Chapter 6

Chapter 6: Operation Mode

Table of Contents

Introduction

This chapter describes the operation of each control mode, including gain adjustment and filters. For position control, you can use an external pulse source or commands from the internal registers. For Speed mode and Torque mode, apart from the commands from the internal registers, you can also control the servo drive by the analog voltage input. In addition to Single mode, Dual mode is also available for meeting the application requirements.

6.1 - Selecting the operation mode

This servo drive provides three basic operation modes: Position, Speed, and Torque. The available communication modes are serial Modbus, Modbus TCP (optional card), and Ethernet/IP (optional card). For basic operation mode, you can choose from Single mode, Dual mode, and Multi-mode. The following table lists all the available control modes in P1.001.

Mode		Short Name	Code	Description
Single mode	Position mode (Terminal block input)	PT	00	The servo drive receives the Position command by high speed pulses and commands the motor to run to the target position. The Position command is communicated through the terminal block and the signal type is pulse and direction, CW/CCW pulses, or A/B Quadrature. Analog positioning mode is also available in PT mode. The motor position is commanded by an analog signal (potentiometer or 0-10V input). No pulse train is required. See P1.064 for more details.
	Position mode (Register input)	PR	01	The servo drive receives the Position command and commands the motor to run to the target position. Position commands are issued from the internal registers (99 sets in total). You can select the register number with DI signals or through communication.
	Speed mode	S	02	The servo drive receives the Speed command and commands the motor to run at the target speed. The Speed command is issued from the internal registers (3 sets in total) or by analog voltage (-10V to +10V) which is communicated through the terminal block. You select the command with DI signals.
	Speed mode (No analog input)	Sz	04	The servo drive receives the Speed command from internal registers that can be populated via communications, through SureServo2 Pro, or the keypad. These values command the motor to run at the target speed.The Speed command can only be issued from the internal registers (3 sets in total) instead of through the analog input. You select the command with DI signals.
	Torque mode	T	03	The servo drive receives the Torque command and commands the motor to run with the target torque. The Torque commands can be issued from the internal registers (3 sets in total) as well as by analog voltage (-10V to +10V) which is communicated through the terminal block. You select the command with DI signals.
	Torque mode (No analog input)	Tz	05	The servo drive receives the Torque command from internal registers that can be populated via communications, SureServo2 Pro, or the keypad. These values command the motor to run at the target torque. The Torque command can only be issued from the internal registers (3 sets in total). You select the command with DI signals.

Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

Mode	Short Name	Code	Description
Dual mode	PT-S	06	You can switch PT and S mode with DI signals.
	PT-T	07	You can switch PT and T mode with DI signals.
	PR-S	08	You can switch PR and S mode with DI signals.
	PR-T	09	You can switch PR and T mode with DI signals.
	S-T	0A	You can switch S and T mode with DI signals.
	-	0B	Reserved
	Communication	0C	Ethernet/IP mode
			PT-PR
	PT-PR	0D	You can switch PT and PR mode with DI signals.
Multi-mode	PT-PR-S	0E	You can switch PT, PR, and S mode with DI signals.
	PT-PR-T	0F	You can switch PT, PR, and T mode with DI signals.

Here are the steps to switch the operation mode:

1) Switch the servo drive to Servo Off status. You can do this by setting DI.SON to OFF.
2) Set P1.001 and refer to the code listed above for the mode selection.
3) After setting the parameter, cycle the power to the servo drive.
4) The following sections describe the operation of each mode, including the mode structure, command source, selection and processing of the command, and gain adjustment.

6.2 - Position mode

Two input modes for position control are available on the SureServo2: external pulses from **T**erminals (P**T** mode = **T**erminals) and internal position **R**egister (P**R** mode = **R**egisters). In PT mode, the SureServo2 can accept pulse input signals: Pulse and Direction, Clockwise/Counterclockwise, and Quadrature (AB phase). See P1.000 for details. The SureServo2 can receive pulse commands of up to 4 Mpps (MHz).

You can also accomplish position control using the internal **R**egister (P**R** mode) without the external high speed pulse command. The SureServo2 provides 99 command registers with two input modes. You can set the 99 registers first before switching the drive to Servo On status and then set DI.POS0 – DI.POS6 of CN1 to call the registers. Or, you can directly set the register values through communication.

6.2.1 - Position command in PT mode

The PT Position command is the pulse input from the **T**erminal block. There are three pulse types - Pulse and Direction, Clockwise/CounterClockwise (CW/CCW), and Quadrature (A/B); and each type has positive and negative logic that you can set in parameter P1.000. Please refer to Chapter 8 for more details.

Parameter	Function
P1.000	External pulse input type

6.2.2 - Position command in PR mode

You select the P**R** command source with (P6.000, P6.001) – (P7.098, P7.099), which are the 99 built-in command **R**egisters. Then, you trigger the Position command with DI.CTRG (0x08). See the following table for more detail.

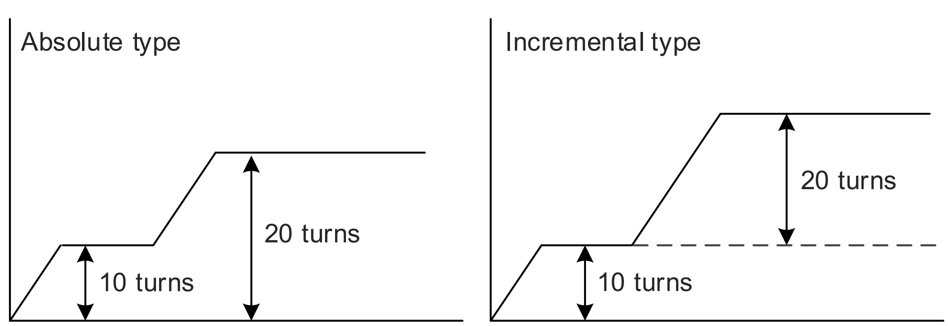
Position Command	*POS6 DI-0x1E*	*POS5 DI-0x1C*	*POS4 DI-0x1B*	*POS3 DI-0x1A*	*POS2 DI-0x13*	*POS1 DI-0x12*	*POS0 DI-0x11*	*CTRG DI-0x08*	*Setting Parameter*
Homing	0	0	0	0	0	0	0	↑	P6.000 P6.001
PR1	0	0	0	0	0	0	1	↑	P6.002 P6.003
~									~
PR49	0	1	1	0	0	1	0	↑	P6.098 P6.099
PR50	0	1	1	0	0	1	1	↑	P7.000 P7.001
~									~
PR99	1	1	0	0	0	1	1	↑	P7.098 P7.099

State of POS0 – POS6: 0 signifies that DI is off; 1 signifies that DI is on. See section 8.4.9 for information on Digital Input functions.

CTRG↑: this signifies the moment that DI is switched from off to on.

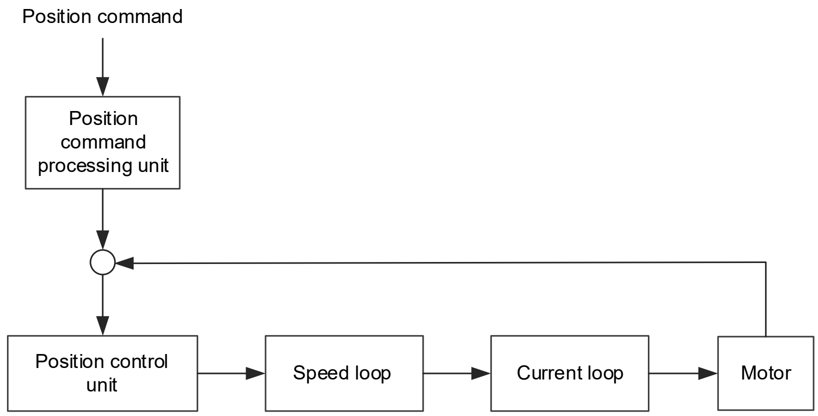
Note: *Instead of selecting the Path and triggering the command by DIs, PR mode can also be initiated by communication. See P5.007 (Modbus/ModTCP) and P5.112 (EtherNet/IP).*

There are many applications for both absolute type and incremental type registers. For example, assume the Position command PR1 is 10 turns and PR2 is 20 turns. The PR1 command is issued first and the motor moves 10 turns. When the PR statement is set for an Absolute type move, the motor will move another 10 turns (it will be at “20” revolutions). If the PR statement was set for Incremental type moves, the motor would rotate an additional 20 turns. The following diagram shows the difference between absolute and incremental positioning. Each set of PR command registers contains the target distance for the move, whether the move is Incremental or Absolute, and several other advanced features. See Chapter 7 for more details. SureServo2 PRO software simplifies setting up the PR command registers.

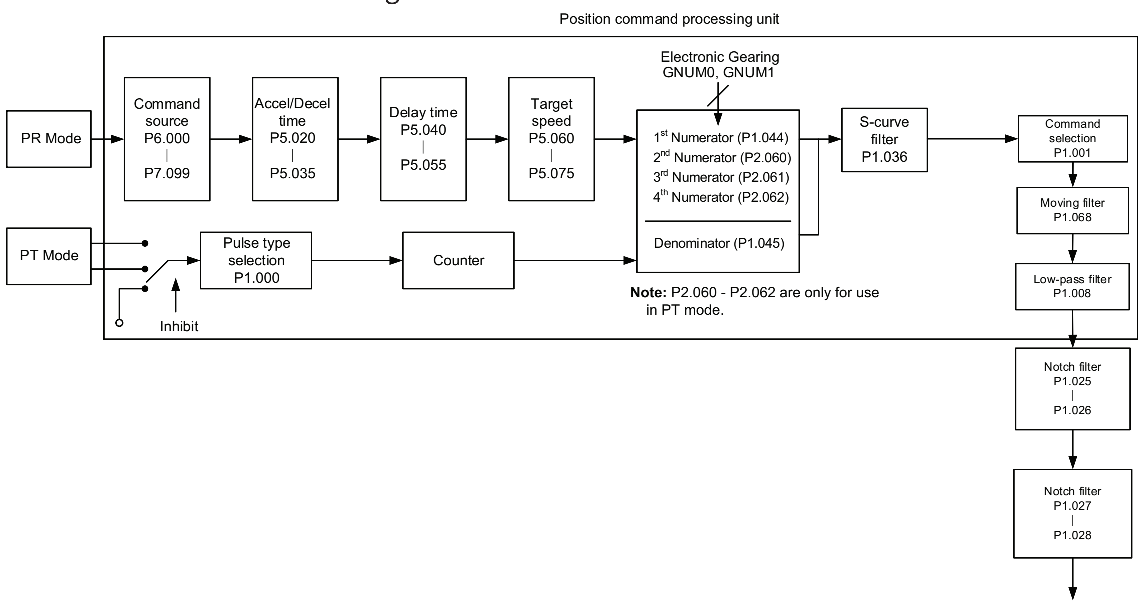


6.2.3 - Control structure of Position mode

The basic control structure is shown in the following flow chart:



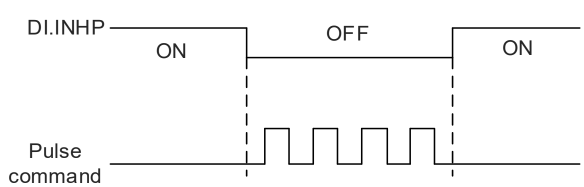
For better control, the pulse signals are processed by the Position Command processing unit. The structure is shown in the diagram below.



The upper path of the above diagram is the PR mode and the lower one is the PT mode that you can select with P1.001. You can set E-Gear ratio in both modes to adjust the positioning resolution. In addition, you can use either an S-curve or low-pass filter to smooth the command (described below).

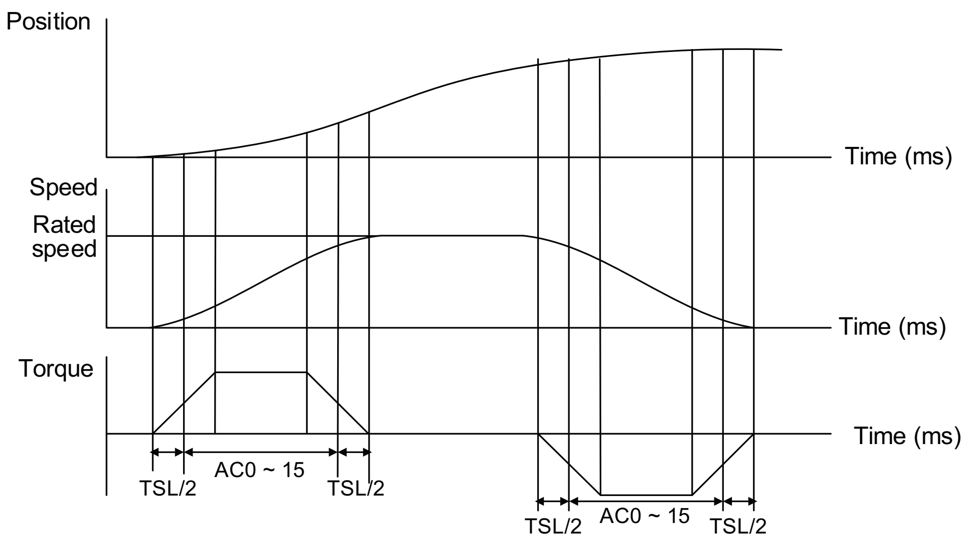
<u>The Pulse Command Input Inhibit (INHP) function</u>

In PT mode, when DI.INHP is on, the servo drive stops receiving external pulse commands and the motor stops running. As this function is only supported by DI 8, setting 0x45 (DI.INHP) to P2.017 (DI 8) is required. This feature is useful when the servo needs to follow an encoder, but also needs to be started and stopped.



6.2.4 - S-curve filter (Position)

The S-curve filter smoothes the motion command in PR mode. This filter softens the speed and acceleration curves and reduces jerk, resulting in a smoother mechanical operation. If the load inertia increases, the motor operation is influenced by friction and inertia when the motor starts or stops rotating. Setting a larger acceleration / deceleration constant for the S-curve (TSL) and for the acceleration / deceleration time in P5.020 – P5.035 can increase the smoothness of operation. When the Position command source is pulse, the speed and angular acceleration are continuous, and the S-curve filter is not necessary.



Position and S-curve speed and time setting

Relevant parameters: please refer to Chapter 8 for detailed descriptions.

Parameter	Function
P1.036	S-curve acceleration / deceleration constant
P5.020 – P5.035	Acceleration / deceleration time (Number #0 – 15)

Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

6.2.5 - Electronic gear ratio (E-Gear ratio)

The resolution of SureServo2 system is 24 bits, which means that the drive generates 16,777,216 pulses per motor rotation. The drive output to the motor always operates in 24-bit for optimum control. The input pulse per revolution resolution can be changed with electronic gearing so the host controller does not need to send 16,777,216 pulses for every rotation.

If you leave the numerator (P1.044) of the gear ratio at the default value of 16,777,216 then just changing the denominator (P1.045) will give you the number of PUU (pulse user units) you need for 1 revolution. EX: If P1.045= 15000 and P1.044 is default then 15000 pulses will result in 1 shaft rotation.

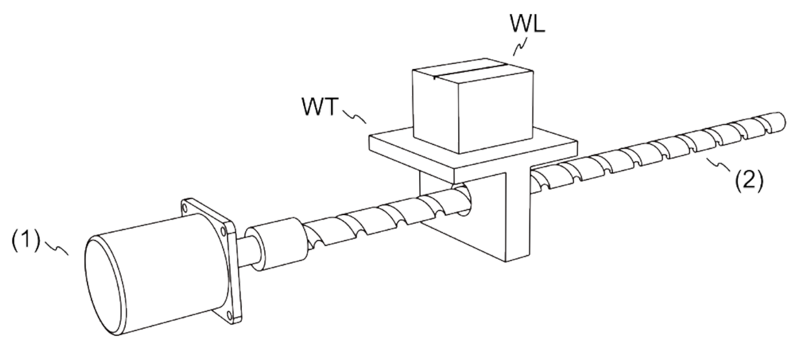
Example 1:

If the E-Gear ratio is 1, the drive will require 16,777,216 input pulses per one motor rotation and send 16,777,216 pulses to the motor to complete this rotation.

Example 2:

If the E-Gear ratio is 0.5 (1/2), then the drive will require 8,388,608 (16,777,216 divided by 2) input pulses per one motor rotation, but the drive still sends 16,777,216 pulses to the motor. In this E_Gear ratio every one pulse from the command controller (PLC) corresponds to two pulses to the motor.

A large E-Gear ratio might create a sharp corner in the profile and lead to a high jerk. To solve this problem, you can apply an S-curve acceleration / deceleration filter, or a low-pass filter to reduce the jerk.



(1) Motor (2) Ball screw pitch: 3mm (equals 3000μm)
WL: Workpiece; WT: platform

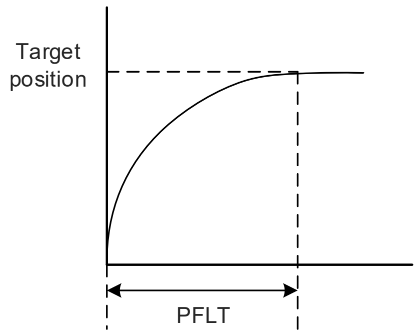
	Gear Ratio	Moving distance per 1 pulse command
Electronic gear is not applied	= 1 / 1	$= \dfrac{3000\ \frac{\mu m}{rev}}{16777216\ \frac{Pulse}{rev}} \times \dfrac{1}{1} = \dfrac{3000}{16777216}$ (Unit: $\frac{\mu m}{Pulse}$)
Electronic gear is applied	= 16777216 / 3000	$= \dfrac{3000\ \frac{\mu m}{rev}}{16777216\ \frac{Pulse}{rev}} \times \dfrac{16777216}{3000} = 1$ (Unit: $\frac{\mu m}{Pulse}$)

Relevant parameters: please refer to Chapter 8 for detailed descriptions.

Parameters	Function
P1.044	E-Gear ratio (Numerator) (N1) (Default=16777216)
P1.045	E-Gear ratio (Denominator) (M) (Default=100000)

The default E-Gear settings (16,777,216 / 100,000) result in 1 motor revolution for every 100,000 input pulses.

6.2.6 - Low-pass filter

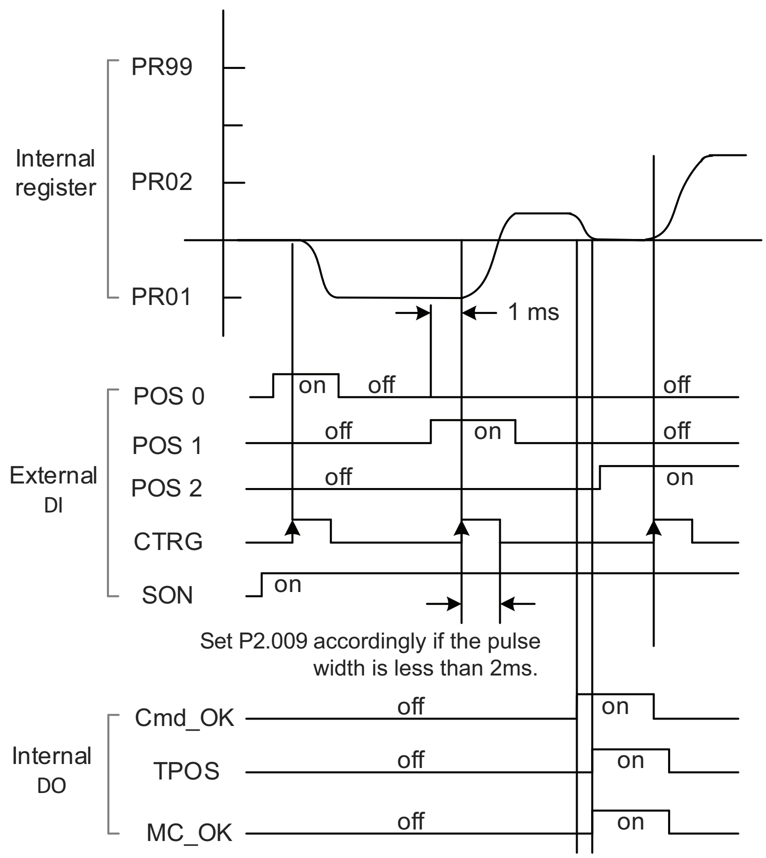


Relevant parameters: please refer to Chapter 8 for detailed descriptions.

Parameter	***Function***
P1.008	Position command smoothing constant (Low-pass filter)

6.2.7 - Timing diagram of PR mode (internal indexing)

In PR mode (**P**osition command from internal **R**egisters), the Position command is issued with DI signals (POS0 – POS6 to select the target position, and CTRG for the Command Trigger. Please refer to Section 6.2.2 for information about the DI signal and its selected register. The timing diagrams are shown below.



Note: *Cmd_OK is on when the PR command completes; TPOS is on when the error is smaller than value set by P1.054; MC_OK is on when Cmd_OK and TPOS are both on.*

6.2.8 - Gain adjustment for the position loop

There are two types of gain adjustment for the position loop: auto and manual.

- *Auto adjustment:*
  The SureServo2 servo drive provides an Auto Tuning function that allows you to easily complete the gain adjustment. Please refer to Chapter 5 Tuning for a detailed description.
- *Manual adjustment:*
  Before setting the position control loop, you have to manually set the speed control PID (P2.004 and P2.006) since a speed loop is included in the position loop. Then set the position loop gain (P2.000) and position feed forward gain (P2.002).

Description of the proportional gain and feed forward gain:

1) Proportional gain: a larger gain increases the response bandwidth of position loop.
2) Feed forward gain: reduces the deviation of phase delay.

Please note that the position loop bandwidth should not be larger than the speed loop bandwidth.

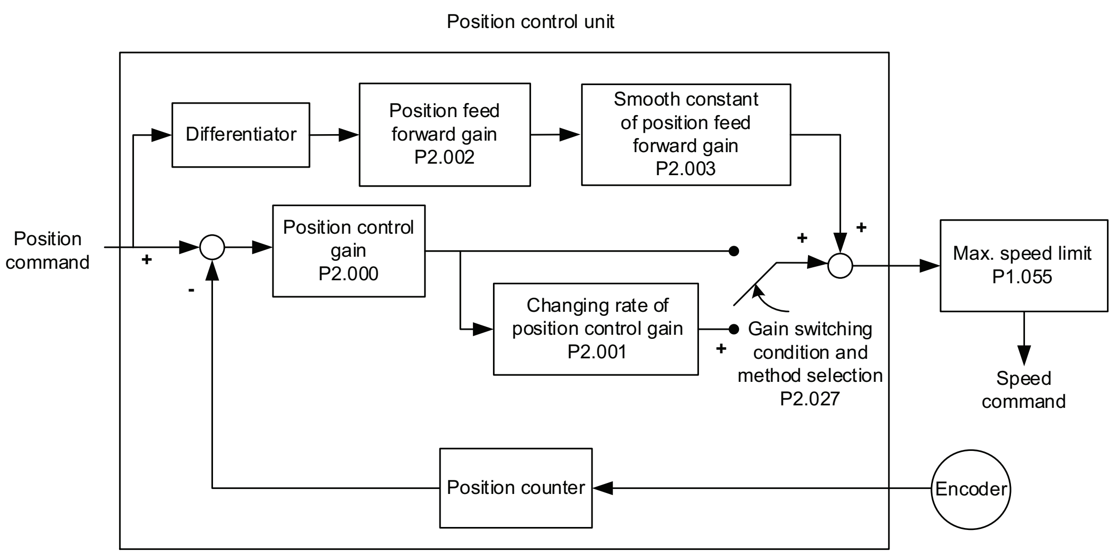
Calculation: $fp \leq \frac{fv}{4}$ (fv: response bandwidth of speed loop (Hz); fp: response bandwidth of position loop (Hz).)

Position Loop Gain setting (P2.000): KPP = $2 \times \pi \times fp$

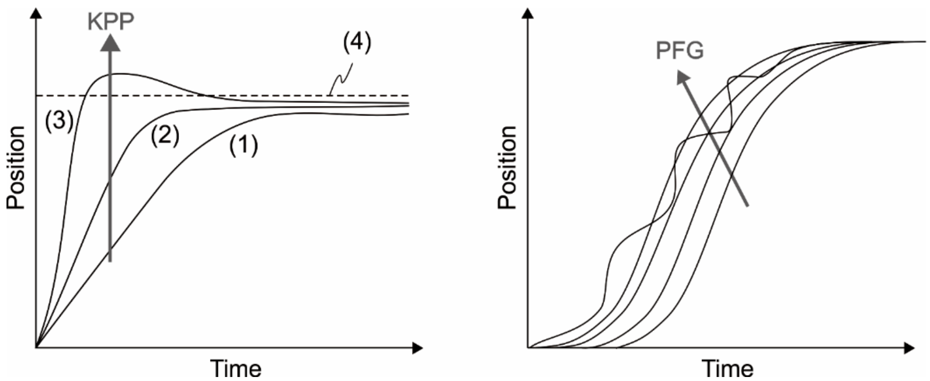
Example: if the desired position bandwidth is 20Hz, then adjust KPP (P2.000) to 125.
($2 \times \pi \times 20Hz = 125$)

Relevant parameters: please refer to Chapter 8 for detailed descriptions.

