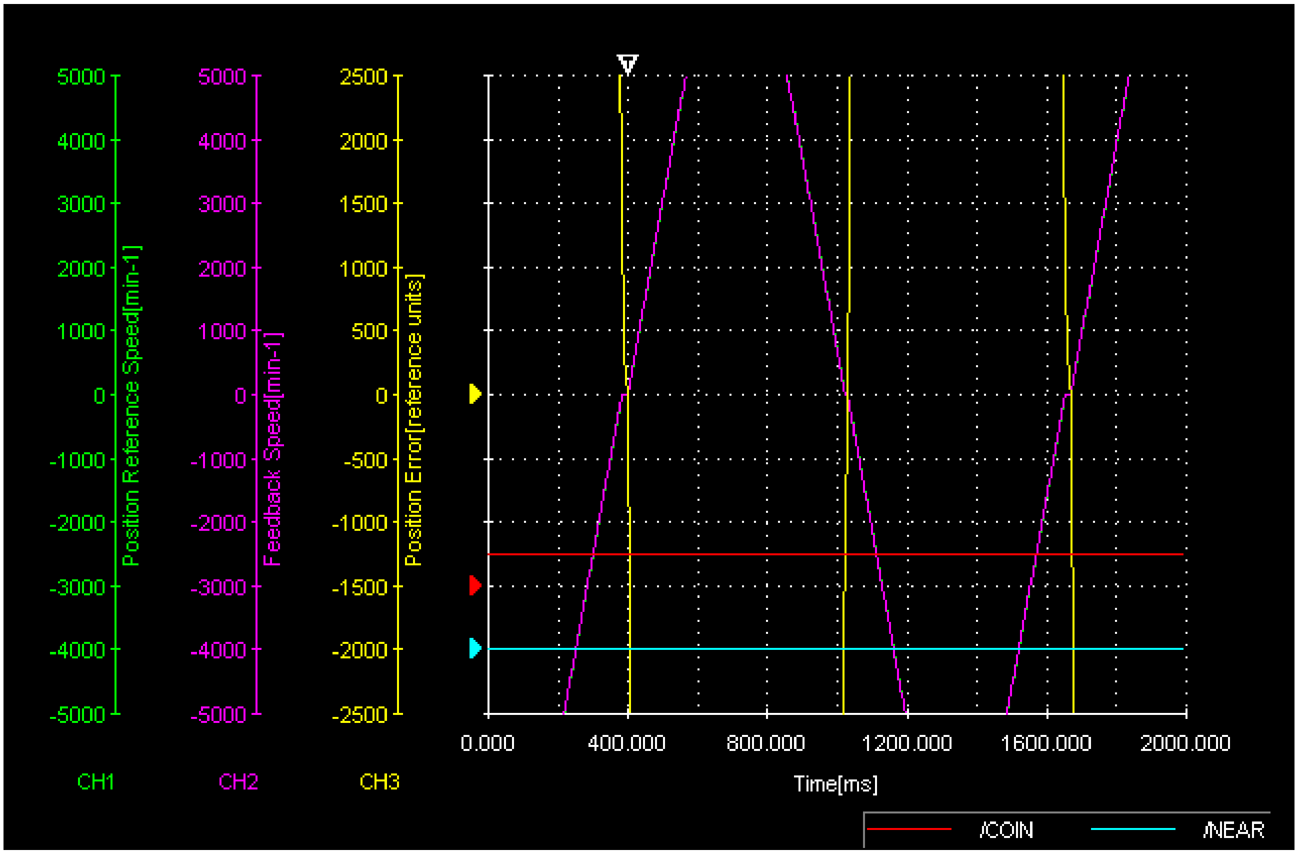


Figure 26: Negative position error

Since the error was negative (due to high feed forward component), reduce the model following gain. Pn141 = 3000. When plotting the following error for the reduced model following gain case, it can be seen that position error changed sign to positive (indicating low feed forward component). This is shown in figure 27.

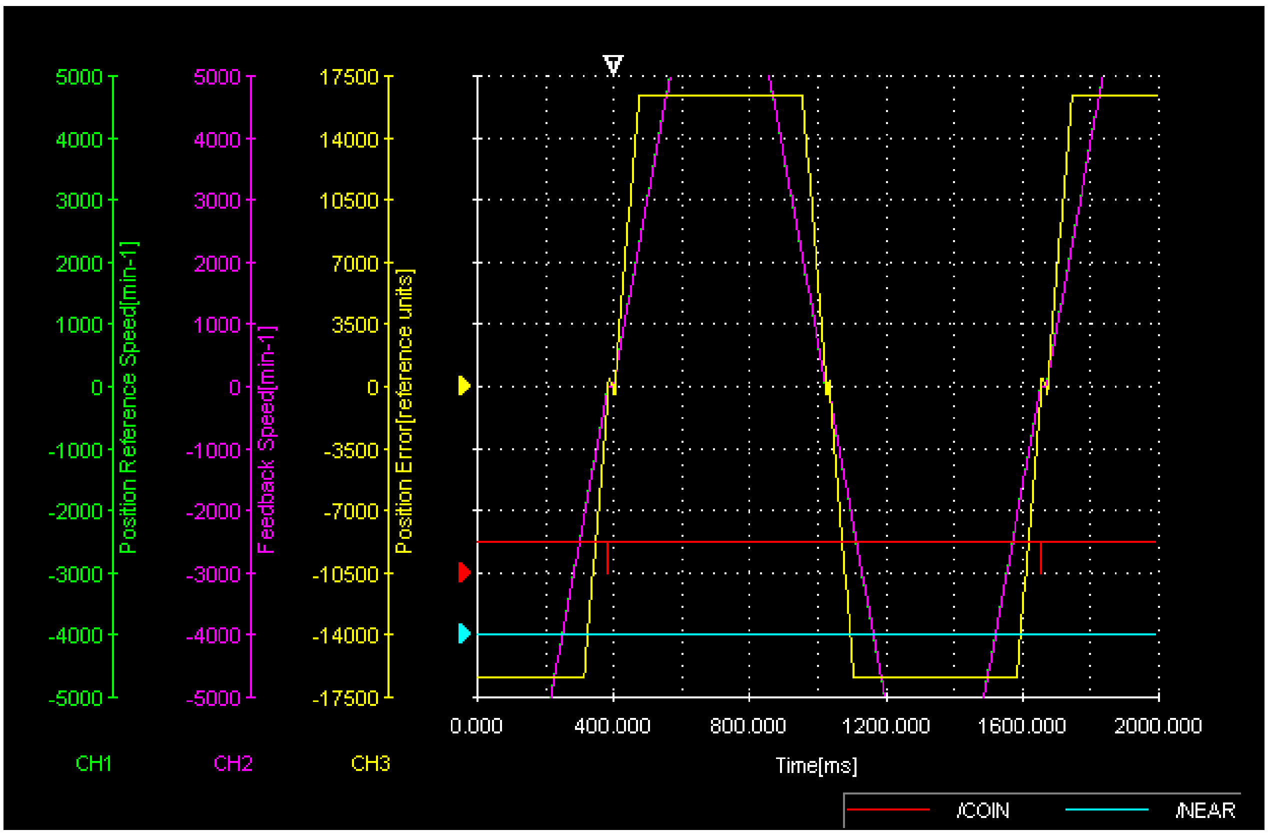


Figure 27: Positive position error (low feed forward component)

To hone in on an optimal feed forward gain, increase the value of Pn141 = 3500. The position error plot as seen in Figure 28 shows positive error, but is getting closer to a zero error condition.

