Homing with the PTi210 Module (Unidrive M700)

Objective

Create an application that demonstrates Homing using the PTi210 module and PowerTools Studio.

Solution Summary

Nearly all motion control applications require a machine/system to be homed prior to beginning the machine sequence. Homing (also referred to as Datum) is a simple routine that is used to establish a known reference position for the machine sequence.

This Application Tool will cover the different options available for the internal homing sequence. Once the basics of homing have been covered, three common home sequence examples will be covered in detail: Home to Sensor, Using a Limit Switch as a Home Sensor, and Home to Sensor between two Limit Switches.

Step 1 – Select Drive Type

Using PowerTools Studio, select File/New and select the drive model you are working with. In this example, I will be using a Unidrive M700 drive.

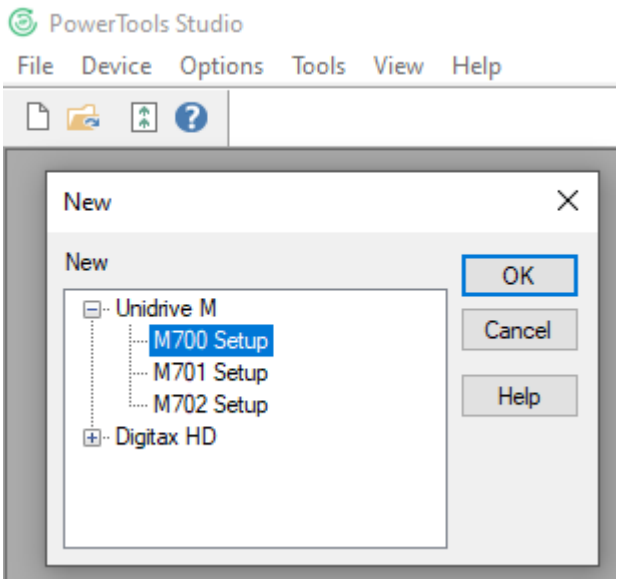

Figure 1

Step 2 – Select Drive/Motor/Feedback

Expand the Hardware branch on the hierarchy tree and select the Drive/Encoder/Motor branch. Doing so will display the view as seen in Figure 2 below. This view is used to select and configure the drive and motor to be used in the application.

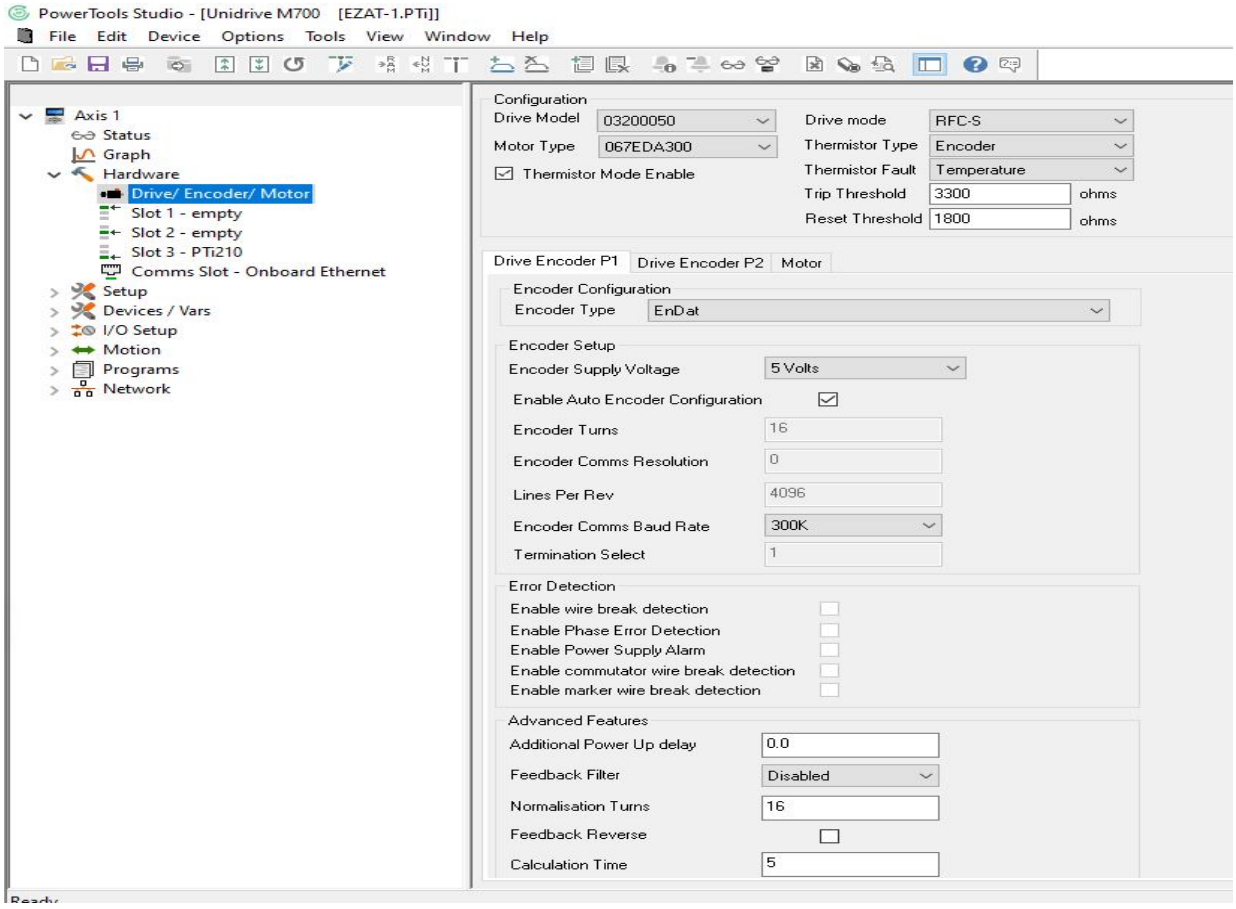

Figure 2

Drive Type - Select the drive model number that matches the drive you are working with from the list box.

Motor Type - Select the servo motor model that matches the motor you are using from the list box.

Drive Mode – Select what type of application the drive will be configured for. Available selections are RFC-S for servo motors and RFC-A for induction motors. Based on the setting of this parameter, information on the Motor tab will change.

Thermistor Type - If the Thermistor mode is enabled, this option allows the thermistor type to be selected.

Thermistor Fault - This option sets the P1 Thermistor Fault Detection parameter (3.123) on the drive.

Trip Threshold - The value of this field is used to set the P1 Thermistor Trip Threshold parameter (3.120) on the drive.

Reset Threshold - The value of this field is used to set the P1 Thermistor Reset Threshold parameter (3.121) on the drive.

Drive Encoder P1 – This tab configures the settings for the encoder if connected to P1 on the drive. The settings are determined by the type of encoder. Refer to your motor/encoder documentation to determine the type of encoder being used.

Drive Encoder P2 – This tab configures the settings for the encoder if connected to P2 on the drive. The settings are determined by the type of encoder. Refer to your motor/encoder documentation to determine the type of encoder being used.

Motor – This tab allows the user to use the motor from the DDF file or manually adjust the motor parameters.