Parameter	Function
P2.000	Position control gain
P2.002	Position feed forward gain



When you set the value of KPP (Position Control Gain: P2.000) too high, the bandwidth for the position loop is increased and the phase margin is reduced. Meanwhile, the motor rotates and vibrates in the forward and reverse directions. In this case, you should decrease KPP until the rotor stops vibrating. When the external torque is too high, a too-low value for KPP cannot meet the demand of reducing position error. In this case, increasing ***Position Feed Forward Gain***, PFG (Position Feed Forward Gain: P2.002), can effectively reduce the following error.



The actual position curve changes from (1) to (3) with the increase in the KPP value. (4) stands for the Position command.

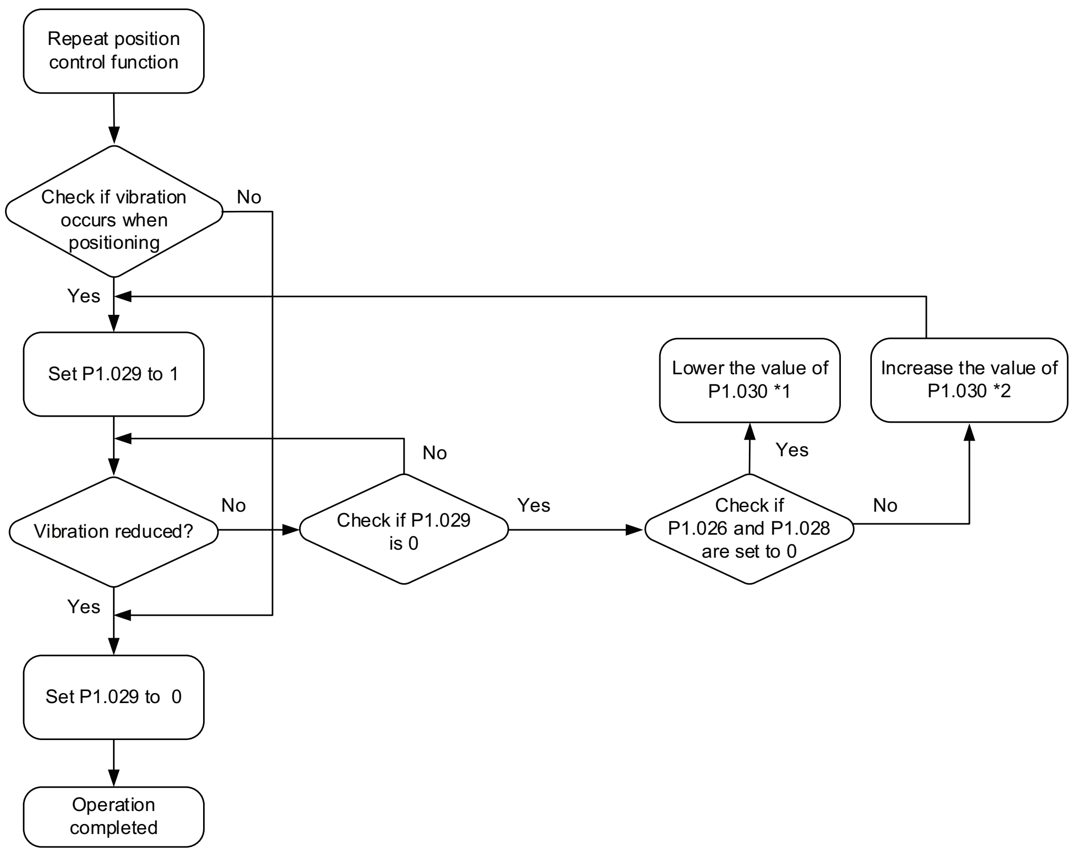
6.2.9 - Low-frequency vibration suppression in Position mode

If the machine is too flexible, vibration persists even when the motor stops after executing the Positioning command. The low-frequency vibration suppression (LFVS) function can reduce the machine vibration. The suppression range is between 1.0 Hz and 100.0 HZ. Both auto and manual functions are available.

Auto Low-frequency Vibration Suppression Mode (P1.029)

If you have difficulty finding the resonance frequency, you can enable the Auto LFVS function (P1.029), which searches for the specific resonance frequency. If you set P1.029 to 1, the system disables any existing LFVS and starts to search the vibration frequency. When the detected resonance frequency remains at the same level, it automatically sets Auto LFVS P1.029 to 0 and sets LFVS Frequency (1) P1.025 to the first frequency and sets LFVS Gain (1) P1.026 to 1. It sets LFVS Frequency (2) P1.027 to the second frequency and then sets LFVS Gain (2) P1.028 to 1. If Auto LFVS P1.029 is automatically reset to 0, but the low-frequency vibration persists, please check that the P1.026 or P1.028 gains are enabled. If the values of P1.026 and P1.028 are both 0, it means no frequency is detected. Please lower the LFVS detection value in P1.030 and set P1.029 to 1 to search the vibration frequency again. Please note that when you set the detection level too low, it might detect noise as low-frequency vibration.

The diagram of the basic control structure is shown in the following flowchart:



Notes:

1) When the gain values of P1.026 and P1.028 are 0, it means that the vibration frequency cannot be found, probably because the detection level is set too high so that the low-frequency vibration is not detected.
2) When the gain values of P1.026 or P1.028 are greater than 0 and the vibration is not reduced, it is probably because the detection level is set too low, and the system detects noise or other frequency as low-frequency vibration.

3) When the auto suppression procedure completes, but the vibration persists, you can manually set LFVS frequencies P1.025 or P1.027 to suppress the vibration if you have identified the problematic low frequency.
4) Physically lowering the inertia ratio of the motor to the machine's power transmission can reduce resonance along with stiffer couplings.

Relevant parameters: please refer to Chapter 8 for detailed descriptions.

Parameter	***Function***
P1.029	Auto low-frequency vibration suppression mode
P1.030	Low-frequency vibration detection level

P1.030 sets the detection range for the magnitude of low-frequency vibration. When the frequency is not detected, it is probably because you set the value of P1.030 too high and it exceeds the vibration range. In this case, it is suggested that you decrease the value of P1.030. Please note that if the value is too small, the system might detect noise as the resonance vibration frequency. You can also use the software Scope in SureServo2 Pro to observe the range of position error (pulse) between the upper and lower magnitude of the curve to adjust the value of P1.030.

Manual Setting

There are two sets of low-frequency vibration suppression: one is parameters P1.025 – P1.026 (LFVS frequency 1) and the other is parameters P1.027 – P1.028 (LFVS frequency 2). You can use these two sets of low-frequency vibration suppression parameters to reduce two different frequency vibrations. Use parameters P1.025 and P1.027 to suppress the low-frequency vibration. The function works only when the low-frequency vibration setting is close to the real vibration frequency. Use gain parameters P1.026 and P1.028 to set the response after frequency filtering. The bigger the values of P1.026 and P1.028, the better the response. However, if you set the values too high, the motor might not operate smoothly. The default values of parameters P1.026 and P1.028 are 0, which means the two filters are disabled by default.

Relevant parameters: please refer to Chapter 8 for detailed descriptions.

Parameter	***Function***
P1.025	Low-frequency vibration suppression frequency (1)
P1.026	Low-frequency vibration suppression gain (1)
P1.027	Low-frequency vibration suppression frequency (2)
P1.028	Low-frequency vibration suppression gain (2)

6.3 - Speed mode

This servo drive includes two types of command inputs: analog and internal register (parameters). The Analog command controls the motor speed by scaled external voltage input. The command register input controls the speed in two ways. The first is to set different speed values in three command registers and then switch the speed by using Digital Inputs SPD0 and SPD1 from CN1. The second is to change the value in the registers by communication. In order to deal with the problem of non-continuous speed when switching registers, you can use the S-curve acceleration / deceleration filter. In a closed-loop system, the servo drive uses gain adjustment, the integrated PI controller, and the two modes (Manual and Auto).

You can use Manual mode to manually set the parameters. In this mode, all auto or auxiliary function are disabled. The gain adjustment function provides different modes for you to estimate load inertia and tune the bandwidth as well as the responsiveness. In addition, the parameter values you set are regarded as the default values. It is a very good idea to apply a torque limit when in speed mode to ensure the torque of the motor does not exceed a safe value of the application. See “6.6.1 - Applying the speed limit” on page 6–31.

6.3.1 - Selecting the Speed command source

There are two types of Speed command sources: analog voltage and internal speed registers (parameters). There are 2 Speed control modes in P1.001:

Mode	P1.001 Setting	Description
S	02	Speed Mode (velocity command = analog input or from internal speed registers)
Sz	04	Speed-Zero Mode (velocity command = 0 or from internal speed registers)

You can select the source by using DI signals from CN1. See the following table for the command source selection:

Speed Command	CN1 DI Signal		Command Source			Content	Range
	SPD1	SPD0					
S0	0	0	Mode	S	External analog signal	Voltage difference between V-REF and GND	-10V to +10V
				Sz	N/A	Speed command is 0	0
S1	0	1	Register parameters			**P1.009 Speed 1**	-60000 – 60000
S2	1	0				**P1.010 Speed 2**	-60000 – 60000
S3	1	1				**P1.011 Speed 3**	-60000 – 60000

Status of SPD0 – SPD1:

0 means that DI is off; 1 means that DI is on.

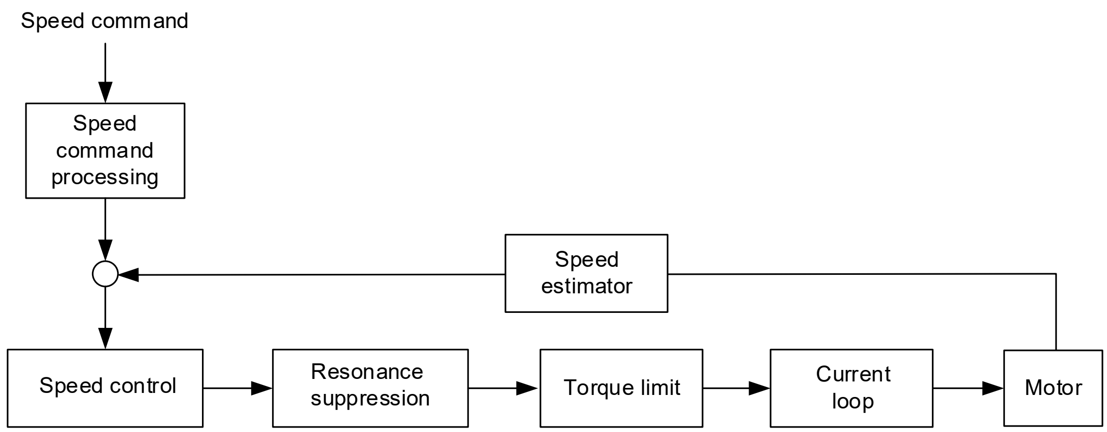
- *When both SPD0 and SPD1 are 0:*
    - *Sz mode: the analog velocity command is void. Thus, if the Speed command using analog voltage type is not required, you can use Sz mode to address the problem of zero drift in the voltage.*
    - *S mode: the voltage command is the voltage deviation between V-REF and GND. The range of the input voltage is between -10V and +10V and you can adjust the scaling by changing the “Maximum Rotation Speed for Analog Speed Command” (P1.040).*
- *When either SPD0 or SPD1 is not 0, the Speed command comes from the internal registers. The command is activated once the status of SPD0 – SPD1 is changed. There is no need to use CTRG for triggering.*

- *The parameter setting range (internal registers) is -60000 – 60000.*
  Setting value = setting range x unit (0.1 rpm). For example, if P1.009 = +30000, then rotation speed = +30000 x 0.1 rpm = +3000 rpm

Use the Speed command registers in Speed mode (S or Sz) to set the command speed. Use the Speed command registers in Torque mode (T or Tz) to set the speed limit. See section "6.6.2 - Applying the torque limit" for detailed explanations on limiting the torque in speed mode.

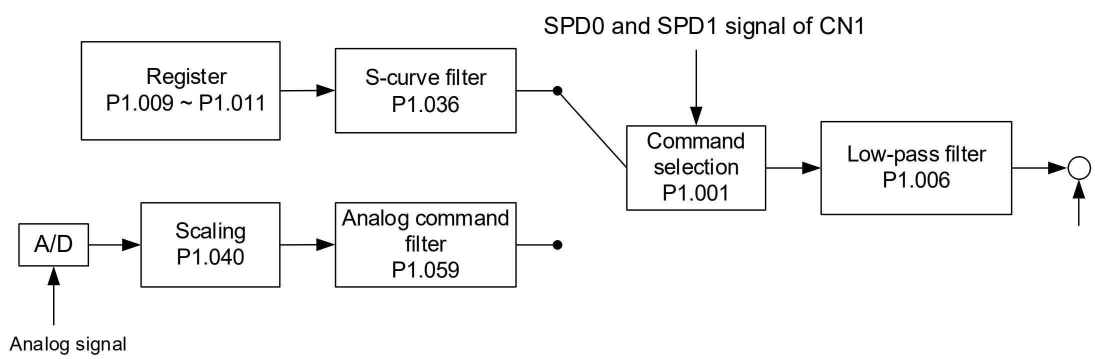
6.3.2 - Control structure of Speed mode

The diagram of the basic control structure is shown in the following flowchart:



The Speed Command processing unit selects the command source (see Section 6.3.1), including the scaling parameter (P1.040) for rotation speed and S-curve parameter for smoothing the speed. The Speed Control unit manages the gain parameters for the servo drive and calculates the current command for servo motor in real-time. The Resonance Suppression unit suppresses the resonance of the machine.

The following diagram introduces the function of the Speed Command unit. Its structure is shown below.



Control in the upper path is from the Speed control registers while the lower path command signal is from the external analog voltage. The command is selected according to the status of SPD0, SPD1 and P1.001 (S or Sz). In this condition, the S-curve and low-pass filters are applied to achieve a smoother response.

6.3.3 - Smooth Speed command

S-curve filter

During the process of acceleration or deceleration, the S-curve filter uses the three-stage acceleration curve and creates a smoother motion trajectory. It avoids jerk (rapid change of acceleration), resonance, and noise caused by abrupt speed variation. You can use the acceleration time constant P1.034 (TACC) to adjust the slope of the change in acceleration; the S-curve deceleration constant P1.035 (TDEC) adjusts the slope of the change in deceleration; and the S-curve acceleration / deceleration constant P1.036 (TSL) improves the status of motor activating and stopping. This can also calculate the total time for executing the command.

Note: *The S-Curve acceleration/deceleration constant P1.036 (TSL) must be greater than 0 for TACC and TDEC to take effect.*

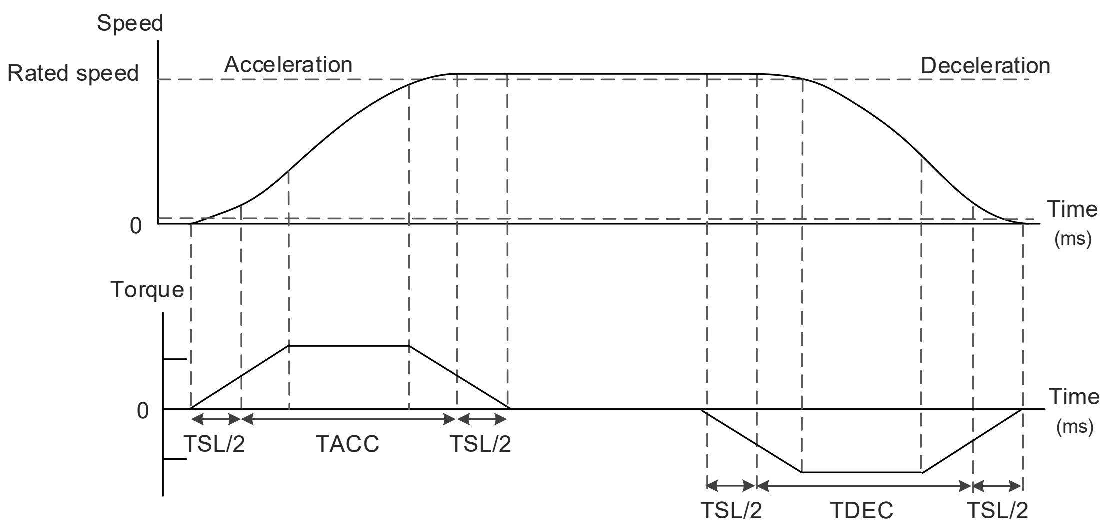


Figure 6-1 S-curve and time setting

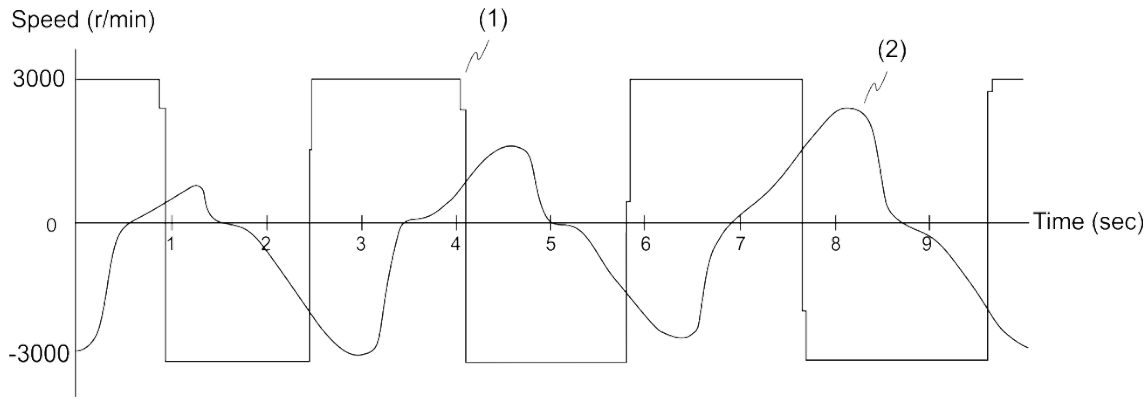
Note: *The graph is from 0 to Rated Speed (not 0 to commanded speed). The Calculated acceleration or deceleration time from 0 - rated speed will not be the same as the time from 0 - any commanded value less than Rated speed.*

Relevant parameters: refer to Chapter 8 for more information.

Parameter	*Function*
P1.034	Acceleration constant (TACC)
P1.035	Deceleration constant (TDEC)
P1.036	S-curve acceleration / deceleration constant (TSL)

Analog Speed command filter

The Analog Speed Command filter helps to stabilize the motor operation when the analog input signal (speed) changes rapidly.



(1) Analog Speed command (2) Motor torque

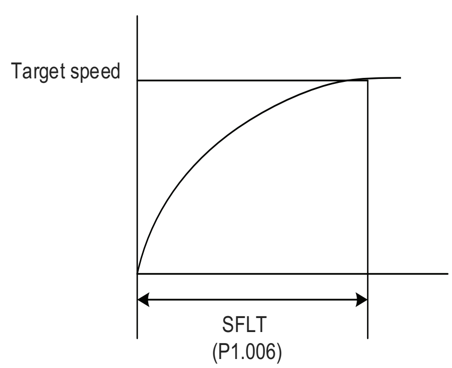
The Analog Speed Command filter smooths the analog input signal. Its time program is the same as the S-curve filter at normal speed. Also, the speed and acceleration curves are both continuous. The above graph shows the curve of the Speed command and the motor torque when you apply the Analog Speed Command filter. In the diagram above, the slopes of the Speed command in acceleration / deceleration are different. You can adjust the time setting (P1.034, P1.035, and P1.036) according to the actual application to improve the performance.

Low-pass filter for commands

You usually use the low-pass filter to remove unwanted high-frequency response or noise so that the speed change is smoother.

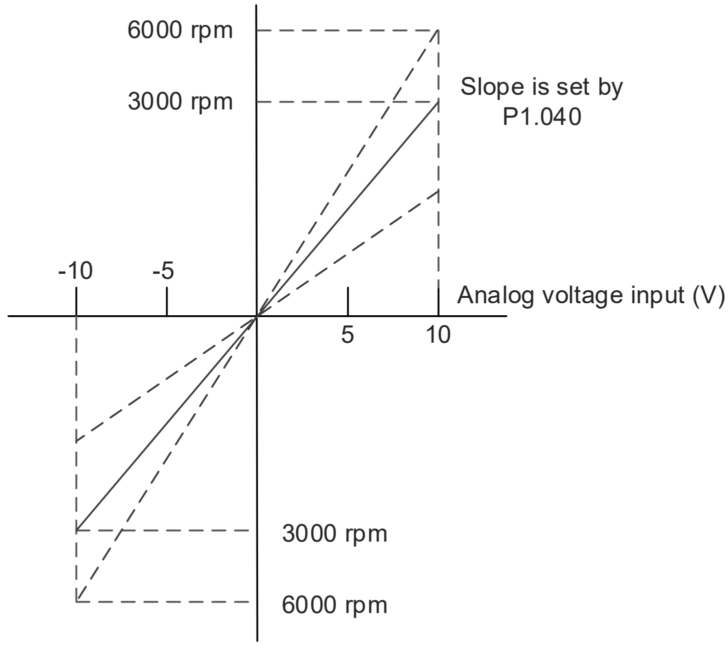
Relevant parameters: refer to Chapter 8 for more information.

Parameter	Function
P1.006	Speed command smoothing constant (Low-pass filter)



6.3.4 - Scaling of the analog command

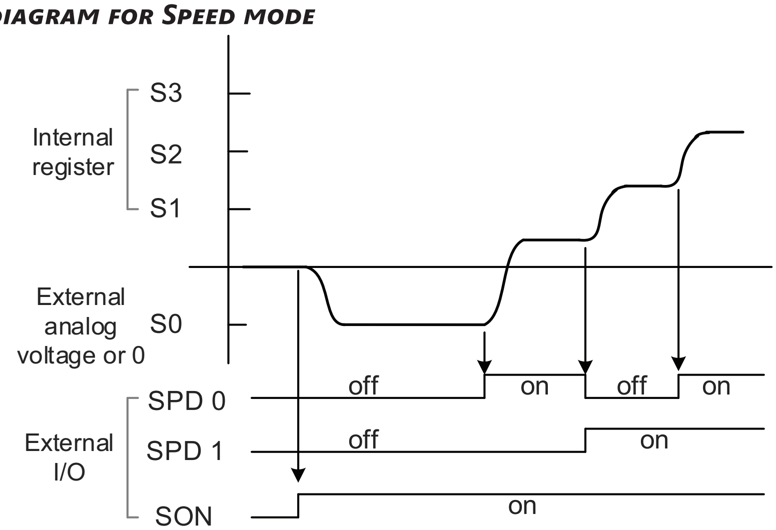
In Analog mode, you control the motor Speed command by the analog voltage difference between V_REF and GND. Use parameter P1.040 (maximum rotation speed for the analog Speed command) to adjust the slope of the speed change and its range.



Relevant parameters: refer to Chapter 8 for more information.

Parameter	Function
P1.040	Maximum rotation speed for the analog Speed command

6.3.5 - Timing diagram for Speed mode

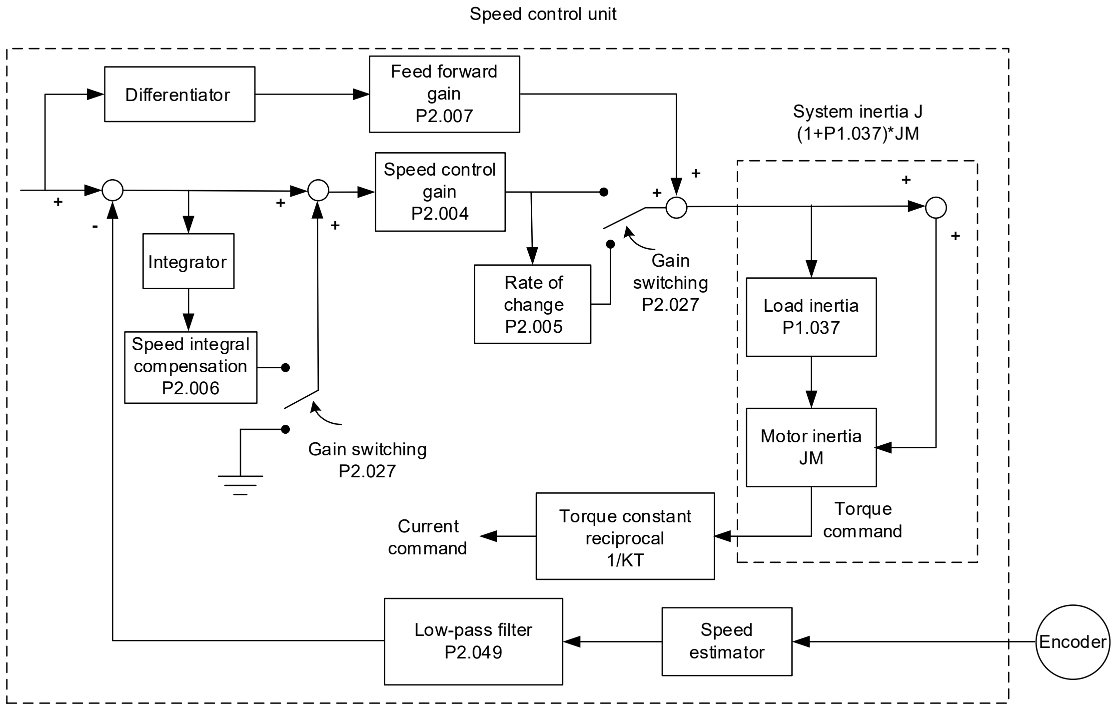


Notes:

1) "Off" means that the contact is open while "On" means that the contact is closed.
2) In Servo On state, the command is selected by the state of SPD0 – SPD1.
3) When in Sz mode (P1.001=4), the Speed command S0 = 0rpm; when in S mode (P1.001=2), the Speed command S1 is the external analog voltage input.