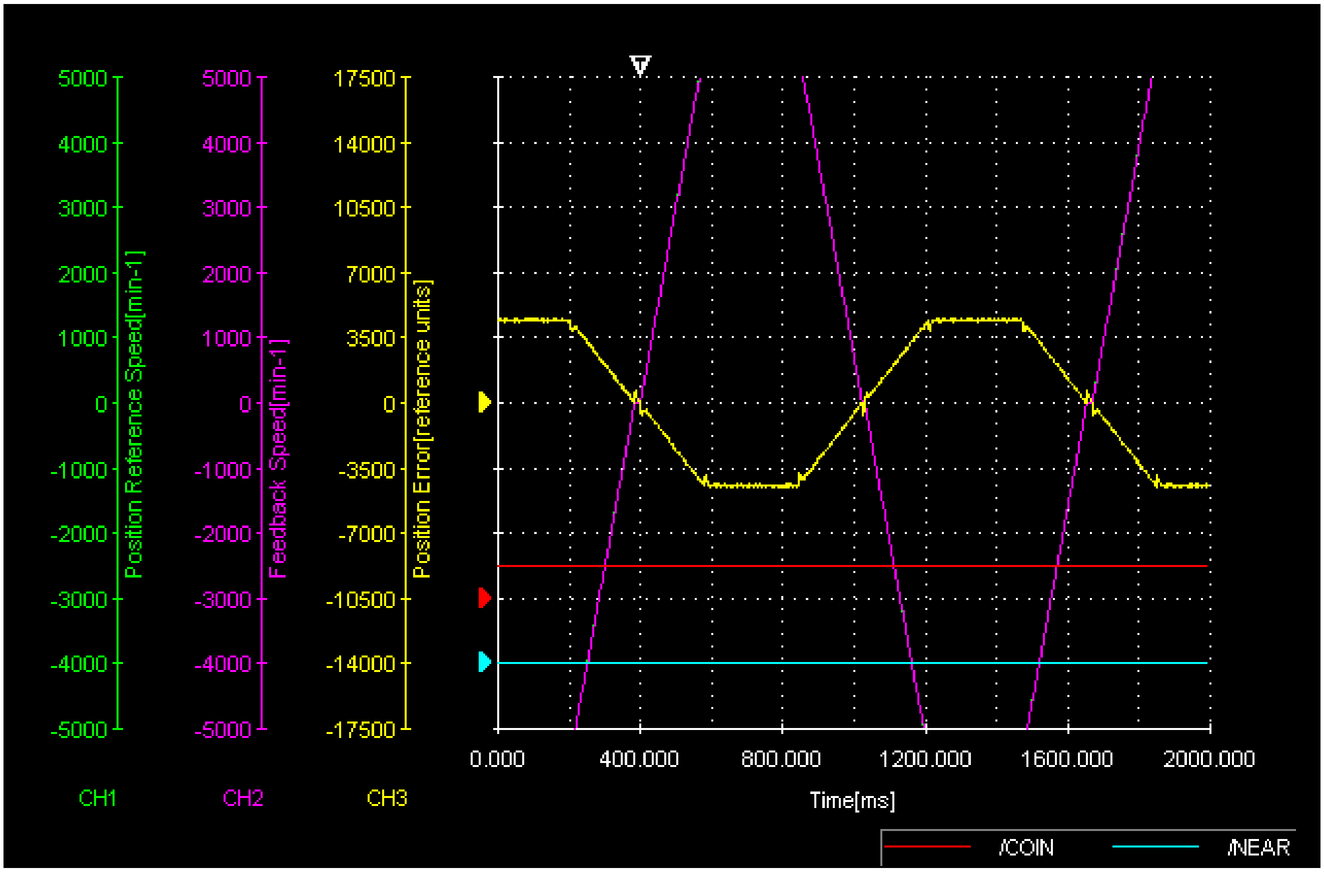


Figure 28: Following error with Pn141 = 3500

By trial and error, it is now clear that the best model following gain lies between 3000 and 3500, closer to 3500. Set Pn141 = 3443. The position following error can now seen to be as low as 1100 pulses (0.001 degrees) as seen in figure 29

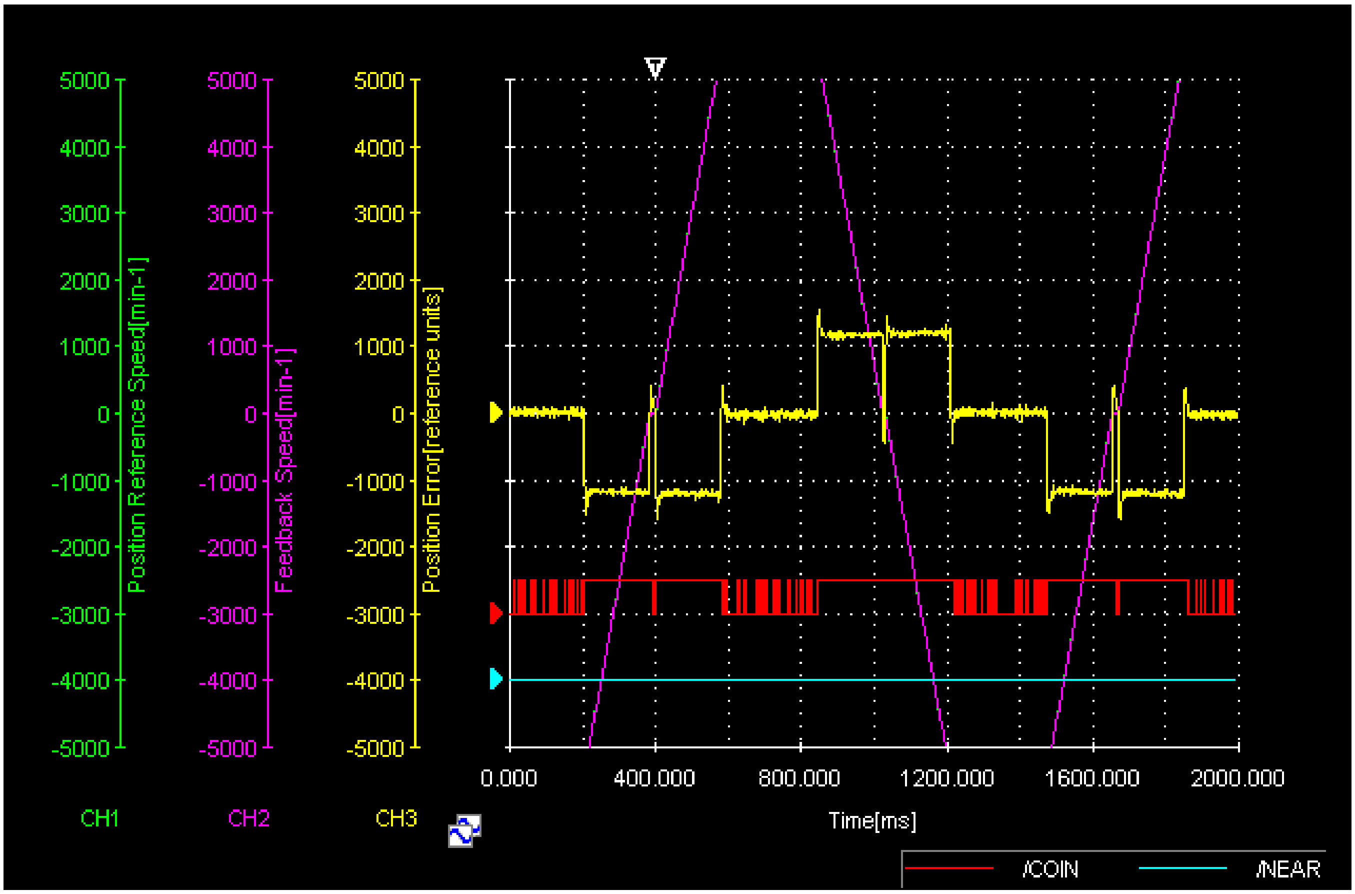


Figure 29: Following error with model following control and controller side feed forward active

Figure 30 is a plot of position following error that compares results when only controller side feed forward component is active and the case when controller feed forward acts with the optimal value of model following control gain.

