

HOMING TECHNIQUES

Subject: Homing
Author: Tom Keller
Date: 8/4/2021

Before You Begin: The ProMotion Windows GUI can be used to verify proper connections related to Encoders, Index, Limit Switches and Home Switches prior to attempting a Homing procedure.

“Homing” is a sequence of moves that is used to establish an absolute reference frame which is typically required in position-based applications. The procedure is repeated for as many axes as necessary. From that point on all future position values are defined as absolute co-ordinates within that reference frame.

The details of the sequence depend on what types of sensors and feedback devices exist in the system. This can be divided into two major common categories depending on whether or not an incremental encoder is present.

- **Incremental Encoder Present**

Option #1 – Move axis toward Home Switch and capture the encoder position of the Home Switch.

Option #2 – Move toward a Limit Switch and stop. Move out of the limit switch and capture the encoder position of the first index.

Option #3- Move toward a mechanical hard limit. Once detected, reference the system to the position of the hard limit.

- **Step Motor with No Encoder**

Move toward negative Limit Switch quickly and stop. Move out of switch slowly and then back into switch slowly and stop.

Homing with Incremental Encoder

Mentioned above are three options available when an encoder is present. All have their own advantages and disadvantages. In some PMD products the encoder index is qualified with the encoder A/B channel, so the captured position of the index has the potential to be more repeatable than the captured position of a Home Switch. However, the relationship between the limit switch and the encoder index can vary from motor to motor depending on how the motor/encoder assembly is performed and how the assembly is installed in the system. The third option does not rely on any sensors but does not account for any mechanical compliance in

the hard limit which can affect the repeatability of the homing procedure. The user must determine which method to use based on system requirements.

The procedures below assume that all necessary initialization commands have been sent to the PMD controller such that it is ready to perform a move. The encoder-based homing methods utilize various inputs such as a home, index, and negative limit. These inputs are considered to be active low by default. If an active high switch/index is to be used then the appropriate bit in the SignalSense register should be set high. During initial prototyping and before attempting a homing routine, GetSignalStatus should be used to verify the switch and/or index is being detected. When SignalSense has been properly used, the appropriate bit in the SignalStatus should go low when the switch is active.

Signal	Signal Status bit#	Signal Sense bit#	Event Status bit#
Index	2	2	3*
Home	3	3	3*
Negative Limit	5	5	6

*When selected as the Capture Source

Homing to a Home Switch

This procedure assumes a home switch and a negative limit switch are present in the system.

Before sending commands to create motion toward the home switch, the GetCaptureValue command should be issued to flush out any previous captures. Likewise, ResetEventStatus should be used to clear the CaptureReceived bit.

The CaptureSource should be set to HOME. GetSignalStatus should be used to determine if the home switch is currently active. If so, a move out of the switch in the positive direction should be commanded and stopped when the home switch is no longer active.

GetSignalSense can also be used to verify if the Negative Limit is active. If so, a move out of the switch in the positive direction should be commanded. When the home switch is detected it will continue to move until the switch is no longer active.

The next step is to place the controller in the velocity contouring profile mode and send appropriate acceleration and velocity values such that motion begins in the negative direction toward the home switch.

After motion begins, GetEventStatus should be polled until bit 3 goes active. At which point GetCaptureValue should be used and the returned value recorded. SetStopMode/Update can be used to stop motion but is not necessary. The home switch can then be set to be the “zero” position by using the command AdjustActualPosition. The argument to this command should be the negative of the recorded value returned by GetCapturePosition. At this point the axis coordinate system has been re-referenced such that the home switch represents the origin (zero).

If the initial position is on the negative side of the home switch, GetEventStatus will eventually detect that the negative limit has been reached (Bit 6). In that case a move in the positive direction is commanded. When the home switch is detected, the move continues until the home switch is no longer active. At which point a move in the negative direction is commanded and polling for the home switch begins as described in the previous paragraph.

Homing to Encoder Index

Note: This method is only available when using an Encoder with an Index and a negative limit switch, but does not require a home switch.

Some of PMD position controllers “gate” the index signal which means it will only be recognized when Encoder channels A and B are low. Not all encoders are manufactured with this relationship and swapping of encoder wires may be necessary. Products that do not gate will instead use edge-based detection which is identical to the home switch detection process.

PMD Controller	Index Capture Method
MC5xxx0	Gated
MC5xxx3	Edge
ION and ION/CME	Gated
Prodigy boards	Gated

During initial prototyping and before attempting a homing routine, care must be taken to ensure the index is being recognized properly. In the context of rotary encoders, the index should appear once and only once per revolution. To verify this, spin the encoder by hand and use the GetCaptureValue command in a tight loop. (or use the POSITION FEEDBACK window in Pro-Motion.) GetCaptureValue will return the same value until a new index appears. The difference between the new value and the old should always be equal to the number of encoder counts per revolution.