Encoder Type – This parameter defines what type of encoder is being used as the feedback device. This encoder could either be an external stand-alone encoder, or it could be an encoder from an upstream motor. If using a standard stand-alone encoder from Control Techniques, this should be set to Quadrature Incremental. If the feedback device is an upstream motor, the Encoder Type should be set to Quadrature Incremental w/ Commutation Outputs. All encoder types supported by the drive can be found in this listbox.

Encoder Supply Voltage – Select the correct supply voltage for the encoder from this list. Valid selections are 5V, 8V, and 15V. For standard quadrature encoders, 5V should be selected.

Enable Auto Encoder Configuration – This checkbox is available only when using an Absolute encoder with a communications protocol. The encoder can be interrogated on power up for Encoder Turn Bits, Lines Per Rev, and Encoder Comms Resolution automatically.

Encoder Turns – This is an Absolute encoder type term. This is how may revolutions will be recorded before the Revolution Counter rolls over. The parameter is the number of bits of resolution and should be provided by the encoder manufacturer.

Encoder Comms Resolution – This is an Absolute encoder type with a communications protocol term. This parameter defines the maximum resolution of the absolute position of the encoder being transmitted.

Lines Per Rev (pre quadrature) – The user should enter the pre-quadrature resolution of the feedback device here. For an encoder such as a SCSLD-4, enter 3000. For a Control Techniques MG/MH/NT/XV motor enter 2048. For most Control Techniques Unimotors, enter 4096. Check the Unimotor nameplate, since it's possible it may have a 2048 device installed.

Lines Per Rev Divider – This is used to scale the equivalent lines per revolution of incremental and SINCOS encoders, without comms, on rotary motors, and all but comms only encoders on linear motors (servo encoders must have the same number and pitch of poles as the motor) The equivalent line per revolution parameter is divided by the Lines Per Rev Divider. This is most often used when an encoder is used with a linear motor where the number of counts or sine waves per pole is not an integer.

Encoder Comms Baud Rate – This is an Absolute encoder type with a communications protocol term. This is the user-defined baud rate of the Absolute encoder.

Termination Select – P1 Termination Select is used to enable or disable the terminations on the position feedback interface inputs. This function depends on the position feedback device type selected. Refer to the drive manual or Online Help in PowerTools Studio for more information.

Step 3 – Configure the Option Module Slots

The next step is to define what type of Solutions Module is fitted in each of the Unidrive M70x/DigitaxHD option slots. Figure 2 below is an example of a Slot Setup view. For this example an M700 is used and the only module necessary is a PTi210 module. Slots 1 and 2 are empty, and the PTi210 is in Slot 3.

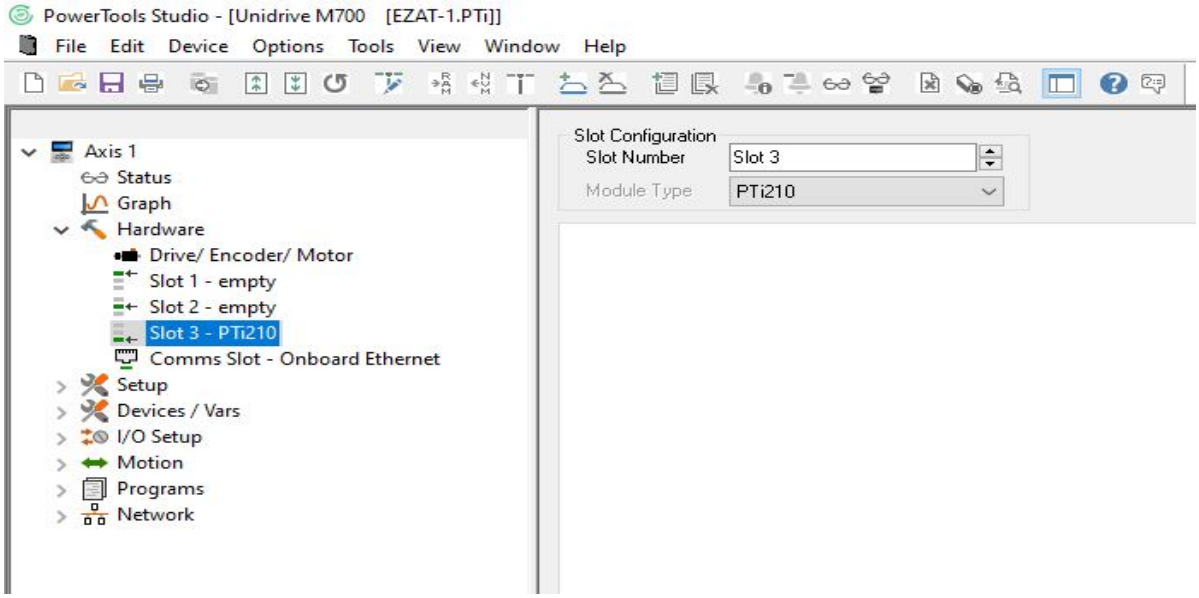

Figure 3

Step 3 – Configure the Assignments

To use the built-in home routine, there are two events that we need to assign to our digital I/O. These events are the Home Initiate, and the Home Sensor. For this example, we will use the I/O on the M700. Figure 4 below shows the Assignments view with the necessary I/O configured.