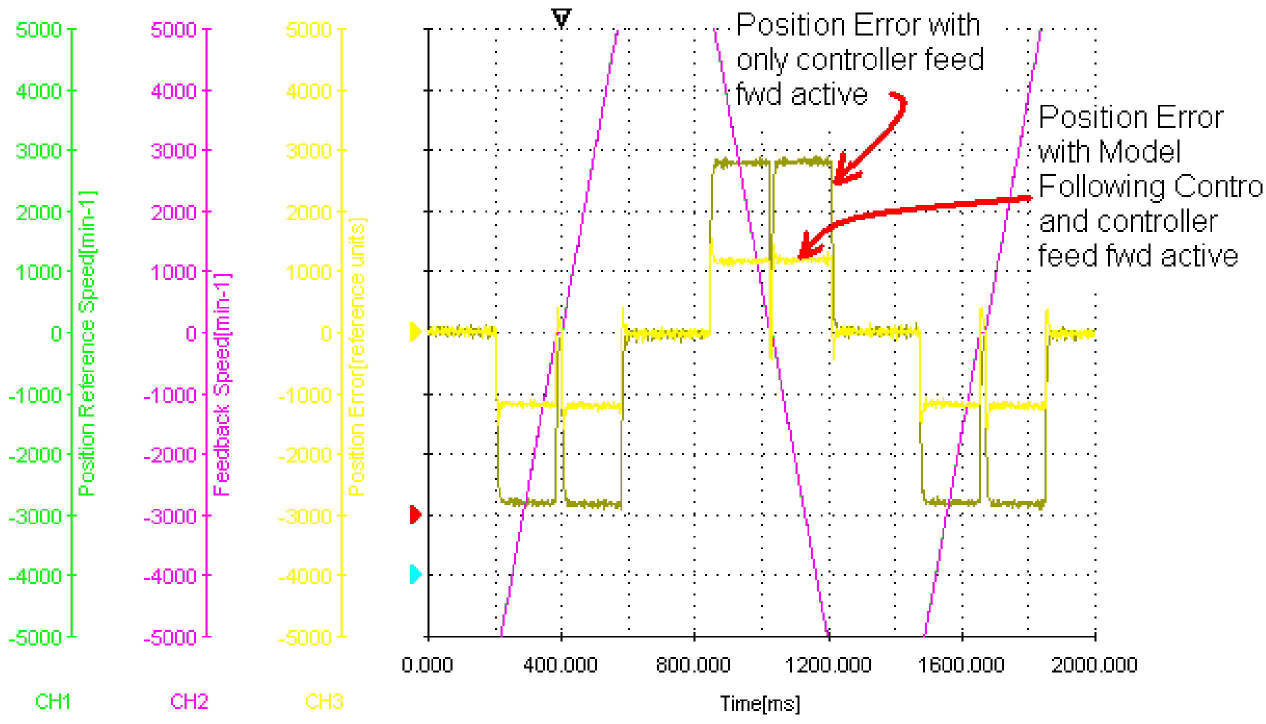


Figure 30: Final result comparison

Step 10: Start Advanced adjustment (One parameter tuning).

In an attempt to verify if one parameter tuning can improve performance more, start tuning with priority given to response. Run the axis from commands from the controller and start the tuning process. Increase the tuning level until the axis shows signs of becoming unstable (Figure 31).

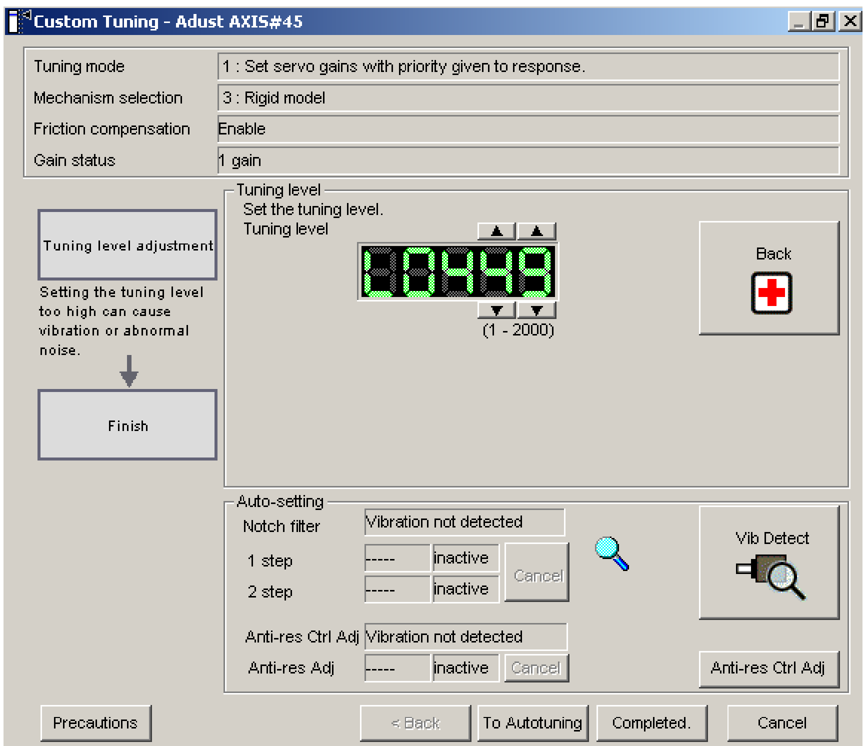


Figure 31: One touch tuning with priority given to response

Check the position following error. The results are shown in figure 32. The maximum position error is 1500 pulses (0.0014 degrees). Figure 33 is a comparison of the best performances. The performance is best when the controller side feed forward is supported by an optimal manually set model following gain in Pn141. The controller side feed forward component is enabled by default. Performance of the axis when controller feed forward and one parameter tuning are used together is also substantially better than the performance when only controller feed forward is used as can be seen from Figure 33.