6.3.6 - Gain adjustment of the speed loop

The structure of the speed control unit appears in the following diagram:



In the Speed Control unit, you can adjust different types of gain. You can adjust the gain manually or use the three gain adjustment modes provided.

Manual: you set values for all the parameter settings. Auto and auxiliary functions are disabled.

For the other Gain Adjustment modes: please refer to Chapter 5 Auto tuning.

Manual mode

When you set Gain Adjustment Mode P2.032 to 0 (manual mode), you must set the Speed Loop gain (P2.004), Integral Compensation (P2.006), and Feed Forward gain (P2.007).

Speed Loop gain: the higher the gain, the larger the bandwidth for the speed loop response.

Integral Compensation gain: increasing this gain increases the low frequency rigidity and reduces the steady-state error. However, the phase margin is smaller. If you set this gain too high, you reduce the system stability.

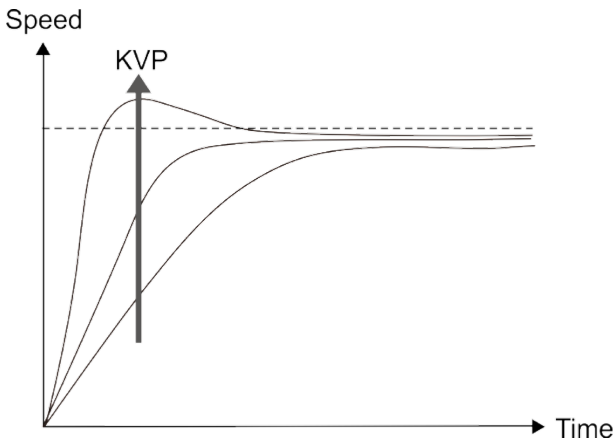
Feed Forward gain: reduce the deviation of the phase delay.

Relevant parameters: refer to Chapter 8 for more information.

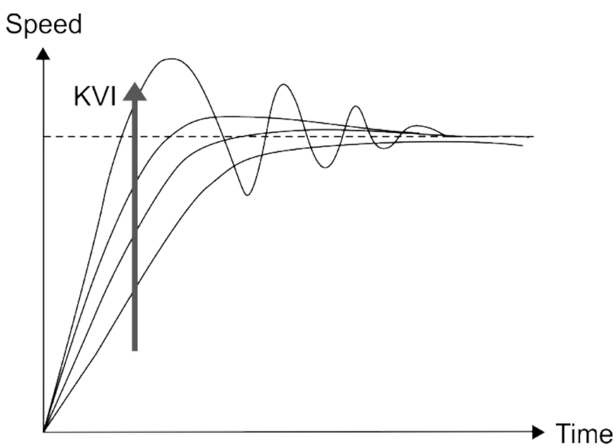
Parameter	Function
P2.004	Speed control gain (KVP)
P2.006	Speed integral compensation (KVI)
P2.007	Speed feed forward gain (KVF)

Theoretically, a step response can be used to explain proportional gain (KVP – P2.004), integral gain (KVI – P2.006), and feed forward gain (KVF – P2.007). Here, the frequency domain and time domain are used to illustrate the basic principle.

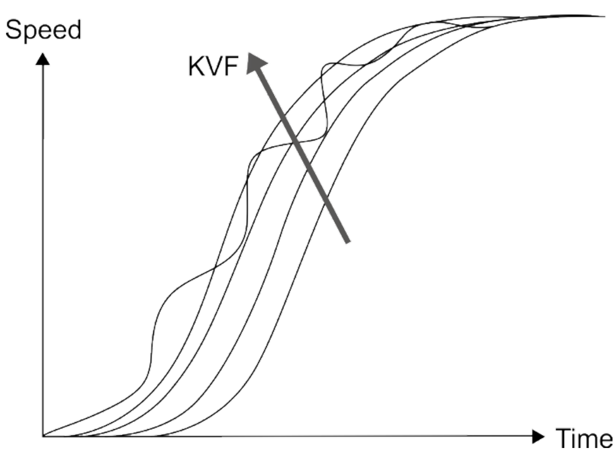
Frequency domain



The higher the KVP value, the larger the bandwidth. The time of the speed increase will also be shorter. However, if the value is set too high, the phase margin is too small. The effect is not as good as KVI for the steady-state error but is better for the effect for the following error.



The higher the KVI value, the larger the low frequency gain. It shortens the time for the steady-state error to reduce to zero. However, it does not significantly reduce the following error.



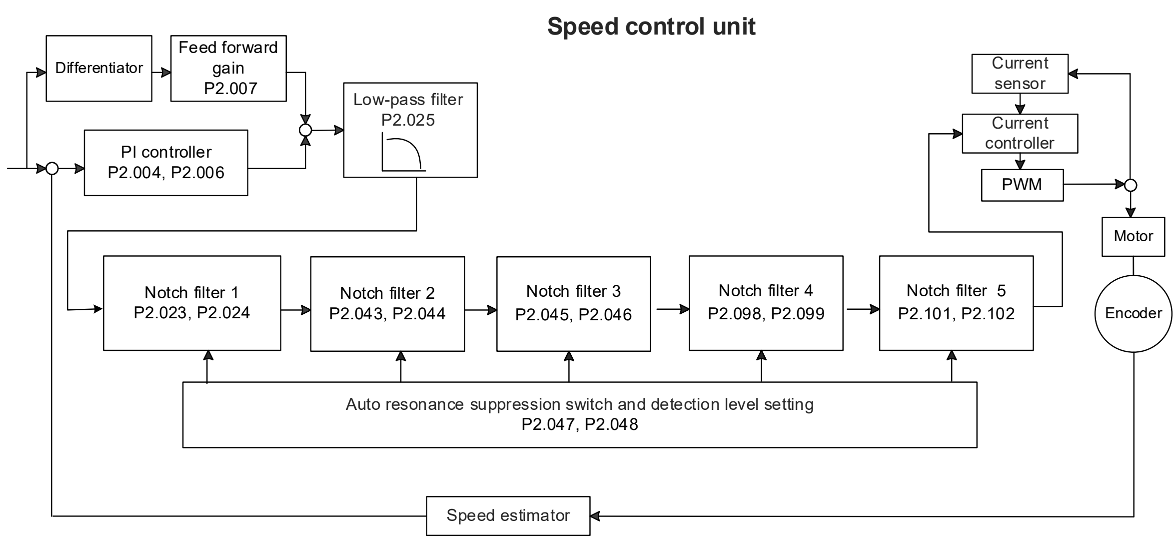
The closer the KVF value is to 1, the more complete the forward compensation. The following error becomes very small. But a KVF value that is set too high also causes vibration.

6.3.7 - Resonance Suppression unit

When resonance occurs, it is probably because the stiffness of the control system is too high or the response is too fast. Eliminating these two factors can improve the situation. In addition, you can use the low-pass filter (parameter P2.025) and Notch filter (parameters P2.023, P2.024, P2.043 – P2.046, P2.095 – P2.103) to suppress the resonance if you want the control parameters to remain unchanged. Each Notch filter consists of a Notch filter frequency (Hz) and a Notch filter attenuation level (-dB)

Relevant parameters: refer to Chapter 8 for more information.

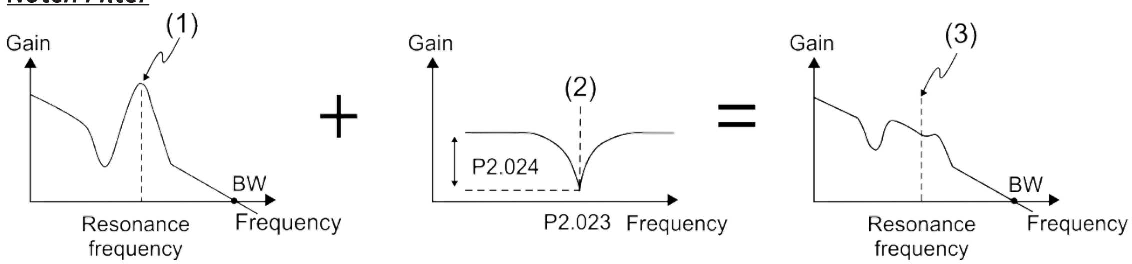
Parameter	Function
P2.023	Notch filter frequency (1)
P2.024	Notch filter attenuation level (1)
P2.043	Notch filter frequency (2)
P2.044	Notch filter attenuation level (2)
P2.045	Notch filter frequency (3)
P2.046	Notch filter attenuation level (3)
P2.095	Notch filter bandwidth (1)
P2.096	Notch filter bandwidth (2)
P2.097	Notch filter bandwidth (3)
P2.098	Notch filter frequency (4)
P2.099	Notch filter attenuation level (4)
P2.100	Notch filter bandwidth (4)
P2.101	Notch filter frequency (5)
P2.102	Notch filter attenuation level (5)
P2.103	Notch filter bandwidth (5)
P2.025	Resonance suppression low-pass filter



SureServo2 provides two types of resonance suppression: one is the Notch filter and the other is the low-pass filter. See the following diagrams for the results of using these filters.

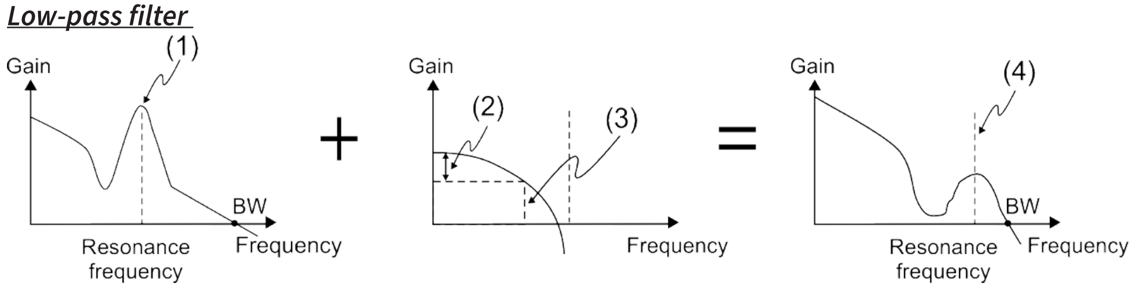
System open-loop gain with resonance:

Notch Filter



(1) Point of resonance (2) Notch filter (3) Point of resonance suppressed by the Notch filter

Low-pass filter

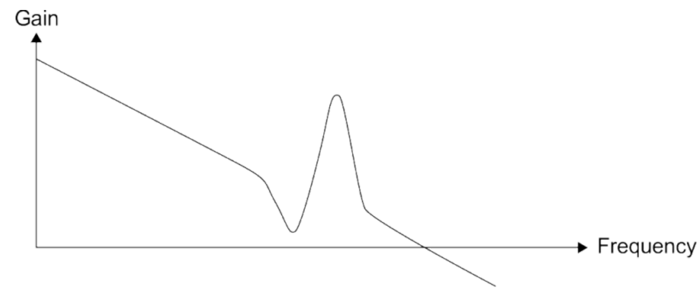


(1) Point of resonance (2) Attenuation rate (-3 dB)
(3) Low-pass filter (Cutoff frequency of low-pass filter = 1000 / P2.025 Hz)
(4) Resonance point suppressed by the low-pass filter

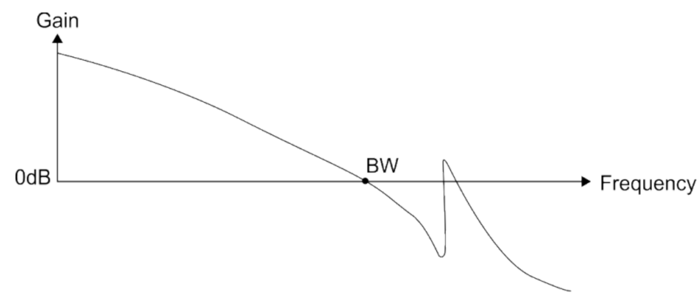
To conclude from these two examples, if you increase the value of the low pass filter P2.025 from 0, the bandwidth (BW) becomes smaller. Although it solves the problem of resonance, it also reduces the response bandwidth and phase margin, and thus the system becomes unstable.

If you know the resonance frequency, you can suppress the resonance by using the Notch filter, which is better than using the low-pass filter in this condition. If the resonance frequency drifts significantly with time or due to some other cause, using the Notch filter is not suggested.

The following figure shows the system open-loop gain with resonance suppression.



When the value of P2.025 (Low-pass Filter) is increased from 0, BW becomes smaller. Although it solves the problem of the resonance frequency, the response bandwidth and phase margin are reduced. Also, the system becomes unstable.



If you know the resonance frequency, the Notch filter can eliminate the resonance directly. The frequency range of the notch filter is 50 – 5000 Hz and the suppression strength is 0 – 32 dB. If the frequency does not meet the Notch filter conditions, then using the low-pass filter to reduce the resonance is suggested.

6.4 - Torque mode

Torque Control mode (T or Tz) is suitable for torque control applications, such as printing machines and winding machines. There are two kinds of command sources: analog input and internal torque register (parameters). The analog command input uses a scaled external voltage to control the torque of the motor while the register mode uses the internal parameters (P1.012 – P1.014) for the Torque command. t is a very good idea to apply a speed limit when in torque mode to ensure the velocity of the motor does not exceed a safe value of the application. See “6.6.1 - Applying the speed limit” on page 6–31.

Mode	P1.001 Setting	Description
T	3	Torque Mode (torque command = analog input or from internal torque registers)
Tz	5	Torque-Zero Mode (torque command = 0 or from internal torque registers)

6.4.1 - Selecting the Torque command source

External analog voltage and internal parameters are the two Torque command sources. You select the command source with CN1’s DI signal. See the table below for more detail.

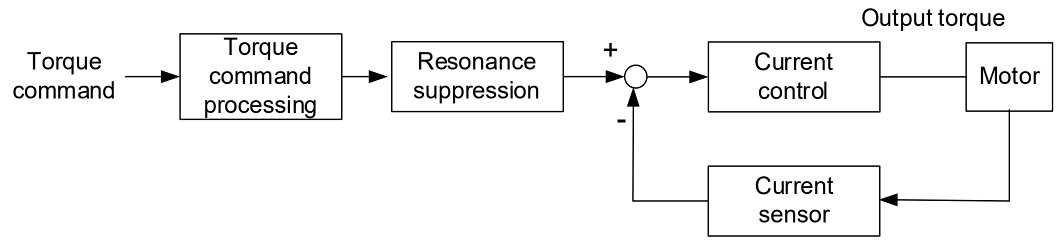
Torque Command	DI signal of CN1		Command Source			Content	Range
	TCM1	TCM0					
T0	0	0	Mode	T	External analog command	Voltage difference between T-REF and GND	-10V to +10V
				Tz	N/A	Torque command is 0	0
T1	0	1	Parameters			P1.012 Torque 1	-300% – 300%
T2	1	0				P1.013 Torque 2	-300% – 300%
T3	1	1				P1.014 Torque 3	-300% – 300%

- *State of TCM0 – TCM1: 0 means that the circuit is open (DI is off); 1 means that the circuit is closed (DI is on).*
- *When TCM0 and TCM1 are 0:*
  - *Tz mode: analog torque command is void. If there is no need to use the analog voltage for the Torque command, then Tz mode is applicable and can avoid the problem of zero voltage drift.*
  - *T mode: the command is the voltage difference between T-REF and GND. Its input voltage range is -10V to +10V, which means you can adjust the corresponding torque (P1.041).*
- *When either one of TCM0 or TCM1 is not 0, the internal parameters become the source for the Torque command. The command is executed after TCM0 – TCM1 are changed. There is no need to use CTRG for triggering.*

You can use the Torque command in Torque mode (T or Tz) and Speed mode (S or Sz). When in Speed mode, the torque registers are torque limits. See section “6.6.1 - Applying the speed limit” for detailed explanations on limiting the speed in torque mode.

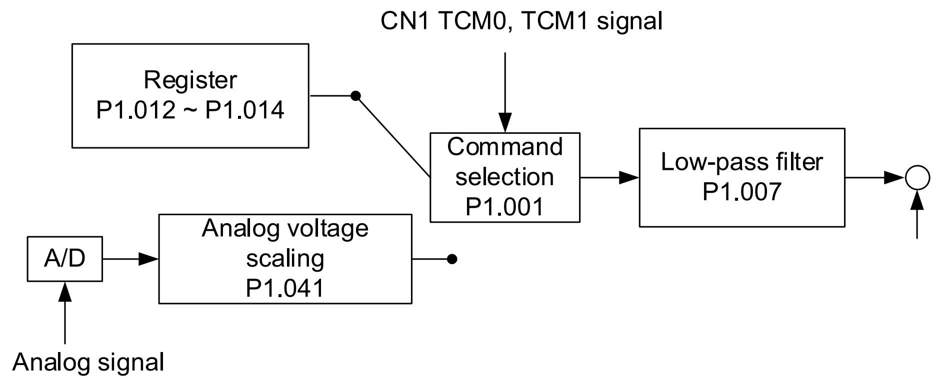
6.4.2 - Control structure of Torque mode

The following diagram shows the basic control structure of Torque mode:



Use the Torque Command unit to specify the Torque command source (mentioned in Section 6.4.1), including the scaling of the analog voltage (P1.041) and the S-curve setting. The current control unit manages the gain parameters for the servo drive and calculates the current for servo motor in real-time; you can only set this by commands.

The structure of Torque Command unit is as the follows:



Control in the upper path is from the torque registers while the lower path control is from the external analog voltage. You select the command according to the status of digital inputs TCM0, TCM1, and Control Mode P1.001 (T or Tz).

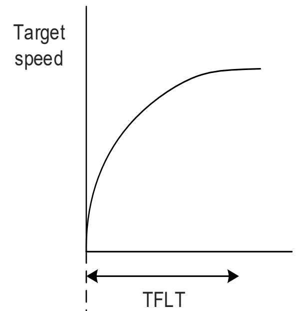
You can adjust the torque with the analog voltage scaling (P1.041) and you can smooth the response with the low-pass filter (P1.007).

6.4.3 - Smooth Torque command

Relevant parameters: refer to Chapter 8 for more information.

Setting the torque command low pass filter to 0ms disables the filter. Setting P1.007 to any other value smooths the torque command transition.

Parameter	Function
P1.007	Torque command smoothing constant (Low-pass filter)

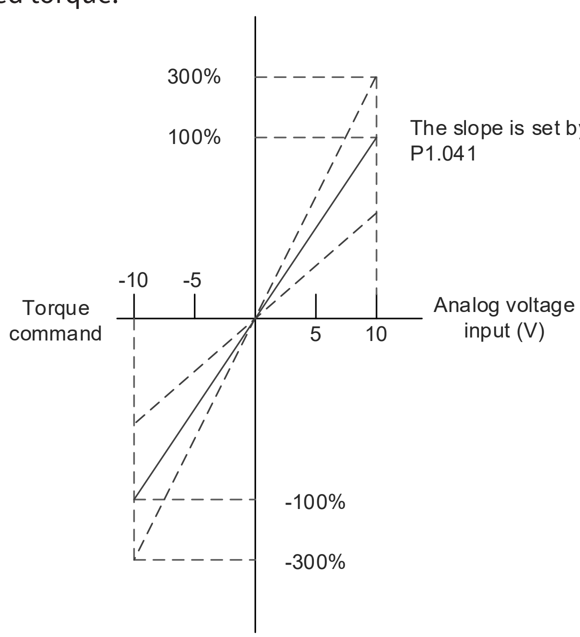


6.4.4 - Scaling of the analog command

The Torque command is controlled by the analog voltage difference between T_REF and GND. You can adjust the torque slope and its range with parameter P1.041 (Maximum output for analog Torque command). This parameter is entered in percent.

For example:

1) If you set P1.041 to 100 and the external input voltage is 10V, the Torque command is 100% of the rated torque.
2) If you set P1.041 to 300 and the external input voltage is 10V, the Torque command is 300% of the rated torque.

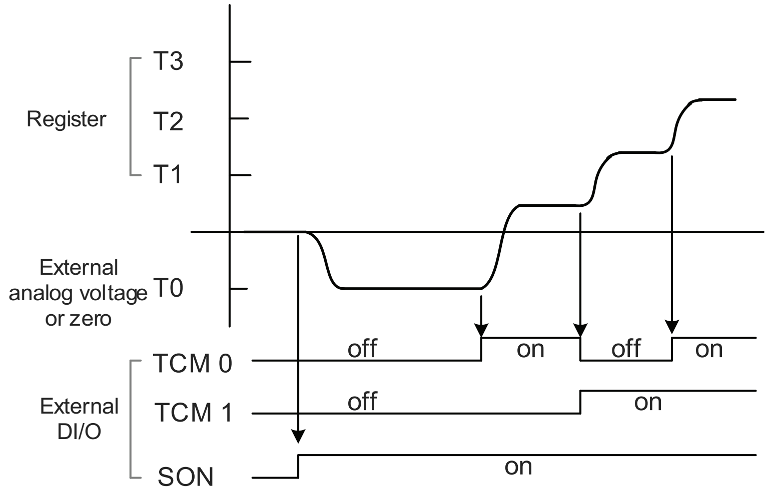


Relevant parameters: refer to Chapter 8 for more information.

Parameter	Function
P1.041	Maximum output for analog Torque command

Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

6.4.5 - Timing diagram in Torque mode



Notes:

1) “Off” signifies the contact is open while “On” signifies the contact is closed.
2) When in Tz mode, the Torque command T0 = 0; when in T mode, the Torque command T0 is the external analog voltage input.
3) In Servo On state, the command is selected according to the state of TCM0 – TCM1.

6.5 - Dual mode

Apart from single modes for position, speed, and torque control, there are eight dual / multiple modes also provided for operation (See Section 6.1).

Mode	Short Name	P1.001 Control Setting	Description
Dual mode	PT-S	06	PT and S can be switched with DI signal, DI.S-P (0x18).
	PT-T	07	PT and T can be switched with DI signal, DI.T-P (0x20).
	PR-S	08	PR and S can be switched with DI signal, DI-S-P (0x18).
	PR-T	09	PR and T can be switched with DI signal, DI.T-P (0x20).
	S-T	0A	S and T can be switched with DI signal, DI.S-T (0x19).
	PT-PR	0D	PT and PR can be switched with DI signal, DI.PT-PR (0x2B).
Multiple mode	PT-PR-S	0E	PT, PR, and S can be switched with DI signal, S_P and PT_PR.
	PT-PR-T	0F	PT, PR, and T can be switched with DI signal, T_P and PT_PR.

Sz and Tz (Speed and Torque Modes with a 0 selection and no analog input) dual mode is not supported. To avoid occupying too many digital inputs in dual mode, Speed and Torque modes can use the external analog voltage as the command source instead of the Speed and Torque parameters to reduce the use of DI points (SPD0, SPD1 or TCM0, TCM1). In addition, Position mode can use the pulse input to reduce the use of DI points (POS0, POS1, POS2, POS3, POS4, POS5, and POS6). Also, PR mode can be executed with communications (saving even more DI points). See P5.007 for more details. Please refer to Section 3.4.2 for the table of DI/O default value in each mode.

If you want to change the settings, the DI/O signals in correspondence with the PINs are defined as above in Section 3.4.2.

6.5.1 - Speed / Position dual mode (PT-S, PR-S, PT-PR)

P**T**-S and P**R**-S are available in Speed / Position dual mode. The command source for P**T**-S comes from the external **T**erminals (pulses) while the source for P**R**-S comes from the internal **R**egisters (P6.000 – P7.099). You can control the Speed command with the external analog voltage or the internal parameters (P1.009 – P1.011). The switch for Speed / Position mode is controlled by DI.S-P (0x18) signal. The switch for PT and PR for Position mode is controlled by DI.PT-PR (0x2B). Thus, you select both Position and Speed commands in PR-S mode with the DI signal. The timing diagram is shown below.

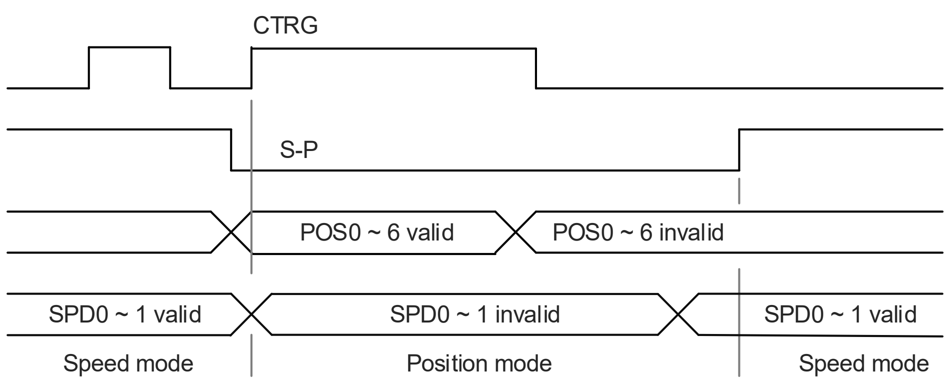


Figure 6-2 Speed / Position dual mode (PR-S)

Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

In Speed mode (DI.S-P is on), you select the Speed command with DI.SPD0 and DI.SPD1. DI.CTRG is not applicable. When switching to Position mode (DI.S-P is off), since the Position command has not been issued (it waits for the *rising edge* of DI.CTRG), the motor stops. The Position command is controlled by DI.POS0–DI.POS6 and triggered by the rising edge of DI.CTRG. When DI.S-P is on, it returns to Speed mode. Please refer to the introduction of single mode for the DI signal and the selected commands for each mode.