Before commanding any motion the state of the negative limit switch should be verified by checking bit 5 in the SignalStatus register. If the switch is active, a slow move out of the switch (positive velocity) should be started and then stopped when bit 5 is no longer active (high).

Place the controller in the velocity contouring profile mode and send appropriate acceleration and negative velocity values such that motion begins in the direction of the negative limit switch.

After motion begins, GetEventStatus should be polled until bit 6 goes active. By default motion will stop when the negative limit switch goes active. The CaptureSource then should be set to INDEX. The GetCaptureValue command should be issued to flush out any previous captures that may have occurred. Likewise, ResetEventStatus should be used to clear the CaptureReceived and NegativeLimit bits.

Next a trajectory will be started that moves the axes out of the negative limit. This is easily achieved by defining a positive velocity. After motion begins, GetEventStatus should be polled until bit 3 goes active. At which point GetCaptureValue should be used and the returned value recorded. SetStopMode/Update can be used to stop motion but is not necessary. The first index out of the negative limit switch can then be set to be the “zero” position by using the command AdjustActualPosition. The argument to this command should be the negative of the recorded value returned by GetCapturePosition. At this point the axis coordinate system has been re-referenced such that the closest index on the positive side of the negative limit switch represents the origin (zero).

Homing to a Hard Stop

This procedure will begin by moving the axis toward a mechanical “hard” limit (or “hard stop”). A hard limit is a position whereby the axis cannot move beyond as a result of a mechanical obstruction. When the hard limit is reached the actual motion will stop, but the profile will continue moving until the position error exceeds the PositionErrorLimit setting. At which point an AbruptStop will occur because the SetEventAction command is used to change the MotionError response to AbruptStop. When the MotionError occurs, a position error will still exist and therefore the axis will still be pushing against the hard limit. If the hard limit is on the far side of a limit switch, the SetEventAction command may be needed again which can be used to temporarily disable the limit switch reaction (Action=None).

Because of the existence of the constant position error, the ActiveMotorCommand will increase until it reaches saturation at the MotorLimit value. Therefore, MotorLimit can be used to place an upper bound on how hard the axis is pushing against the hard limit. When the host detects the MotionError using GetEventStatus, the host will then poll GetActiveMotorCommand until the absolute value returned is equal to the MotorLimit. *Hint: If the ActiveMotorCommand does not saturate at the MotorLimit value than the Position Loop Integration Limit needs to increase.*

The SetActualPosition command is used to reference the position of the hard limit. Then the ClearPositionError command is used to servo the axis to the hard limit position. The axis is now referenced to the hard stop and normal operation can begin. It is recommended that MotorLimit, PositionErrorLimit and SetEventAction be returned to their original values if altered.

Homing a Step Motor to a Limit Switch (no Encoder)

Many step motor applications do not have an encoder, however if a step motor with an encoder is being use, the Encoder based homing routines described above should be used because of better repeatability.

As is the case with the Encoder based homing methods, the procedure below assumes that all necessary commands have been sent to the PMD controller such that it is ready to perform a move.

To optimize the repeatability of this procedure the axis should have a small constant velocity when it encounters the limit switch. The reasons for this are explained below in the [Deep Dive into Step Motor Accuracy](#) section. At initialization the distance to the negative limit switch is unknown, it may take an excessive amount of time to encounter the limit switch using a slow velocity. An alternative is to do a fast

move to the negative limit switch, back out of the limit switch a small amount, and then move into the limit switch slowly.

Before commanding any motion the state of the negative limit switch should be verified by checking bit 5 in the SignalStatus register. If the switch is active, a slow move out of the switch (positive velocity) should be started. When bit 5 is no longer active (high) the slow move into the switch can be performed. The “fast” move described in the two paragraphs below can be skipped.

After verifying the negative limit is not active place the controller into the velocity contouring profile mode and specify an appropriate acceleration value and a nominal negative velocity value. ResetEventStatus should be used to clear the NegativeLimit bit just in case it is set.