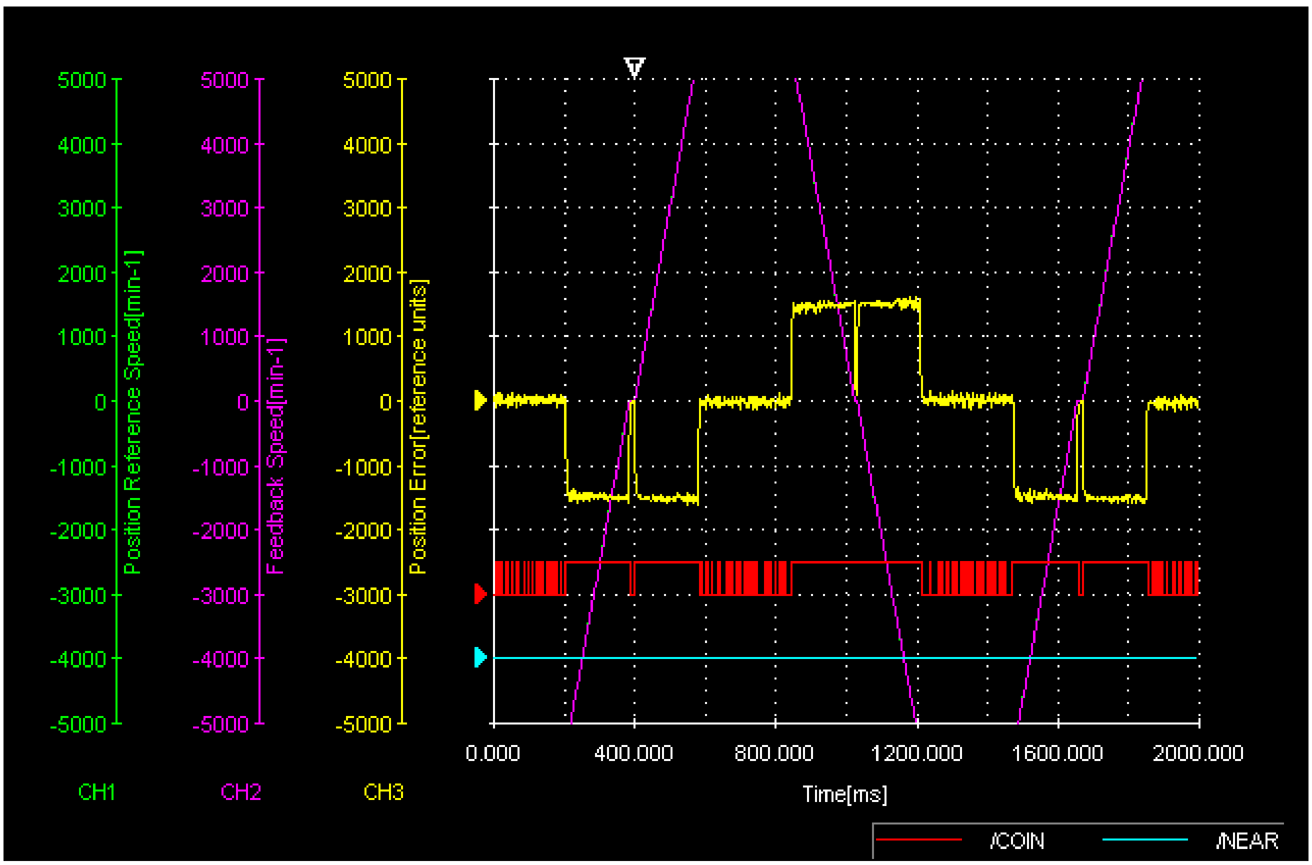


Figure 32: Position following error after one touch tuning

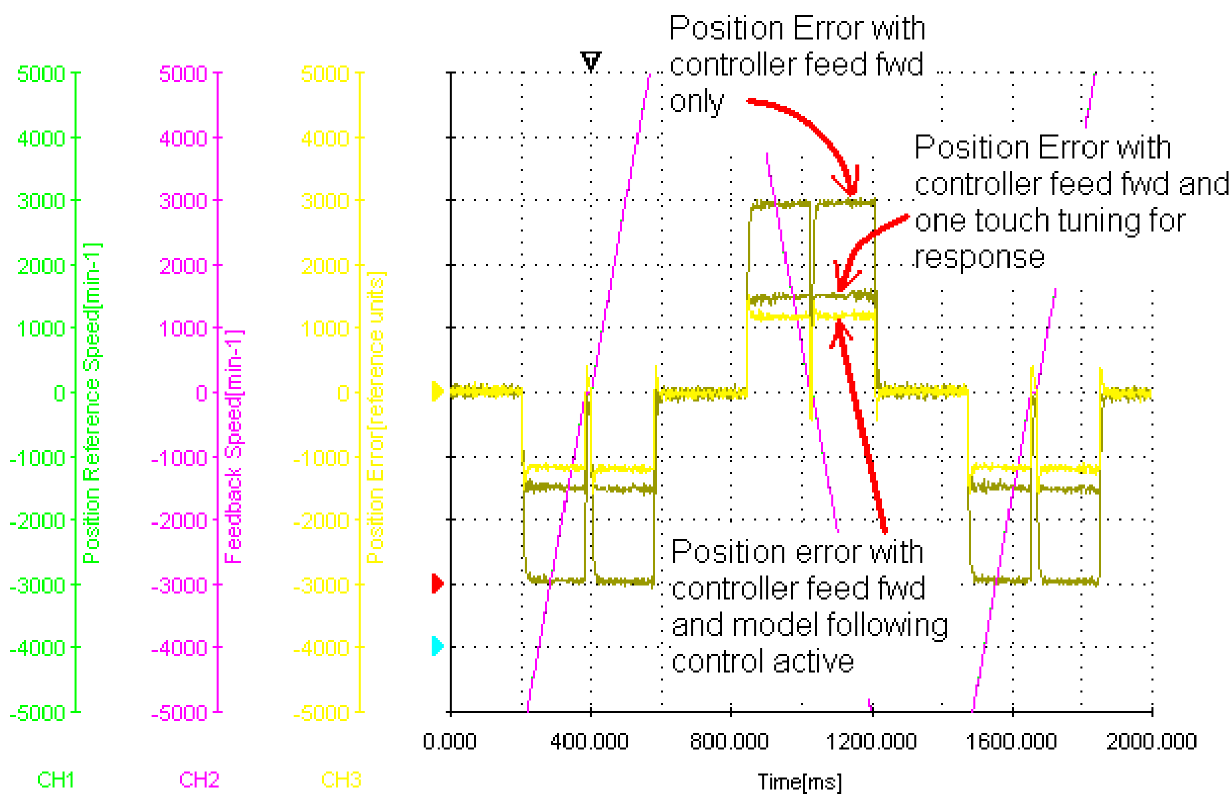


Figure 33: Comparison of position errors

Step 11: In one parameter tuning mode, change the tuning level to check if dynamic response is improved. Constantly monitor axis performance through the trace function. If no improvement in tuning performance, cancel custom tuning and use model following control results from step 9.

Tuning Challenge # 2

Move Profile:

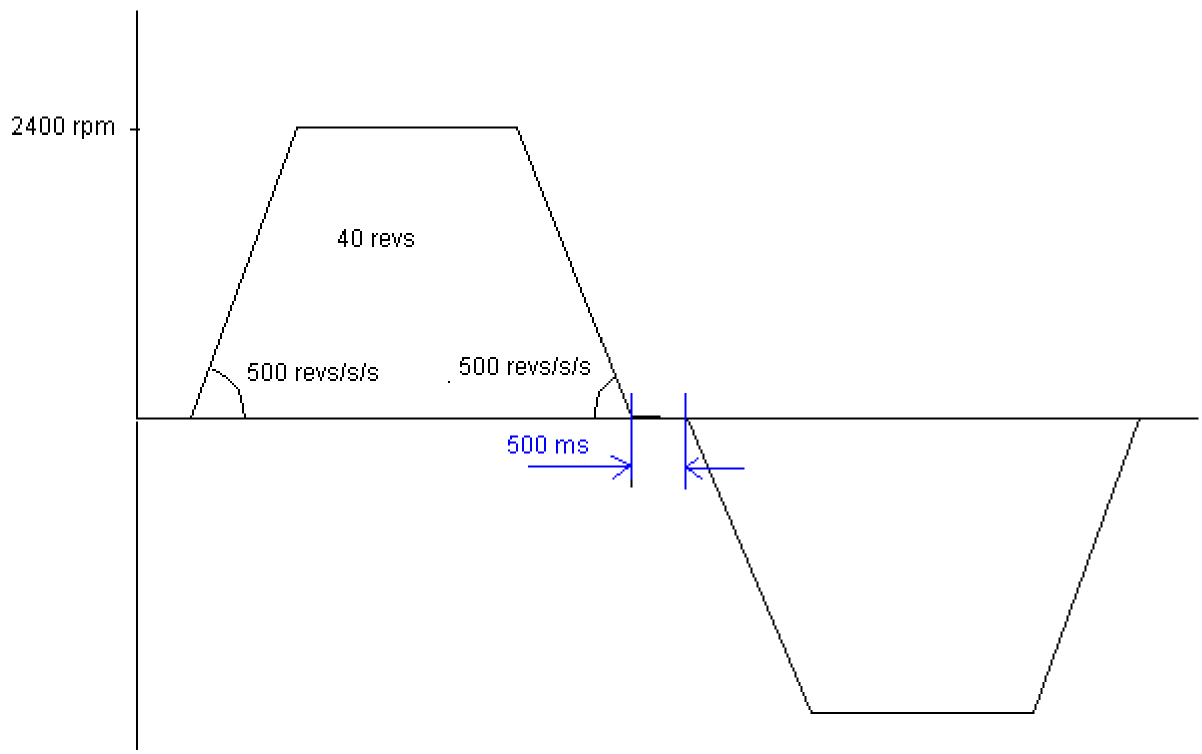


Figure 34: Test 2 Move profile

Inertia Mismatch: 20:1

Mechanical coupling: Directly coupled load

Hardware used:

SGDV-R90F11A servopack

SGMAV-01A3A61 motor

Objective: To reduce maximum position error to less than 0.01 degrees.

Tuning Method:

Step 1: **Turn tuneless mode off (Pn170.0 = 0)**

Step 2: **Start Autotuning with no reference input**

Choose positioning as the tuning objective and select the system to be a rigid system.

Step 3: Start autotuning

Step 4: On Mechatrolink systems, set controller parameter 1311 to 1 or 2. Set servopack parameters 811 and 812 to the Mechatrolink update. On an MP2600iec system set Pn 216 or Pn217. There is no 1311 parameter that needs to be set in the case of MP2600iec.