6.5.2 - Speed / Torque dual mode (S-T)

Speed / Torque dual mode includes only the S-T mode. You control the Speed command with the external analog voltage and the internal parameters (P1.009 – P1.011), which you select with DI.SPD0 – DI.SPD1. Similarly, the source of the Torque command can be the external analog voltage or the internal parameters (P1.012 – P1.014), and is selected by DI.TCM0 – DI.TCM1. The switch between Speed and Torque mode is controlled by DI.S-T (0x19) signal. The timing diagram is shown below.

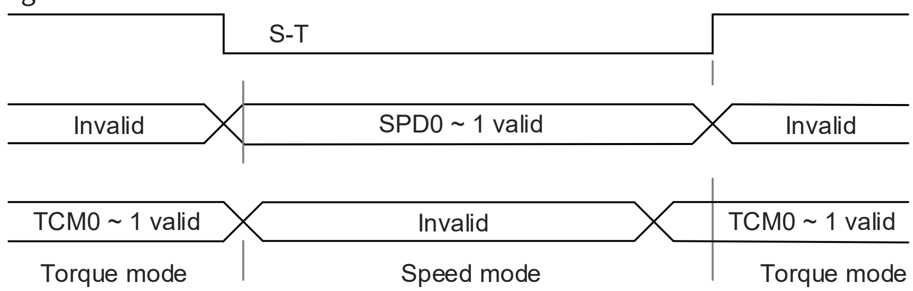


Figure 6-3 Speed / Torque dual mode

In Torque mode (DI.S-T is on), you select the Torque command with DI.TCM0 and DI.TCM1. When switching to Speed mode (DI.S-T is off), you select the Speed command with DI.SPD0 and DI.SPD1. The motor operates according to the Speed command. When DI.S-T is ON, it returns to the Torque mode. Please refer to the introduction of single mode for the DI signal and the selected commands for each mode.

6.5.3 - Torque / Position dual mode (PT-T, PR-T)

Torque / Position dual mode includes PT-T and PR-T. The command source for PT-T comes from the external pulse while the source for PR-T comes from internal parameters (P6.000 – P7.099). You control the Torque command with the external analog voltage or the internal parameters (P1.012 – P1.014). The switch between Torque and Position mode is controlled by DI.T-P (0x20) signal. You select PT and PR in Position mode with DI.PT-PR (0x2B). Thus, you select both Position and Torque commands in PR-T mode with the DI signal. The timing diagram is shown below.

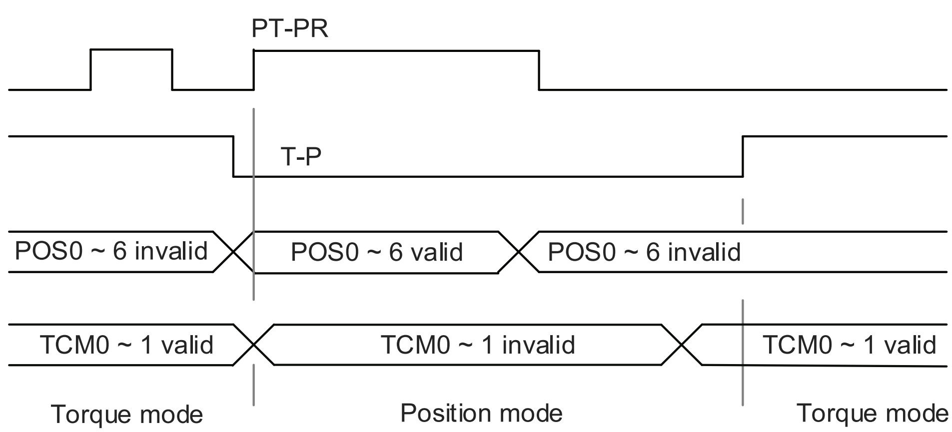


Figure 6-4 Torque / Position dual mode

In Torque mode (DI.T-P is on), you select the Torque command with DI.TCM0 and DI.TCM1. DI.CTRG is not applicable. When switching to Position mode (DI.T-P is off), since the Position command has not been issued (it waits for the rising edge of DI.CTRG), the motor stops.
The Position command is determined by DI.POS0 – DI.POS6 and triggered by rising edge of DI.CTRG. When DI.T-P is on, it returns to Torque mode. Please refer to the introduction of single mode for the DI signal and the selected commands for each mode.

6.6 - Others

6.6.1 - Applying the speed limit

The maximum speed in each mode (Position, Speed, Torque) is determined by the internal parameter P1.055 – Maximum Speed Limit (rpm). You use the same method for the Speed Limit and Speed commands. You can use either the external analog voltage or the internal parameters (P1.009 – P1.011). Please refer to Section 6.3.1 for descriptions.

If you are using the external analog voltage in Torque mode, the DI signals are available and you can set SPD0–SPD1 for the motor speed limit value (internal parameters). If not, you can use the analog voltage input for the Speed Limit command. When you set P1.002.X (disable / enable speed limit function) to 1, you enable the Speed Limit function.

Example Speed limit settings and behavior In Torque mode

When P1.001=3:

- *P1.041 is valid for limiting max torque when using analog torque control.*
- *P1.012 - P1.014 are valid torque selections and override the analog torque setting when TCM0 and TCM1 are not 0.*
- *P1.002.Y (enable torque limit) is not valid.*
- *DI.SPDLM [0x10] =1 will also enable speed limit. Speed limit is enabled in torque mode when P1.002.X=1 OR DI.SPDLM=1.*

When P1.002.X=1 , Enable speed limit:

- *P1.040 is valid for speed limitation when using analog torque control and SPD0 and SPD1 are 0.*
- *P1.009 - P1.011 are valid speed limits when properly selected with SPD0 and SPD1 and are not 0. Speed limit P1.040 is no longer valid.*

When P1.002.X =0, Disable speed limit:

- *P1.040 and P1.009 - P1.011 are not valid speed limits.*

6.6.2 - Applying the torque limit

The method for using the Torque Limit command and Torque command are the same. You can use either the external analog voltage or the internal parameters (P1.012 – P1.014). Please refer to Section 6.4.1 for descriptions.

You can use the torque limit in Position mode (PT, PR) or Speed mode (S) to limit the motor torque output. When you execute the command in Position mode using the external pulse or execute the command in Speed mode using the external analog voltage, DI signals are available and you can set TCM0 – TCM1 to determine the Torque Limit command (internal parameters). If there is not enough DI signal available, you can execute the Torque Limit command using the analog voltage. When you set the Torque Limit function (P1.002.Y) to 1, you enable the Torque Limit function. P1.041 determines the torque limit for the analog input and TCM0-TCM1 determine the limit when using DI selections.

Example Torque limit settings and behavior In Speed mode

When P1.001 =2:

- *P1.040 is valid for speed limitation when using analog speed control.*
- *P1.009 - P1.011 are valid speed selection when properly selected with SPD0 and SPD1 and are not 0. Speed limit P1.040 is no longer valid*
- *P1.002.X (enable speed limit) is not valid*
- *DI.TRQLM [0x09] =1 will also enable torque limit. Torque limit is enabled in speed mode when P1.002.Y=1 OR DI.TRQLM=1*

When P1.002.Y =1, Enable torque limit:

- *P1.041 is valid for limiting max torque when using analog torque control.*
- *P1.012 - P1.014 are valid torque setting and override the analog torque setting when TCM0 and TCM1 are not 0.*

When P1.002.Y=0, Disable torque limit:

- *P1.041 and P1.012 - P1.014 are not valid torque limits.*

6.6.3 - Analog monitoring

You can externally monitor drive conditions with analog outputs. Two ±8V analog channels are provided by the servo drive and located on terminals 15 and 16 of CN1. P0.003 can set the two analog outputs to monitor motor speed (command or actual), motor torque (command or actual), pulse command frequency, DC bus voltage, or the outputs can be set as a generic analog voltage output (equal to P1.101 and P1.102). The analog output resolution is 10-bit. Please refer to Chapter 8 for more information about the relevant parameters.

Parameter	***Function***
P0.003	Analog output monitoring
P1.003	Analog and Encoder pulse output polarity
P1.004	MON1 analog monitor output proportion
P1.005	MON2 analog monitor output proportion
P1.101	Analog monitor output voltage 1
P1.102	Analog monitor output voltage 2

Example:

Specify a motor speed of 1000 rpm, which corresponds to analog voltage output of 8V with the maximum speed of 5000 rpm. The setting is as follows:

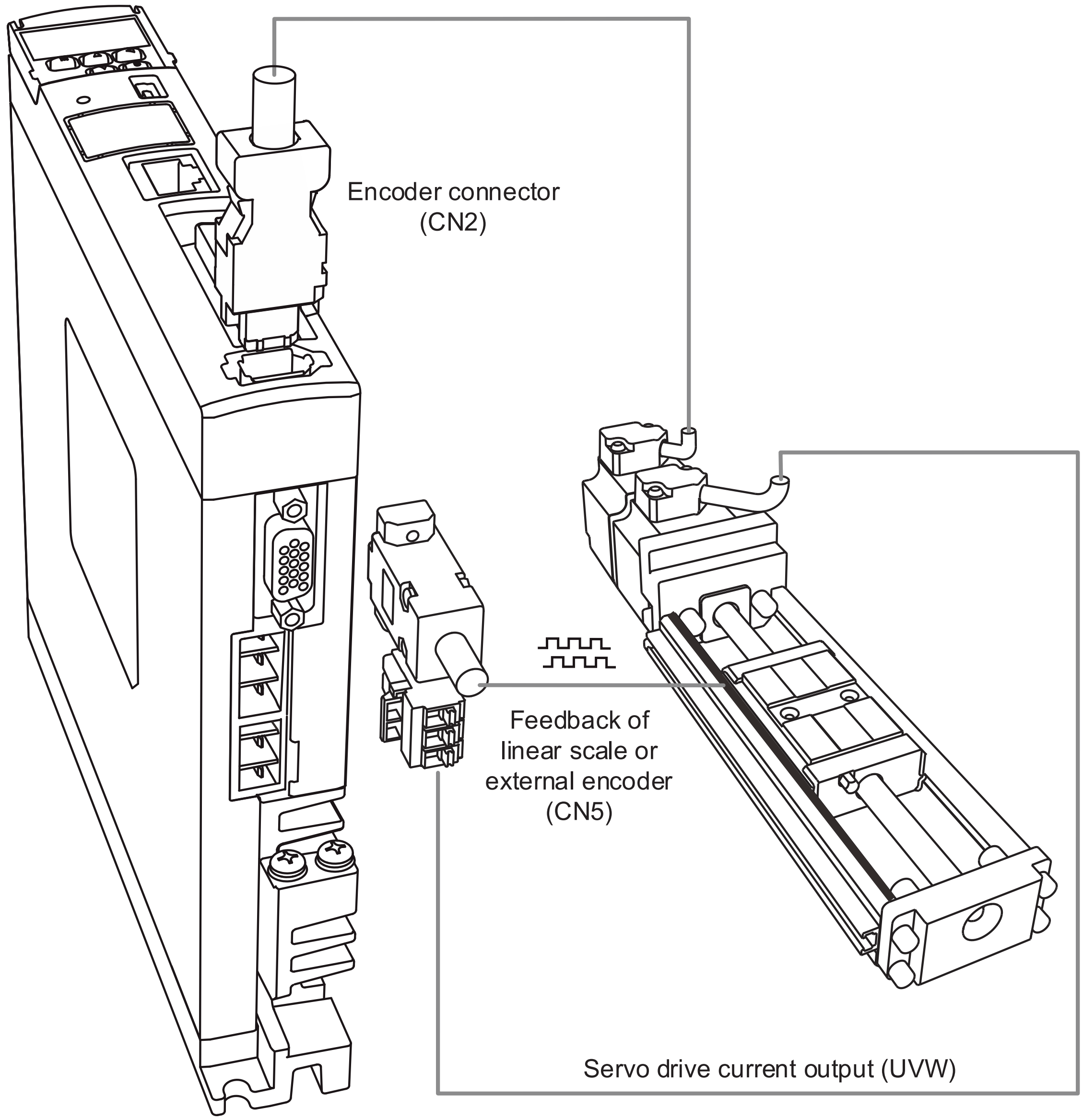
$$P1.004 = \frac{\text{Required speed}}{\text{Maximum speed}} \times 100\% = \frac{1000\text{ rpm}}{5000\text{ rpm}} \times 100\% = 20\%$$

You can calculate the corresponding voltage output for the current motor speed with the formula below.

Motor speed	***Mon1 Analog monitoring output***
300 rpm	$MON1 = 8V \times \frac{\text{Current speed}}{(\text{Maximum speed} \times \frac{P1.004}{100})} \times 100\% = 8V \times \frac{300\text{ rpm}}{5000\text{ rpm} \times \frac{20}{100}} \times 100\% = 2.4V$
900 rpm	$MON1 = 8V \times \frac{\text{Current speed}}{(\text{Maximum speed} \times \frac{P1.004}{100})} \times 100\% = 8V \times \frac{900\text{ rpm}}{5000\text{ rpm} \times \frac{20}{100}} \times 100\% = 7.2V$

6.7 - Full-Closed Loop Control System

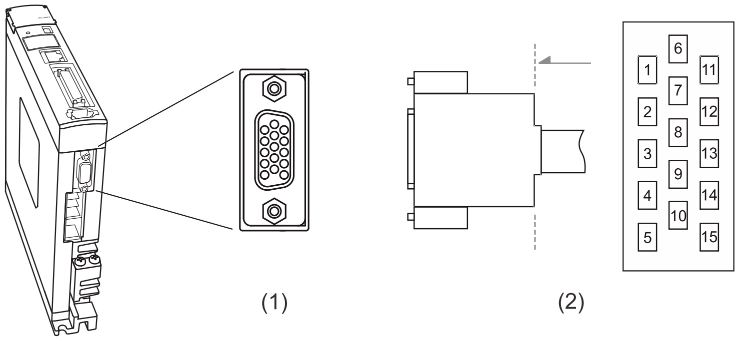
The auxiliary encoder (CN5) returns the actual position of the machine end to the servo drive in the full-closed loop system, which improves the conditions of lead screw backlash, flexibility or stretching of couplings or belts, and thermal expansion, linearity, and sliding end of the transmission system, achieving high-precision positioning.



Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

6.7.1 - Hardware configuration

The CN5 connector is for connecting to the auxiliary encoder (A, B, and Z) and forms a full-closed loop with the servo system. If the Z pulse input for CN5 is not wired, then the full-closed loop function will not work. If there is not a Z pulse available, then wire Z to +5VDC and /Z to 0VDC.



(1) CN5 connector (female); (2) CN5 connector (male)

NOTE: This only supports AB phase signal and an encoder of 5V. The maximum single-phase (Phase A or Phase B) pulse frequency for the encoder cannot exceed 1MHz.

NOTE: Use ZL-HD15M-CBL-DB15F cable + ZL-RTB-DB15 ZIPLink breakout board, or use ZL-HD15M-CBL-2P cable (HD15 to flying leads).

WARNING: Do not use a standard VGA HD15 cable. The typical VGA cable does not include a connection on pin 8.

Pin Assignment:

Pin Number	Color	Signal	Function
1	Black/White	Opt_/Z	/Z pulse input
2	Blue/White	Opt_/B	/B phase input
3	Blue	Opt_B	B phase input
4	Green	Opt_A	A phase input
5	Green/White	Opt_/A	/A phase input
6	Yellow Yellow/Black	GND	Encoder grounding
7	Red/White	GND	Encoder grounding
8	Red	+5V	Encoder power
9	Black	Opt_Z	Z pulse input
10	Orange	Reserved	Reserved
11	Orange/White	Reserved	Reserved
12	Brown	Reserved	Reserved
13	Brown/White	Reserved	Reserved
14	Purple	Reserved	Reserved
15	Purple/White	Reserved	Reserved

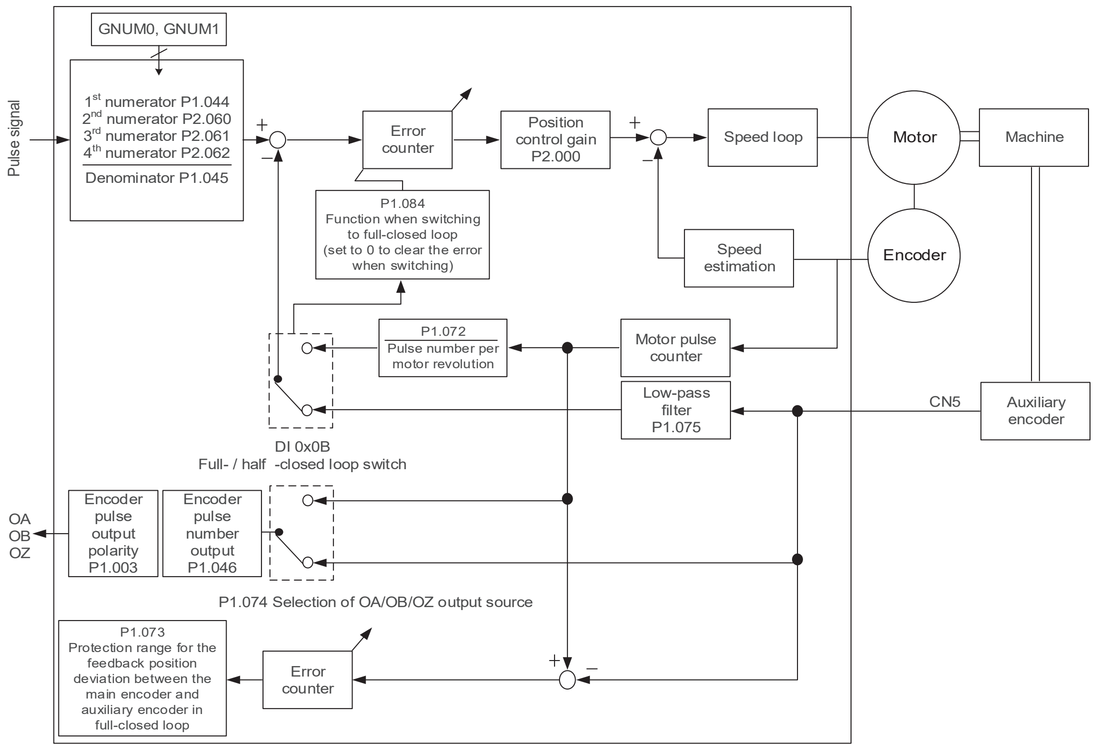
Specifications and Wiring Descriptions for the CN5 Signals:

Signal Type	A,B,Z Phase Signal
Operating voltage	5V
Signal format	Differential
Encoder power (5V) output	≤ 300mA
Max. pulse frequency	Single-phase pulse frequency: 4MHz

6.7.2 - Control structure

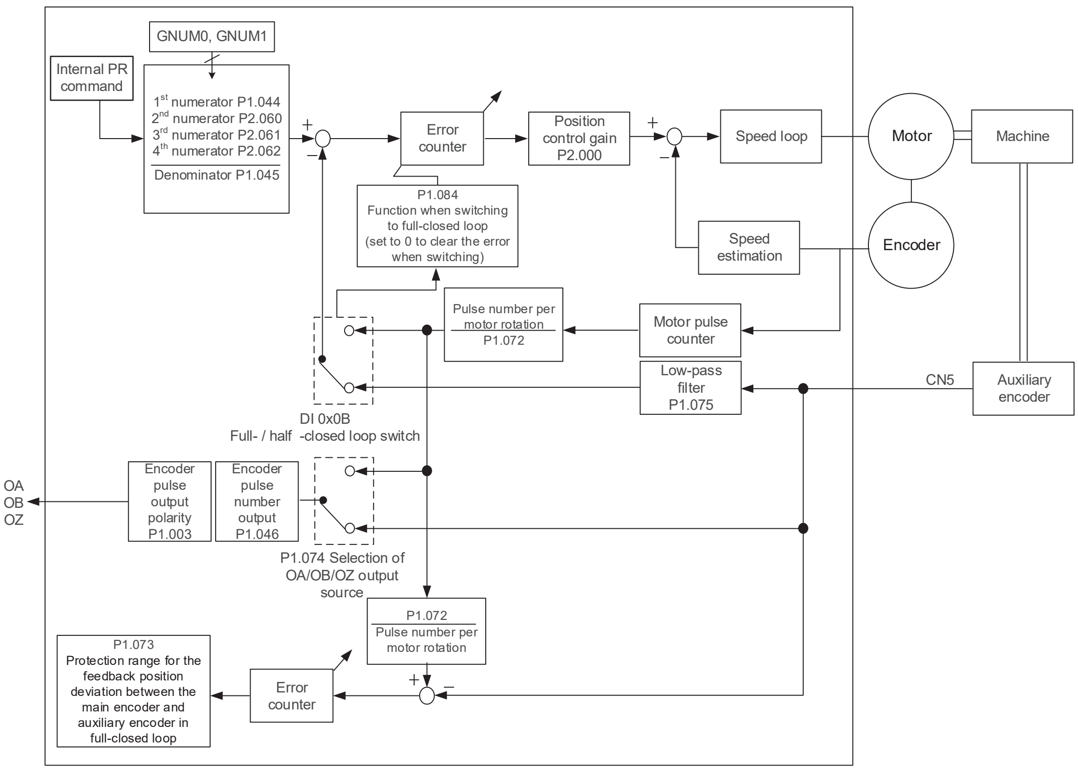
Full closed Loop Control Structure in PT Mode

When the servo is in full-closed loop control in PT mode, if the E-Gear ratio is set to $\frac{1}{1}$, one pulse from the command corresponds to one quadruple-frequency pulse from the auxiliary encoder. If the E-Gear ratio is set to $\frac{2}{1}$, one pulse from the command corresponds to two quadruple-frequency pulses from the auxiliary encoder.

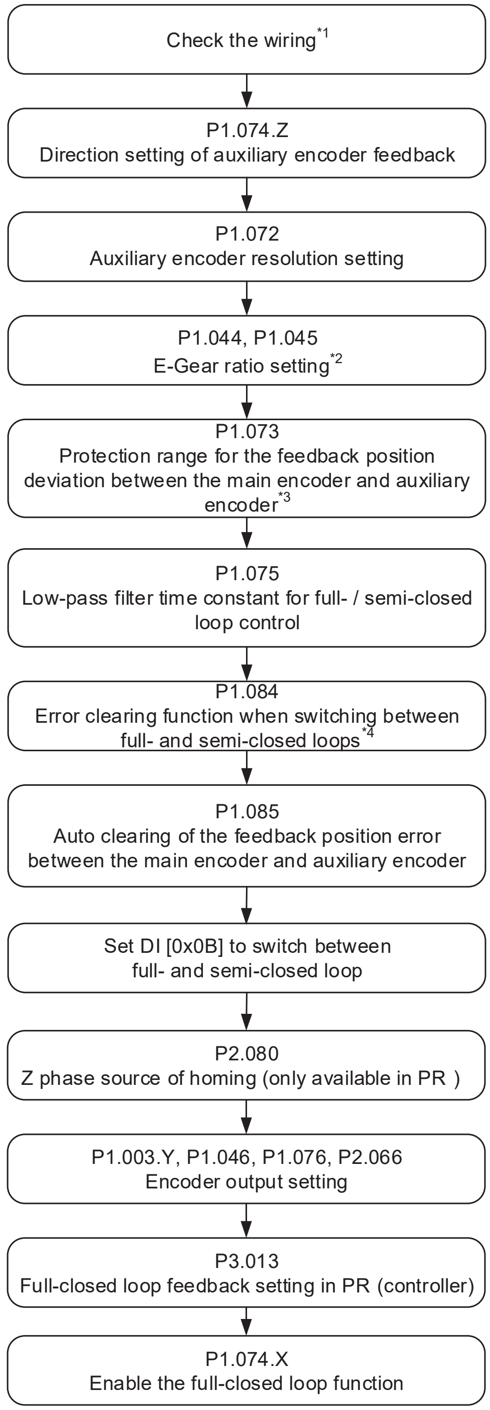


Full-closed Loop Control Structure in PR Mode

The servo is in full-closed loop control in PR mode (P1.001=1), full-closed loop function is enabled (P1.074.X=1), and full-closed loop to half-closed loop selection (DI [0x0B]) is set to OFF. When the servo is in full-closed loop control in PR mode, the E-Gear ratio should be set to $\frac{1}{1}$ for ease of use. With this setting, one PUU position command from the PR path corresponds to one quadruple-frequency pulse from the auxiliary encoder. If the E-Gear ratio is set to $\frac{2}{1}$, one PUU position command from the PR path corresponds to two quadruple-frequency pulses from the auxiliary encoder.