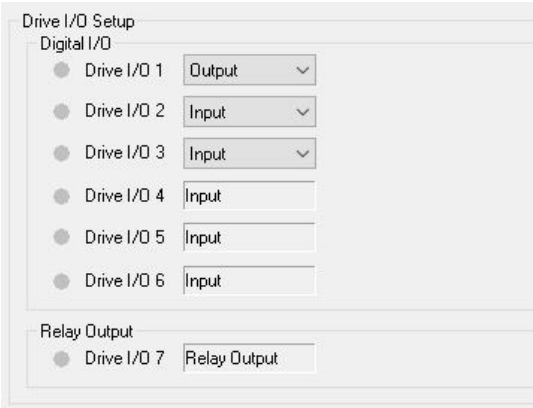

Figure 4

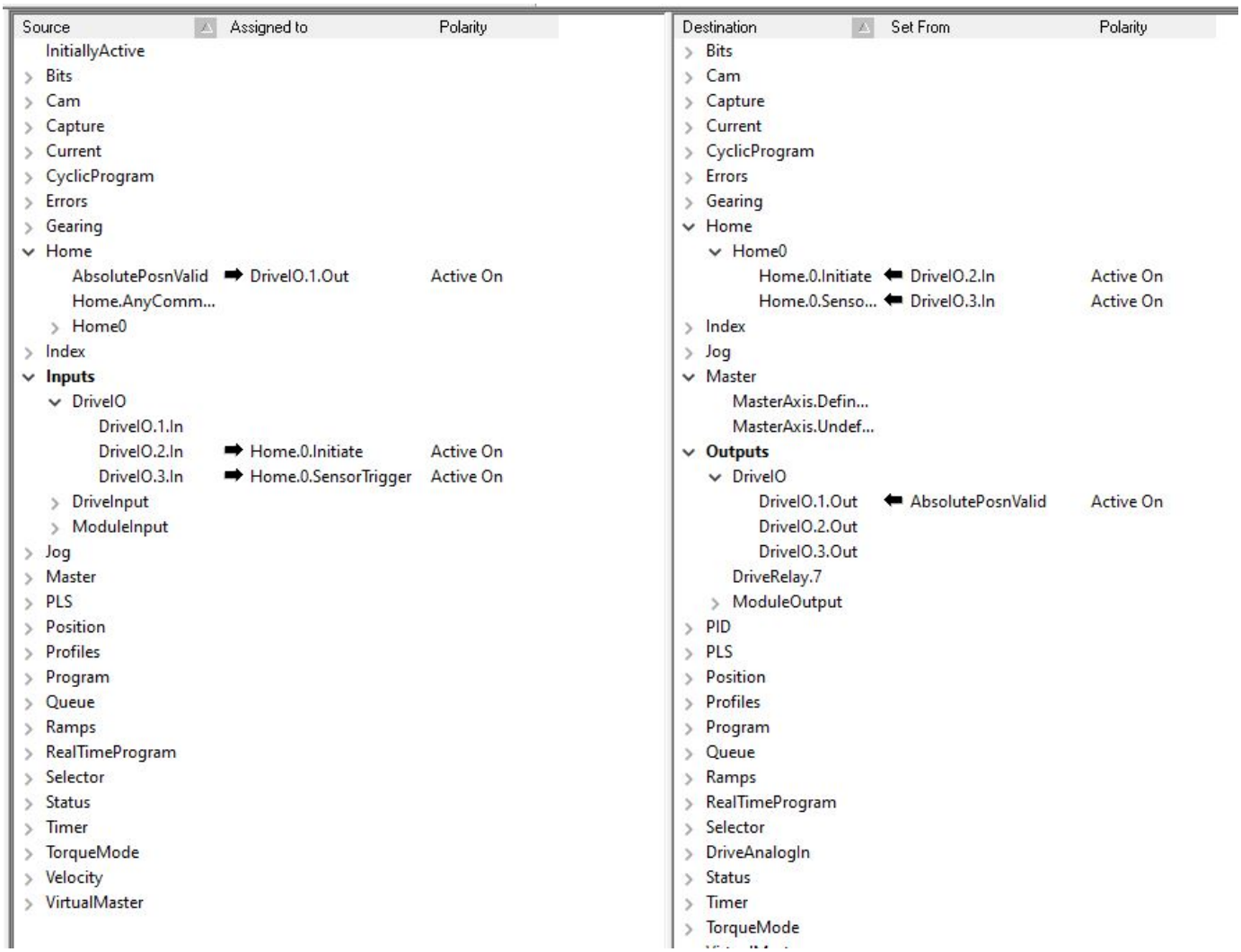

Figure 5

Following is a description of each of the assignments created:

DriveIO.1.IN – Home.0.Initiate

This assignment is used to start the homing sequence. When a rising edge of DriveIO.1.IN is detected, it causes Home.0.Initiate to activate, and therefore the home sequence begins.

DriveIO.2.IN – Home.0.SensorTrigger

When homing to a sensor, the home profile travels at the specified velocity until the Home.0.SensorTrigger event activates. DriveIO.2.IN is used to activate the Home.0.SensorTrigger event. Therefore, the actual hardware sensor should be wired to Input 2 on the M700. When the sensor activates, DriveIO.2.IN will see a rising edge, and will cause Home.0.SensorTrigger to activate, and then the Home profile will begin its Offset routine.

AbsolutePosnValid – DriveIO.1.Out

This assignment is used to activate a digital output indicating that the home sequence is complete, and is entirely optional. Absolute Position Valid is a very important signal for many features. If Absolute Position Valid is not

active, then Software Limit Switches and PLS's will not function. This signal indicates the system has been homed, and there has been no error/trip that has caused the system to lose position. If a following error fault or encoder fault occurs, AbsolutePosnValid will automatically deactivate.

Step 4 – Configure the Home Profile

The Home profile can be configured within PowerTools Studio by navigating to the Home branch underneath the Motion group in the Application Hierarchy. Figure 6 below shows an example of the Home view.

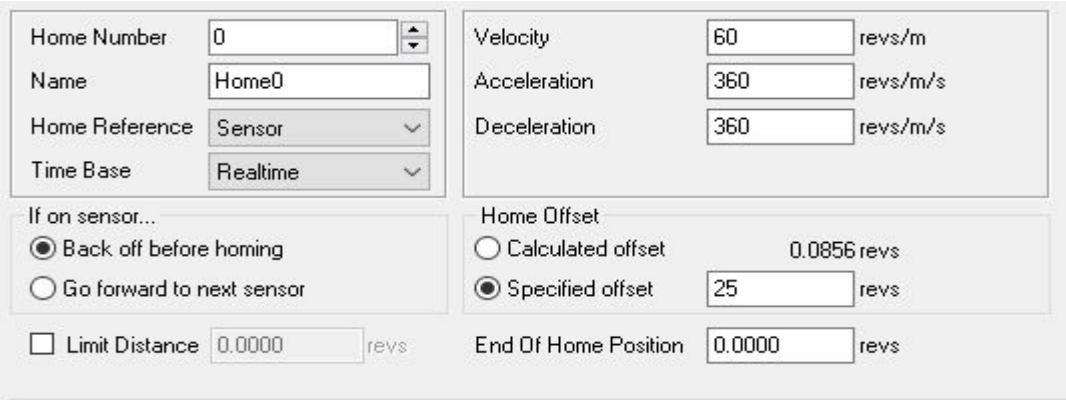

Figure 6

This view is used to configure the motion performance of the home sequence, and how to handle certain sensor conditions. Following is a description of each of the Home parameters:

Home Number – The Home Number (or Home Instance) allows the user to select from multiple Home routines. However, at this point in time, only one home instance is available in the PTi210 module. More Home Instances may be added in the future.

Name – The Home Name parameter allows the user to give a descriptive name to the home sequence. The Name may be up to twelve characters in length. Alpha and Numeric characters may be used, however no spaces are allowed (underscore should be used instead).

Home Reference – The Home Reference is the signal that activates to tell the system that it is "at the home position". The user can select from three different Home Reference types including: Marker, Sensor, and Sensor then Marker.

The term "Marker" refers to the motor encoder marker pulse (or Z channel). The user can use the marker pulse of the motor as the zero point or home position of the system.

The term "Sensor" refers to some external device (i.e. proximity sensor, switch, electric eye) that activates when the machine is at its desired home position.