The results of autotuning without external reference commands are shown in figure 35.

Table 1: Tuning and model following parameters after tuning without a reference

Pn	Value
100	663
101	960
102	994
401	46
140	0101
141	2016

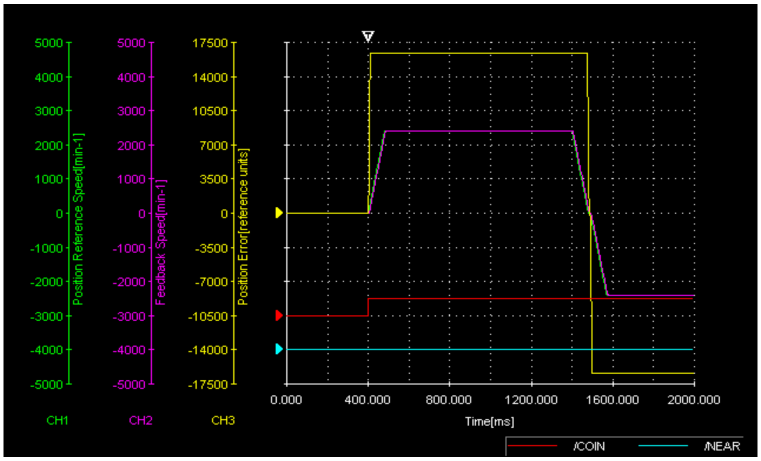


Figure 35: Position error results from tuning without reference

Step 5: Start Autotuning with position reference input from host controller.

Choose positioning as the tuning objective and select the system to be a rigid system

The results of tuning with the controller reference commands are shown in figure 36. Table 2 lists the parameters at the end of controller reference tuning. It can be seen that after autotuning, the model following control is activated (Pn140.0) and any external feed forward gains are ignored (Pn140.3). The results from figure 36 show that a feed forward component is required as the actual velocity is lagging behind the commanded velocity.

Table 2: Tuning and model following parameters after tuning with the controller reference

Pn	Value
100	733
101	868
102	1099
401	40
140	0101
141	1999

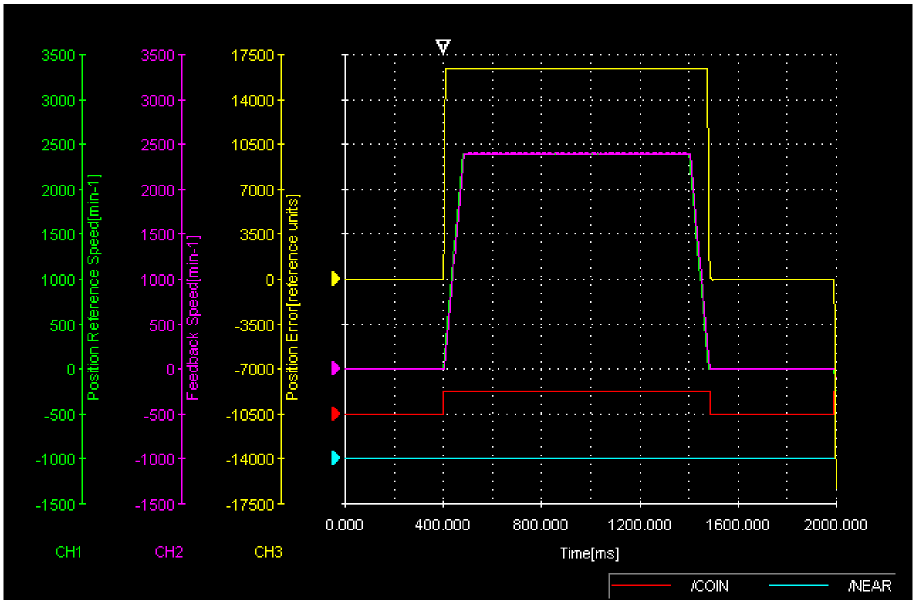


Figure 36: Position error results from tuning with controller reference

Step 6: **Once reference based autotuning is done, set Pn 140.0 = 0.**

The results from turning off the servopack model following control and using only the controller feed forward are shown in Figure 37. The parameters used in this scheme are shown in table 3. It can be seen in figure 37 that the maximum following error has been brought down to close to 6000 pulses (0.005 degrees)

Table 3: Parameters used when model following controller is turned off

Pn	Value
100	733
101	868
102	1099
401	40
140	0100
141	1999

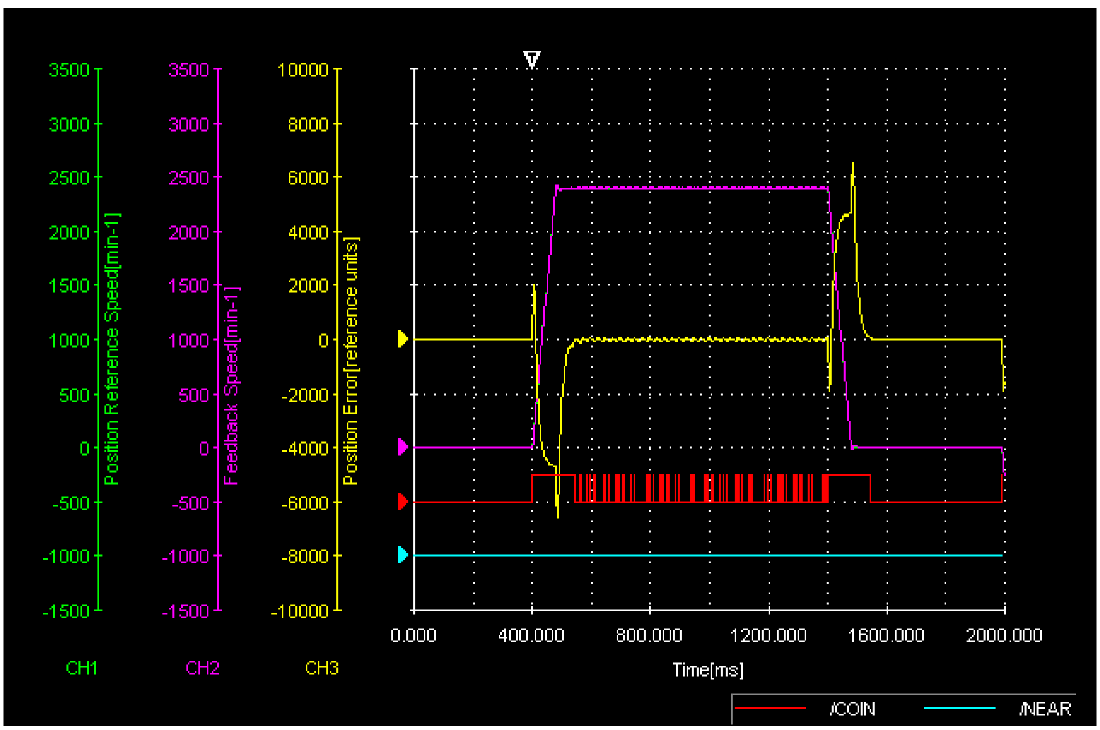


Figure 37: Controller feed forward only

Step 7: Monitor dynamic position error.

Step 8: Set Pn 140.0 = 1, Pn 140.3 = 1 to enable model following gain and controller feed forward component

When both model following control and controller feed forward are turned on together (Pn140.3 =1), the position error plot (figure 38) shows a negative trend indicating that the actual velocity is leading the command velocity. This means feed forward velocity is higher than optimal.