6.7.3 - Steps for setting the full-closed loop function



Notes:

1 - The auxiliary encoder (A, B, Z) is connected to the CN5 on the servo drive to form a full-closed loop. You can monitor whether the drive receives the feedback position from the auxiliary encoder with P5.017 or LED display.

2 - Set the E-Gear ratio to 1:1 to make PUU calculations easy.

3 - When setting the full-closed loop function for the first time, setting P1.073 too high can cause the auxiliary encoder to disconnect or trigger inverse direction causing motor continuous operation.

4 - This parameter is not available in PR mode. In PR mode, the error is automatically cleared when the system switches between full- and half-closed loops.

6.7.4 - Auxiliary encoder direction setting

P1.074	***Full-closed loop control for secondary or auxiliary encoder***		***Hex Address***	***Dec Address***
			0194H 0195H	40405 40406
Default:	0x0000	Control mode:	PT / PR (full-closed loop)	
Unit:	-	Setting range:	0000h–F132h	
Format:	HEX	Data size:	16-bit	

Settings:

0002

U Z Y X

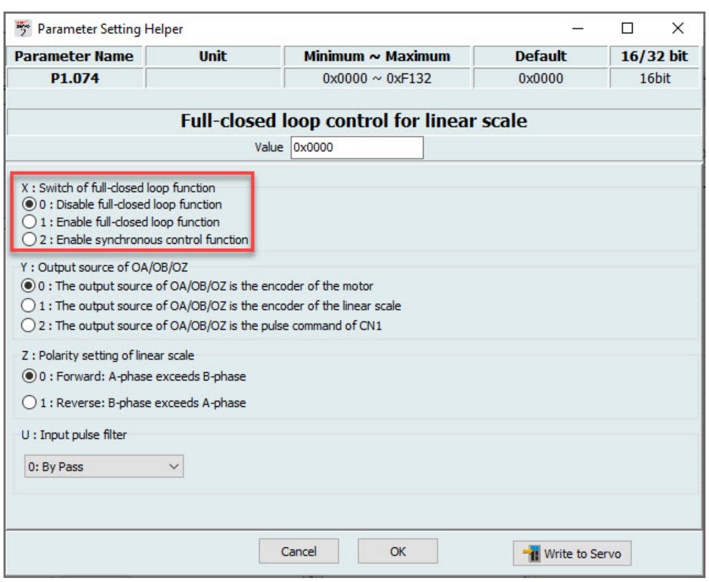
X	Full-closed loop control switch	Z	Positive / negative direction selection of auxiliary encoder feedback
Y	Source for OA / OB / OZ output	U	Auxiliary encoder filter function

- *X: full-closed loop*
  - *0: disable full-closed loop function*
  - *1: enable full-closed loop function*
- *Z: positive / negative direction selection of auxiliary encoder feedback*
  - *0: positive direction when A phase leads B phase of auxiliary encoder*
  - *1: positive direction when B phase leads A phase of auxiliary encoder*

Before using the full-closed loop control function, check if the feedback pulse of the auxiliary encoder increases or decreases in the same direction as the motor encoder. If the directions for the two feedback pulses are inverse, change the setting value of P1.074.Z to reverse the direction for the signal of the auxiliary encoder.

Here are the steps for checking the directions.

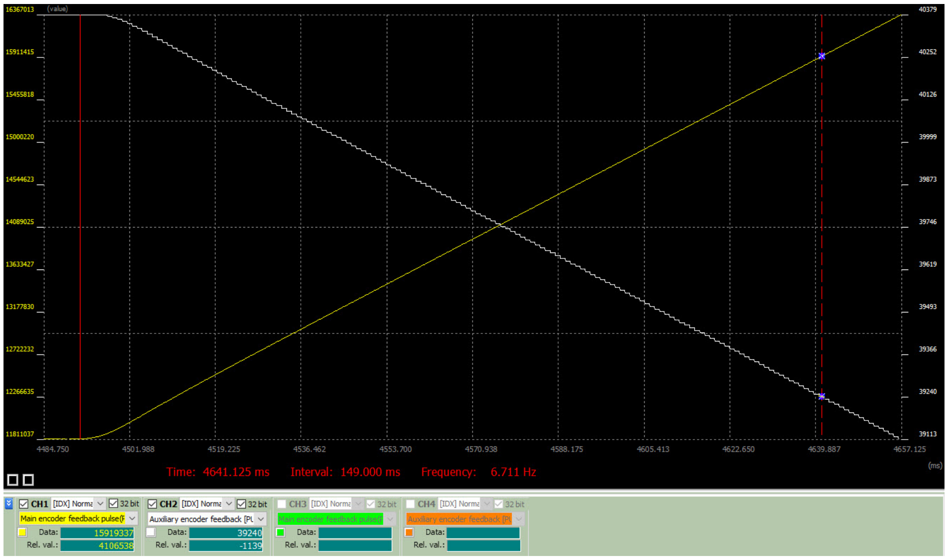
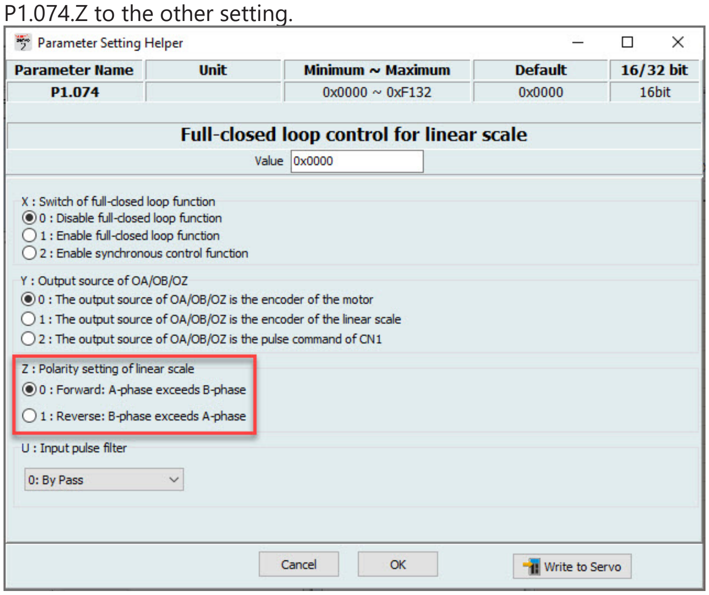
Step	***Action***
1	Disable the full-closed loop function by setting P1.074.X to 0.

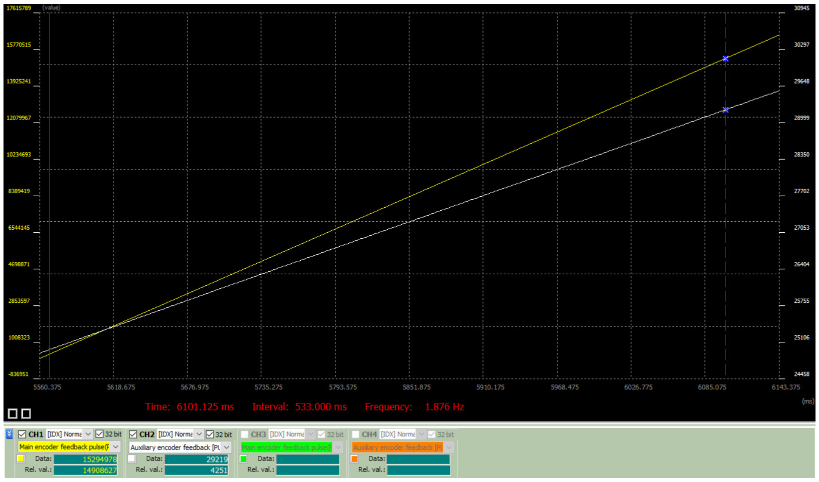


<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>Open the software scope, select Main encoder feedback pulse [Pulse] for CH1 and Auxiliary encoder feedback [Pulse] for CH2, and then click Start to start the scope.

Use the JOG function to operate the motor in a single direction at a low speed. If you get two pulse signals which are in inverse directions (shown as follows), go on to Step 3 to adjust the parameter.

If the motor direction output control (P1.001.Z) is set to 1 then the trace will always have a negative slope when running in the forward direction when a scope channel is set to “Main encoder feedback pulse [pulse]”. Use “Feedback position [PUU]” if you have P1.001.Z=1.


</td></tr>
<tr><td>3</td><td>If the two pulse signals are in the inverse direction due to the previous setting of P1.074.Z. Switch P1.074.Z to the other setting.

</td></tr>
</table>

Step	Action
4	Go back to Step 2 to ensure the feedback pulses of the motor encoder and auxiliary encoder increase or decrease in the same direction. 

6.7.5 - Auxiliary encoder resolution setting

P1.072	Resolution of auxiliary encoder for full-closed loop control		Hex Address	Dec Address
			0190H 0191H	40401 40402
Default:	5000	Control mode:	PT / PR (full-closed loop)	
Unit:	pulse / rev	Setting range:	200–1280000	
Format:	DEC	Data size:	32-bit	

Settings:

A/B pulse count from the auxiliary encoder that equates to one revolution of the motor shaft. The 4x value needs to be entered here.

There are two methods for calculating the corresponding pulse number of the auxiliary encoder per motor revolution. One method calculates the theoretical value from hand calculations. The other calculates the actual value with the software scope of SureServo2 Pro. If the resolution of auxiliary encoder for full-closed loop control (P1.072) is incorrectly set, the position error between the auxiliary encoder feedback and the motor encoder accumulates during long-term operation, triggering AL040.

The encoder must be a line driver output AB Quadrature encoder with a Z pulse. If there is no Z pulse and one is not needed then the Z and Z/ signals need to be tied to +5V and 0V respectively to avoid an alarm.

Method 1: Calculating the Theoretical Value

To calculate the theoretical value for a machine using a screw transmission with an external encoder for full-closed loop control, the pitch and all gear ratios must be known as well as the resolution of the auxiliary encoder. The calculation will determine the corresponding number of pulses from the auxiliary encoder that equal one motor revolution. When the specifications of the screw and auxiliary encoder are known, you can calculate the value of P1.072.

Example 1:

If the screw pitch is 5mm (one revolution will translate to 5mm of linear travel) and the resolution of the auxiliary encoder is 0.5 µm between pulses, the calculation is as follows.

$$\frac{5\ mm}{0.5\ \mu m} = \frac{5000\ \mu m}{0.5\ \mu m} = 10000\ pulse = \ \ \text{P1.072}$$

When the motor turns one revolution, the auxiliary encoder should produce 10,000 pulses.

Example 2:

Using a roll-on encoder that directly tracks a conveyor belt. The encoder is sold as a 1024ppr encoder. The drive will use the 4x multiplier of the AB quadrature signals so the true pulse count is 4096 ppr of the encoder for positioning calculations. For this example, the encoder wheel circumference is 317.6 mm.

$$\text{So } \frac{317.6\,mm}{4096\,pulses} = 0.0775\,mm/pulse$$

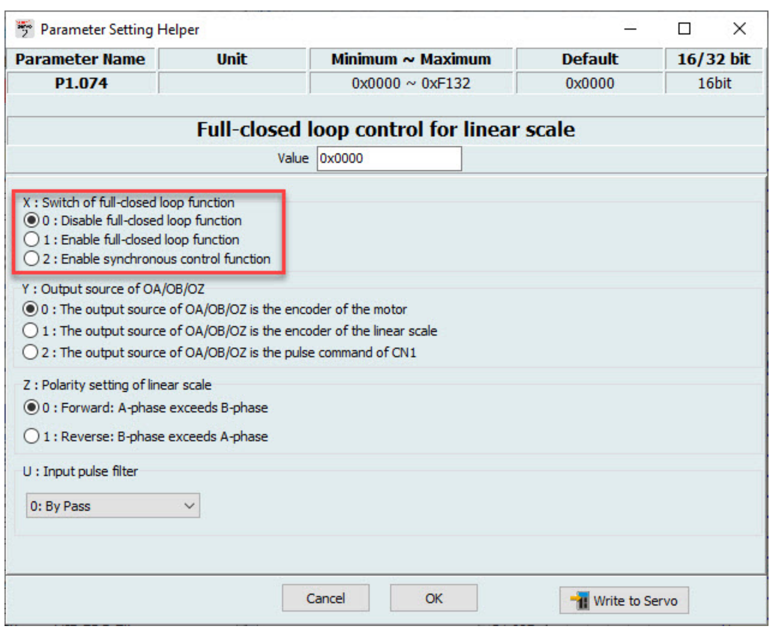
This means for every 0.0775 mm the conveyor travels, the aux encoder will output one pulse. If the distance the conveyor moves for one motor revolution is 375.23 mm then we can now calculate what to put into P1.072.

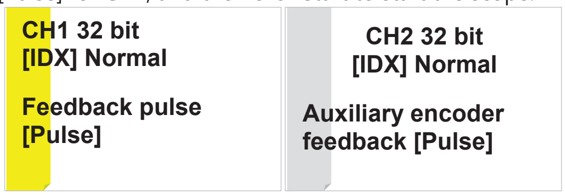
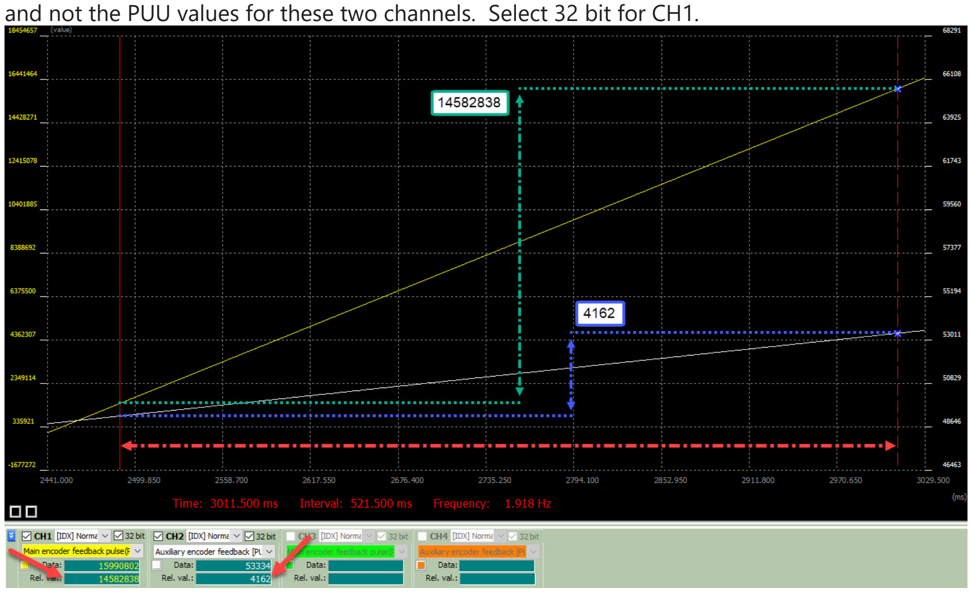
$$\frac{375.23\,mm/\textbf{motor revolution}}{0.0775\,mm/pulse} = 4841.70\ pulses$$

of the aux encoder per 1 motor rev revolution. Only integer values are allowed so P1.072 = 4842.

Method 2: Measuring the Actual Value with Motor Pulse Number (Short Distance)

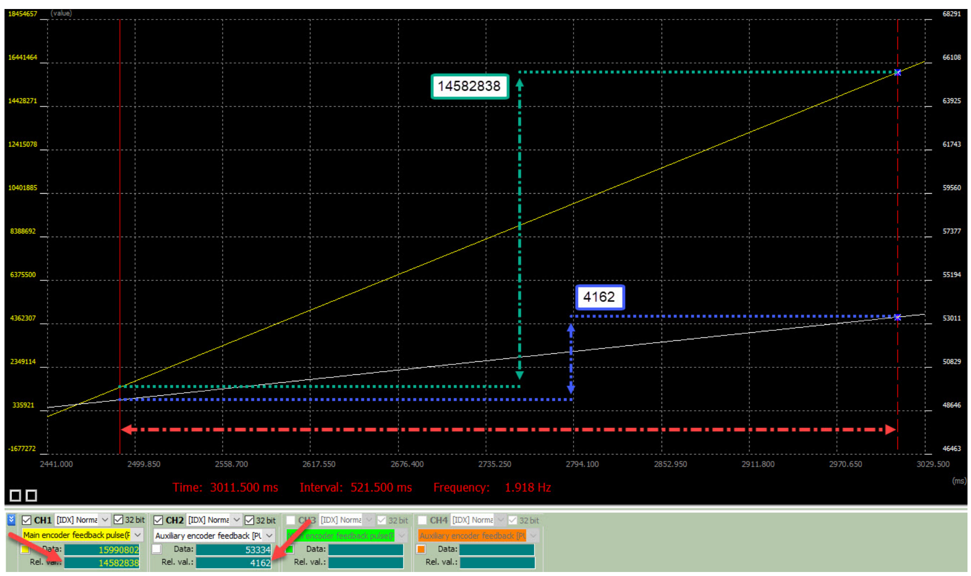
Calculating theoretical values is infeasible if the system does not use screws for transmission or the system consists of complex mechanical parts. In this case, use the JOG function to operate the motor in a single direction at low speed in the non full-closed loop mode, and calculate the value of P1.072 by using the software scope to monitor the feedback pulse number of the motor encoder and auxiliary encoder.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Disable the full-closed loop function by setting P1.074.X to 0.

</td></tr>
</table>

Step	Action
2	Open the software scope, select Feedback pulse [Pulse] for CH1 and Auxiliary encoder feedback [Pulse] for CH2, and then click Start to start the scope. 
3	Use the JOG function to operate the motor in a single direction at low speed, and observe the feedback pulse number from the two channels as shown in the following figure. Ensure that no slip is occurring on the auxiliary encoder during motion. Be sure you are reading the Pulse values and not the PUU values for these two channels. Select 32 bit for CH1. 
4	Use the two cursors to find the relative values (Rel. val.) of the two channels. Here you can see the motor moved 14582838 pulses. The auxiliary encoder traveled 4162 pulses in the same distance. According to the following formula, when the motor runs one revolution, the auxiliary encoder outputs 4788 pulses. This is not far off from the hand calculations of 4852 performed in the previous step. $\left\vert\frac{\text{Auxiliary encoder (linear scale) pulse number} \times 16777216}{\text{Motor encoder pulse number}}\right\vert = \frac{4162 \times 16777216}{14582838} \approx 4788$

Method 3: Measuring the Actual Value with Motor PUU Number (Long Distance)

If more accuracy is needed than provided by Method 2, then the auxiliary encoder should be allowed to travel a long distance before performing the calculations. Instead of using the JOG function from Method 2, you can use a PR path INC move that will move the motor a set number of exact rotations.

Step	***Action***
1	Enable full-closed loop function by setting P1.074.X to 1. You must also disable the full-closed/half-closed loop switching by assigning a DI to 0x0B and setting to ON so the drive is still in half-closed loop. Set P1.072 to an easy number such at 5000 and P1.044=1 and P1.045=1. This will cause the motor to have 5000 PUU per revolution for PR commands. If you have a more accurate number from Method 2, you can use that number instead. Configure a PR path to perform an incremental move of x10 or x100 the value entered in P1.074. In this example the PR path will move 100 exact revolutions of the motor by issuing a position command of 500000 (x100).
2	Open the software scope, select the following: CH1: Command Position [PUU] (32 bit) CH2: Auxiliary encoder feedback [Pulse] (32 bit) Then click Start to start the scope.
3	Run the PR path to have the motor run exactly 10 or 100 revolutions as was determined in step 1. Observe the Auxiliary encoder feedback [Pulse] number (Ref. value) with the first and second cursor placed at the zero speed zone at the beginning and end of the position move. 
4	Use the two cursors to find the relative values (Rel. val.) of the two channels. Here you can see the motor moved 500000 PUU which is 100 motor revolutions. The auxiliary encoder traveled 478073 pulses from start to stop. Since we know the move was exactly 100 motor revolutions then we just need to divide the aux encoder count by 100. The new more accurate value of P1.072 is 4781.

Method 1 resulted in 4852 pulses. This method is useful for calculating the value when all known ratios of the power transmission are known.

Method 2 resulted in 4788 pulses. This method is useful when the mechanics are unknown or hard to calculate and only a short position move is possible for the machine to perform.

Method 3 resulted in 4781 pulses. This method is useful when the mechanics are unknown or hard to calculate and a long position move is possible. Method 3 is the most accurate.

6.7.6 - Troubleshooting the full-closed loop function

Below are some helpful monitoring variables for troubleshooting the full-closed loop function. These can be monitored like any other monitoring variable using the LED display (P0.002), status monitoring registers (P0.009-P0.013), the status monitoring window, or using any of the channels in the Scope window. These codes can be found in section 8.4.11.

Variable	Code	Description
Feedback Position (PUU)	000	• Current feedback position of the motor encoder. Unit: (PUU). • When full-closed loop function is on (P1.074.X=1) the Feedback Position is referencing the PUU value set up in P1.044, P1.045, and P1.072. It does not matter if DI [0x0B] is set to full-closed or half-closed loop. • When in P1.074.X=1 and P3.013=1, this monitoring variable will show the feedback pulses of the auxiliary encoder, code 029. • When full-closed loop function is off (P1.074.X=0) the Feedback Position is referencing the PUU value set up in P1.044 and P1.045. Feedback Position will not reference P1.072.
Command Position (PUU)	001	• Current coordinate of the Position command. Unit: (PUU). • PT mode: number of pulse commands received by the drive. • PR mode: absolute coordinates of the Position command.
Auxiliary Encoder Feedback (PUU)	029	• Pulse counts directly from the auxiliary encoder (CN5). Unit: (PUU) • Same as code 048 but will get reset during homing routine.
Auxiliary Encoder CNT	048	• Pulse counts directly from the auxiliary encoder (CN5). Unit: (PUU) • Same as code 029 but will NOT get reset during homing routine or SV_OFF. Requires a power cycle to reset this value.
Main/Auxiliary Encoder Poistion Error (PUU)	031	• Feedback position deviation between the Command Position (Code 001) and auxiliary encoder CNT (Code 048). • This will show the accumulated error of the Auxiliary Encoder Feedback (Code 029) and/or Feedback Position (Code 000) compared to the Command Position (Code 001).
Auxiliary Encoder Position Error (PUU)	030	• Feedback position deviation between the Command Position (Code 001) and auxiliary encoder CNT (Code 048). • This will show the instantaneous error of the Auxiliary Encoder Feedback (Code 029) and/or Feedback Position (Code 000) compared to the Command Position (Code 001).
Main/Aux Encoder Deviation	115	• This variable is used for P1.073 and P1.085 fault protection. Like Code 031 but gets reset after the P1.085 number of motor revolutions has elapsed. • When using full closed loop, this variable will display the deviation between main encoder and auxiliary encoder. It will reset to zero once the number of motor revolutions entered in P1.085 have elapsed. If this value reaches the value entered in P1.073 before the P1.085 number of motor revolutions has been reached, then AL040 will occur. • A homing routine will not reset this value to 0. Setting SV_OFF will reset this value to 0.

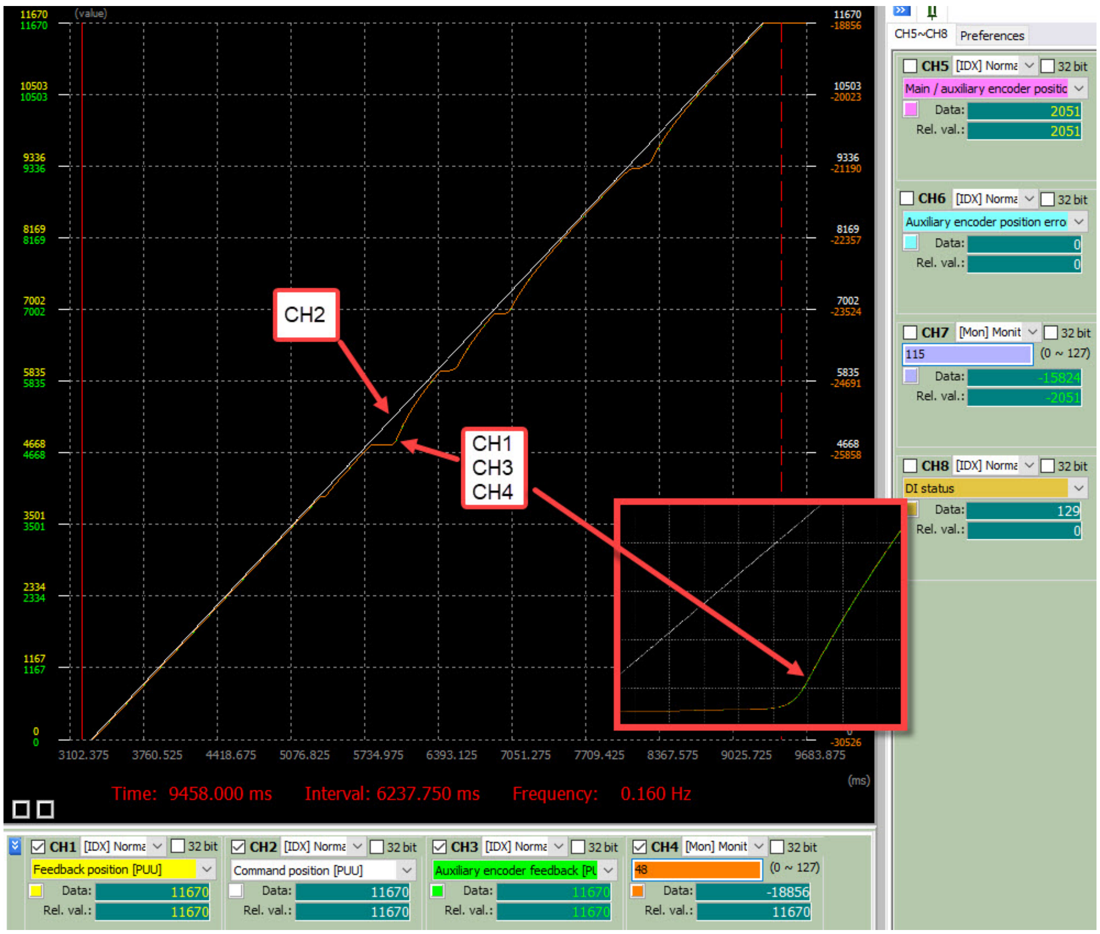
Example Trace:

Demonstration of traces below during a position move in full-closed loop mode.