Following is a graphical interpretation of each of the three Home Reference Types:

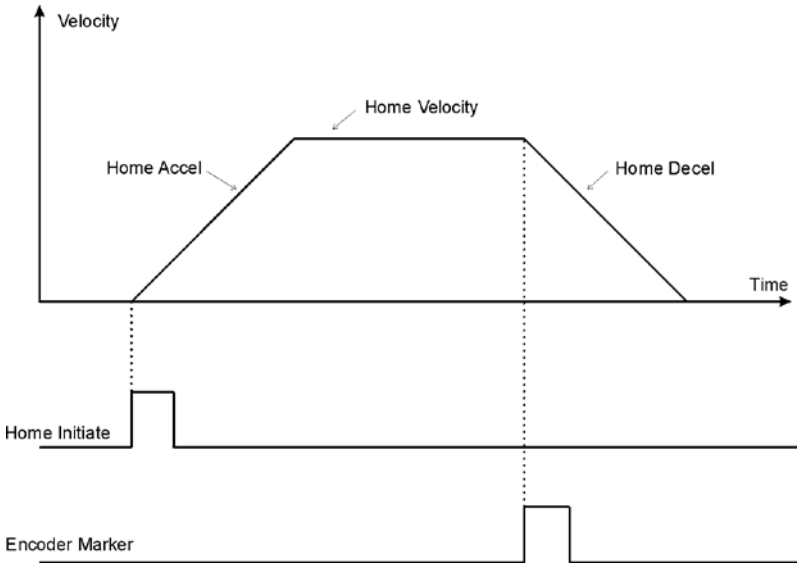

Figure 7 – Home to Marker Profile

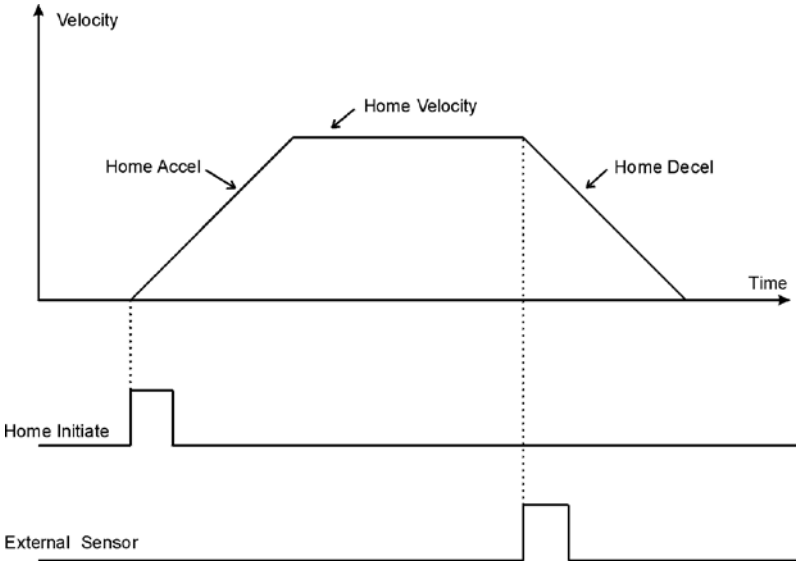

Figure 8 – Home to Sensor Profile

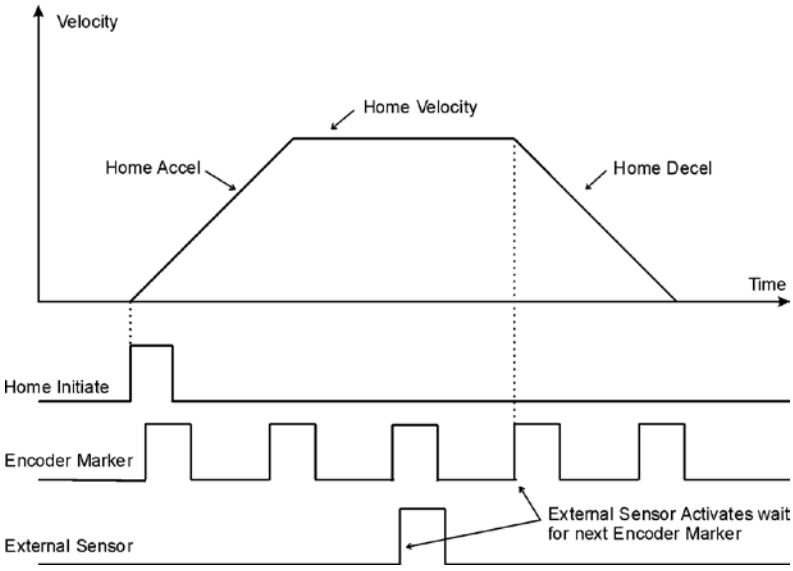

Figure 9 – Home to Sensor then Marker Profile

Time Base – The Time Base parameter allows the user to define the velocity and position reference for the Home motion profile. The user can select from Realtime or Synchronized. Realtime indicates that the internal clock is used as the time reference (*t*) in the velocity and accel/decel units ($v = d / t$ *and* $a = v / t$). If the user selects Synchronized, then the master encoder signal is used as the time reference (*t*) for velocity units ($v = d /$ *MasterDist* and $a = v /$ *MasterDist*).

Velocity – This parameter defines the target speed for the Home profile. The motor may or may not be able to reach the programmed velocity depending on the Accel/Decel ramps specified. The Home Velocity is a signed parameter that determines the direction of travel for the homing routine. If the value is positive, the home will travel in the positive direction. If the value is negative, the home will travel in the negative direction.

Acceleration – This parameter defines the ramp used to reach the target velocity when the home is initiated.

Deceleration – This parameter defines the ramp used to decelerate to zero velocity when the home sequence is complete.

If On Sensor Options
In the condition where the Home sequence is initiated while the Home reference is already active, the following two options define how the sequence will react:

Back Off Before Homing – If the Home Sensor is active when the Home sequence is initiated, and this option is selected, then the motor will move in the opposite direction as specified by the Home Velocity. The motor will continue in the opposite direction until the Home.0.SensorTrigger event deactivates. When the Home Reference deactivates, then the profile will decelerate to a stop, and then begin homing in the direction

originally specified by the Home Velocity. The profile will continue in the specified direction until a rising edge is seen on the Home.0.SensorTrigger event.

Go Forward to Next Sensor - If the Home Sensor is active when the Home sequence is initiated, and this option is selected, then the motor will move in the direction as specified by the Home Velocity. The motor will continue in the specified direction until the next rising edge of the Home.0.SensorTrigger event.

Home Offset Options

The Home Offset is the distance that is traveled after the Home Reference activates. The user has the ability to choose between two different offset types including Calculated Offset and Specified Offset.

Calculated Offset – If the user selects Calculated Offset, then the motor will begin decelerating immediately after the rising edge of the Home Reference. The distance of the offset is automatically calculated based on the Home Velocity and the Home Deceleration parameters. The motor decelerates using the specified deceleration ramp and comes to a stop. When the motor reaches zero velocity, the position feedback is set to the Define Home Position. Figure 9 below shows a Home to Sensor using a Calculated Offset. Compare Figure 9 to Figure 10 to see the difference between Calculated and Specified Offsets.

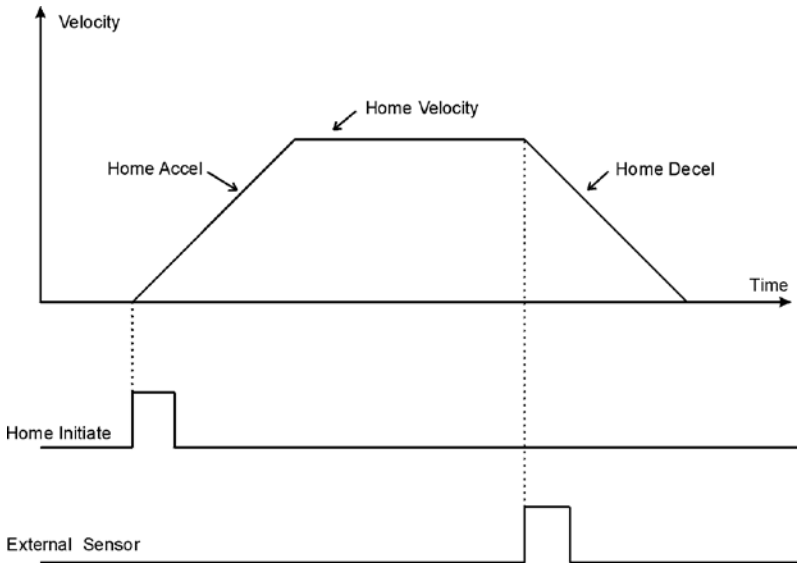

Figure 10 – Home to Sensor using Calculated Offset

Specified Offset - If the user selects Specified Offset, then the motor will continue at the Home Velocity for a specified distance after the rising edge of the Home Reference. The user can enter the desired distance in the Specified Offset value parameter. The Offset parameter is a signed value, so the user can specify a positive or negative offset distance. If the Offset value has a different sign that the Home Velocity, then the motor will decelerate to a stop immediately after the rising edge of the Home Reference, and then the motor will back up until the motor is exactly the Home Offset Value distance from where the Reference activated.

Figure 10 below shows a Home to Sensor using a Calculated Offset. Compare Figure 11 to Figure 10 above to see the difference between Calculated and Specified Offsets.

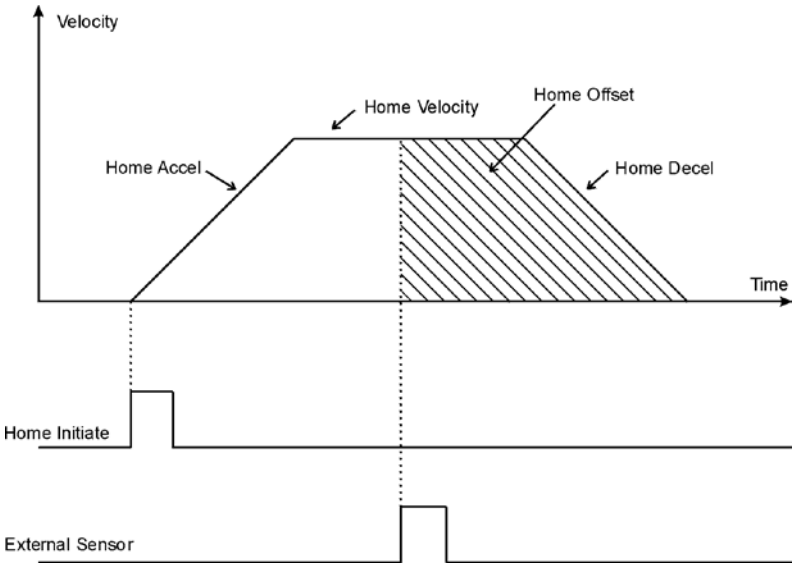

Figure 11 – Home to Sensor using Specified Offset

Limit Distance Enable – If the user wishes to set a maximum distance that the Home profile will travel in search of the Home Reference, simply check this checkbox to Enable the Limit Distance.

Limit Distance – If the Limit Distance Enable is activated, the user must specify the maximum distance that the motor will travel in search of the Home Reference. If the Home is initiated, and the profile reaches this specified distance without seeing a rising edge of the Home Reference signal, then the Home.0.LimitDistHit signal will activate. This signal could be used to indicate to a master controller that the Home Reference was not found properly.

End of Home Position – When the Home sequence is finished (Offset is complete), then the Position Feedback and Position Command parameters are automatically set to the End of Home Position value. Typically, the End of Home Position has a value of 0, but it is possible to specify any non-zero value. This is often used when some long offset is preformed, and the ending position of the home is not the actual "machine zero position".

Now that the details have covered on the Homing profile, let's look at the three examples mentioned earlier:

- Home to Sensor
- Using a Limit Switch as a Home Sensor
- Home to Sensor Between Two Limit Switches

Homing Example #1 – Homing to a Sensor

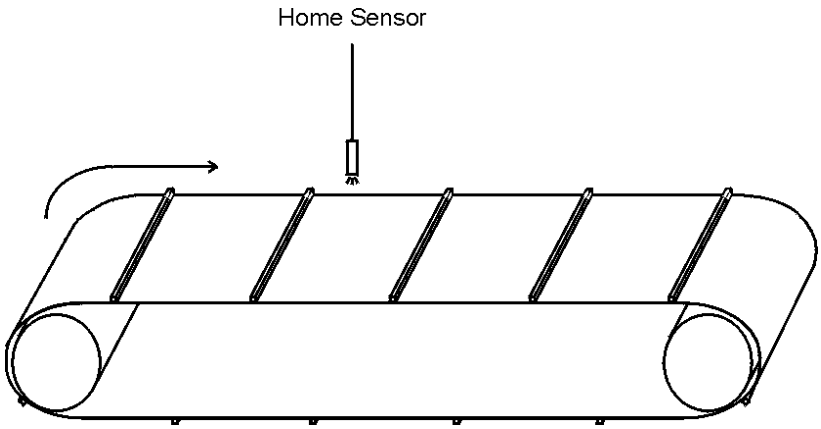

Figure 12 – Home to Sensor Example

On many machines, there is an external sensor that is used to define the Home position or zero cycle position for the machine. The sensor can be used to detect a lug on a belt, gap between products, a printed mark on material, or any other configuration.

In the example shown in Figure 12 above, the motor turns a conveyor belt with evenly spaced flights on the conveyor. Some product needs to be placed in between the flights so that the product can then be loaded into a box. In order receive parts in-between the flights, the belt needs to be in a known position with respect to a flight when the system starts. Therefore, the Home routine is configured to turn the conveyor until the proximity sensor activates indicating that a flight (or lug) is beneath the sensor. The home then decelerates to a stop, and that is defined as the Home Position of the machine.

The user has a push-button on a panel that is used to initiate the Home. That push-button should be wired to a digital input on the Unidrive M700/DigitaxHD, or the PTi210 module. No program is needed to run this Home profile, so on the Assignments view, the input that the push-button is wired to must be Assigned to the Home.0.Initiate destination.

Now, the external sensor must be wired to a digital input on the M700. Then, on the Assignment view in PowerTools Studio, the input that the sensor is wired to must be Assigned to the Home.0.SensorTrigger destination.

An example of the Assignments listed above can be seen in Figure 4.

Next, the user must navigate to the Home view in PowerTools Studio and specify the Home Reference to be “Sensor”. Then enter a Velocity, Acceleration and Deceleration for the home profile.

To initiated the home, the user then must activate the push-button. The motor will then turn the conveyor until a flight passes the sensor, which causes the sensor to activate. When the sensor activates, the

Home.0.SensorTrigger activates because of the Assignment created above. The home then performs the Offset (either Specified or Calculated) and the home is complete.

Homing Example #2 – Using a Limit Switch as a Home Sensor

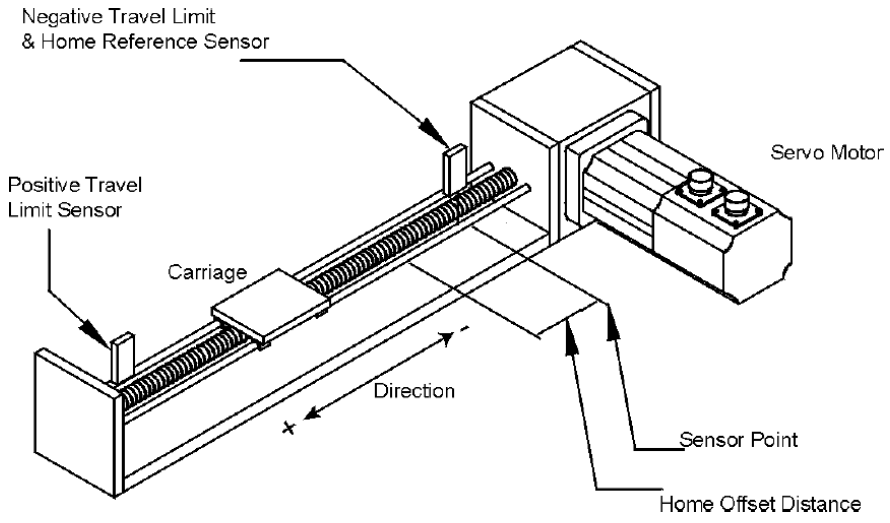

Figure 13 – Using a Limit Switch as a Home Sensor

In this example, a servo motor is used to control the position of a carriage on a leadscrew. The machine has two sensors that act as limit switches for the machine. The Limit Switches are used to protect against the motor running into the hardstops on either end of the leadscrew, which could damage the mechanics. The user wishes to use the Negative Travel Limit Sensor as the Home Sensor as well. In order for this to work, we must disable the functionality of the Travel Limits while the home is in progress. Otherwise, the travel limit would activate and the home sequence would stop. One of the keys to this application is that when the home is complete, it must be clear of the Travel Limit Sensor so that when travel limits are reactivated, a travel limit doesn't turn on immediately after the home.

Assignments

Figure 14 below shows a screen capture of the Assignments required for this example.

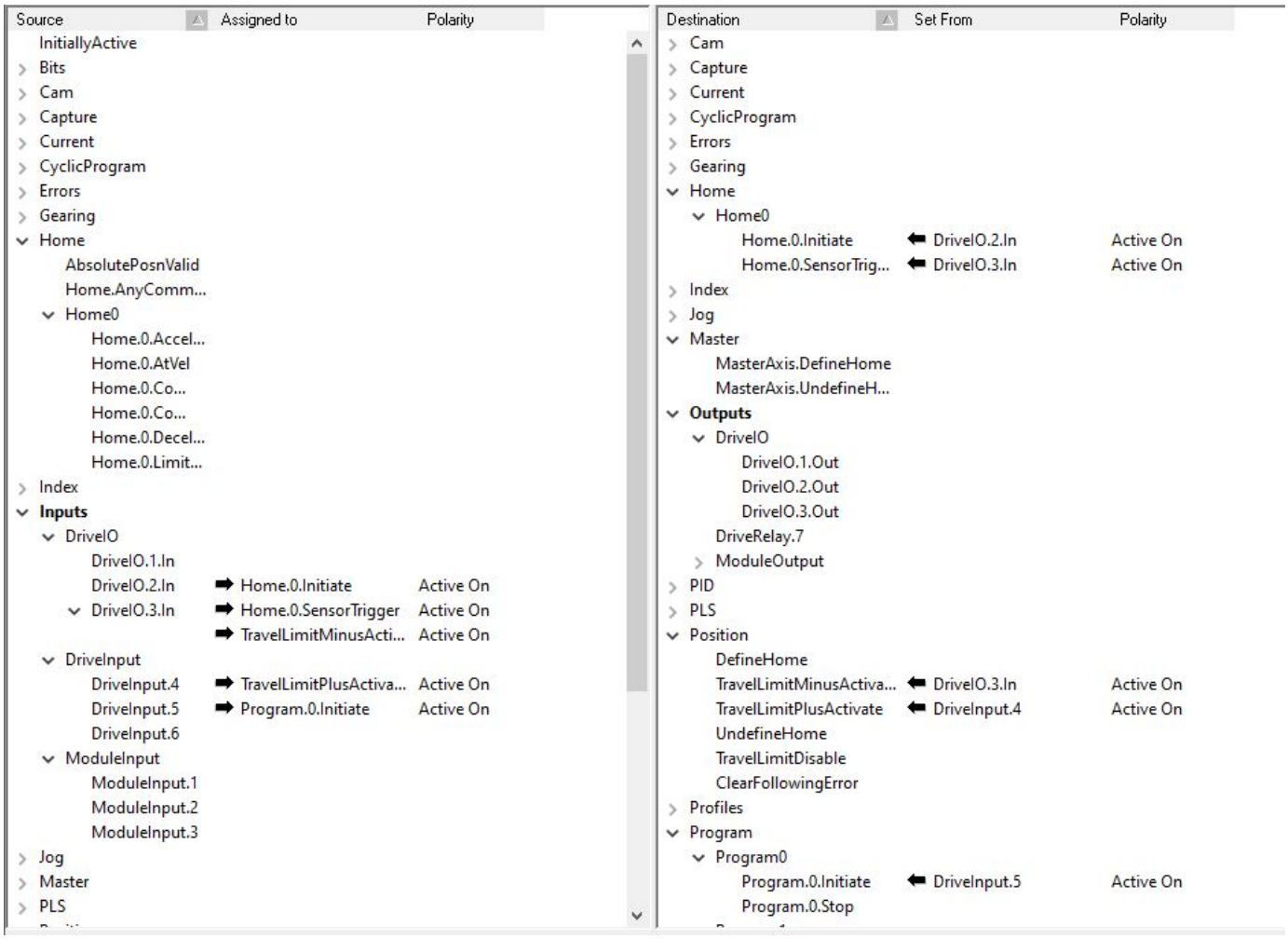

Figure 14 – Assignments for the Using a Limit Switch as a Home Sensor Example

DriveIO.3.In – **Home.0.SensorTrigger**

TravelLimitMinusActivate

This digital input is used for multiple functions. Under normal conditions, the input acts as the Negative Travel Limit sensor input. Therefore, the Negative Limit Sensor should be wired to this digital input. While the system is homing, however, this input will also act as the Home.0.SensorTrigger for the condition where the carriage is on the negative side of the Home Sensor when the home begins. Once the home is complete, we check if this input is active. If so, then we need to back off of the travel limit until we pass the home sensor, and the re-home the machine.

DriveInput.4 – TravelLimitPlusActivate

This input is used to signal that the carriage has reached the Positive Travel Limit position. When this input activates, motion in the positive direction will be prevented.

DriveInput.5 – Program.0.Initiate

Since this example requires the use of a program, we need a way to start the user program. In this example, digital input 4 on the Unidrive M700 is used to initiate the program. This could be a push-button wired to input 4 on the module.

Motion – Home View

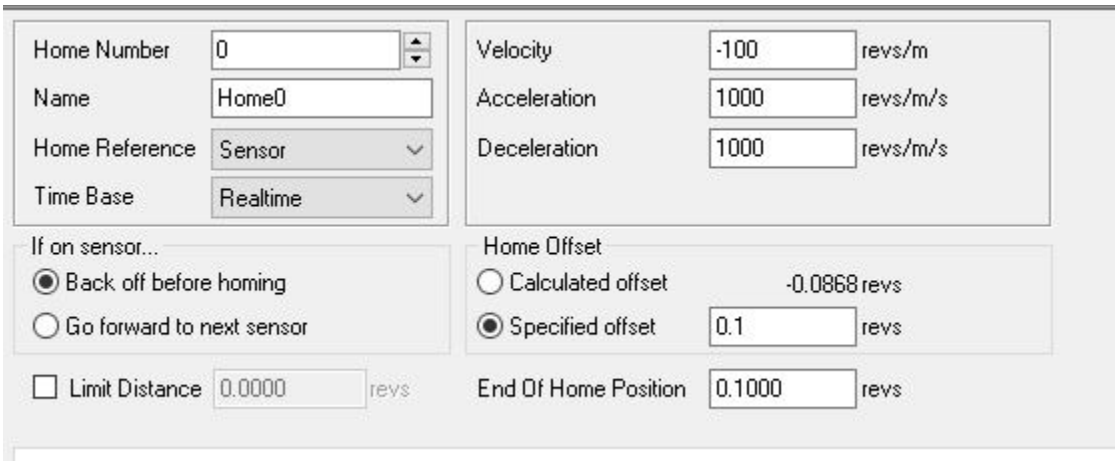

Figure 15 – Parameter setup for Home profile

In order to move off of the travel limit at the end of the home sequence, we select the Offset type of Specified Offset, and enter a value of 0.1 revs. Doing so causes the carriage to move off of the negative travel limit 0.1 revolutions in the positive direction. This is done so that when the travel limits are re-enabled, a travel limit trip does not occur instantly.

Notice also that the Home Velocity is negative meaning that the home will move in the negative direction in search of the Home Reference.

Program0 – Actual Program Code

```
'------------------------------------------------------------------------
'Program 0                                                              -
'Description: Home Sensor Between Two Limit Switches                    -
'Filename(s): EZAT1.PTi                                                  -
'Revision 1 - 3/22/2021, Created using PowerTools Studio 1.2            -
'Min. Software Level Required: PowerTools Studio 1.1 or higher          -
'Min. Firmware Required: V01.01.00.20  -
'------------------------------------------------------------------------
TravelLimitDisable = ON
Home.0.Initiate    'Home0,Sensor,SpecifiedOffset=0.0000 revs,Vel=10 revs/m
Wait For Home.AnyCommandComplete
```

```
TravelLimitDisable = OFF
```

Description of Program Code Used

The first instruction in the program is used to turn off the travel limit functionality during the home routine. This is done because it is possible that the negative travel limit sensor will activate during the home. By turning off the travel limits, if the negative travel limit is reached during the home, the user program will continue to process when the TravelLimitMinusActive function turns on. If travel limits were not disabled, the program would stop when the travel limit activates.

Next, the Home sequence is initiated using the Home.0.Initiate instruction. This is followed by the Wait For Home.AnyCommandComplete instruction, which causes the program to wait on that line until the home sequence is complete.

Once the Home is complete, then the program re-enables the Travel Limits using the TravelLimitDisable = OFF instruction.

Once the program is complete, the home is done, and the carriage will always be in the same position. This program could then be used to call another program, or the rest of the users program could be added to this program at the end.

Homing Example #3 – Homing to a Sensor in-between two Limit Switches

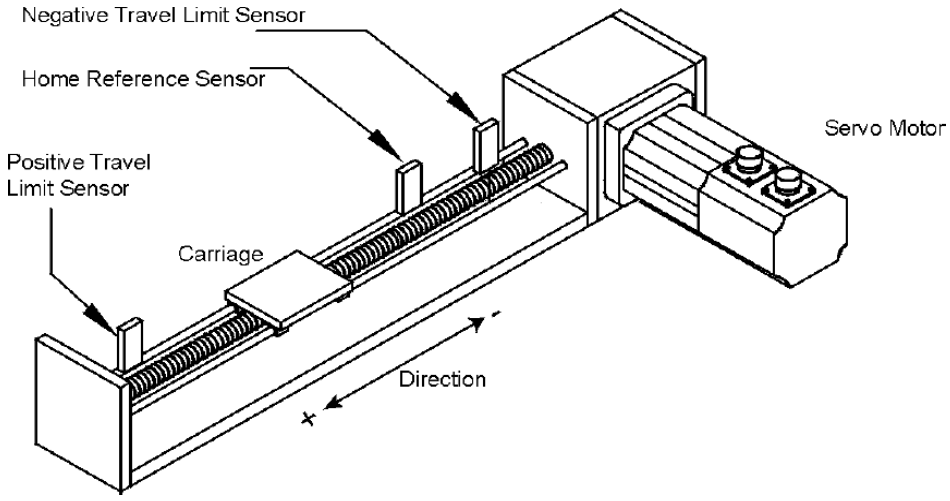

Figure 16 – Home to Sensor Between Two Limit Switches

In this example, a servo motor is used to control the position of a carriage on a leadscrew. The machine has two sensors that act as limit switches for the machine. The Limit Switches are used to protect against the motor running into the hardstops on either end of the leadscrew, which could damage the mechanics. The user wishes to home to a Sensor that is located in-between the two travel limit sensors. This can be a challenging task because it is not possible to know which side of the Home Sensor that the carriage is on when the Home profile begins. Since the Home Sensor is located near the Negative Travel Limit Sensor, it is most likely that the carriage is on the "positive side" of the Home Sensor. Therefore, we will choose to Home in the negative direction by default. We must create a program that handles the condition when the home is initiated when the carriage is on

the “negative side” of the Home Sensor. In this case, the Home will cause the carriage to move into the Negative Travel Limit. The program must move the carriage off of the travel limit, and then past the Home Sensor in the positive direction, and then again Home in the negative direction. This is done because it is desirable to always home in the same direction (otherwise the Home Position would be off by the width of the carriage).

Assignments

Figure 15 below shows a screen capture of the Assignments required for this example.

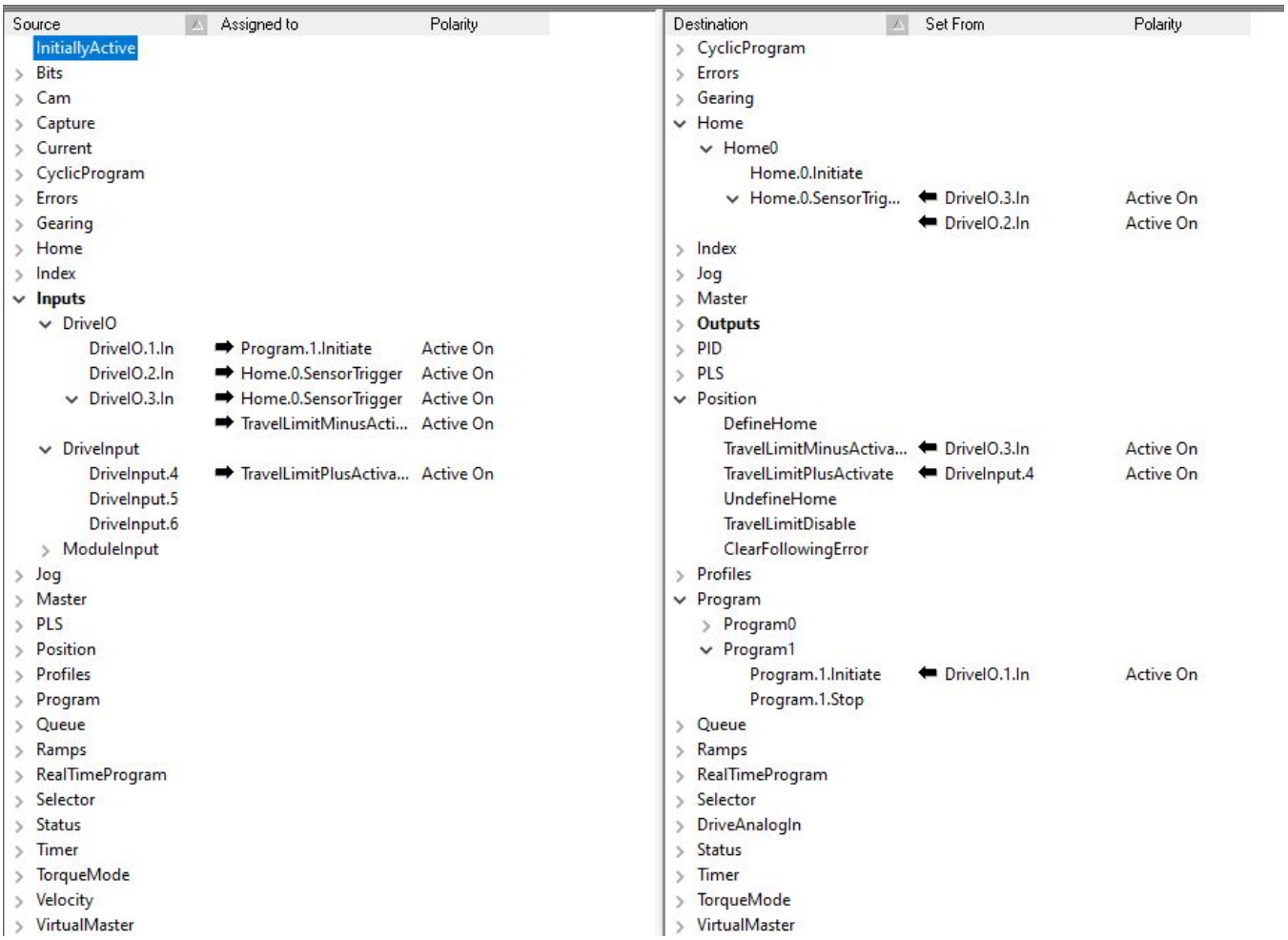

Figure 17 – Assignments for the Home to Sensor Between Two Limit Switches

DriveIO.1.In – Program.1.Initiate

Since this example requires the use of a program, we need a way to start the user program. In this example, digital input 1 on the M700 is used to initiate the program. This could be a push-button wired to input 1 on the module.

DriveIO.2.In – Home.0.SensorTrigger

When homing to a sensor, the home profile travels at the specified velocity until the Home.0.SensorTrigger event activates. DriveIO.2.In is used to activate the Home.0.SensorTrigger event. Therefore, the actual hardware sensor should be wired to Input 2 on the M700. When the sensor activates, DriveIO.2.In will see a rising edge, and will cause Home.0.SensorTrigger to activate, and then the Home profile will begin its Offset routine.

DriveIO.3.In – Home.0.SensorTrigger
TravelLimitMinusActivate

This input is used for multiple functions. Under normal conditions, the input acts as the Negative Travel Limit sensor input. Therefore, the Negative Limit Sensor should be wired to this digital input. While the system is homing however, this input will also act as the Home.0.SensorTrigger for the condition where the carriage is on the negative side of the Home Sensor when the home begins. Once the home is complete, we check if this inputs is active. If so, then we need to back off of the travel limit until we pass the home sensor, and the re-home the machine.

DriveInput.4 – TravelLimitPlusActivate

This input is used to signal that the carriage has reached the Positive Travel Limit position. When this input activates, motion in the positive direction will be prevented.

Program1 – Actual Program Code

```
'--------------------------------------------------------------------
'Program 1                                                          -
'Description: Home Sensor Between Two Limit Switches                -
'Filename(s): EZAT1.PTi                                             -
'Revision 1 - 3/22/2021, Created using PowerTools Studio 1.2        -
'Min. Software Level Required: PowerTools Studio 1.1 or higher      -
'Min. Firmware Required: V01.01.00.20  -
'--------------------------------------------------------------------
TravelLimitDisable = ON

Home.0.Initiate    'Home0,Sensor,SpecifiedOffset=0.0000 revs,Vel=10 revs/m
Wait For Home.AnyCommandComplete


If (TravelLimitMinusActive) Then
    Jog.0.PlusInitiate    'Jog0,Vel=100 revs/m
    Wait For DriveIO.3.In = ON  'This is the input the Home Sensor is wired to
    Wait For DriveIO.3.In = OFF
    Wait For Time 1.0 'Seconds
    Jog.Stop
    Wait For Time 1.0 'Seconds

    Home.0.Initiate      'Home0,Sensor,SpecifiedOffset=0.0000 revs,Vel=10 revs/m
    Wait For Home.AnyCommandComplete
Endif

TravelLimitDisable = OFF
```

The first instruction in the program is used to turn off the travel limit functionality during the home routine. This is done because it is possible that the negative travel limit sensor will activate during the home. By turning off the travel limits, if the negative travel limit is reached during the home, the user program will continue to process when the TravelLimitMinusActive function turns on. If travel limits were not disabled, the program would stop when the travel limit activates.

Next, the Home sequence is initiated using the Home.0.Initiate instruction. This is followed by the Wait For Home.AnyCommandComplete instruction, which causes the program to wait on that line until the home sequence is complete.

The program then checks if the TravelLimitMinusActive event is on indicating that the carriage reached the negative travel limit sensor. In this condition, we wish to back off of the travel limit until the Home Sensor activates and then deactivates (meaning that the carriage has passed to the positive side of the Home Sensor). If the TravelLimitMinusActive is not on, then the program skips to the EndIf and the Travel Limits are again enabled. if the TravelLimitMinusActive is on, then we need to back off of the travel limit.

The program initiates a Jog profile in the positive direction to clear the travel limit sensor.

The next instruction waits until DriveIO.2.In activates meaning that the carriage has reached the home sensor, and then it again waits until DriveIO.2.In is OFF indicating that the carriage has passed the Home Sensor and is now on the positive side of the sensor.

The program then waits 1 second and then stops the Jog profile using the Jog.Stop instruction.

Once the Jog is stopped, the program waits 1 second and then initiates the Home sequence again. This is done because it is desirable to always home onto the Home Sensor from the same direction. If we instead homed in the positive direction after the limit switch was hit, then the Home position would be off by the width of the carriage.

The program then waits until the Home sequence is complete, and then re-enables the Travel Limits using the TravelLimitDisable = OFF instruction.

Once the program is complete, the home is done, and the carriage will always be in the same position. This program could then be used to call another program, or the rest of the users program could be added to this program at the end.

Resources:

M700-702 User Guide & Advanced User Guide

DigitaxHD Control User Guide & Power Installation Guide

Guides, can be found on our website: www.controltechniques.com

You can email to techsupport.cta@mail.nidec.com

You can call Technical Support at 952-995-8000, 24/7/365