Table 4: Parameters when model following control and controller feed forward are both used

Pn	Value
100	733
101	868
102	1099
401	40
140	1101
141	1999

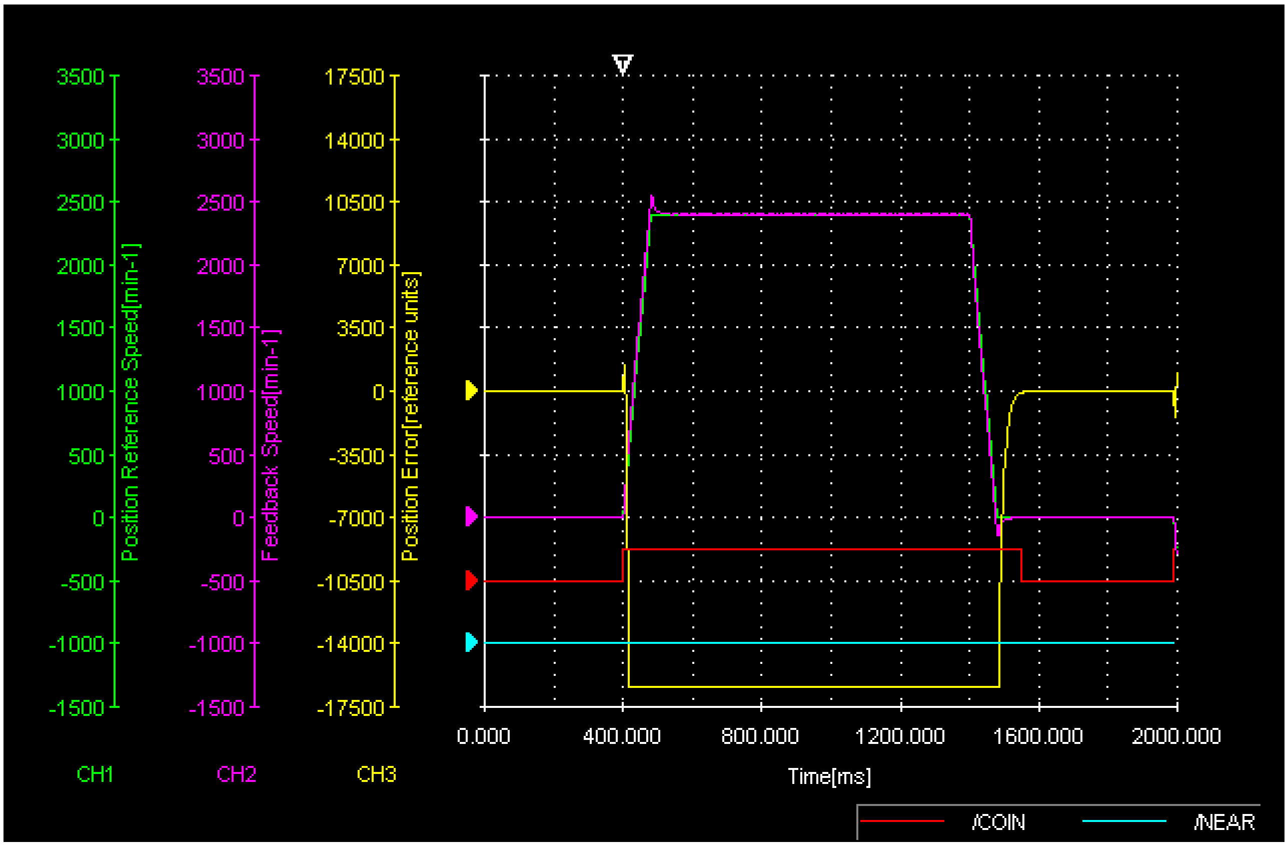


Figure 38: Position error plot with model following control and controller feed forward gain

Step 9: Adjust Pn 141 to bring the absolute value of dynamic position error as close to zero as possible

To reduce the negative position error, reduce the model following gain Pn141 to 1084. Figure 39 illustrates the result of the optimal model following gain used with the controller feed forward term. It can be seen that the maximum position error is 12000 pulses (0.011 degrees).

Table 5: Optimal model following gain with tuning parameters

Pn	Value
100	733
101	868
102	1099
401	40
140	1101
141	1084

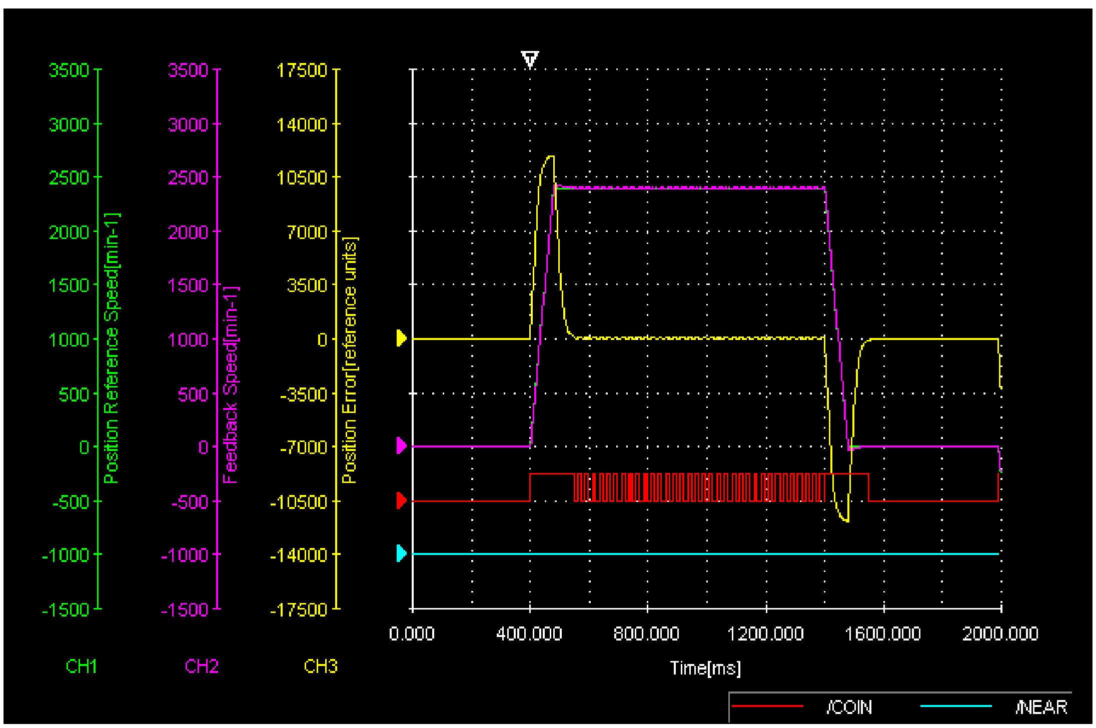


Figure 39: Optimal model following gain alongside controller feed forward gain

Step 10: Start Advanced adjustment (One parameter tuning).
Tune with priority given to response. Select the rigid model.
In an attempt to verify if one parameter tuning can produce better results, use advanced adjustment as shown in figure 40. Choose custom tuning with priority given to response (figure 42).