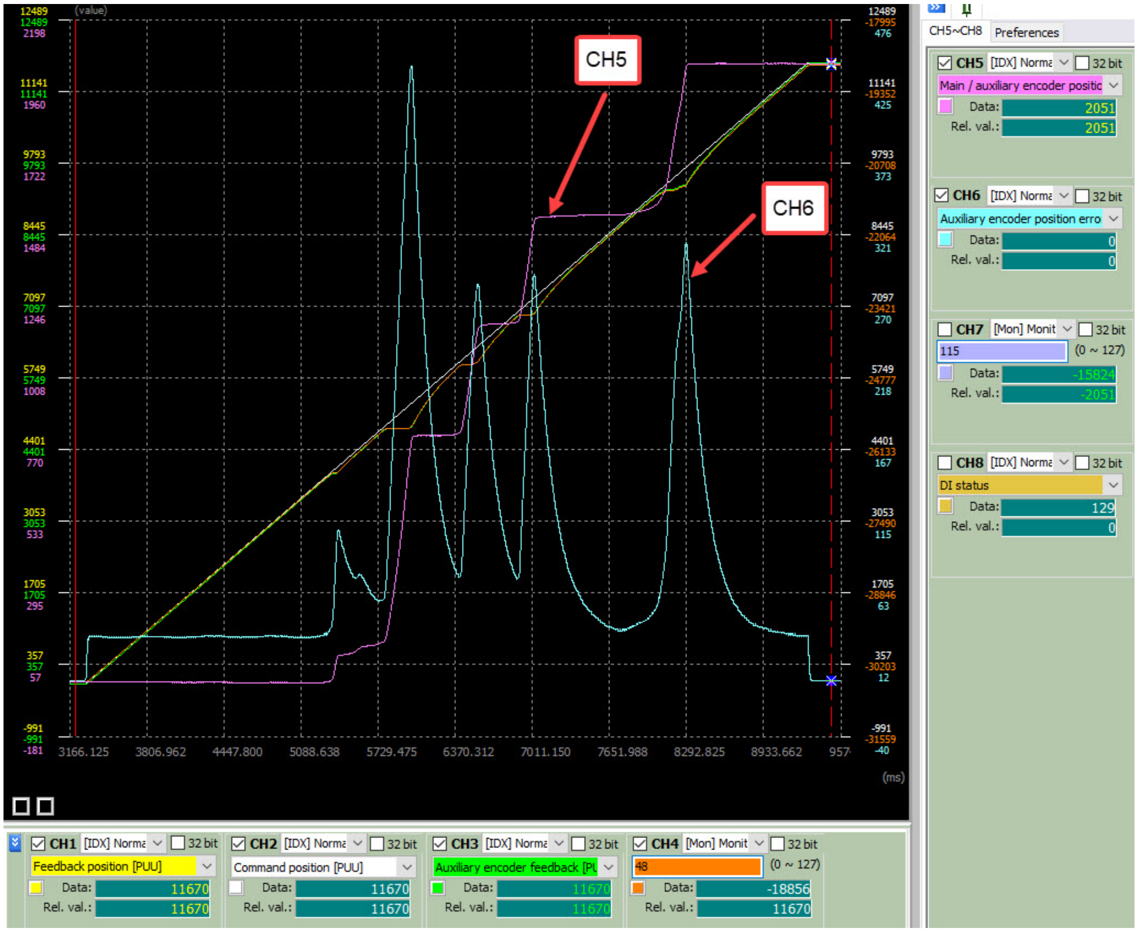
Parameter settings:

- *P1.072.X = 1*
- *P3.013.X = 1*
- *DI [0x0B] = Off*

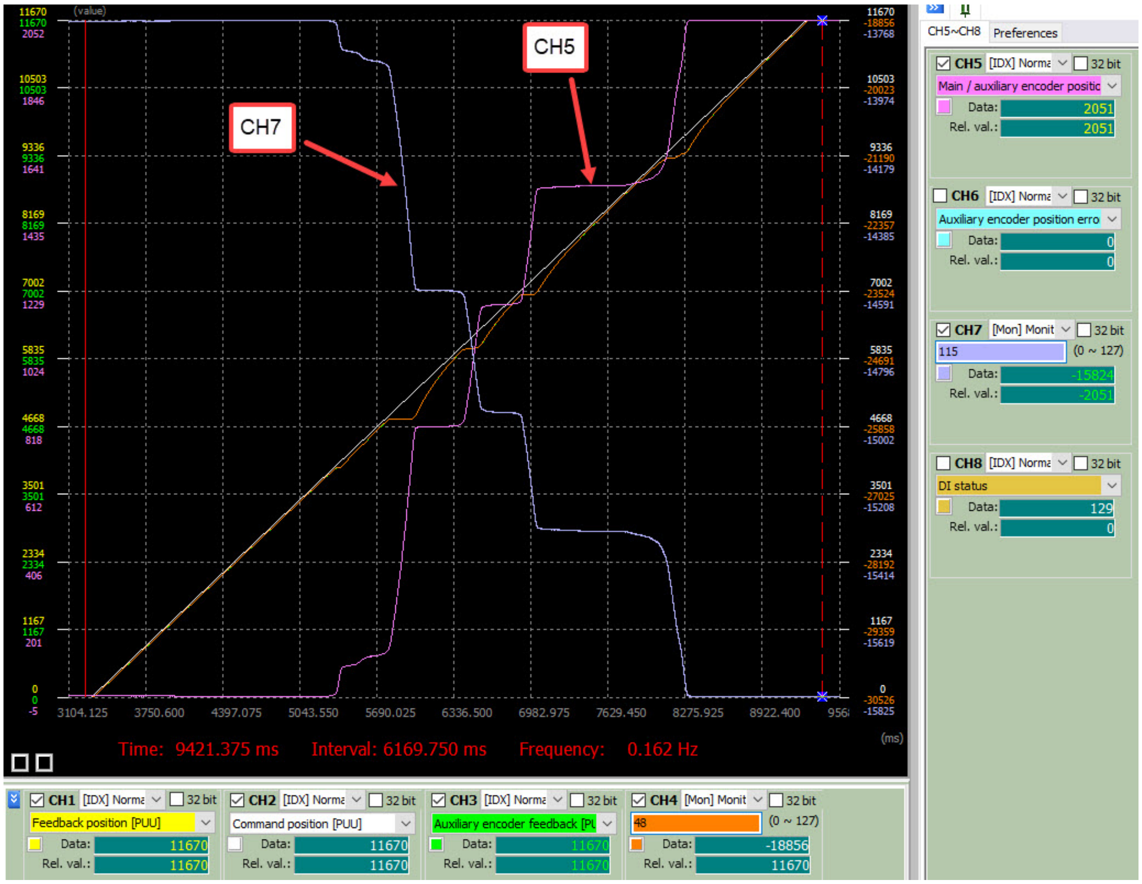
In the below full-closed loop Scope capture you can see the Commanded position is linear and does not change (CH2). The encoder slips at several points along the move. Channels 1, 2, and 4 all show this and the traces are nearly identical.



In the same full-closed loop Scope capture below, with channel 5 and channel 6 turned on you can see how the Main/Auxiliary Encoder Position Error (CH 5) shows the accumulated error and the Auxiliary Encoder Position Error (CH 6) is showing the deviation at one moment in time.



Again, in the same full-closed loop Scope capture below, with channel 5 and channel 7 turned on you can see how the Main/Auxiliary Encoder Position Error (CH 5) shows the accumulated error and the Main/Aux Encoder Deviation (CH 7) shows a similar trace in the opposite direction. Channel 7 shows the value that is monitored by P1.073. Once this absolute value reaches the value set in P1.073 then AL040 will occur. Positive or negative movement of this value just depends on the direction of travel during the deviation. The absolute value is all that P1.073 monitors. CH 5 gets reset during homing but CH 7 does not–a power cycle is required to reset CH 7.



6.7.7 - E-Gear settings

When the servo is in full-closed loop control, set both P1.044 and P1.045 to 1, this will make the value of P1.072 equal the motor PUU for one revolution when P1.074.X= 1.

6.7.8 - Protection range for feedback position error

P1.073	**Error protection range for full-closed loop control**	**Hex Address**	**Dec Address**
		0192H 0193H	40403 40404
Default:	30000	Control mode:	PT / PR (full-closed loop)
Unit:	Pulse or PUU (based on the feedback of full-closed loop)	Setting range:	1 to (2^{31}-1)
Format:	DEC	Data size:	32-bit

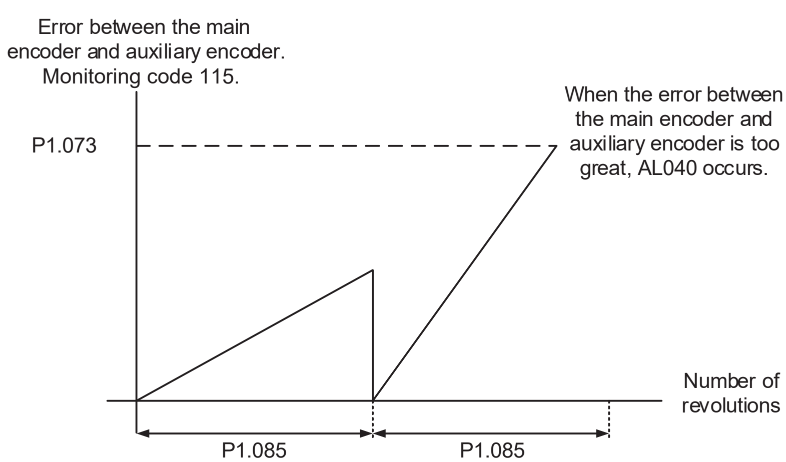
Settings:

Stopping the motor may be necessary when the deviation between the auxiliary encoder and the motor encoder feedback position is excessive due to a loose connector, an encoder failure, or other mechanical problems. This deviation can be monitored using monitoring variable code 115 in P0.001 or in the SureServo2 Pro Scope.

This parameter works in conjunction with P1.085. After the number of motor revolutions has occurred as defined in P1.085 the accumulated error measured in monitoring code [115] gets reset to 0.

When the deviation is greater than the value of P1.073, AL040 (excessive deviation of full closed-loop position control) occurs. To completely avoid this alarm, set P1.073 high and P1.085=1.

$$\mathrm{P1.073} < (\mathrm{Main\ \ encoder\ \ feedback} \times \frac{P1.072}{16777216}) - \mathrm{Auxiliary\ \ encoder\ \ feedback}$$



6.7.9 - Setting low-pass filter time constant

P1.075	**Low-pass filter time constant for full- and half-closed loop control**	**Hex Address**	**Dec Address**
		0196H 0197H	40407 40408
Default:	100	Control mode:	PT / PR (full-closed loop)
Unit:	ms	Setting range:	0–1000
Format:	DEC	Data size:	16-bit

Settings:

When the stiffness of the mechanical system between full-closed and half-closed loops is insufficient, set the proper time constant to enhance the stability of the system. In other words, this filter temporarily blends full-closed loop and half-closed loop feedback to establish a stable start and stop position, and after stabilizing, the full-closed loop effect is in 100% control. When the stiffness is sufficient, set to disable.

A half-closed loop is referring to the encoder on the back of the servo motor being the feedback device for the drive to close the velocity and position loop. For a fully-closed loop system, this is referring to the motor's encoder being used to close the velocity loop and an external encoder connected to CN5 to close the position loop.

Set the value to 0 to disable the low-pass filter (bypass) function.

If the stiffness of the mechanical system is high, decrease the value of P1.075, or set the value to 0 to disable. If the stiffness of the mechanical system is low, increase the value of P1.075.

When an extremely flexible mechanism is using full-closed loop control and when the motor starts turning, the external encoder might get some unstable feedback due to the flexible structure. So increasing the low pass filter (P1.075) can decrease the unstable level of the feedback.

This parameter will mitigate any fluctuations seen in the external encoder when in full-closed loop control when starting and stopping. The servo will partially act as a half-closed loop system and use the servo motor's encoder to reduce instability of the load, and after the motion of the load is stabilized, the full-closed loop function is turned back on. This filter blends the two encoder feedback signals during feedback instability.

6.7.10 - Setting error clearing function

P1.084	**Error clearing function when switching between full- and half-closed loops**		**Hex Address**	**Dec Address**
			01A8H 01A9H	40425 40426
Default:	0x0000	Control mode:	PT (full-closed loop)	
Unit:	-	Setting range:	0x0000 – 0x0001	
Format:	HEX	Data size:	16-bit	

Settings:

This parameter is not available in PR mode. In PR mode, the error is automatically cleared when the systems switches between full- and half-closed loops.

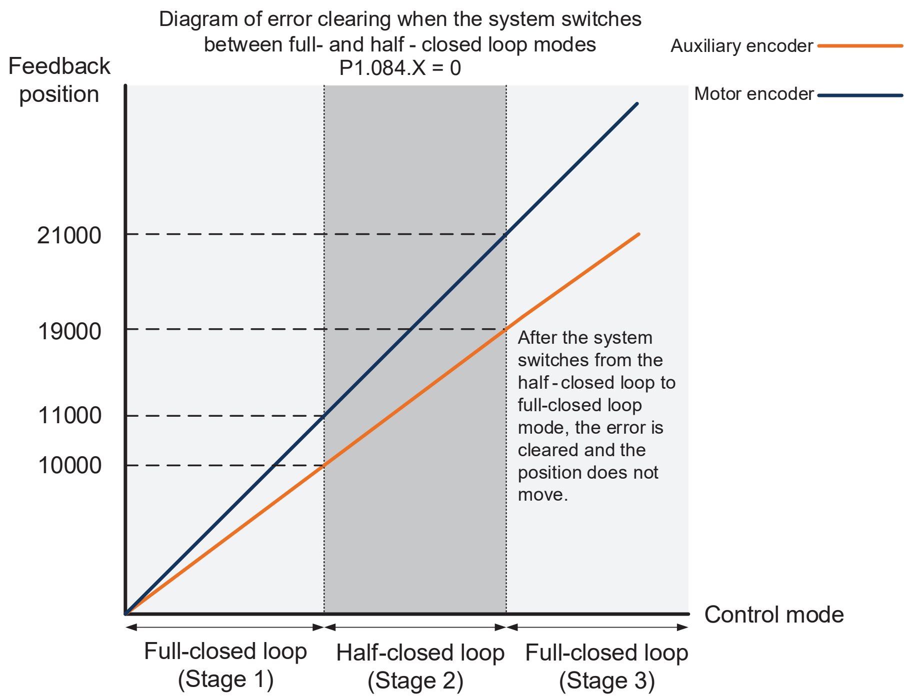
X	Error clearing function when the system switches from half-closed loop to full-closed loop	Z	Reserved
Y	Reserved	U	Reserved

- *X: Error clearing function when the system switches from half-closed loop to full-closed loop*
  - *0: clear the error when switching.*
    When the system is in half-closed loop, the command refers to the motor encoder and the position does not move after the system switches to full-closed loop.
  - *1: no clearing of the error when switching.*
    When the system is in half-closed loop control, the command refers to the motor encoder. After the system switches to full-closed loop, the command issued in half-closed loop becomes the full-closed loop command, and thus the position moves.

Note: Use DI [0x0B] to switch between full- and half-closed loop modes (P1.074.X must equal 1).

Examples:

Error Cleaning Enabled (P1.084.X=0)



Stage 1: full-closed loop control (feedback position of the auxiliary encoder)

If the servo drive issued a position command of 10,000 PUU and the feedback position of the auxiliary encoder is 10,000 PUU, the final feedback position of the motor encoder is 11,000 PUU due to the backlash and sliding of the mechanical parts.

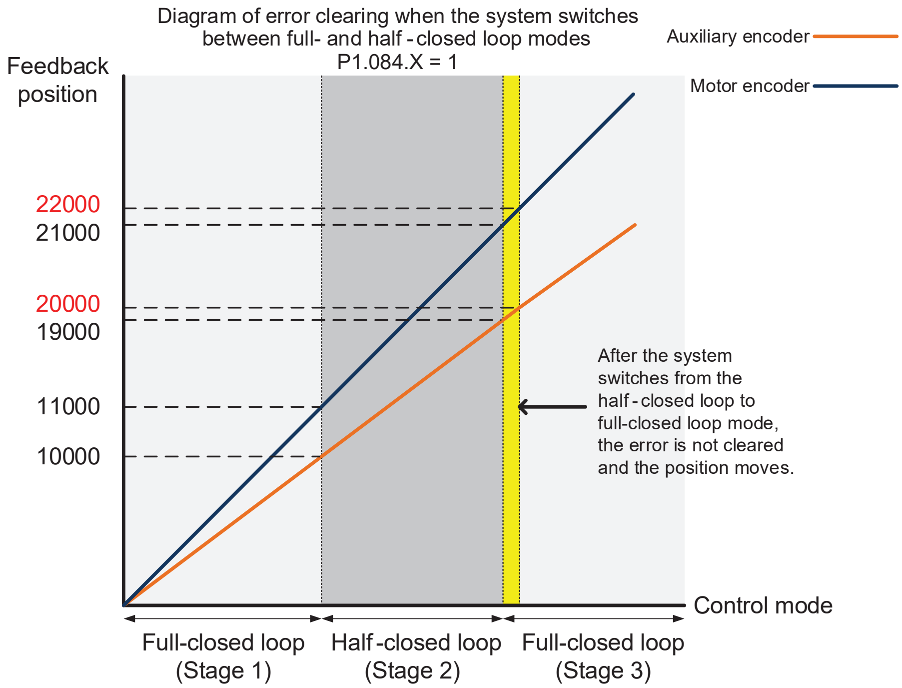
Stage 2: half-closed loop control (feedback position of the motor encoder)

Use DI [0x0B] to switch the control mode from full-closed loop to half-closed loop, and then issue the position command of 10,000 PUU again. In half-closed loop control, since the command refers to the position of the motor encoder, the feedback position of the motor encoder is 21,000 PUU, but the feedback position of the auxiliary encoder is 19,000 PUU. In this mode, there is an error of 1,000 PUU between the auxiliary encoder (19,000 PUU) and the position command (20,000 PUU).

Stage 3: full-closed loop control (feedback position of the auxiliary encoder)

When you set P1.084 to 0, the error will be cleared. Thus, after using DI [0x0B] to switch the control mode from half-closed loop to full-closed loop, the feedback position of the auxiliary encoder is not corrected.

__Error Clearing Disabled (P1.084.X=1)__



Stage 1: full-closed loop control

If the servo drive issued a position command of 10,000 PUU and the feedback position of the auxiliary encoder is 10,000 PUU, the final feedback position of the motor encoder is 11,000 PUU due to the backlash and sliding of the mechanical parts.

Stage 2: half-closed loop control

Use DI [0x0B] to switch the control mode from full-closed loop to half-closed loop, and then issue the position command of 10,000 PUU again. In half-closed loop control, since the command refers to the position of the motor encoder, the feedback position of the motor encoder is 21,000 PUU, but the feedback position of the auxiliary encoder is 19,000 PUU. In this mode, there is an error of 1,000 PUU between the auxiliary encoder (19,000 PUU) and the position command (20,000 PUU).

Stage 3: full-closed loop control

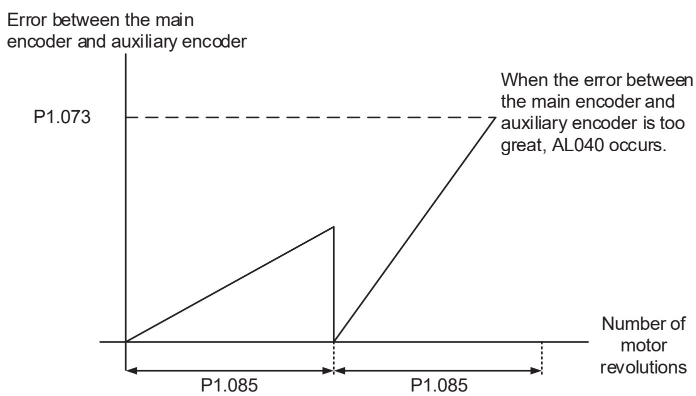
When you set P1.084 to 1, the error will not be cleared. Thus, after using DI [0x0B] to switch the control mode from half-closed loop to full-closed loop, the feedback position of the auxiliary encoder is corrected and the motor moves to the corresponding position (yellow area as shown in the above figure). The previous half-closed loop command becomes the full-closed loop command and refers to the auxiliary encoder to move the mechanical part to the position corresponding to the actual command. The final feedback position of the auxiliary encoder is 20,000 PUU.

6.7.11 - Auto clearing feedback position error

P1.085	**Auto clearing position deviation between motor and auxiliary encoder**		**Hex Address**	**Dec Address**
			01AAH 01ABH	40427 40428
Default:	0	Control mode:	PT/PR (full-closed loop)	
Unit:	rev	Setting range:	0–32767	
Format:	DEC	Data size:	16-bit	

Settings:

This parameter sets the upper limit of the feedback position error between the main encoder and auxiliary encoder. When the number of motor revolutions is greater than or equal to this parameter value, the system automatically clears the error. When set to 0 the parameter is disabled. The deviation value will not reset regardless of the number of motor revolutions. Once the deviation reaches P1.073 then an AL040 will occur.



6.7.12 - Set DI [0x0B] to Switch Between Loop Modes

DI [0x0B] is effective only when the full-closed loop function is enabled (P1.074.X = 1). When the full-closed loop function is disabled (in P1.074.X), the PUU setting of P1.072 is ignored.

DI [0x0B] must be OFF for a full-closed loop feedback from the aux encoder.

Example 1 illustrates the half-closed loop mode when the full-closed loop function is enabled (P1.074.X=1). Example 2 illustrates the half-closed loop function which is normally used in servo applications (P1.074.X=0).

The PUU value of P1.072 is effective when the full-closed loop function is enabled (P1.074.X=1) whether DI [0x0B] is ON or OFF.

Example 1:

Enable the full-closed loop function (P1.074.X = **1**), set DI [0x0B] to ON, E-Gear ratio to 1:1, and P1.072 = 5000.

To have the motor run a cycle when the full-closed loop function is enabled, the position command has to be 5000.

Example 2:

Disable the full-closed loop function (P1.074.X = **0**), set DI [0x0B] to ON, E-Gear ratio to 1:1, and P1.072 = 5000.

To have the motor run a cycle when the full-closed loop function is disabled, the position command has to be 16777216 because the DI [0x0B] setting is ignored and the setting of P1.072 is ignored.

Value: 0x0B			
DI Name	***Description***	***Triggering Method***	***Control Mode***
FHS	Switch between full- and half-closed loop modes.	Level triggered	PT / PR full-closed loop

6.7.13 - Z pulse source of homing

P2.080	***Z pulse source of homing***		***Hex Address***	***Dec Address***
			02A0H 02A1H	40671 40672
Default:	0x0000	Control mode:	PR (full-closed loop)	
Unit:	–	Setting range:	0x0000 – 0x0011	
Format:	HEX	Data size:	16-bit	

Settings:

When you execute homing and have the servo look for the Z pulse, use this parameter to set either the Z pulse of the motor or the Z pulse of the auxiliary encoder as the homing origin. Select the auxiliary encoder to achieve higher positioning precision. Note this is only available in PR mode.