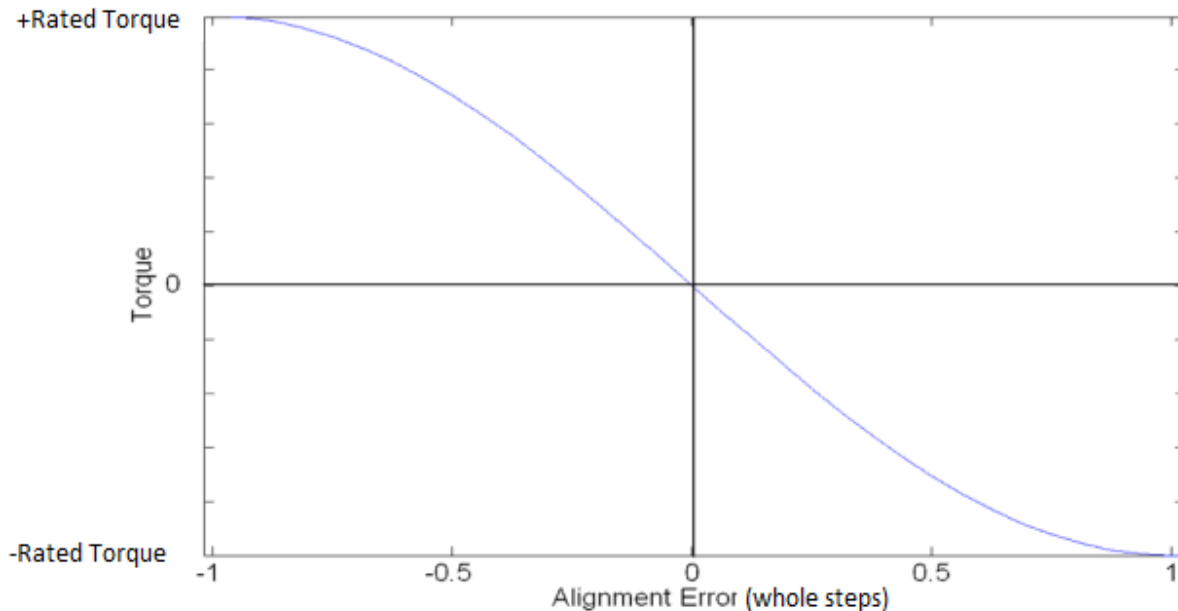
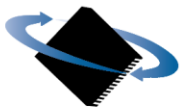
After motion begins, GetEventStatus should be polled until bit 6 goes active. By default motion will stop when the negative limit switch goes active. Next a trajectory will be started that moves the axes out of the negative limit. This is achieved by defining a slow positive velocity. Before this happens, ResetEventStatus should be used again to clear the NegativeLimit bit. After motion begins, GetSignalStatus should be polled until bit 5 goes inactive and then SetStopMode/Update can be used to stop motion.

To begin motion back into the limit switch a small constant velocity (< 1.0 steps/cycle) or less is specified. Once again the motion will stop when the negative limit switch is encountered. When the limit switch has been detected using GetEventStatus, the ‘SetActualPosition 0’ command can be used to define the position of the negative limit switch. At this point the axis coordinate system has been re-reference such that the negative limit switch represents the origin (zero).

The torque the step motor is exerting when stopped at the negative limit will affect the value of commanded position at that location. If there is a force acting on the step motor which pushes it in the positive direction, the commanded position will need to move farther in the negative direction to counter act that force. Ideally the load on the step is zero at that location, otherwise the load should be consistence at that location. For more information see below.

Deep Dive into Step Motor Accuracy:

As mentioned above the torque relationship between the “Commanded Position” and real position of the motor has an impact on the accuracy of homing. Control of a step motor attempts to drive the motor shaft (rotor) to a specific alignment with respect to the motor housing (stator). This point is represented in the figure below where the Alignment error is zero. However note that the torque at this location is zero. This implies that **if there is non-zero torque acting on the motor shaft, the alignment error will have to be non-zero in order to balance that torque.**



This property will affect the accuracy of the homing procedure. The “Commanded Position” described in the procedure above is the desired rotor/stator alignment (which changes when in motion). The amount of torque on the shaft will affect how far away the “commanded position” is from the real motor position. Therefore the motor should be at constant velocity when the limit switch is tripped to minimize the inertial torque on the motor (higher torque exist during acceleration and deceleration). Also, the motor velocity should be small in order to minimize bearing (viscous) friction. After taking these optimization steps there will likely still be some torque on the shaft when the limit switch is seen which means the “Commanded Position” when the limit goes active will be beyond the limit switch in most cases. If the torque (alignment error) at this time is far below the rated torque then this effect will be negligible. If the torque is repeatable then the homing procedure will be repeatable. However, if during the homing procedure, the torque at the time the limit switch goes active is variable, the repeatability of the homing process and thus positional reference may be compromised.

Please contact PMD for additional information.
support@pmdcorp.com +1.978.266.1210