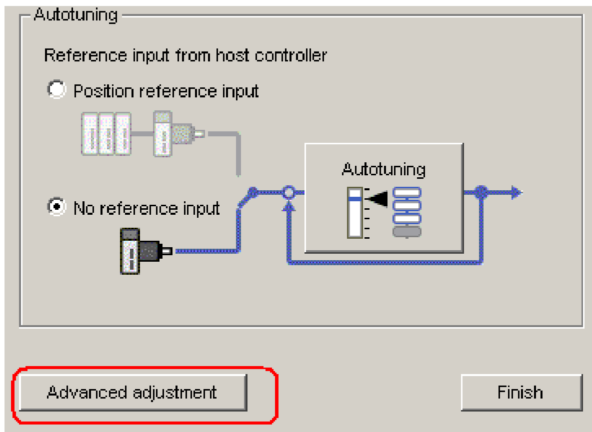


Figure 40: One parameter tuning

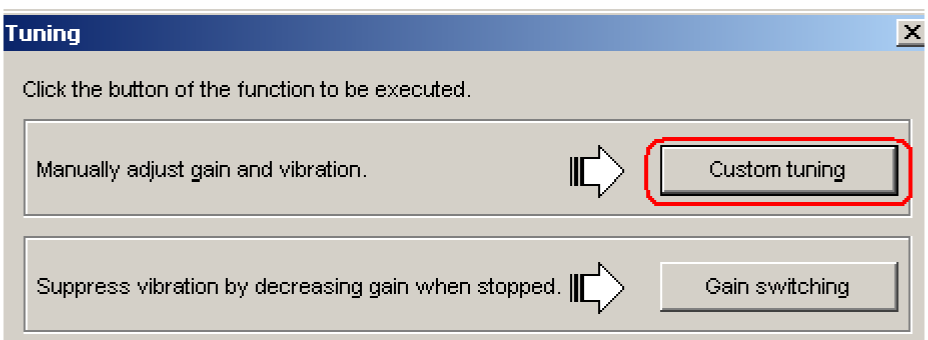


Figure 41: Custom tuning selection

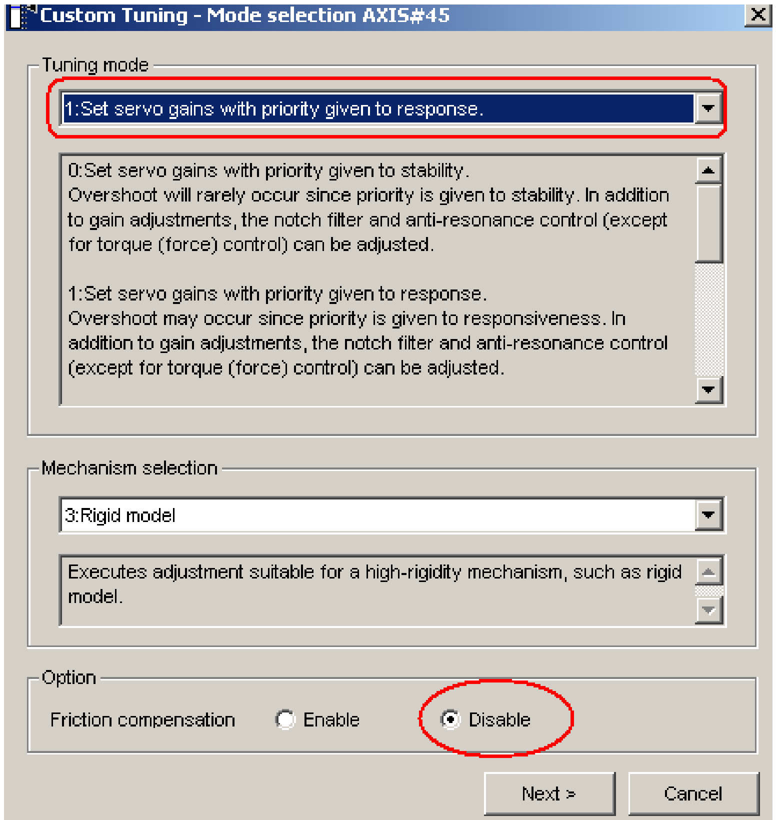


Figure 42: Tuning mode selection

Perform tuning using the one parameter tuning button until the axis shows signs of instability (overshoot or excessive noise). Plot the position error at this gain level. One thing to note is that while the one parameter custom tuning is being performed, plots of error can be taken using the trace function in SigmaWin+. Also, when one parameter custom tuning is started, Pn140.0 is automatically turned off shutting down the model following control algorithm.

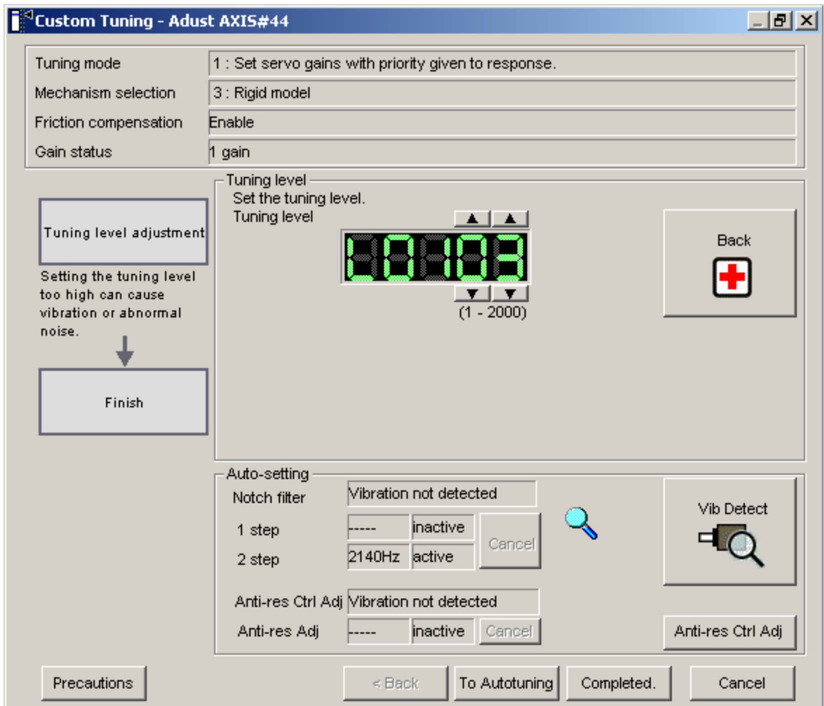


Figure 43: One parameter tuning screen

Table 6: Parameters after one parameter custom tuning

Pn	Value
100	1090
101	734
102	1557
401	22
140	1100
141	1084

Figure 44 shows the results of one parameter custom tuning. The maximum error was reduced to 4000 pulses (0.0038 degrees)

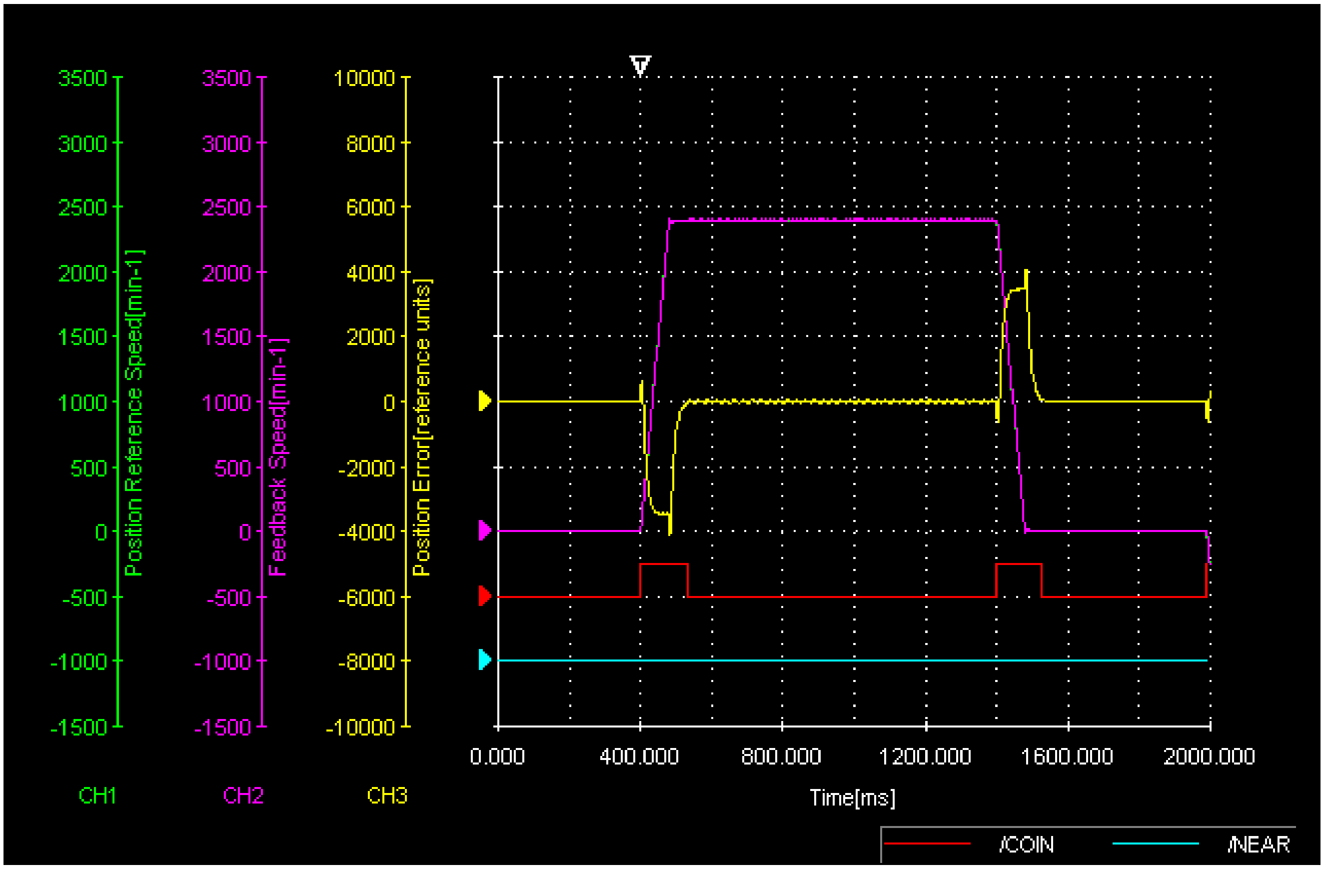


Figure 44: Position error after one touch custom tuning

Figure 45 is a comparison of position error results after runs with only controller feed forward active, controller feed forward and optimal model following gain, and controller feed forward and custom one touch tuning. It can be seen that in this test, best performance was obtained by using controller feed forward and performing one parameter custom tuning.

Step 11: In one parameter tuning mode, change the tuning level to check if dynamic response can be improved. Constantly monitor axis performance through the trace function. If no improvement in tuning performance is seen, cancel custom tuning and use model following control results from step 9.

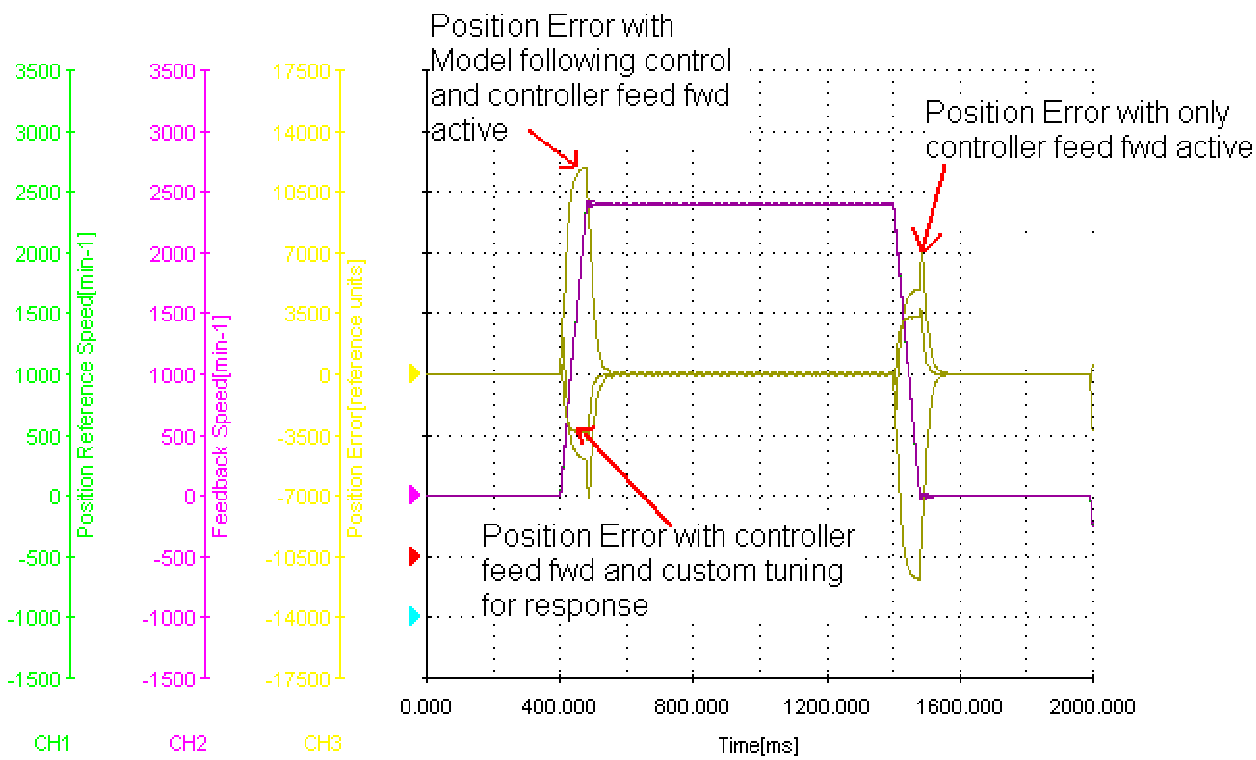


Figure 45: Performance comparison for test 2

Figure 46 is a tuning work flow for tuning the Sigma-5 servopack with commands from the MPiec controller for position following error for aggressive moves with high inertia loads.

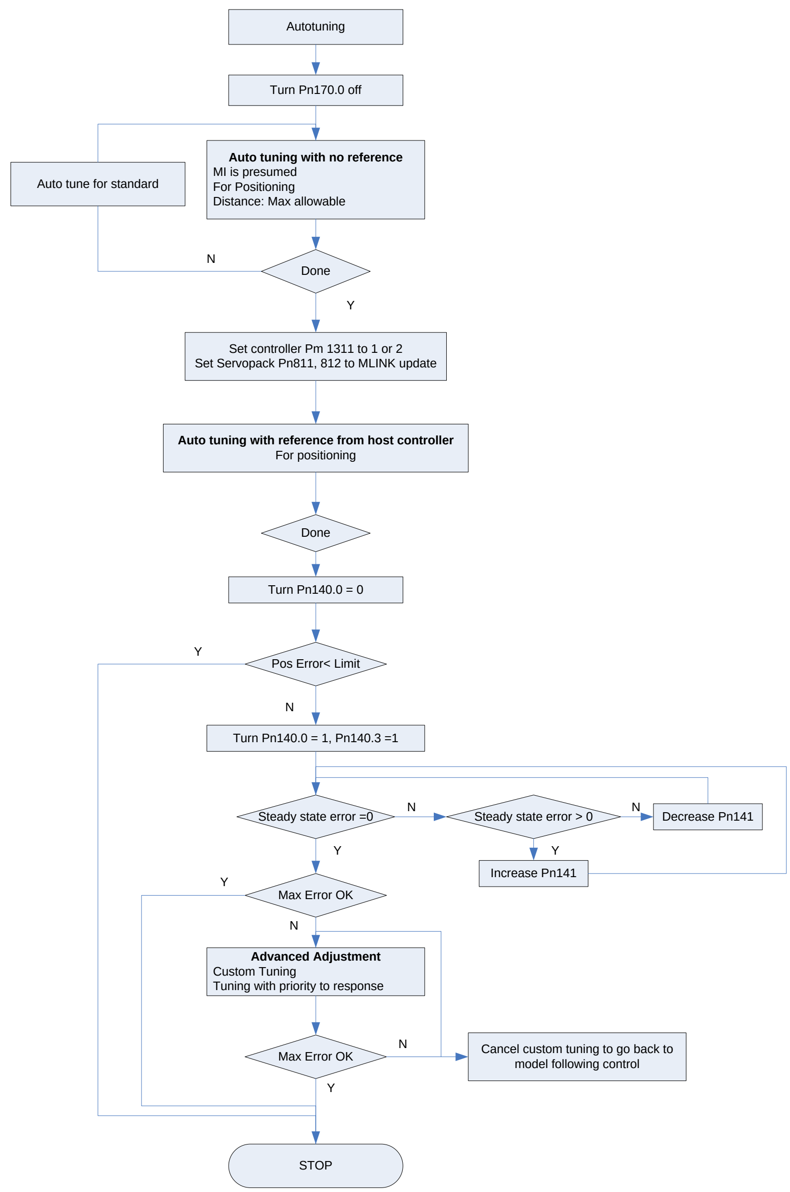


Figure 46: Tuning work flow for minimizing position following error