0002

U Z Y X

X	Z pulse source of full-closed loop homing	Z	Reserved
Y	Z pulse source of half-closed loop homing	U	Reserved

- *X: Z pulse source of full-closed loop homing*
  - *0: auxiliary encoder*
  - *1: motor*
- *Y: Z pulse source of half-closed loop homing*
  - *0: motor*
  - *1: auxiliary encoder*

6.7.14 - Encoder output settings

P1.003	**Analog and Encoder Pulse Output Polarity**			**Hex Address**	**Dec Address**
				0106H 0107H	40263 40264
Default:	0x0000	Control mode:	All		
Unit:	-	Setting range:	0–13		
Format:	HEX	Data size:	16-bit		
Related Parameters	P0.003, P1.004, P1.005				

Settings:

0002

U Z Y X

X	Polarity of monitor analog output	Y	Polarity of encoder pulse output	UZ	Reserved

- *X: polarity of monitor analog output*
  The MON1 and MON2 analog output terminals have a max output of ±8V. This equals an 8-volt swing about the center of the GND reference. The X nibble does not slide the center point of the analog output but can invert the MON1 or MON2 output signal. A (+) means normal polarity and (-) means inverted polarity. P1.004 and P1.005 can adjust the proportional output.
  - *0: MON1(+), MON2(+)*
  - *1: MON1(+), MON2(-)*
  - *2: MON1(-), MON2(+)*
  - *3: MON1(-), MON2(-)*
- *Y: polarity of encoder pulse output*
  - *0: pulse output in forward direction*
  - *1: pulse output in reverse direction*
- *UZ: reserved*

P1.046▲	**Encoder Pulse Number Output**			**Hex Address**	**Dec Address**
				015CH 015DH	40349 40350
Default:	2500	Control mode:	All		
Unit:	Pulse	Setting range:	20–536870912		
Format:	DEC	Data size:	32-bit		
Related Parameters	P1.074, P1.076, P1.097				

Settings:

The number of single-phase pulse outputs per revolution for **OA and OB terminals**; the maximum output frequency of the hardware is 19.8 MHz.

Notes:

The following circumstances may result in exceeding the maximum allowable output pulse frequency of the drive, causing AL018:

1) Encoder error
2) The motor speed is faster than P1.076
3) Source= Motor Encoder: If P1.074.Y = 0 and P1.097 = 0, motor speed (rpm)/60 x P1.046 x 4 > 19.8 x 10^6
   Source= Auxiliary Encoder: if P1.074.Y = 1 and P1.097 = 1, motor speed (μm/s)*1000/16777216 x P1.046 > 19.8 x 10^6

P1.076▲	Maximum speed for encoder output (OA, OB)		Hex Address	Dec Address
			0198H 0199H	40409 40410
Default:	5500	Control mode:	All	
Unit:	rpm	Setting range:	0–6000	
Format:	DEC	Data size:	16-bit	

Settings:

Input the actual maximum speed of the motor or the maximum speed of the application. When you set the value to 0, the smoothing function is disabled.

The setting of P1.076 and P1.046 should follow the two requirements below:

- *P1.076 > motor speed*
- $\frac{\text{Motor speed}}{60} \times \text{P1.046} \times 4 < 19.8 \times 10^6$

P2.066	Special bit register 2		Hex Address	Dec Address
			0284H 0285H	40645 40646
Default:	0x0000	Control mode:	PT / PR / S / Sz	
Unit:	-	Setting range:	0x0000–0x182F	
Format:	HEX	Data size:	16-bit	

Settings:

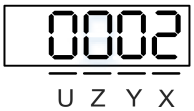
Bit	7	6	5	4	3	2	1	0
Bit	15	14	13	12	11	10	9	8

- *Bit 0–1, Bit 3, Bit 7–8, Bit 10–11, Bit 13–15: reserved*
- *Bit 2: cancel low-voltage error latch function.*
  - *0: enable the low-voltage error AL003 latch function; the error is not cleared automatically.*
  - *1: disable the low-voltage error AL003 latch function; the error is cleared automatically.*
- *Bit 4: disable AL044 detection (servo function overload warning).*
  - *0: enable AL044 detection.*
  - *1: disable AL044 detection.*
- *Bit 5: enable AL041 disconnection detection of linear encoder (only when the full-closed loop control function is activated).*
  - *0: enable AL041 detection.*
  - *1: disable AL041 detection.*
- *Bit 6: RST power error (AL022) latch*
  - *0: disable the latch; RST power error (AL022) is cleared automatically.*
  - *1: enable the latch; RST power error(AL022) is not cleared automatically.*
- *Bit 9: set AL003 Low-voltage as a warning or an alarm.*
  - *0: set AL003 as WARN.*
  - *1: set AL003 as ALM.*
- *Bit 12: set AL022 (RST power error) as ALM or WARN*
  - *0: set AL022 as WARN.*
  - *1: set AL022 as ALM.*

6.7.15 - Full-closed loop feedback source for the controller

P3.013	Full-closed Loop Feedback Source for the Controller		Hex Address	Dec Address
			031AH 031BH	40794 40795
Default:	0x0000	Control mode:	PR (full-closed loop)	
Unit:	-	Setting range:	0x0000 – 0x0022	
Format:	HEX	Data size:	16-bit	

Settings:



X	Encoder feedback source in full-closed loop control	Y	Z pulse offset source in full-closed loop mode (motor/auxiliary encoder)

- *X: encoder feedback source in full-closed loop control.*
  - *0: feedback pulse number from the motor*
  - *1: feedback pulse number from the auxiliary encoder*
  - *2: in half-closed loop control, the feedback pulse is from the motor; in full-closed loop control, the feedback pulse is from the auxiliary encoder*
- *Y: Z pulse offset source in full-closed loop mode (motor/auxiliary encoder)*
  - *0: motor*
  - *1: auxiliary encoder*
  - *2: in half-closed loop control, the motor's Z pulse offset is used; in full-closed loop control, the auxiliary encoder's Z pulse offset is used.*

Note: This parameter setting is different from P1.074.Y (switch between motor encoder and auxiliary encoder). This parameter only modifies the feedback signal source uploaded to the controller. Set P3.013 to 0x0022 to avoid misoperation when the motor is in the Servo On state.

Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

6.7.16 - Troubleshooting full-closed loop alarms

AL040 Excessive Deviation of Full Closed-Loop Position Control	
Trigger condition and causes	Condition: excessive deviation of full closed-loop position control. Cause: 1) The setting value of P1.073 is too low. 2) The connector may be loose or there is a problem when the connector connects to the mechanical parts. 3) The input value for P1.072 can only be an integer. However, when the motor runs a cycle, if the number of A/B pulses in a full-closed loop is not an integer, the position error between the motor encoder and the auxiliary encoder accumulates. Thus, you need to set P1.085 to avoid triggering AL040.
Checking methods and corrective actions	1) Check the value for P1.073. If the value is too low, please set a higher value. 2) Make sure the connector is firmly connected and there is no problem in connecting the mechanical load. 3) Check if the value of P1.085 is set properly.
How to clear the alarm?	DI.ARST

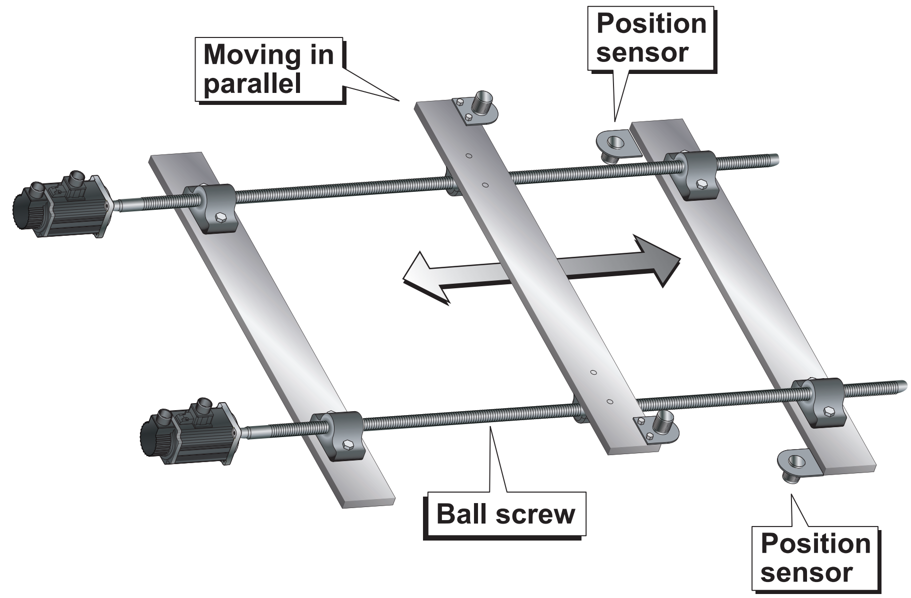
AL041 CN5 Encoder is Disconnected	
Trigger condition and causes	CN5 communication is cut off.
Checking methods and corrective actions	1) Check the communication circuit of CN5. 2) When CN5 is not in use, ensure P1.074.X is set to 0.
How to clear the alarm?	DI.ARST

6.8 - Gantry Mode

This section explains the gantry setting and how gantry works when it is used on SureServo2. The Gantry function allows both motors to watch the other's position and if they get too far out of alignment then both drives will fault out and stop.

6.8.1 - How does gantry mode work?

When two axes control a platform, they must move with the same speed as significant speed deviation between the two axes could damage the mechanism. Synchronizing the motion of the two axes is the top priority. See the example below:



The built-in gantry control function for SureServo2 drives allows the controller to synchronize motion automatically. If position devation exceeds the permitted range, an AL081 alarm will occur and the system will stop working. In this application, an open-loop control is used by the host controller and the servo system; the function of the host controller is to send position commands. The host controller is in charge of the alignment and homing control of two axes. If using the Z pulse as the homing origin, the host controller requires the capability to respond to the fast Z pulse signal of 66μs from the drive.

If misalignment of two axes does not occur or is not possible on the user's mechanism, then the positioning or homing function is not needed. Otherwise, positioning or homing is required before gantry mode is enabled because after the machine is moving the alignment between the two axes cannot occur.

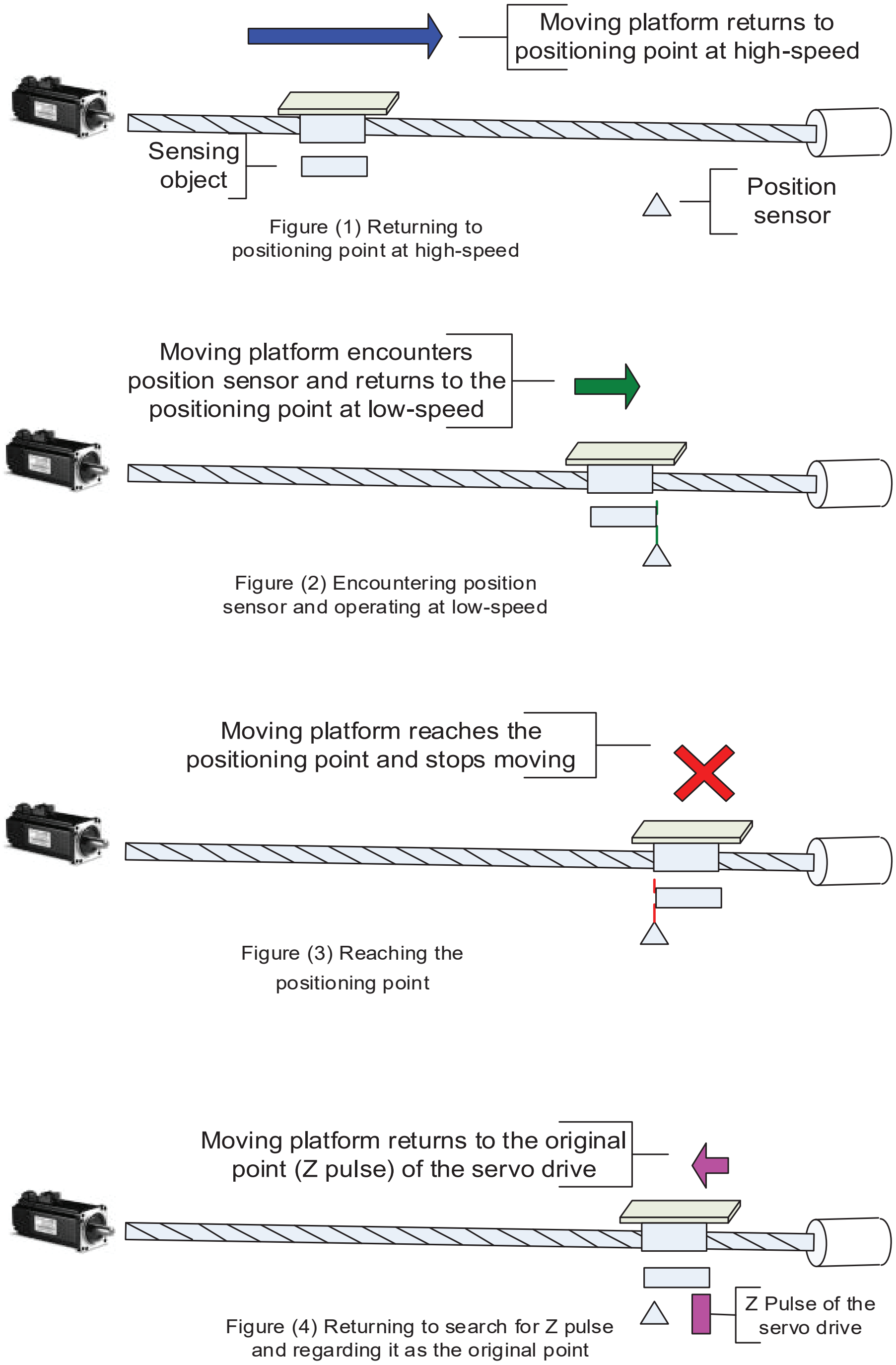
Gantry mode can be used in two ways. The simplest way is just to monitor the position deviation between the two axes and if the position difference is greater than the preset limit then an alarm will activate and the axes will stop. This is the easiest protection method in a difficult to tune system.

The second and more difficult method is to use the synchronous control bandwidth which blends the error of the two axes speed loops together so they sync to each other. Applying this method can cause the individual tuning of each drive to perform different and resonate.

6.8.2 - Positioning and Homing of Gantry

Homing and proper alignment of the two axes by the host controller must be done before using the Gantry function. It is up to the user to create a proper homing or position routine that aligns the two axes. Homing is completed by a position sensor installed on the side of each axis. This position sensor must be precisely installed since it is this sensor that ensures the gantry axes are aligned correctly. Adjust the length and running speed of the sensor according to system requirements. The illustration “Returning to Positioning Point and Homing Origin” on page 6–60 shows the positioning control; after positioning is finished, this point can be regarded as the homing origin (shown in figure (3)). Or, as shown in figure (4), the nearest Z pulse can also be the homing origin (either moving forward or backward to look for Z). The homing method will vary depending on the application.

After alignment, the host controller should then take control of both axes simultaneously to home the gantry and look for just one of the axes Z pulse signals or home sensors.



Returning to Positioning Point and Homing Origin

The figure below demonstrates the gantry misalignment before positioning. If position deviation between two axes exists, one of the axes will arrive at the low-speed zone earlier than the other. When any of the axes reaches the low-speed zone, the entire system will operate at low speed. Due to the deviation, the axis entering the low-speed zone first will reach the positioning point earlier. In figure 1 below, Axis 1 reaches the positioning point first and stops and waits for Axis 2 to arrive. After both axes reach the positioning point, both axes can then move forward (or backward) at the same time and look for Z pulse as the homing origin. The positioning point can also be regarded as homing origin as needed for different applications and demands.

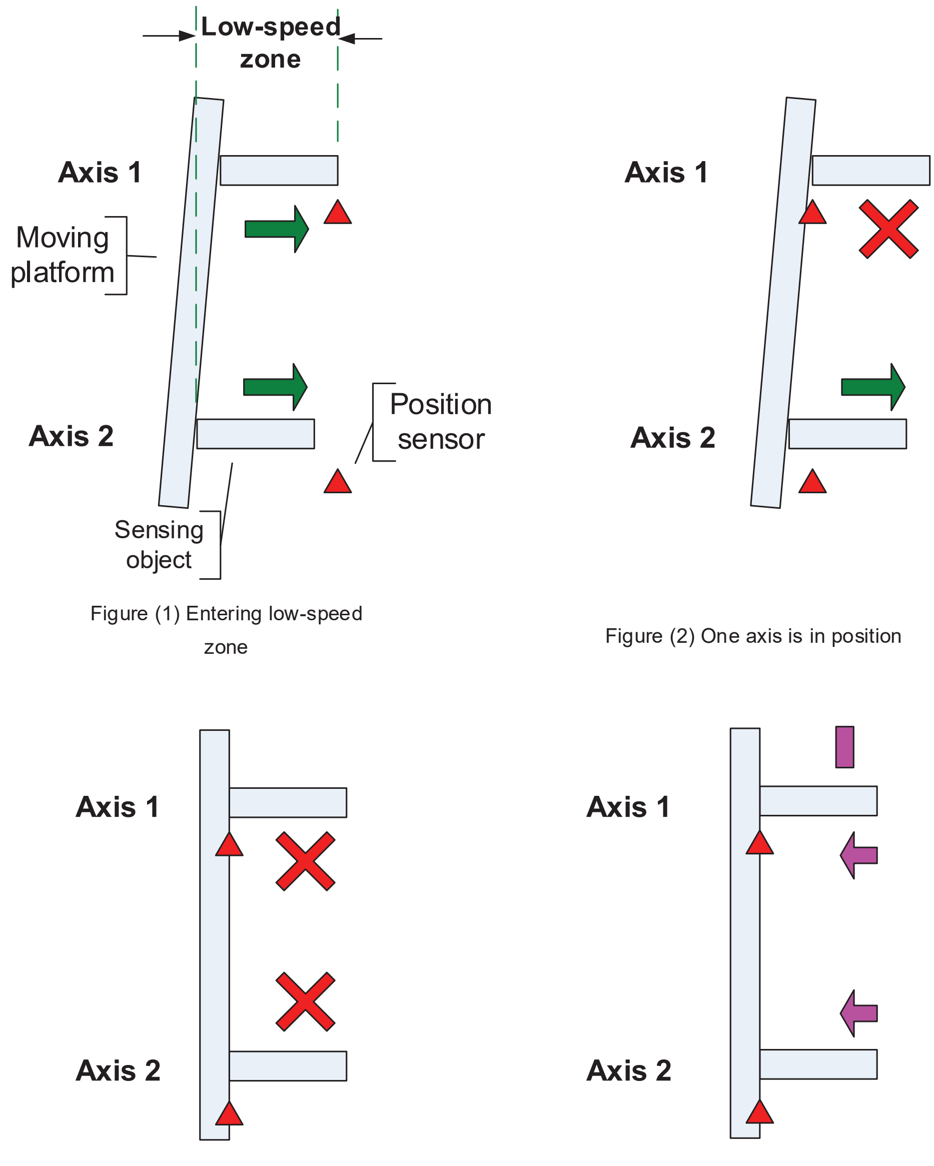


Figure (1) Entering low-speed zone

Figure (2) One axis is in position

Figure (3) Both axes are in position

Figure (4) Both axes synchronously return to Z pulse

System Positioning and Homing

6.8.3 - Motion Following

When gantry alignment is complete and the axes have returned to the home origin, the host controller sends the same position commands (pulses) to each drive so the two axes are synchronized. The drive must be operated in PT mode. The host controller is responsible for all acceleration/deceleration times.

6.8.4 - Servo System Wiring

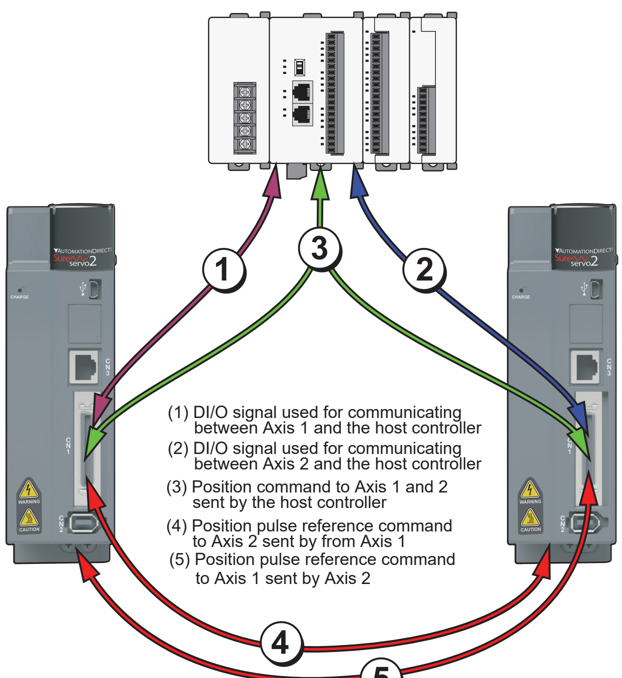
The System Connection Diagram (below) shows the basic connections of the entire system. Users may apply different applications according to actual needs. The "System Wiring Diagram" on page 6–64 shows detailed wiring.

DI Signal

- *SON (0x01): Servo On.*
- *CCLR (0x04): Pulse Clear.*
- *ARST (0x02): Alarm Reset.*
- *GTRY (0x0A): Gantry Stop (Pause); when this input is activated it changes P1.074.X (synchronous control function) from 2 (enable) to 0 (disable) so the two axes can be driven separately and not result in an AL081 alarm.*
- *EMGS (0x21): Emergency Stop; external switch. Make sure that both axes can synchronously receive this signal.*
- *INHP (0x45): Pulse Input Inhibit; when this signal is on, any input pulse signal will not be* admitted. Please note that this signal can only be used on DI8.

DO Signal

- *TPOS (0x105): Reach the Target Position, a reference for the host controller.*
- *SRDY (0x101): System Ready, waiting for the start-up command.*
- *SON (0x102): Servo on; servo system is able to receive commands from the host controller.*



System Connection Diagram

Pulse signal of position command

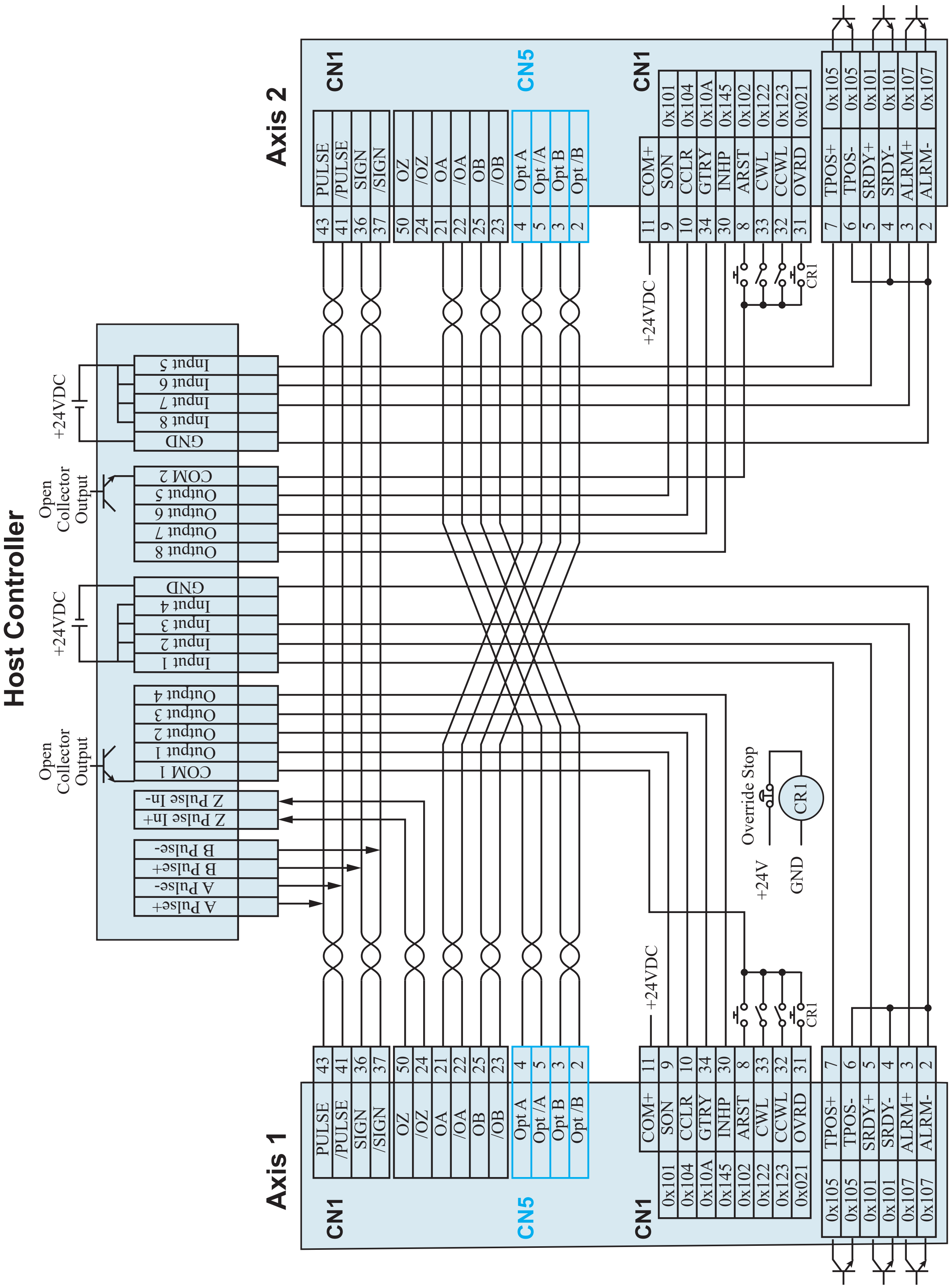
The pulse signal from the host controller should be directly parallel-connected and fed to both axes simultaneously. If a parallel connection is not used, then two separated high speed pulse output channels can be used if they are electronically locked together to provide the exact same pulse train profile move. If using open collector, please carefully apply the wire and the power to avoid short circuit. If Z pulse is used as the homing origin, then the Z pulse from only one of the axes should be sent back to the host controller.

The pulse signal communication between two axes

On Axis 1, CN1 will send pulse signals OA, /OA, OB, and /OB to OptA, /OptA, OptB, and /OptB of CN5 on Axis 2. On Axis 2, pulse signals OA, /OA, OB, and /OB from CN1 have to be sent back to CN5 of Axis 1, received by OptA, /OptA, OptB, and /OptB. This wiring is specially designed for use with Gantry mode. For ease of wiring to the CN5 connector use ZL-HD15M-CBL-DB15F cable + ZL-RTB-DB15 ZIPLink breakout board, or use ZL-HD15M-CBL-2P cable (HD15 to flying leads). See section 3.9 for cable pinout and wire color.

A detailed reference for wiring

The "System Wiring Diagram" on page 6–64 is the detailed wiring reference. This reference is for the wiring of the servo system only. Direct signal input to the host controller such as position sensor is not included in this diagram.



System Wiring Diagram

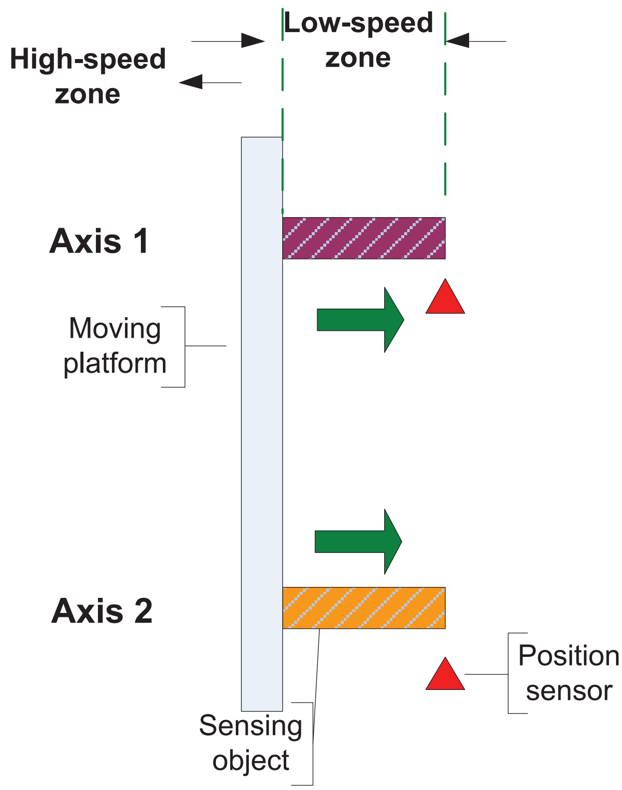
6.8.5 - Sequential Logic Control of Positioning and Homing

For gantry control, the positioning and homing control logic has to be completed by the host controller. The control sequence of a host controller and how it works is explained in previous sections. Detailed timing diagrams are shown on the following pages. Users can decide whether to use either the positioning point or Z pulse as the homing origin.

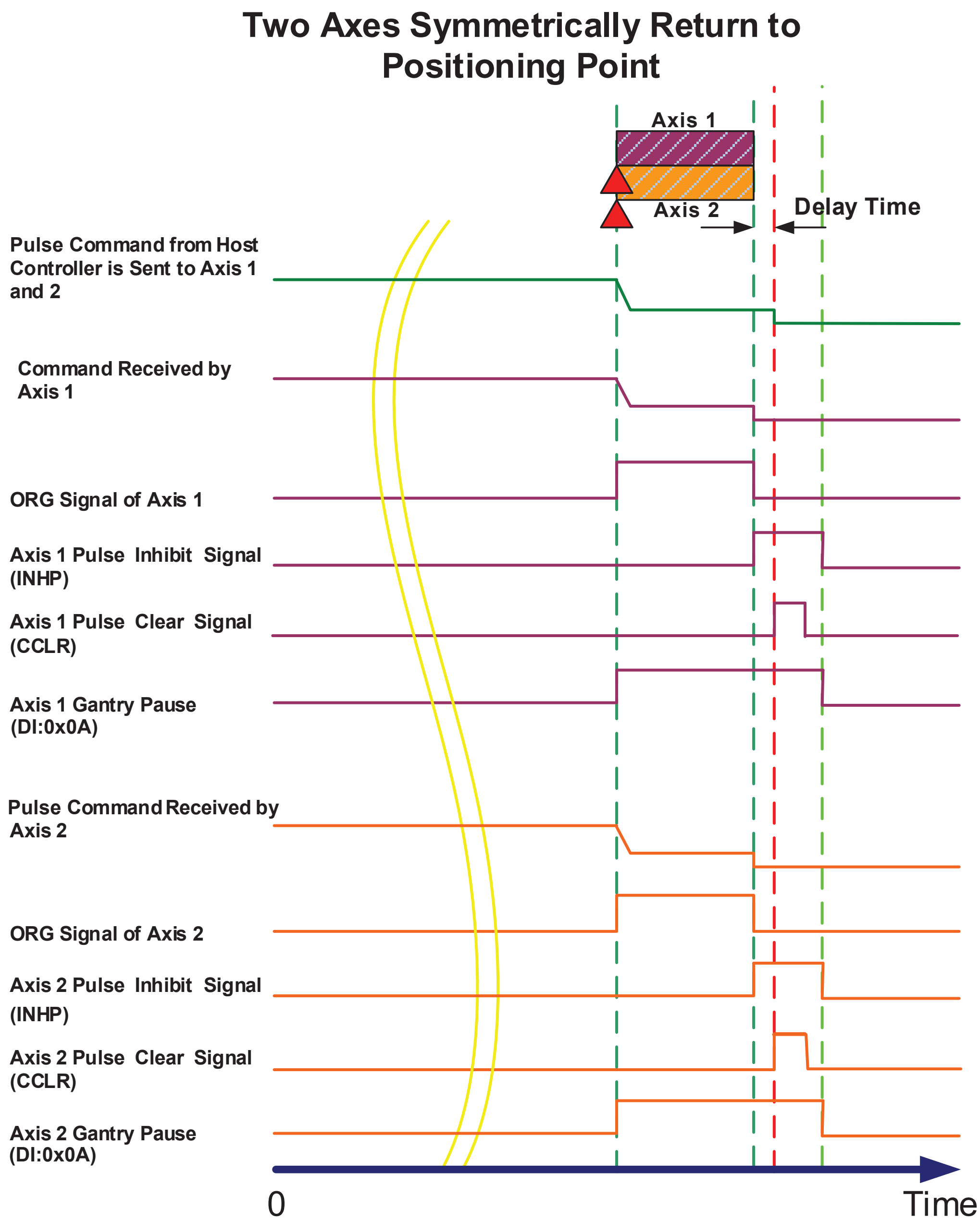
The homing example below has the host controller sending parallel pulses to each drive with the Gantry Pause DI activated so that Gantry mode is temporarily suppressed. This allows each axis to home individually if this method is desired.

Two axes symmetrically return to positioning point

If no abnormality occurs when gantry is working, two axes will be in symmetry when performing homing as shown below.

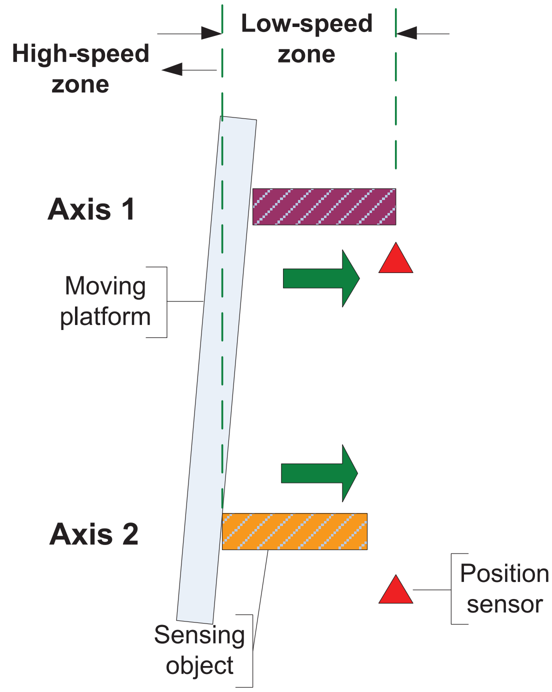


The timing diagram of two axes symmetrically return to positioning point

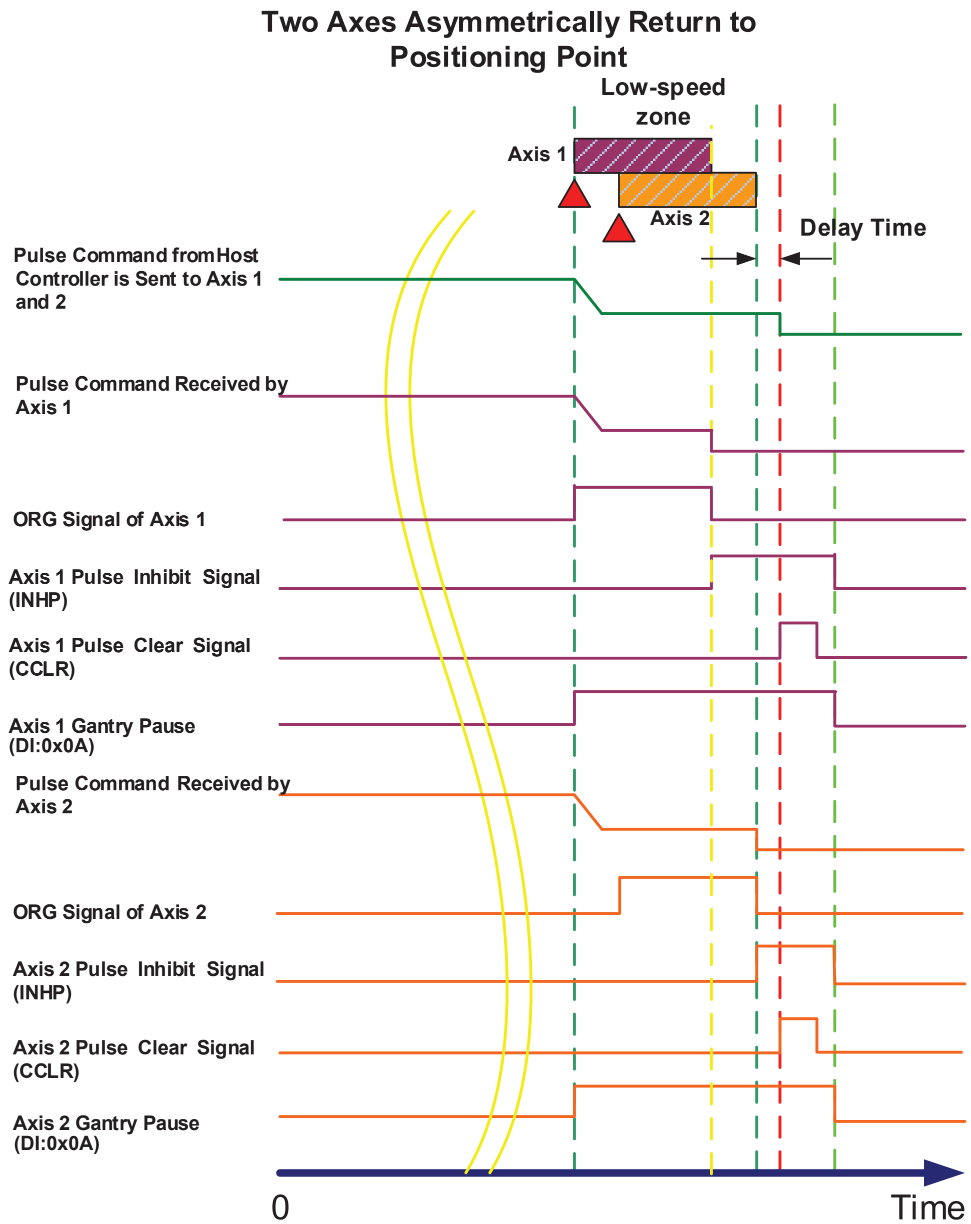


Two axes return to positioning point asymmetrically

As shown below, if any unexpected problem occurs during the operation that results in asymmetry of two axes, the position of two axes can be corrected by homing.



Timing diagram of two axes return to positioning point asymmetrically

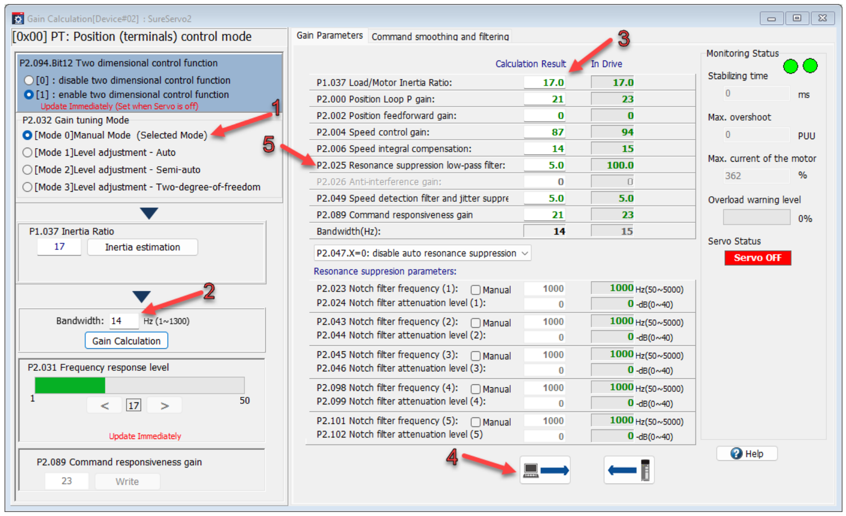


6.8.6 - Steps for Adjusting the Servo When Using Gantry Control

The following steps are about the gantry setting and parameter adjusting.

Note: Before configuring for gantry synchronous mode Set P2.025=100 and P2.057=0. This will reduce the risk of unintended position correction and motor resonance.

Step	Action
1	***Check the wiring.*** Please refer to section "6.8.4 - Servo System Wiring" on page 6–62 and make sure the wiring is correct.
2	***Tune each axis individually.*** This should be done if it is possible to decouple the two axes' moving platforms from each other. This will give you a better starting point for tuning when the complete gantry mechanism is attached. Ensure P1.074.X=0 to deactivate gantry mode. Open the Auto Tuning window and perform the proper tuning needed while ensuring emergency stop switches are working and the motion profile and speed is adequate for the tuning.
3	***Set up the inertia ratio of the system.*** Pause the gantry function (DI=0x0A activated). Check that all emergency stop and positive/negative limits are working properly. Open the Gain Calculation window in SureServo2 Pro and set P2.032=1 this will allow the drive to estimate the inertia value real time. Set P0.002=15, this will allow you to see the real time inertia on the front display of the drive. Let the host controller issue a position command to both axes to make the gantry mechanism move back and forth with low speed to make sure the mechanism is working fine. Then, gradually speed up the gantry to the speed you will likely be using in the final application. The gantry should be moving back and forth in the desired tuning profile. Once the inertia value has stabilized, enter this value in P1.037. Change P2.032, Gain Tuning Mode, to 2 so the inertia will not change. The inertia ratio is the calculation basis for servo motors' operation; this value must be correct. Both sides of the same gantry mechanism will have slightly different inertias. After setting both axes to Mode 2 for P2.032, take note of the Bandwidth in each Gain Calculation window. For Synchronous control to work both BW values must be the same. Lower the higher BW value to match the lower BW value in the other Axis. *Note: Add both drives to the device list and connect SureServo2 Pro to each drive with a separate USB connection to easily monitor and configure each drive.* *WARNING: NEVER AUTO TUNE (FROM THE DRIVE COMMAND OR A HOST CONTROLLER COMMAND) OR USE THE INERTIA ESTIMATION WINDOW FOR EITHER MOTOR INDEPENDENTLY WHEN ATTACHED TO THE MECHANICAL GANTRY. THIS CAN CAUSE MECHANICAL DAMAGE BECAUSE THE TWO MOTORS WILL NOT STAY SYNCHRONIZED* ***(Steps on next page)***

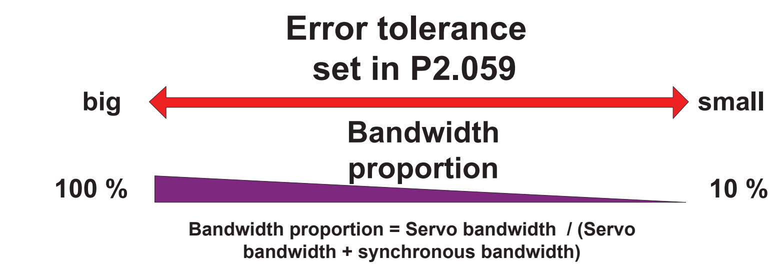
Step	Action
3	***(Continued from previous page)*** To manually lower the BW of the higher axis, perform the following in the image below: *1) Change P2.032 to Mode 0 in the Gain Calculations window and click confirm if not already in Mode 0.* *2) Enter the lower bandwidth value from the other drive and click the Gain Calculation button.* *3) The new values will populated in the Calculation Result column.* *4) Click the Transfer from PC to Drive button to download the Gain parameters to the drive and also to populate them into the Parameter Editor window.* *5) P2.025 will automatically drop to a low value. For Gantry Synchronous control this should be high during commissioning the system and lowered according to the application. In the Parameter Editor window, set this back to a high number and download to the drive.* *6) Set P2.032 back to Mode 2 and click Confirm.* 
4	***Output pulse setting for monitoring.*** For gantry synchronization, the controller's monitoring pulse receiving speed (which is CN5's capability of pulse receiving) needs to be set. The limit is calculated as below. $\frac{MotorSpeed}{60} \times P1.046 \times 4 < 8 \times 10^6$ <u>*Calculation Example:*</u> The pulse command from the host controller at maximum speed is 50000 pulse/s. And the E-gear ratio is 20 times. When the encoder completes a full rotation, the feedback pulse is 1280000 (without going through the E-gear). (Pulse command * E-gear ratio) / Actual pulse number * 60 sec. = Motor speed per minute (RPM), (50000 * 20) / 1280000 * 60 = 46.875 RPM = the maximum motor speed set by the control command Based on the above formula, the permitted maximum setting value of P1.046 (Pulse Number of Encoder Output) can be decided. Improper gain adjusting on a controller might lead to speed overshoot when the motor is operating; meanwhile, the motor speed might exceed the maximum speed set by the command; therefore, the overshoot level should be taken into consideration. For instance, if a margin of 10% is reserved, it has to be increased depending on the circumstances when using special mechanism. $(46.875 * 110\%) / 60 * (P1.046) * 4 < 8 * 10^6$ Two servos' setting of P1.046 has to be the same; this is the output resolution of the motor. The higher the resolution is, the better control of the gantry will be. But, if it exceeds the controllable range, accuracy would be affected when calculating the deviation between two axes. After setting up P1.046, please set up P1.072 (Resolution of Auxiliary encoder): (P1.072) = (P1.046) * 4, both settings of the two servos should be the same.

Step	Action
5	***Set up the permitted deviation value of synchronization.*** Set P2.059 for the max permitted deviation between the two servos. This is the maximum PUU count that the two servos can be off from each other. When the deviation exceeds the range, AL081 will occur. Thus, be sure to consider the position displacement deviation of two axes that the actual mechanism can tolerate. If the set deviation value goes beyond the actual mechanism's tolerance, the mechanical system may be damaged. For instance, the pitch of the ball screw is 10 mm, P1.046 = 60000 and P1.072= 240000. If P2.059 is set to 30000 pulse, the deviation of two axes can be calculated as: $\frac{30000}{240000} \times 10 = 1.25 \text{ mm}$ When the deviation of two axes is over 1.25 mm, the alarm will occur.

Step		Action
6		***Check if motor feedback and auxiliary pulses are incrementing correctly.*** Open the Scope in the SureServo2 Pro software. Both servos can be connected in the same instance of SureServo2 Pro by right clicking on "Device List" and selecting "Add Device". This way you can see the scope trace of both motors at the same time in two different scope windows. Ensure P1.074.X=0 for this step.
	A	At the bottom of the scope, select [IDX] Normal and 32-bit on CH1, and select Auxiliary encoder feedback [Pulse], which is the feedback pulse input from the other drive. This is monitoring the auxiliary encoder port (CN5) in the servo drive. This would be a 32-bit value (monitoring the moving direction of the other servo motor).
	B	Select [IDX] Normal and 32-bit on CH2, and select Feedback Position [PUU]. This is the feedback position of the motor (monitoring the moving direction of the servo drive that connected to the scope).
	C	Let the host controller issue position command to make the two motors move at the same time and then monitor the variation in the SV2-PRO scope. If CH1 is increasing and CH2 is decreasing then the CN5 pulses are not moving in the correct direction. To fix the direction of CH2 to match CH1, the value of P1.074.Z must be changed to the other direction.
	D	If the setting is correct, the signals will be incrementing in the same direction.
	E	Then, connect the PC scope to the other servo drive (if both were not connected to earlier) and make sure the direction of feedback pulse is correct.
7		***Activate the synchronous control.*** Disable both drives, set to SV_OFF. Activate the synchronous control via P1.074X to 2, the synchronous control of gantry function will be activated. Ensure DI.0x0A=OFF.
8		***Trial Runs***
	A	Keep the Gantry mode either deactivated (P1.074.X=0) or paused (DI.0x04=On) while conducting trail runs.
	B	After setting the Synchronous Control Bandwidth (P2.057) to a proper value (adjust from small to large), let host controller issue position commands and observe the position deviation and synchronization of two axes via the SV2 scope. Select CH1, [Mon], and 32-bit and then enter 071; this would be the position deviation between both axes and the unit is pulse (using P2.059 as a basis). If the deviation of two axes exceeds the setting value, alarm AL081 will occur. There is no chance that the loading, tuning, and inertia conditions of two axes are exactly identical, so the acceleration/deceleration times will cause a noticeable position deviation during acceleration and deceleration positions.
	C	When conducting trial runs, be sure to adjust the parameters to proper values; the individual axis bandwidth settings of the two controllers has to be identical so as to avoid alignment deviation due to their different response time. When executing the acceleration/deceleration command from the host controller, the position deviation has to be within the setting range of P2.059; otherwise, the alarm will occur.

Wiring
Parameters
DI/DO Codes
Monitoring
Alarms

Step		Action
9	***Synchronizing test and parameters adjustment***	
	A	Be sure to complete the steps described above. Then, use parallel pulses to control the gantry mechanism between two motors, and then start testing the gantry.
	B	Please do Step 3 mentioned above again if the inertia has changed due to mechanical changes. Re-estimate the system inertia; otherwise, the system setting will not be accurate and unable to work properly. If the mechanism is not symmetric, inertia ratio of two axes will be different.
	C	The system must be protected by a proper value in P2.059 to avoid any errors that could cause mis-alignment of the gantry mechanics.
	D	The bandwidth settings of both servo drive and gantry synchronous control bandwidth has to be identical. The bandwidth of the servo drive can be calculated and set in the Gain Tuning window via SureServo2 Pro; Gantry bandwidth can be set via P2.057 (Synchronous Control Bandwidth). See figure below:



Servo bandwidth	100%	90%	80%	70%	60%	50%	40%	30%	20%	10%
Synchronous bandwidth	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%

Regarding the synchronous bandwidth, P2.057 is the only related parameter that needs to be set. The system will automatically calculate the value of P2.054~P2.056 (the related parameters of synchronous control). When the bandwidth setting of synchronous control is wider (higher) than the bandwidth of servo drive, the synchronous following between the two motors is tighter. However, the following error for the host controller command pulses will be relatively worse. Please note that when Bandwidth of synchronous control + Bandwidth of the servo drive is greater than the Permitted bandwidth of the system, it is easier to cause resonance in the motor.

Bandwidth proportion example 1:

Drive 1 and 2 BW = 14
Synchronous BW (P2.057) = 80

$$\text{Bandwidth proportion: } \frac{14}{14+80} = 14.8\%$$

According to the scale above the synchronous feedback setting are more dominant.

(Step 9D continues on next page)

Step	Action	
9 (cont'd)	***Synchronizing test and parameters adjustment (continued)***	
	D (cont'd)	***Bandwidth proportion example 2:*** Drive 1 and 2 BW = 14 Synchronous BW (P2.057) = 20 $$Bandwidth\ proportion: \frac{14}{14+20} = 41.2\%$$ According to the scale above the servo feedback settings are slightly more dominant than the synchronous feedback setting. If bandwidth cannot be increased to achieve better following, please try to increase the value of P2.055 (Synchronous speed integral compensation). However, if the value of P2.055 is set too high, system vibration will occur. When deciding the bandwidth, be sure that the setting value of P2.025 (Resonance suppression lowpass filter) is much bigger than the bandwidth setting; otherwise, the result might not be satisfactory, and the system might become unstable. When adjusting the bandwidth of the synchronous control, start from small to large. The synchronous control of the gantry is shown below. The synchronous controller affects the speed loop of the cascade below and is directly added to or subtracted from the individual drive's speed loop component. The synchronous control of the gantry is shown below: