

This document contains technology subject to the U.S. Export Administration Regulations (E-AR) and may not be exported or transferred to any unlicensed person, country, or company, by any means, or for end-uses requiring a license, without prior written approval from the U.S. Department of Commerce, Bureau of Industry and Security (BIS). Diversion contrary to U.S. law is prohibited.

APPLICATION NOTE FOR

FULLY INTEGRATED SERVO MOTORS

HOMING PROCEDURES AND METHODS
FOR CLASS 5 AND 6 SMARTMOTOR™
WITH COMBITRONIC™ TECHNOLOGY



Rev. A February 2026

DESCRIBES HOMING PROCEDURES AND
METHODS FOR USE WITH SMARTMOTOR™
SERVOS

www.animatics.com



Copyright Notice

Copyright © 2022 - 2026, Moog Inc.

Moog Animatics SmartMotor™ Homing Procedures and Methods Application Note, Rev. A.
PN:SC80100023-001.

This document, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice and should not be construed as a commitment by Moog Inc., Animatics. Moog Inc., Animatics assumes no responsibility or liability for any errors or inaccuracies that may appear herein.

Except as permitted by such license, no part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Moog Inc., Animatics.

The programs and code samples in this manual are provided for example purposes only. It is the user's responsibility to decide if a particular code sample or program applies to the application being developed and to adjust the values to fit that application.

Moog Animatics and the Moog Animatics logo, SmartMotor and the SmartMotor logo, Combitronic and the Combitronic logo are all trademarks of Moog Inc., Animatics. Other trademarks are the property of their respective owners.

Please let us know if you find any errors or omissions in this manual so that we can improve it for future readers. Such notifications should contain the words "SmartMotor™ Homing Procedures and Methods Application Note" in the subject line and be sent by e-mail to: animatics_sales@moog.com. Thank you in advance for your contribution.

Contact Us:

Moog Animatics
1995 NC Hwy 141
Murphy, NC 28906
USA

Website: www.animatics.com

Email: animatics_sales@moog.com

Table Of Contents

Introduction	6
Purpose	7
Combitronic Technology	7
Safety Information	8
Safety Symbols	8
Other Safety Considerations	8
Motor Sizing	8
Environmental Considerations	8
Machine Safety	9
Documentation and Training	9
Additional Equipment and Considerations	10
Safety Information Resources	10
Additional Documents	11
Related Guides	11
Other Documents	11
Additional Resources	12
Homing Overview	13
Homing Control Variables	13
Example: Homing Method 1	14
Example: Homing Method 3	15
(Optional for Class 6) Homing Operational Checks and Error Reporting Variables	15
(Optional for Class 6) Post Homing Operational Variables	16
Animatics Style vs. CAN Interruptible Homing	17
Homing Methods	18
Methods by Motor Class	18
CANopen Methods	18
Method 1: Homing on negative limit switch and index pulse	18
Method 2: Homing on positive limit switch and index pulse	19
Method 3 and 4: Homing on positive home switch and index pulse	19
Method 5 and 6: Homing on negative home switch and index pulse	19
Method 7 to 14: Homing on home switch and index pulse	20
Method 15 and 16: Reserved	21
Methods 17 to 30: Homing without index pulse	21
Method 17: Home to Negative Limit Switch	21
Method 18: Home to Positive Limit Switch	21

Methods 19 and 20: Home to Positive Home Switch	21
Methods 21 and 22: Home to Negative Home Switch	22
Methods 23 to 26: Homing on Home Switch (Positive Initial Movement)	22
Methods 27 to 30: Homing on Home Switch (Negative Initial Movement)	23
Methods 31 and 32: Reserved	23
Methods 33 and 34: Homing on Index Pulse	23
Method 35: Homing on Current Position	23
Vendor Specific Methods	24
Homing to Hard Stop Methods	24
Method -1: Homing to Hard Stop (Negative) + Index Pulse	24
Method -2: Homing to Hard Stop (Positive) + Index Pulse	24
Method -17: Homing to Hard Stop (Negative)	24
Method -18: Homing to Hard Stop (Positive)	25
Communicating with the SmartMotor	26
Homing Objects	26
Class 6 Homing Objects (Optional Homing Operational Checks and Fault Reporting)	27
Class 6 Homing Objects (Optional Post Homing Operation)	28
CAN Controlword Homing Bits	28
Animatics Homing CAN Controlword Usage	28
Description:	28
CANopen Interruptible Homing CAN Controlword Usage	29
Description:	29
CAN Statusword Homing Bits	29
Examples	30
Configure the Homing Operation, Homing Method 1	30
Start a New Homing Operation	32
Halt the Motor During a Homing Operation	33
Resume the Halted Homing Operation - Animatics Homing	34
Resume the Halted Homing Operation - CAN Interruptible Homing	35
Homing-Specific CANopen Objects	36
EtherNet/IP Homing (Class 6 Only)	37
Example 1: Configure and Start a Homing Operation	37
Example 2: Halt Motor During a Homing Operation	38
Example 3: Resume the Halted Homing Operation	38
Example 4: SM6_Home_Find - Find the Axis Home Position	38
Description	39
Operands	39

Structure	39
Operation	39
Example 5: SM6_Set_Attribute2 - Set Attribute 1, Get Attribute 2	40
Description	40
Operands	40
Structure	40
Operation	41
AniBasic Homing (Class 6 Only)	42
Homing Command Usage	43
Start a Homing Operation - G command	44
Halt a Homing Operation - X command (Animatics Style)	45
Halt a Homing Operation - X command (CAN Interruptible Style)	46
Halt a Homing Operation - S command (Animatics Style)	47
Halt a Homing Operation - S command (CAN Interruptible Style)	48
Example: AniBasic (Class 6 Only)	49
AniBasic Parameters (Class 6 Only)	51
Homing Fault Codes (Class 6 Only)	52
Using Mixed Mode Operations After Homing	53
Homing to a Hard Stop (Class 6 Only)	54
Homing to Hard Stop Control Variables	54
Homing to Hard Stop - Communicating with the Class 6 SmartMotor	55
Homing to Hard Stop - AniBasic Homing	55
AniBasic Example (Homing Method -1)	58
Revision History	59

Introduction

This chapter provides information on the purpose of this document. It also provides information on safety, and where to find related documents and additional resources.

Purpose	7
Combitronic Technology	7
Safety Information	8
Safety Symbols	8
Other Safety Considerations	8
Motor Sizing	8
Environmental Considerations	8
Machine Safety	9
Documentation and Training	9
Additional Equipment and Considerations	10
Safety Information Resources	10
Additional Documents	11
Related Guides	11
Other Documents	11
Additional Resources	12

Purpose

This document describes the Homing procedures and methods available for the Moog Animatics SmartMotor™ servos. The purpose is to explain the basic concepts that system builders and programmers should keep in mind when integrating a SmartMotor into the design. Because system design is quite complex, this document focuses on only those items that directly affect the integration and performance of the SmartMotor.

This document applies to Class 5 and Class 6 SmartMotor servos. Note that Class 6 motors require these firmware versions: for Class 6 M-style, 6.0.2.57 or later; for Class 6 D-style, 6.4.2.57 or later.

Combitronic Technology

The most unique feature of the SmartMotor is its ability to communicate with other SmartMotors and share resources using Moog Animatics' Combitronic™ technology. Combitronic is a protocol that operates over a standard CAN interface. It may coexist with CANopen and other protocols. It requires no single dedicated controller¹ to operate. Each SmartMotor connected to the same network communicates on an equal footing, sharing all information, and therefore, sharing all processing resources.

For additional details, see the [SmartMotor™ Developer's Guide](#).

¹Moog Animatics has replaced the terms "master" and "slave" with "controller" and "follower", respectively.

Safety Information

This section describes the safety symbols and other safety information.

Safety Symbols

The manual may use one or more of these safety symbols:



WARNING: This symbol indicates a potentially nonlethal mechanical hazard, where failure to comply with the instructions could result in serious injury to the operator or major damage to the equipment.



CAUTION: This symbol indicates a potentially minor hazard, where failure to comply with the instructions could result in slight injury to the operator or minor damage to the equipment.

NOTE: Notes are used to emphasize non-safety concepts or related information.

Other Safety Considerations

The Moog Animatics SmartMotors are supplied as components that are intended for use in an automated machine or system. As such, it is beyond the scope of this manual to attempt to cover all the safety standards and considerations that are part of the overall machine/system design and manufacturing safety. Therefore, this information is intended to be used only as a general guideline for the machine/system designer.

It is the responsibility of the machine/system designer to perform a thorough "Risk Assessment" and to ensure that the machine/system and its safeguards comply with the safety standards specified by the governing authority (for example, ISO, OSHA, UL, etc.) for the site where the machine is being installed and operated. For more details, see Machine Safety on page 9.

Motor Sizing

It is the responsibility of the machine/system designer to select SmartMotors that are properly sized for the specific application. Undersized motors may: perform poorly, cause excessive downtime or cause unsafe operating conditions by not being able to handle the loads placed on them. The *System Best Practices* document, which is available on the Moog Animatics website, contains information and equations that can be used for selecting the appropriate motor for the application.

Replacement motors must have the same specifications and firmware version used in the approved and validated system. Specification changes or firmware upgrades require the approval of the system designer and may require another Risk Assessment.

Environmental Considerations

It is the responsibility of the machine/system designer to evaluate the intended operating environment for dust, high-humidity or presence of water (for example, a food-processing environment that requires water or steam wash down of equipment), corrosives or chemicals that may come in contact with the machine, etc. Moog Animatics manufactures specialized IP-rated motors for operating in extreme conditions. For details, see the *Moog Animatics Product Catalog*.

Machine Safety

In order to protect personnel from any safety hazards in the machine or system, the machine/system builder must perform a "Risk Assessment", which is often based on the ISO 13849 standard. The design/implementation of barriers, emergency stop (E-stop) mechanisms and other safeguards will be driven by the Risk Assessment and the safety standards specified by the governing authority (for example, ISO, OSHA, UL, etc.) for the site where the machine is being installed and operated. The methodology and details of such an assessment are beyond the scope of this manual. However, there are various sources of Risk Assessment information available in print and on the internet.

NOTE: The next list is an example of items that would be evaluated when performing the Risk Assessment. Additional items may be required. The safeguards must ensure the safety of all personnel who may come in contact with or be in the vicinity of the machine.

In general, the machine/system safeguards must:

- Provide a barrier to prevent unauthorized entry or access to the machine or system. The barrier must be designed so that personnel cannot reach into any identified danger zones.
- Position the control panel so that it is outside the barrier area but located for an unrestricted view of the moving mechanism. The control panel must include an E-stop mechanism. Buttons that start the machine must be protected from accidental activation.
- Provide E-stop mechanisms located at the control panel and at other points around the perimeter of the barrier that will stop all machine movement when tripped.
- Provide appropriate sensors and interlocks on gates or other points of entry into the protected zone that will stop all machine movement when tripped.
- Ensure that if a portable control/programming device is supplied (for example, a hand-held operator/programmer pendant), the device is equipped with an E-stop mechanism.

NOTE: A portable operation/programming device requires *many* additional system design considerations and safeguards beyond those listed in this section. For details, see the safety standards specified by the governing authority (for example, ISO, OSHA, UL, etc.) for the site where the machine is being installed and operated.

- Prevent contact with moving mechanisms (for example, arms, gears, belts, pulleys, tooling, etc.).
- Prevent contact with a part that is thrown from the machine tooling or other part-handling equipment.
- Prevent contact with any electrical, hydraulic, pneumatic, thermal, chemical or other hazards that may be present at the machine.
- Prevent unauthorized access to wiring and power-supply cabinets, electrical boxes, etc.
- Provide a proper control system, program logic and error checking to ensure the safety of all personnel and equipment (for example, to prevent a run-away condition). The control system must be designed so that it does not automatically restart the machine/system after a power failure.
- Prevent unauthorized access or changes to the control system or software.

Documentation and Training

It is the responsibility of the machine/system designer to provide documentation on safety, operation, maintenance and programming, along with training for all machine operators, maintenance technicians, programmers, and other personnel who may have access to the machine. This documentation must include proper lockout/tagout procedures for maintenance and programming operations.

It is the responsibility of the operating company to ensure that:

- All operators, maintenance technicians, programmers and other personnel are tested and qualified before acquiring access to the machine or system.
- The above personnel perform their assigned functions in a responsible and safe manner to comply with the procedures in the supplied documentation and the company safety practices.
- The equipment is maintained as described in the documentation and training supplied by the machine/system designer.

Additional Equipment and Considerations

The Risk Assessment and the operating company's standard safety policies will dictate the need for additional equipment. In general, it is the responsibility of the operating company to ensure that:

- Unauthorized access to the machine is prevented at all times.
- The personnel are supplied with the proper equipment for the environment and their job functions, which may include: safety glasses, hearing protection, safety footwear, smocks or aprons, gloves, hard hats and other protective gear.
- The work area is equipped with proper safety equipment such as first aid equipment, fire suppression equipment, emergency eye wash and full-body wash stations, etc.
- There are no modifications made to the machine or system without proper engineering evaluation for design, safety, reliability, etc., and a Risk Assessment.

Safety Information Resources

Additional SmartMotor safety information can be found on the Moog Animatics website; open the topic "Controls - Notes and Cautions" located at:

<https://www.animatics.com/support/downloads/knowledgebase/controls---notes-and-cautions.html>

OSHA standards information can be found at:

<https://www.osha.gov/law-regs.html>

ANSI-RIA robotic safety information can be found at:

<http://www.robotics.org/robotic-content.cfm/Robotics/Safety-Compliance/id/23>

UL standards information can be found at:

<http://ulstandards.ul.com/standards-catalog/>

ISO standards information can be found at:

<http://www.iso.org/iso/home/standards.htm>

EU standards information can be found at:

http://ec.europa.eu/growth/single-market/european-standards/harmonised-standards/index_en.htm

Additional Documents

The Moog Animatics website contains additional documents that are related to the information in this manual. Please refer to these lists.

Related Guides

- Moog Animatics SmartMotor™ Installation & Startup Guides
<http://www.animatics.com/install-guides>
- *SmartMotor™ Developer's Guide*
<http://www.animatics.com/smartmotor-developers-guide>
- *SmartMotor™ System Best Practices Application Note*
<http://www.animatics.com/system-best-practices-application-note>

In addition to the documents listed above, guides for fieldbus protocols and more can be found on the website: <https://www.animatics.com/support/downloads.manuals.html>

Other Documents

- SmartMotor™ Certifications
<https://www.animatics.com/certifications.html>
- *SmartMotor Developer's Worksheet*
(interactive tools to assist developer: Scale Factor Calculator, Status Words, CAN Port Status, Serial Port Status, RMODE Decoder and Syntax Error Codes)
<https://www.animatics.com/support/downloads.knowledgebase.html>
- *Moog Animatics Product Catalog*
<http://www.animatics.com/support/moog-animatics-catalog.html>

Additional Resources

The Moog Animatics website contains useful resources such as product information, documentation, product support and more. Please refer to these addresses:

- General company information:
<http://www.animatics.com>
- Product information:
<http://www.animatics.com/products.html>
- Product support (Downloads, How-to Videos, Forums and more):
<http://www.animatics.com/support.html>
- Contact information, distributor locator tool, inquiries:
<https://www.animatics.com/contact-us.html>
- Applications (Application Notes and Case Studies):
<http://www.animatics.com/applications.html>

Homing Overview

The Moog Animatics SmartMotor has an integrated Homing feature based on the CiA 402 CANopen specification. Homing is the process of finding the zero position of an application relative to a home switch or either mechanical hard stop (i.e., physical end of travel) on an axis. The home switch can be one of the limit switches or another switch placed within the range of travel.

"Homing to a Hard Stop", which requires no mechanical switch, is also available. For details, see Homing to a Hard Stop (Class 6 Only) on page 54.

Homing Control Variables

The homing operation is controlled by the following variables:

Homing Method - the homing method determines the following behavior:

- The type of homing signal (positive limit switch, negative limit switch, home switch, mechanical hard stop)

NOTE: The electrical polarity of the limit switches is not configurable (refer to the relevant SmartMotor Installation and Startup Guide for more details)

- The initial direction of motion
- If the home position is relative to an encoder index mark or the edge of the home switch
- The style of home switch to use

Homing Speeds - there are two homing speeds (values are given in motor units):

- The speed used to find the home switch, usually the faster of the two.
- The speed used to find the home position, either an index pulse or edge of the home switch. It is usually the slower of the two speeds and provides the most accuracy and repeatability.

Homing Acceleration - indicates the acceleration and deceleration values to be used during the homing operation. The values are given in motor units.

Home Offset - the difference between the home position found during the homing operation and the zero position for the application.

Halt Option* - determines the action of the motor when the CAN Controlword Halt bit is set.

- 1 - Decelerate on the profile deceleration
- 2 - Slow down on Quick Stop Deceleration

*Used with the [CAN Interruptible Homing](#) style

Quick Stop Deceleration* - the deceleration value used to stop the motor if the Halt Option is set to 2 and the CAN Controlword Halt bit is set to 1.

*Used with the [CAN Interruptible Homing](#) style

Once these values have been set, the homing operation is controlled and monitored using bits in the CAN Controlword and CAN Statusword.

The following two examples illustrate possible homing scenarios.

Example: Homing Method 1

The following figure illustrates Homing Method 1, homing on the first index mark to the right of the negative limit switch. A positive Home Offset value sets the application zero position to the right of the home position; a negative Home Offset value sets it to the left.

Homing Method 1: Home on the Negative Limit switch, End on an Encoder Index pulse

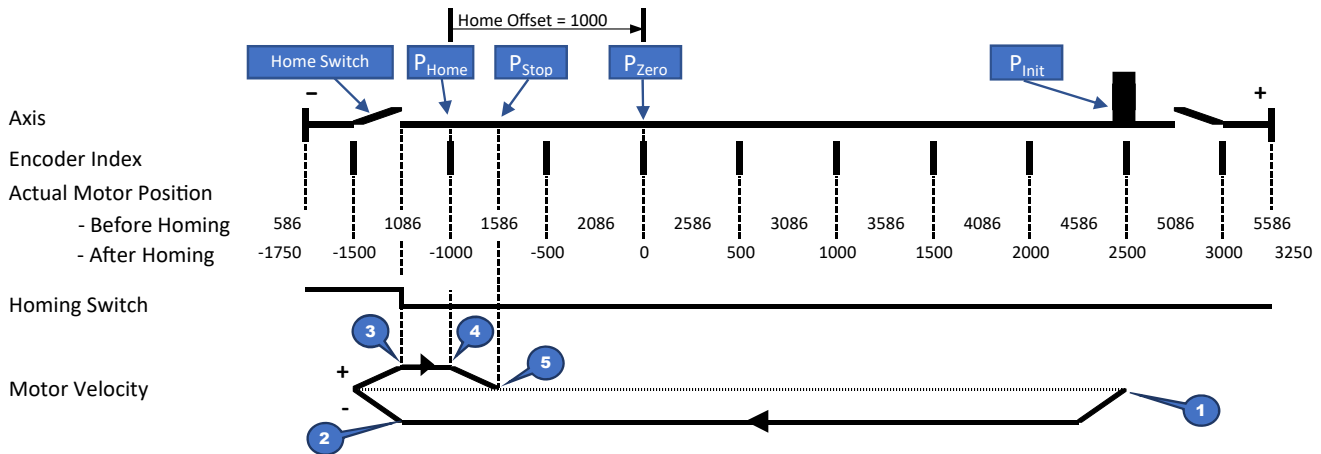
P_{Init} = The position of the actuator before the homing operation starts

P_{Home} = The machine home position found during the homing operation

P_{Stop} = The position of the actuator after the homing operation completes.

P_{Zero} = The zero position for the application, $P_{Zero} = P_{Home} + HomeOffset$

Home Offset = The difference between the machine home position, found during homing, and the zero position for the application



Homing Example - Homing Method 1

Homing Steps:

1. The homing operation starts, and the motor moves towards the negative limit switch.
2. The negative limit switch is detected, stopping the motor.
3. The motor moves in the positive direction until the limit switch releases and then starts looking for the next encoder index mark.
4. The position of the index mark is captured, the motor ramps to a stop and the application zero position is calculated. The actual and calculated motor positions are updated so that P_{Zero} location reports a value of zero (0).
5. The motor comes to rest, which completes the homing operation.

Example: Homing Method 3

The following figure illustrates Homing Method 3, homing on the first index mark to the left of the positive home switch. The Home Offset value is negative, moving the application zero position to the left of the home position.

Homing Method 3: Homing On Low Side of a Positive Home Switch and Index Pulse

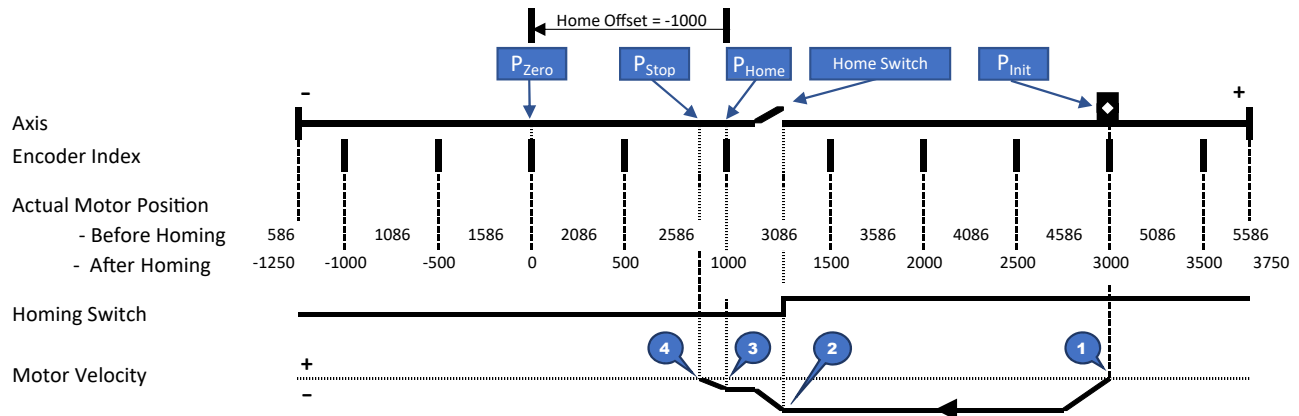
P_{Init} = The position of the actuator before the homing operation starts

P_{Home} = The machine home position found during the homing operation

P_{Stop} = The position of the actuator after the homing operation completes.

P_{Zero} = The zero position for the application, $P_{Zero} = P_{Home} + HomeOffset$

HomeOffset = The difference between the machine home position, found during homing, and the zero position for the application



Homing Example - Homing Method 3

Homing Steps:

1. The homing operation starts, and the motor moves towards the negative limit switch.
2. The home switch goes low, the motor slows and starts looking for the next encoder index mark.
3. The position of the index mark is captured, the motor ramps to a stop and the application zero position is calculated.
4. The motor comes to rest, which completes the homing operation.

(Optional for Class 6) Homing Operational Checks and Error Reporting Variables

Operational checks for all the homing methods can include a time check and a distance check. These Operational Checks include the following variables:

Homing Time Limit - an optional parameter that allows the operator to specify the maximum time (in seconds) for the homing method to complete. The operation stops if it does not complete within this time limit. The valid range of values are 0 to 65535.

- CAN Interruptible Homing style does allow a homing operation to Halt and Resume. If the Homing time limit checking is enabled and the CAN Interruptible Halt is commanded, the Homing time limit check is also halted.

Homing Distance Limit - an optional parameter that allows the operator to specify the maximum distance (in encoder counts) for the homing method to complete. The operation stops if it does not complete within this distance limit. If this check is enabled, the zero position is where homing was commanded and the maximum distance is the plus or minus encoder counts from this starting position. The valid range of values are 0 to 2147483647.

Homing Fault Code - in instances where the homing operation failed, this fault code can be queried by the user to assist in troubleshooting the cause of the failure.

(Optional for Class 6) Post Homing Operational Variables

The following apply after the homing operation has completed:

Home Offset (definition) - the difference between the homing operation's "home" (found) position and the application's zero position.

Post Homing Operation - with the above definition in mind, this optional parameter allows the actuator to either stay at the found home position OR move to the zero position for the application.

NOTE: The post homing move uses the default values associated with the SmartMotor. For example, if the post homing value is set to move to the zero position but the VT variable is set to zero, the post homing move to the zero position will not occur.

Animatics Style vs. CAN Interruptible Homing

The **Class 6** SmartMotor supports two styles of homing:

- Moog Animatics ("Animatics Style") Homing - controls the motor in a manner similar to the Moog Animatics command interface but uses the CAN Controlword homing bits in a non-standard manner. This homing style does not allow for interrupting or resuming a homing operation.
- CAN Interruptible Homing - controls the motor using the homing bits of the CAN Controlword according to the CANopen specification. This homing style allows the homing operation to be interrupted and resumed.

The difference between the two styles is in how the operation is interrupted or stopped.

The Class 5 SmartMotor supports the Animatics Style Homing logic and a limited number of homing methods. For more details, see Homing Methods on page 18.

When the motor is configured for Animatics Style Homing:

- The homing operation can be halted using the default motor ramp, thus aborting the homing operation, similar to issuing an X-Stop command from the Terminal View in the SMI software.
- The homing operation can be halted using the Quick Stop deceleration, thus aborting the homing operation, similar to issuing an S-stop command from the Terminal View in the SMI software.
- The homing operation cannot be interrupted and then resumed. If it is interrupted, that homing operation is aborted and a new one must be started.

When the motor is configured for CAN Interruptible Homing, the homing operation can be:

- Halted using the motor ramp selected by the Halt option. The homing operation can be resumed or aborted depending on the Homing CAN Controlword bit settings.
- Interrupted and resumed. When the homing operation is interrupted, the motor will continue moving until it halts at the end of the current segment. When resumed, if the homing operation was not completed in that segment, the motor will continue from that point.
- The interruptible mode was designed to use the Homing CAN Controlword bits to initiate, halt, and potentially resume operation. The G, X, and S commands can be used to start and stop operation, but only consider doing this as a backup / emergency stop scenario. For example, an S command can immediately stop a homing operation, but an X command *is required* to clear the emergency stop operation.

For more details, see CAN Controlword Homing Bits on page 28.

The motor defaults to Animatics Style Homing on power up. CAN Interruptible Homing can be selected issuing one of the following commands: (the commands can be placed at the beginning of a user program or entered from the SMI software Terminal View)

- CANCTL(61,0), enables Animatics Style Homing
- CANCTL(61,1), enables CAN Interruptible Homing

Homing Methods

This section describes the available homing methods for the Class 5 and Class 6 SmartMotor servos. There is also a set of methods available on Class 6 motors for homing to a hard stop. For details, see Homing to a Hard Stop (Class 6 Only) on page 54.

Methods by Motor Class

Class 5 SmartMotors support Homing Methods:

- 1 and 2: Home on a limit switch, the home position is relative to an index pulse
- 17 and 18: Home on a limit switch, the home position is relative to the edge of the home switch
- 33 and 34: Home on the nearest index pulse
- 35: Home on the current position

Class 6 SmartMotors support Homing Methods:

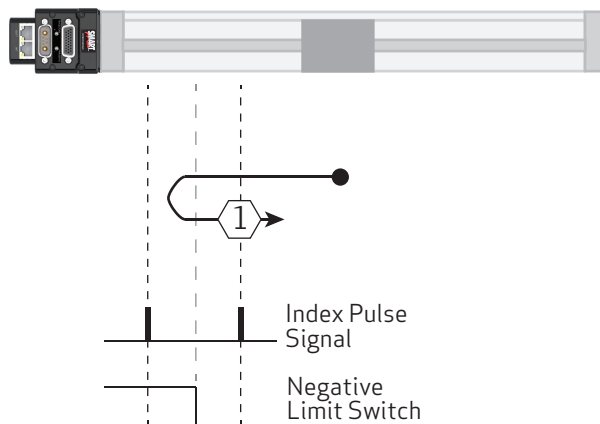
- -1, -2, -17, -18: The home position is relative to the mechanical hard stop
- 1 to 14: The home position is relative to an index pulse
- 17 to 30: The home position is relative to the edge of the home switch
- 33 and 34: Home on the nearest index pulse
- 35: Home on the current position

CANopen Methods

NOTE: Methods 1 and 2 are supported on both Class 5 and Class 6 motors; methods 3 to 14 are supported on Class 6 motors only.

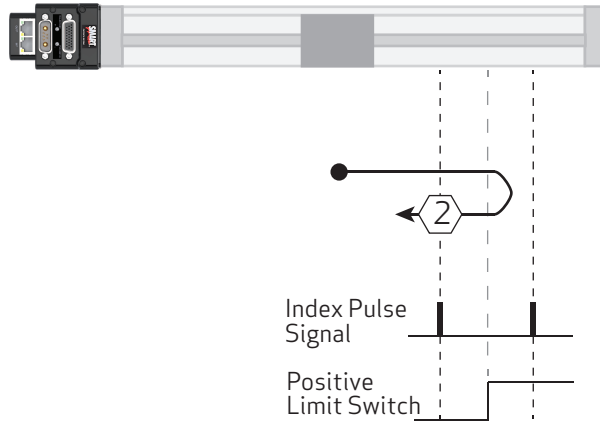
Method 1: Homing on negative limit switch and index pulse

The initial direction of movement shall be to the left if the negative limit switch is inactive. The home position shall be at the first index pulse to the right of the position where the negative limit switch becomes inactive.



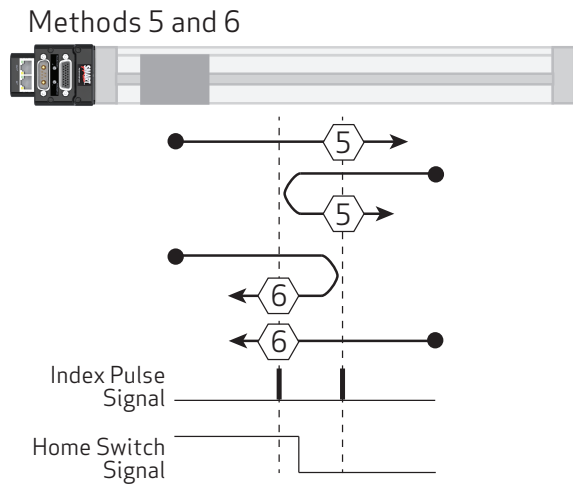
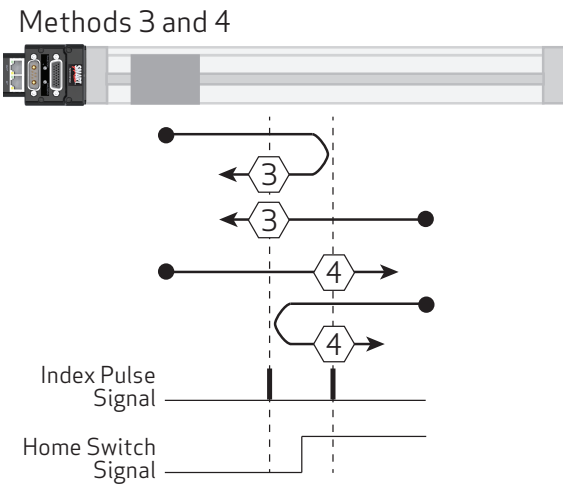
Method 2: Homing on positive limit switch and index pulse

The initial direction of movement shall be rightward if the positive limit switch is inactive. The position of home shall be at the first index pulse to the left of the position where the positive limit switch becomes inactive.



Method 3 and 4: Homing on positive home switch and index pulse Method 5 and 6: Homing on negative home switch and index pulse

The initial direction of movement depends on the state of the home switch. The home position will be at the index pulse to the left or right of the point where the home switch changes state. If the initial position is situated so that the direction of movement reverses during homing, the reversal may take place anywhere after the home switch changes state.



Method 7 to 14: Homing on home switch and index pulse

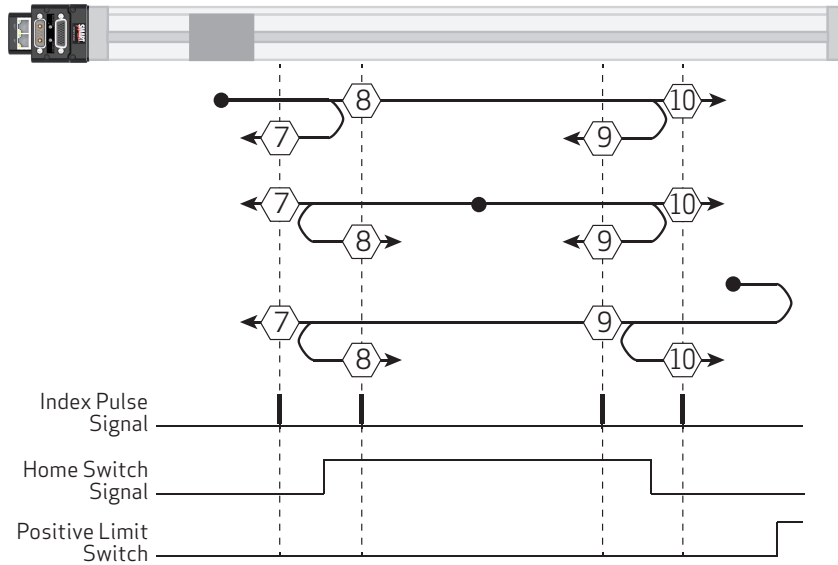
These methods use a home switch in mid-travel. The switch has a "momentary" action as the axis's position sweeps past the switch. The initial direction of movement depends on the state of the home switch.

If the home switch is not active at the start of homing:

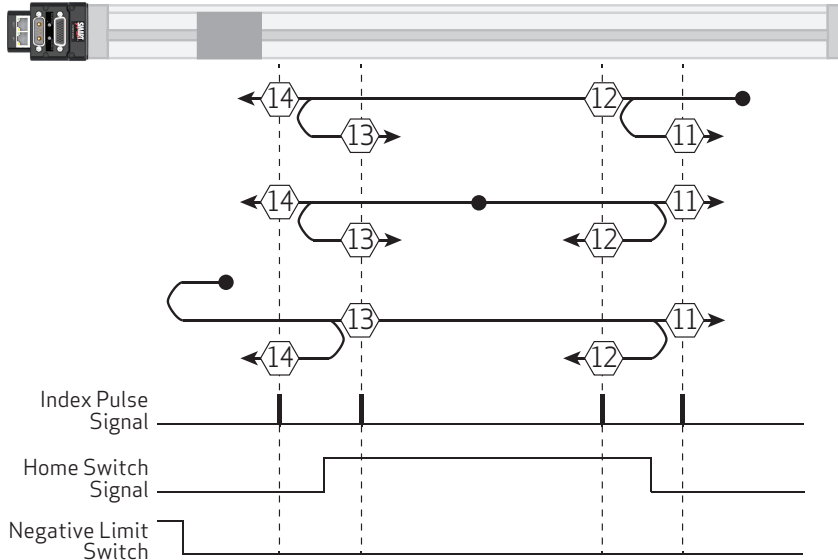
- Methods 7 to 10, the initial direction of movement shall be to the right
- Methods 11 to 14, the initial direction of movement shall be to the left

If the home switch is active at the start of the motion, the initial direction of motion shall be dependent on the edge being sought. The home position shall be at the index pulse on either side of the rising or falling edges of the home switch as shown in the following figures. If the initial direction of movement leads away from the home switch, the drive shall reverse on encountering the relevant limit switch.

Methods 7-10: Positive Initial Motion



Methods 11-14: Negative Initial Motion



Method 15 and 16: Reserved

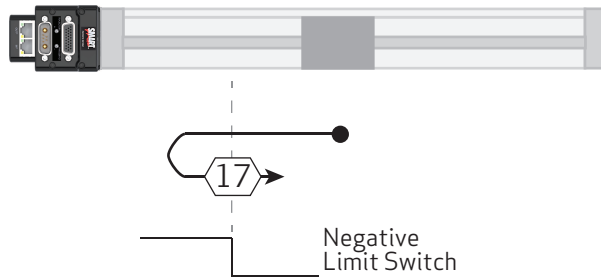
These methods are reserved.

Methods 17 to 30: Homing without index pulse

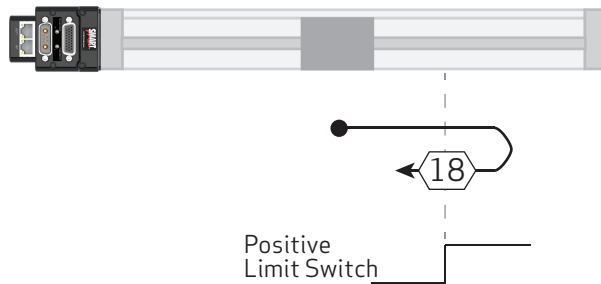
Methods 17 to 30, shown in the next sections, are the same as methods 1 to 14 except that the home position is relative to the edge of the home switch instead of an encoder index mark.

NOTE: Methods 17 and 18 are supported on both Class 5 and Class 6 motors; methods 19 to 30 are supported on Class 6 motors only.

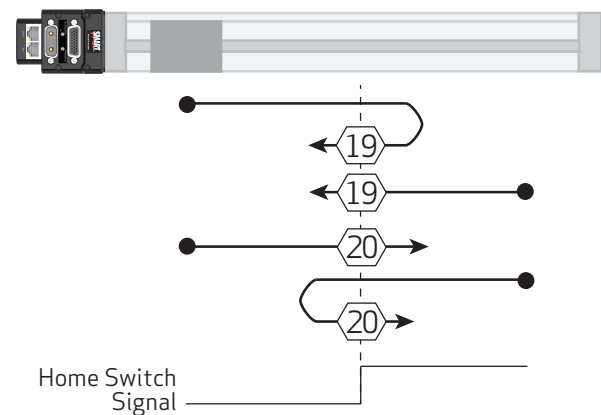
Method 17: Home to Negative Limit Switch



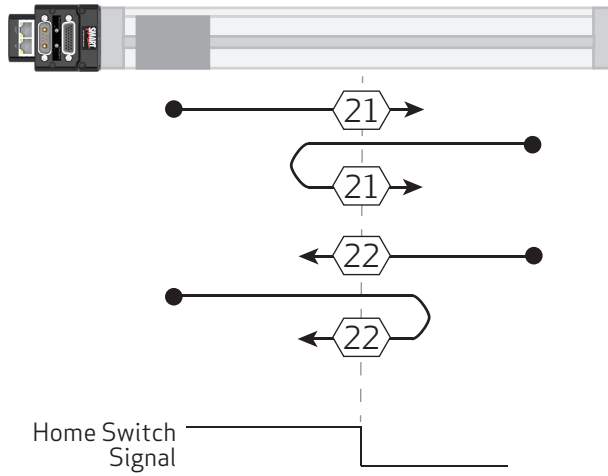
Method 18: Home to Positive Limit Switch



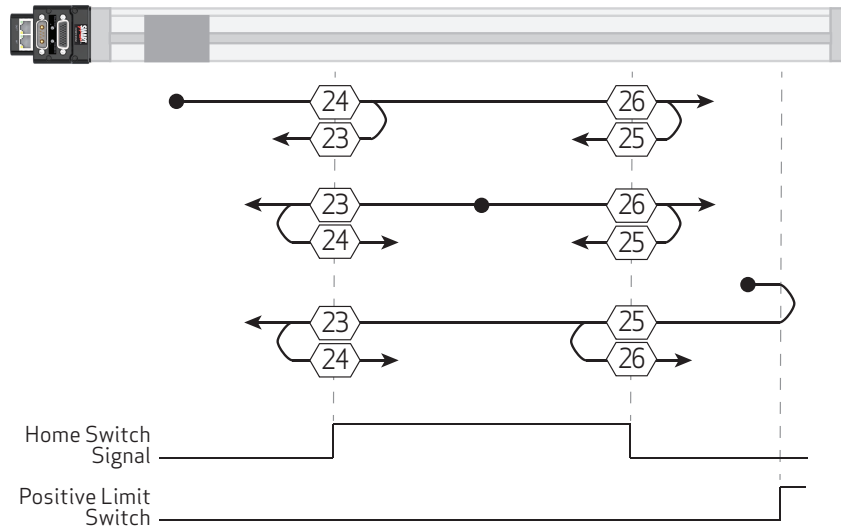
Methods 19 and 20: Home to Positive Home Switch



Methods 21 and 22: Home to Negative Home Switch

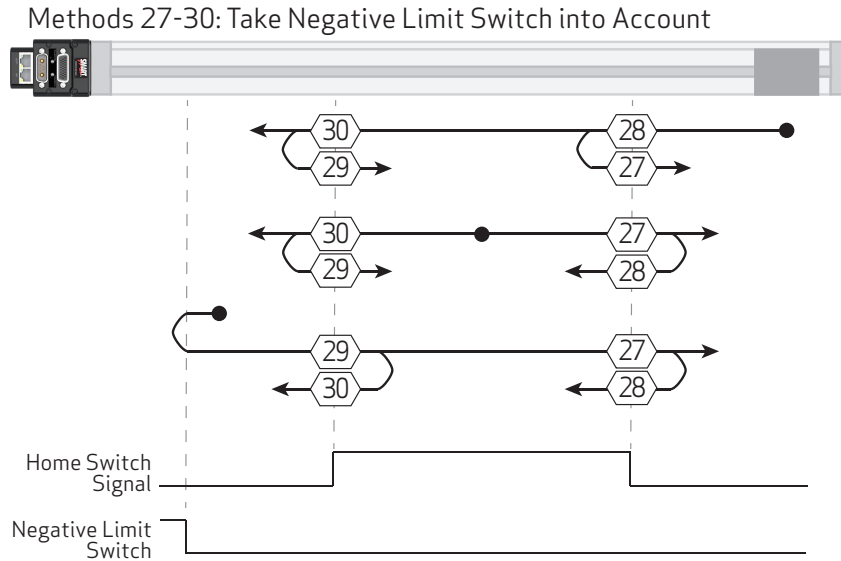


Methods 23 to 26: Homing on Home Switch (Positive Initial Movement)



This document contains technology subject to the U.S. Export Administration Regulations (EAR) and may not be exported or transferred to any unlicensed person, country, or company, by any means, or for end-uses requiring a license, without prior written approval from the U.S. Department of Commerce, Bureau of Industry and Security (BIS). Diversion contrary to U.S. law is prohibited.

Methods 27 to 30: Homing on Home Switch (Negative Initial Movement)

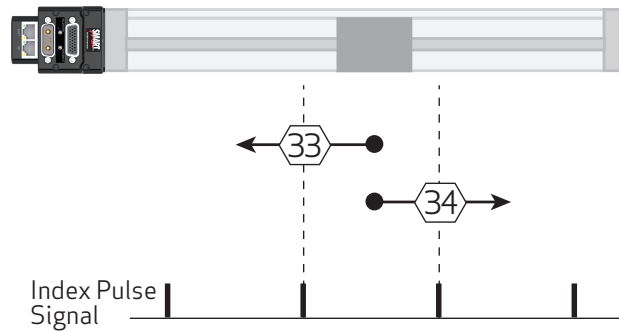


Methods 31 and 32: Reserved

These methods are reserved.

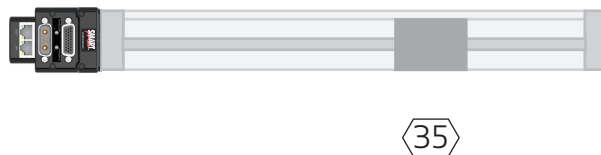
Methods 33 and 34: Homing on Index Pulse

The initial direction of homing is left or right, respectively. The home position shall be at the index pulse found in the selected direction as shown.



Method 35: Homing on Current Position

In this method, the current position shall be taken to be the home position. This method does not require the drive device to be in enabled operational state.



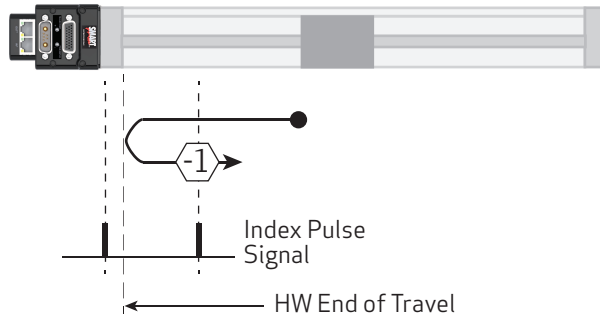
Vendor Specific Methods

The next section shows homing to a hard stop methods for Class 6 only. These methods set the home position based on the mechanical hard stop at either end of the actuator. For details, see Homing to a Hard Stop (Class 6 Only) on page 54.

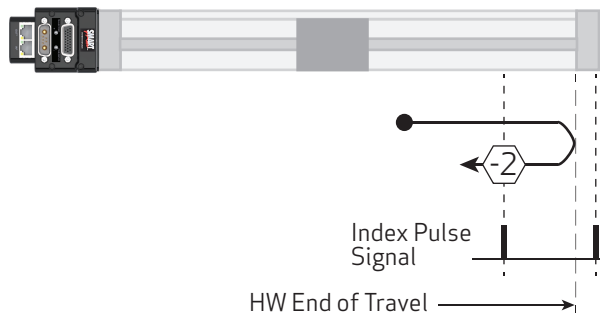
Homing to Hard Stop Methods

Homing to a hard stop supports the following four methods for the Class 6 SmartMotor servos:

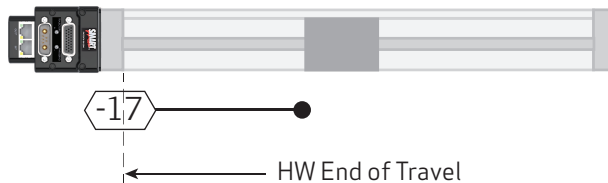
Method -1: Homing to Hard Stop (Negative) + Index Pulse



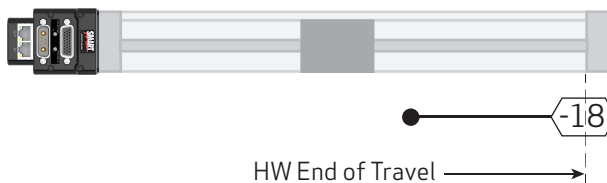
Method -2: Homing to Hard Stop (Positive) + Index Pulse



Method -17: Homing to Hard Stop (Negative)



Method -18: Homing to Hard Stop (Positive]



Communicating with the SmartMotor

CANopen communicates using a set of well-defined objects. Each homing parameter has an assigned Object Number, Object Index and Data Type.

EtherNet/IP communicates using a Device Profile Object and Device Attributes. The Class 6 SmartMotor has defined vendor specific attributes for the Position Control Supervisor object to communicate with the SmartMotor homing logic.

Homing Objects

Description	CANopen					EtherNet/IP Position Control Supervisor	AniBasic Command (Class 6 Only)
	Object Index / Sub- Index	Data Type	Access	Required	Default	Attribute (Class 6 Only)	
Homing Com- patibility	2230h. 0	Unsigned 8-bit	rw	No	0	129	RCAN(61) CANCTL (61,x)
Homing Input State	60fdh. 0***	Unsigned 32-bit	ro	No	0	16**	RIN(6)
CAN Controlword	6040h. 0	Unsigned 16-bit	rw	Yes		120	G,X,S CANCTL (68,x) CANCTL (69,x)
CAN Statusword*	6041h. 0	Unsigned 16-bit	ro	Yes		121	RCAN(3)
Homing method	6098h. 0	Signed 8-bit	rw	Yes	0	122	HM_MTHD
Homing accel/decel	609Ah. 0	Unsigned 32-bit	rw	No	4	123	HM_ADT
Homing speeds - during search for switch	6099h. 1	Unsigned 32-bit	rw	Yes	0	124	HM_VTS
Homing speeds - during search for zero	6099h. 2	Unsigned 32-bit	rw	Yes	0	125	HM_VTZ
Home offset	607Ch. 0	Signed 32-bit	rw	No	0	126	HM_OSET
Halt option	605Dh. 0	Signed 16-bit	rw	No	1	127	RCAN(62) CANCTL (62,x)

Class 6 Homing Objects (Optional Homing Operational Checks and Fault Reporting)

Description	CANopen					EtherNet/IP Position Control Supervisor	AniBasic Command (Class 6 Only)
	Object Index / Sub- Index	Data Type	Access	Required	Default	Attribute (Class 6 Only)	
Quick stop deceleration	6085h, 0	Unsigned 32-bit	rw	No	0x7FFFFFFF	128	RCAN(63) CANCTL (63,x)
<p>rw = read/write access, ro = read only access</p> <p>*Moog specific-bit 15 of the CAN Statusword (6041h) indicates the status of the home input invert option. The AniBasic command CANCTL(60,x) is used to set the value of this option. For more details, see AniBasic Homing (Class 6 Only) on page 42.</p> <p>**Reading attribute 16 will invert the input state if the homing invert was enabled CANCTL(60,1).</p> <p>*** Reading object 60fd bit 2 will represent the homing input state and invert the state if the homing invert was enabled CANCTL(60,1).</p>							

Class 6 Homing Objects (Optional Homing Operational Checks and Fault Reporting)

Description	CANopen					EtherNet/IP Position Control Supervisor	AniBasic Command (Class 6 Only)
	Object Index / Sub- Index	Data Type	Access	Required	Default	Attribute (Class 6 Only)	
Homing Time Limit	2235h, 0	Unsigned 16- bit	rw	No	0	134	HM_TIML
Homing Distance Limit	2236h, 0	Signed 32-bit	rw	No	0	135	HM_DSTL
Homing Fault (Read Only)	2237h, 0	Signed 16- bit	ro	No	0	136	RHM_FLT
Homing State (Read Only)	2239h, 0	Signed 32- bit	ro	No	0	139	RHM_STAT
<p>rw = read/write access, ro = read only access</p>							

Class 6 Homing Objects (Optional Post Homing Operation)

Description	CANopen					EtherNet/IP Position Control Supervisor	AniBasic Command (Class 6 Only)
	Object Index / Sub- Index	Data Type	Access	Required	Default	Attribute (Class 6 Only)	
Homing Post Operation	2238h, 1	Unsigned 8-bit	rw	No	0	137	HM_POST(0,x)
Homing Post Off- set	2238h, 2	Signed 32-bit	rw	No	0	138	HM_POST(1,x)

rw = read/write access, ro = read only access

CAN Controlword Homing Bits

Two bits of the CAN Controlword are used to control the homing operation.

15							9	8	7	6	5	4		3	2	1	0
								Halt					Homing Start				

CAN Controlword Bits for Homing Operation

NOTE: Changing the motor from Homing mode to Position, Velocity or Torque mode will abort an in-progress homing operation.

Animatics Homing CAN Controlword Usage

Bit 4	Bit 8	Action
0	0	Homing is not active
0->1	0	Start homing operation
1->0	0	Halt homing operation using default ramp (X Stop), abort homing operation
1	0->1	Halt homing operation using a hard stop (S Stop), abort homing operation
1	1->0	No operation, Bit 4 must be cycled (1->0->1) to start a new homing operation

Description:

- Setting bit 4 will start a homing operation, the value must transition from a 0 to a 1. Homing operations cannot be interrupted and resumed. A Homing operation will always start in the initial direction dictated by the homing Method.
- Clearing bit 4 will stop the homing operation, halting the motor near the current location, equivalent to an X Stop command
- Setting bit 8 will stop the homing operation, halting the motor near the current position, equivalent to an S Stop command

4. Clearing bit 8 will allow a homing operation to be started. In order to start the homing operation bit 4 must transition from a 0 to a 1.

CANopen Interruptible Homing CAN Controlword Usage

Bit 4	Bit 8	Action
0	0	Homing is not active
0->1	0	Start a new homing operation or resume an in-progress homing operation
1->0	0	Interrupt the homing operation at the end of the current segment.
1	0->1	Halt the motor according to the halt option code. (See CAN Object 605Dh)
1	1->0	Resume the in-progress homing operation
0	1->0	Abort any in-progress homing operation

Description:

1. Setting bit 4 will start a homing operation if one is not in-progress or will continue an in-progress homing operation from the current position.
2. Clearing bit 4 will interrupt the active homing operation. The motor will continue to move until the end of the current Homing Method segment. If the current segment is the last segment, then the homing operation is complete and will no longer be in-progress.
3. Setting bit 8 will immediately halt the motor near the current position. Any changes of Bit 4 will be ignored while Bit 8 is set.
4. Clearing bit 8 will:
 - Continue an in-progress homing operation from the current position if bit 4 is set
 - Abort an in-progress homing operation if bit 4 is clear.

CAN Statusword Homing Bits

Three bits of the CAN Statusword are used to monitor the state homing operation.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		Homing Error	Homing Attained		Target Reached										

CAN Statusword Bits for Homing Operation

The following table shows the definition of CAN Statusword bits 10, 12 and 13.

Bit 13	Bit 12	Bit 10	Definition
0	0	0	Homing operation is in progress
0	0	1	Homing operation is interrupted or not started
0	1	0	Homing is attained, but target is not reached
0	1	1	Homing operation is completed successfully
1	0	0	Homing error occurred; velocity is not 0
1	0	1	Homing error occurred; velocity is 0
1	1	X	Reserved

Examples

These examples demonstrate how to configure and control the homing operation using the CANopen 402 motion profile CAN Controlword and CAN Statusword. See the EtherNet/IP and AniBasic Command sections for details on using these alternate interfaces.

NOTE:

- The user must configure the parameters listed in the Homing Parameters table before starting the homing operation.
- The examples assume the motor is configured for Homing Method 1.
- Limit switches must be physically connected in this example. The negative limit switch will be used as the home reference.

Configure the Homing Operation, Homing Method 1

The following table provides a summary of CAN objects to configure before the homing operation is started.

Description	Read / Write	Index Object Code	Sub Index	Data	
				Length	Dec / Hex / Binary
Reset CAN Statusword	W	6040h	00	02	128 0080h 0000 0000 1000 0000b
Set homing method	W	6098h	00	01	1 01h
Set homing speed 1	W	6099h	01	04	10000 000186A0h
Set homing speed 2	W	6099h	02	04	10000 00002710h
Set homing acceleration	W	609Ah	00	04	100 00000064h
Set homing offset	W	607Ch	00	04	1000 000003E8h

Configure the Homing Operation, Homing Method 1

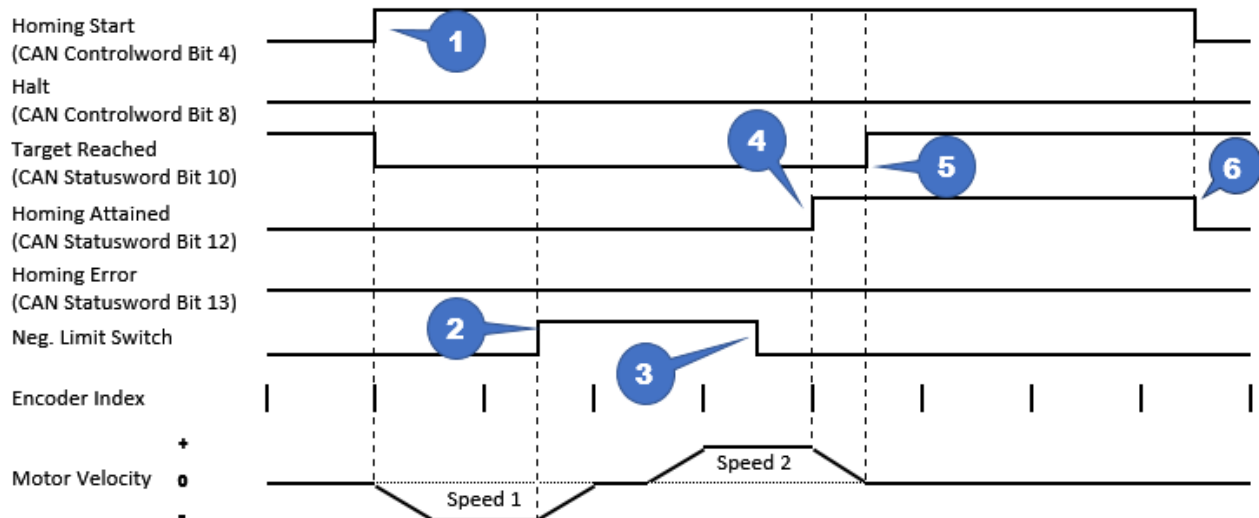
Description	Read / Write	Index Object Code	Sub Index	Data	
				Length	Dec / Hex / Binary
*Set Motor Mode to Homing (HM)	W	6060h	00	01	6 06h
*Change state: Ready to switch on	W	6040h	00	02	6 0006h 0000 0000 0000 0110b
*Change state: Switched on	W	6040h	00	02	7 0007h 0000 0000 0000 0111b
*Enable operation	W	6040h	00	02	15 000Fh 0000 0000 0000 1111b
*These objects are required for the CANopen operation and not used in the EtherNet/IP implementation (refer to the CiA 402 Drive State machine (PDSFSA) for more details).					

This document contains technology subject to the U.S. Export Administration Regulations (EAR) and may not be exported or transferred to any unlicensed person, country, or company, by any means, or for end-uses requiring a license, without prior written approval from the U.S. Department of Commerce, Bureau of Industry and Security (BIS). Diversion contrary to U.S. law is prohibited.

Start a New Homing Operation

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
0	0	0	0	1	Before the homing operation starts:
1	0	0	0	0	<ol style="list-style-type: none"> 1. Set the Homing Start bit. 2. The motor moves in the negative direction, at Homing Speed 1, until the negative limit switch is detected and then ramps to a stop. 3. The motor then moves in a positive direction, at Homing Speed 2, while looking for the negative limit switch to release and then for the next encoder index mark.
1	0	0	1	0	<ol style="list-style-type: none"> 4. The index mark is detected. <ul style="list-style-type: none"> • The home position is set to the encoder index position. • The zero position is calculated using the home position and home offset. • The motor begins ramping to a stop.
1	0	0	1	1	5. The motor comes to a halt; the homing operation is complete.
0	0	0	0	1	6. The Homing Attained status bit resets when the Homing Start CAN Controlword bit is reset.

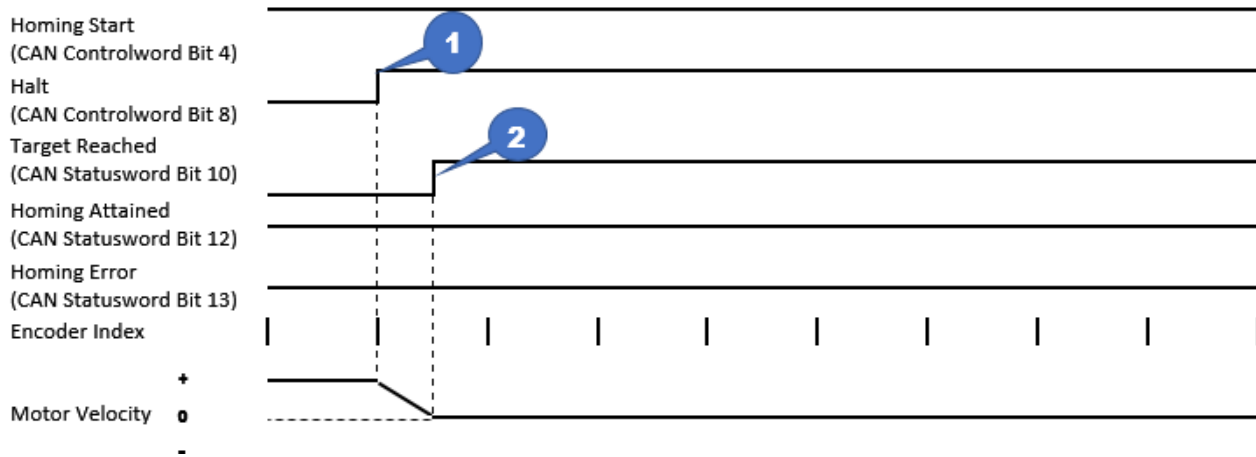
The event timing is as follows:



Halt the Motor During a Homing Operation

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	0	0	0	0	A homing operation is in progress.
1	1	0	0	0	1. Set the Halt bit to stop the motor.
1	1	0	0	1	2. The motor comes to a halt; the motor velocity is zero.

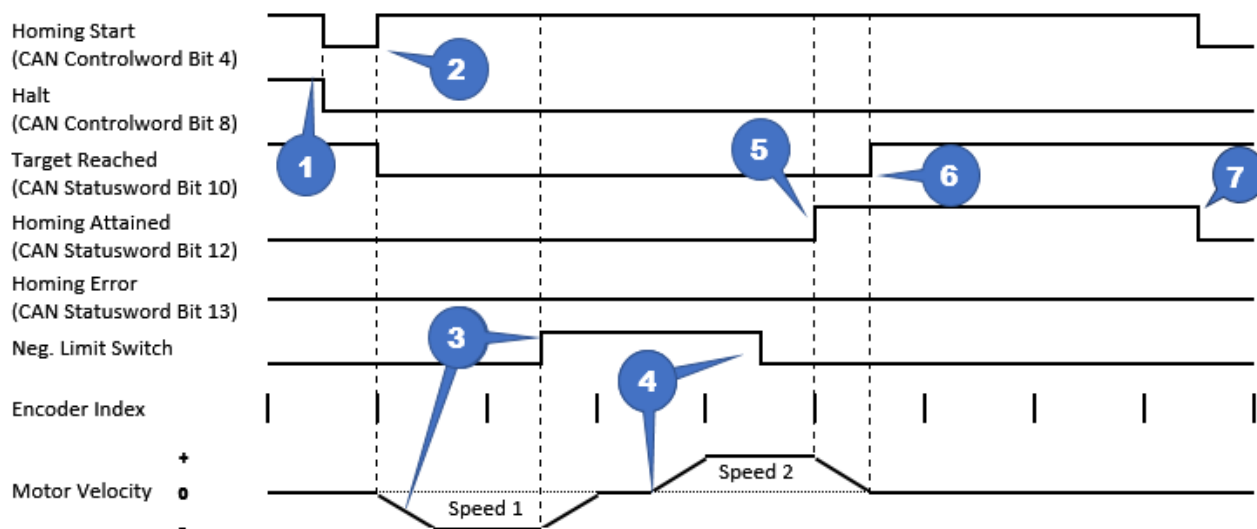
The event timing is as follows:



Resume the Halted Homing Operation - Animatics Homing

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	1	0	0	1	The motor is halted; the homing operation is aborted.
0	0	0	0	1	1. Reset homing CAN Controlword Homing Start and Halt bits. No action - the previous homing operation is aborted.
1	0	0	0	0	2. Set the Homing Start bit. 3. The motor moves in the negative direction, at Homing Speed 1, until the negative limit switch is detected and then ramps to a stop. 4. The motor then moves in a positive direction, at Homing Speed 2, while looking for the negative limit switch to release and then for the next encoder index mark.
1	0	0	1	0	5. The index mark is detected. <ul style="list-style-type: none"> The home position is set to the encoder index position. The zero position is calculated using the home position and home offset. The motor begins ramping to a stop.
1	0	0	1	1	6. The motor comes to a halt; the homing operation is complete.
0	0	0	0	1	7. The Homing Attained status bit resets when the Homing Start CAN Controlword bit is reset.

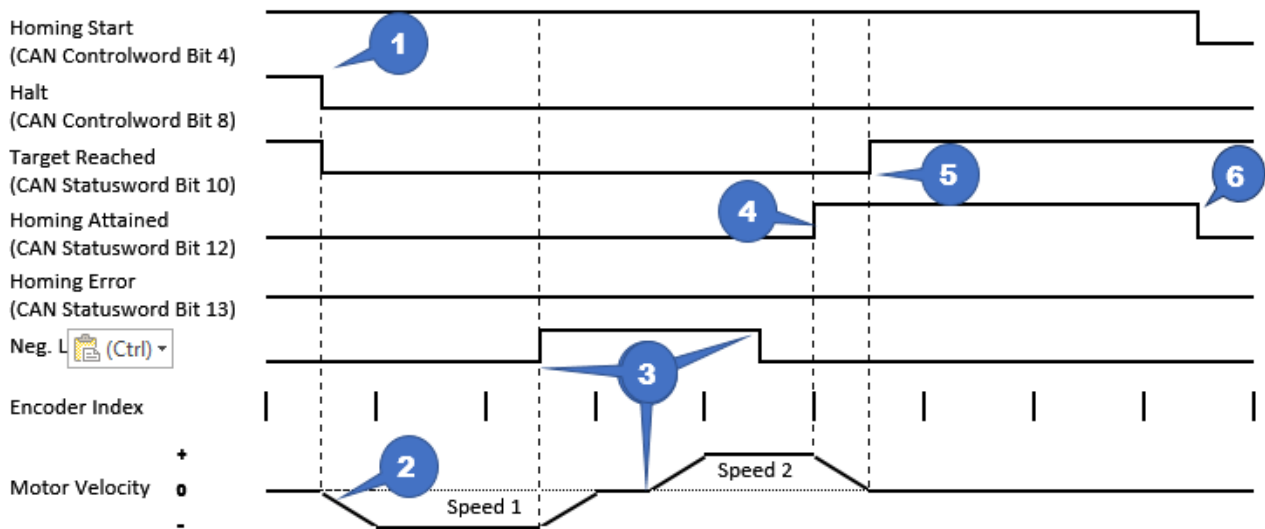
The event timing is as follows:



Resume the Halted Homing Operation - CAN Interruptible Homing

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	1	0	0	1	The motor is halted; the homing operation is suspended.
1	0	0	0	0	<ol style="list-style-type: none"> 1. Reset the CAN Controlword Halt bit. 2. The homing operation immediately continues moving in the direction/speed it was traveling when stopped, in this case toward the negative limit switch at Homing Speed 1. 3. The negative limit switch is detected. The motor then moves in a positive direction at Homing Speed 2, while looking for the negative limit switch to release and then for the next encoder index mark.
1	0	0	1	0	<ol style="list-style-type: none"> 4. The index mark is detected. <ul style="list-style-type: none"> • The home position is set to the encoder index position. • The zero position is calculated using the home position and home offset. • The motor begins ramping to a stop.
1	0	0	1	1	5. The motor comes to a halt; the homing operation is complete.
0	0	0	0	1	6. The Homing Attained status bit resets when the Homing Start CAN Controlword bit is reset.

The event timing is as follows:



Homing-Specific CANopen Objects

Refer to the following table for a list and description of the homing-specific CANopen objects. For complete details on these objects, see the [SmartMotor CANopen Guide](#).

CANopen Object	Description
Homing Compatibility Object 2230h	This object allows the writing / reading of the current Animatics Mode or CAN Interruptible mode
CAN Controlword Object 6040h	This object is the primary method of commanding motion in the SmartMotor.
CAN Statusword Object 6041h	This object indicates the current state of the drive
Halt Option Code Object 605Dh	This object determines what action should be taken if the halt bit (bit 8) is set in CAN Controlword object (6040h).
Home Offset Object 607Ch	This object shifts the origin of the actual position when the Homing (HM) mode is executed.
Quick Stop Deceleration Object 6085h	This object is used to stop the drive with the Quick Stop function, which is commanded from bit 2 of the CAN Controlword object (6040h).
Homing Method Object 6098h	This object selects the method used in Homing (HM) mode. This must be set before starting a homing process, and it should not be changed while HM mode is actively seeking home.
Homing Speeds Object 6099h	This object only applies to Homing (HM) mode. The homing profile will accelerate to these speeds depending on the segment of the homing routine that is in use.
Homing Acceleration Object 609Ah	This object is the acceleration and deceleration in Homing (HM) mode.

EtherNet/IP Homing (Class 6 Only)

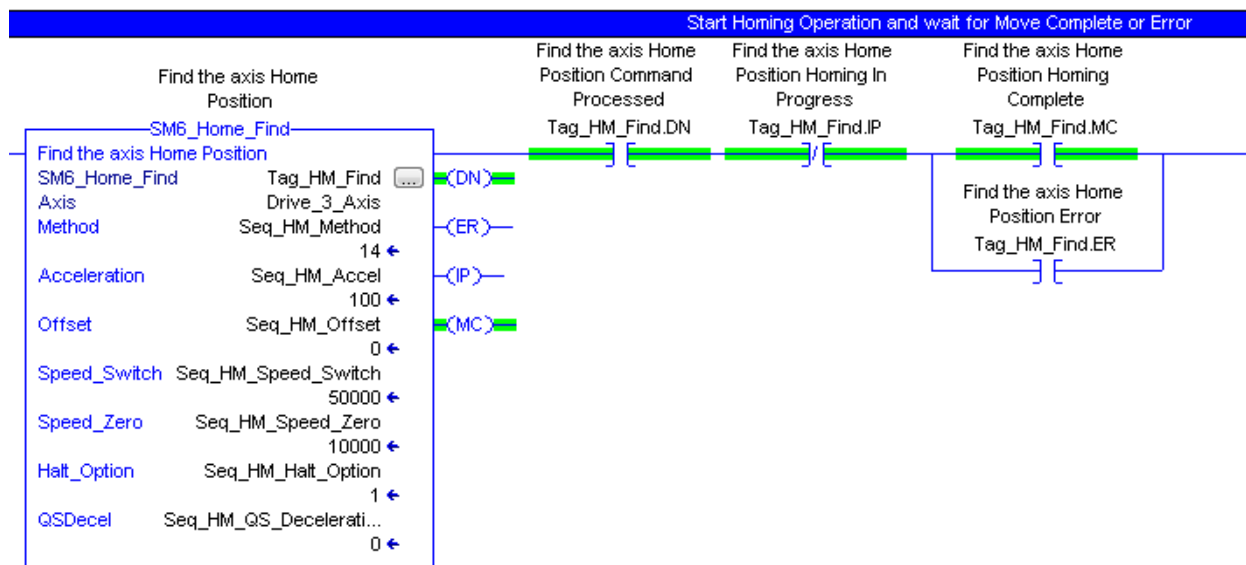
The Moog Animatics Class 6 SmartMotor communicates on EtherNet/IP networks using the Position Controller Device profile consisting of the Position Controller Supervisor Object and the Position Controller Object. Vendor Specific Attributes 120 - 129 have been added to the Position Controller Supervisor Object to communicate with the built-in CiA 402 CANopen homing logic.

Moog Animatics provides example Add On Instructions (AOIs) for the Rockwell Studio 5000 Logix Designer PLC programming environment to communicate with Rockwell PLCs. Two AOIs have been added to demonstrate the homing functionality: SM6_Home_Find to setup and initiate a homing operation, and SM6_Set_Attribute2 to manipulate the CAN Controlword and monitor the CAN Statusword.

The following examples provide ladder logic diagram snippets that demonstrate their use.

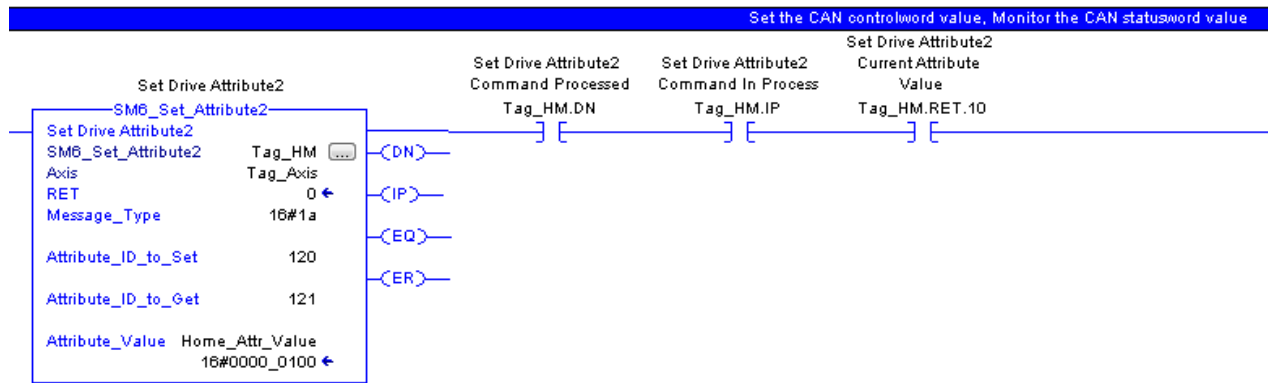
Example 1: Configure and Start a Homing Operation

This example configures and starts a homing operation and then waits for the operation to be complete.



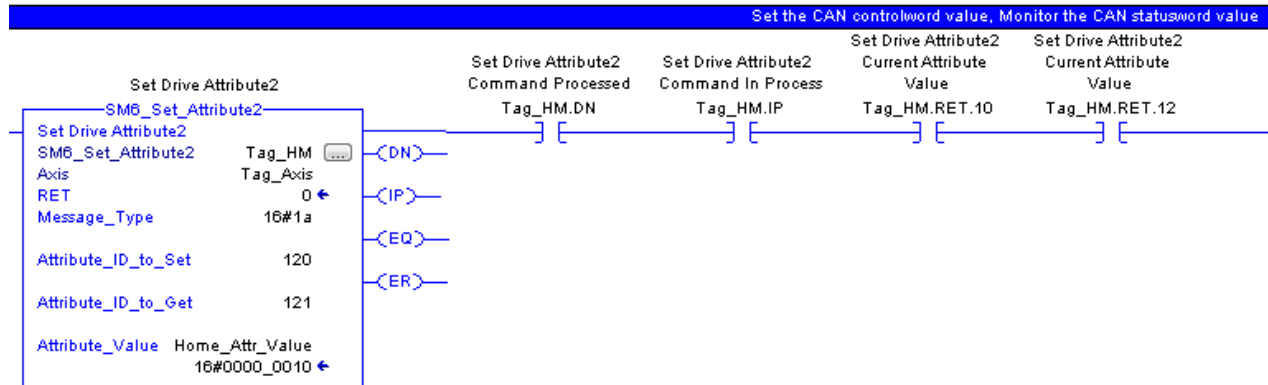
Example 2: Halt Motor During a Homing Operation

This example halts (stops) the motor during a homing operation and then waits for the motor to be halted.



Example 3: Resume the Halted Homing Operation

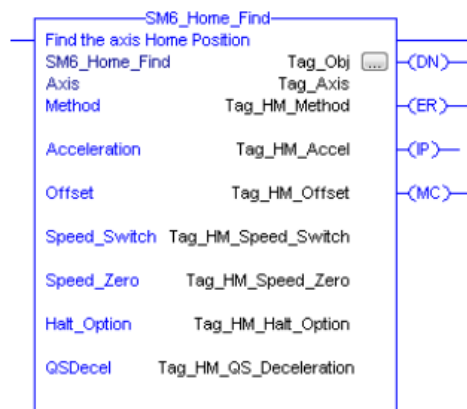
This example resumes the halted homing operation and then waits for it to be completed



NOTE: When using the Animatics Homing style, the Attribute value must cycle the Home Start bit from a 0 to 1 to restart the homing operation.

Example 4: SM6_Home_Find - Find the Axis Home Position

This example finds the home position of the axis.



Description

The Find Home feature of the Animatics motor uses the built-in CiA 402 homing logic in the motor.

Operands

Operand	Data Type	Operand Type	Description
SM6_Home_Find	SM6_Home_Find	Tag	Control tag for this AOI
Axis	SM6_Axis	Tag	The targeted drive
Method	DINT	Immediate Value	Homing method
Acceleration	DINT	Immediate Value	Homing acceleration
Offset	DINT	Immediate Value	Homing offset
Speed_Switch	DINT	Immediate Value	Speed to homing switch
Speed_Zero	DINT	Immediate Value	Speed to zero position
Halt_Option	SINT	Immediate Value	Halt option code: 1 - Slow ramp 2 - Quick stop ramp
QSDecel	DINT	Immediate Value	Quick stop deceleration

Structure

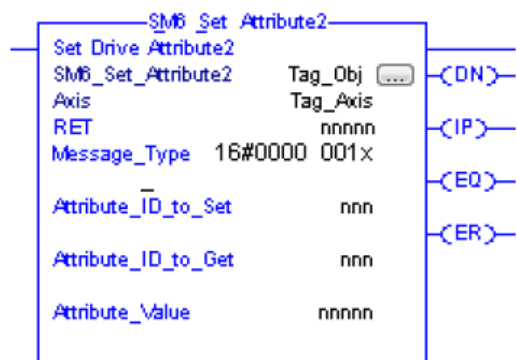
Field	Type	Description
.DN	BOOL	Set when all of the commands to setup and execute the find Home operation are sent Cleared when a Find Home operation is started/restarted
.ER	BOOL	Set when an error response is received from the motor Cleared when a Find Home operation is started/restarted
.IP	BOOL	Set while sending commands to the motor Cleared when the motor comes to a stop for any reason
.MC	BOOL	Set when the Find Home status indicates the operation has completed Cleared when a Find Home operation is started/restarted

Operation

Scan Mode	Description
Prescan	Initialize variables
Rung-condition-in is FALSE	Monitors for error conditions or changing of the drive mode
Rung-condition-in is TRUE	Updates the homing attributes in the drive with the specified values and starts the Find Home operation

Example 5: SM6_Set_Attribute2 - Set Attribute 1, Get Attribute 2

This example is used to set Attribute 1 and to get Attribute 2.



Description

Sets the value of the Attribute_ID_to_Set and monitors the value of the Attribute_ID_to_Get. The AOI will continue to monitor the Axis response while the IP flag is set, which allows the ladder logic to check for the value of the Attribute_ID_to_Get on another rung.

Operands

Operand	Data Type	Operand Type	Description
SM6_Set_Attribute	SM6_Set_Attribute	Tag	Control tag for this AOI
Axis	SM6_Axis	Tag	Targeted drive
RET	DINT	Output Value	Value of the attribute returned by the drive
Message_Type	SINT	Immediate Value	16#0000_001a - Position Controller Supervisor 16#0000_001b - Position Controller
Attribute_ID_to_Set	DINT	Immediate Value	ID of the attribute being set
Attribute_ID_to_Get	DINT	Immediate Value	ID of the attribute whose value is being monitored
Attribute_Value	DINT	Immediate Value	Requested value to which the attribute is being set

Structure

Field	Type	Description
.DN	BOOL	Set when the command is sent to the motor Cleared when instruction is not enabled or attribute ID is cleared
.IP	BOOL	Set when the response for the requested attribute ID is received Cleared when one of the input parameter values change, an error response is received, or the attribute id being returned is changed

Operation

Field	Type	Description
.EQ	BOOL	<p>Set when the value returned from the drive for the Attribute_ID_to_Get is equal to the Attribute_Value parameter</p> <p>Cleared when the attribute value returned by the motor is for a different attribute ID or does not match the Attribute_Value parameter</p> <p>Flag is only useful if the Attribute_ID_to_Set is the same as the Attribute_ID_to_Get</p>
.ER	BOOL	<p>Set when an error response is received from the motor</p> <p>Cleared when a Get Attribute operation is started or one of the input parameter values change</p>

Operation

Scan Mode	Description
Prescan	Initialize variables and reset timeout
Rung-condition-in is FALSE	Monitor/update the Attribute_Value until the Attribute_ID_to_Get is cleared/changed.
Rung-condition-in is TRUE	Send a message to the drive to set the Attribute_ID_to_Set to Attribute_Value and monitor the value of the Attribute_ID_to_Get; the instruction will continue to monitor the Axis response and updates the RET value while the IP flag is set

AniBasic Homing (Class 6 Only)

The commands listed in the following table are used for homing operations through AniBasic on the Moog Animatics SmartMotor. For details on the commands in the shaded rows, see the [SmartMotor™ Developer's Guide](#).

Command	Read Command	Description
MH	RMODE	Write/Read the Homing Mode
HM_VTZ=n	RHM_VTZ	Write/Read the Homing Velocity to Zero
HM_VTS=n	RHM_VTS	Write/Read the Homing Velocity to Switch
HM_ADT=n	RHM_ADT	Write/Read the Homing Accel/Decel
HM_OSET=n	RHM_OSET	Write/Read the Homing Offset
HM_MTHD=n	RHM_MTHD	Write/Read the Homing Method
	RCAN(2)	Read the CAN Controlword object (6040h) CAN(2)&272 extracts Homing bits
	RCAN(3)	Read the CAN Statusword object (6041h) CAN(3)&13312 extracts Homing bits
CANCTL(60,x), x=0,1	RCAN(60)	Write/read the Homing Invert flag: 0 - Active high 1 - Active low
CANCTL(61,x), x=0,1	RCAN(61)	Write/read the Homing style: 0-Animatics 1-CAN Interruptible
CANCTL(62,x), x=1,2	RCAN(62)	Write/read the CAN Halt Option code: 1 - Decelerate on the profile deceleration 2 - Slow down on Quick Stop deceleration
CANCTL(63,x)	RCAN(63)	Write/Read the CAN Quick Stop Deceleration value: 0-2147483647 (0h-7FFFFFFFh) Default: 7FFFFFFFh
CANCTL(68,x), x=0,1	RCAN(68)	Write/read the CAN Controlword Homing Start bit
CANCTL(69,x), x=0,1	RCAN(69)	Write/read the CAN Controlword Homing Halt bit
G		Start a homing operation
X		Halt a homing operation
S		Stop a homing operation
NOTE: All of these commands support Combitronic communications.		

Homing Command Usage

This section describes the steps for using the AniBasic homing commands.

1. Use the G, X and S commands
 - G - Starts the homing operation in both Homing styles
 - Clears and then Sets the CAN Controlword Homing Start bit
 - The firmware remembers that the homing operation was started from the G command so that the X and S commands will know to abort the homing operation
 - X - Halts the homing operation
 - Clears the CAN Controlword Homing Start bit
 - Animatics Style - halts the motor on the normal ramp; aborts the homing operation
 - CAN Interruptible Style- Interrupts the homing operation; halts the motor using the ramp specified by the CAN Halt Option code
 - S - Halts the homing operation
 - Animatics Style - (Sets the CAN Controlword HALT bit) Halts the motor on the Quick Stop Deceleration; aborts the homing operation
 - CAN Interruptible Style- Processed as a normal S command; the homing machine signals a Homing Error (CAN Statusword bit 13); note that an X command *is required* to clear the error
2. The following commands manipulate the CAN Controlword directly. The Homing machine will not clear these bits when the operation completes, so the user is responsible for resetting these flags when needed.
 - CANCTL(68,1) - Set the CAN Controlword Homing Start bit
 - Animatics Style - Starts a new homing operation when the Start bit transitions from 0 to 1
 - CAN Interruptible Style- Starts a new or Resumes an in-process homing operation when the Start bit transitions from 0 to 1
 - CANCTL(68,0) - Clear the CAN Controlword Homing Start bit
 - Animatics Style - halts the motor on the normal ramp; aborts the homing operation.
 - CAN Interruptible Style- Interrupts the homing operation, which stops at the end of the current segment using the normal ramp; it resumes from that point when the bit is set using the G command or the CANCTL(68,x) command
 - CANCTL(69,1) - Set the CAN Controlword Homing Halt bit
 - Animatics Style - Halts the motor; aborts the homing operation
 - CAN Interruptible Style- Starts a new or resumes an in-process homing operation when the Start bit transitions from 0 to 1
 - CANCTL(69,0) - Clear the CAN Controlword Homing Halt bit
 - Animatics Style - halts the motor on the normal ramp; aborts the homing operation
 - CAN Interruptible Style- Interrupts the homing operation at the end of the current segment using the normal ramp; it resumes from that point on a G command

Start a Homing Operation - G command

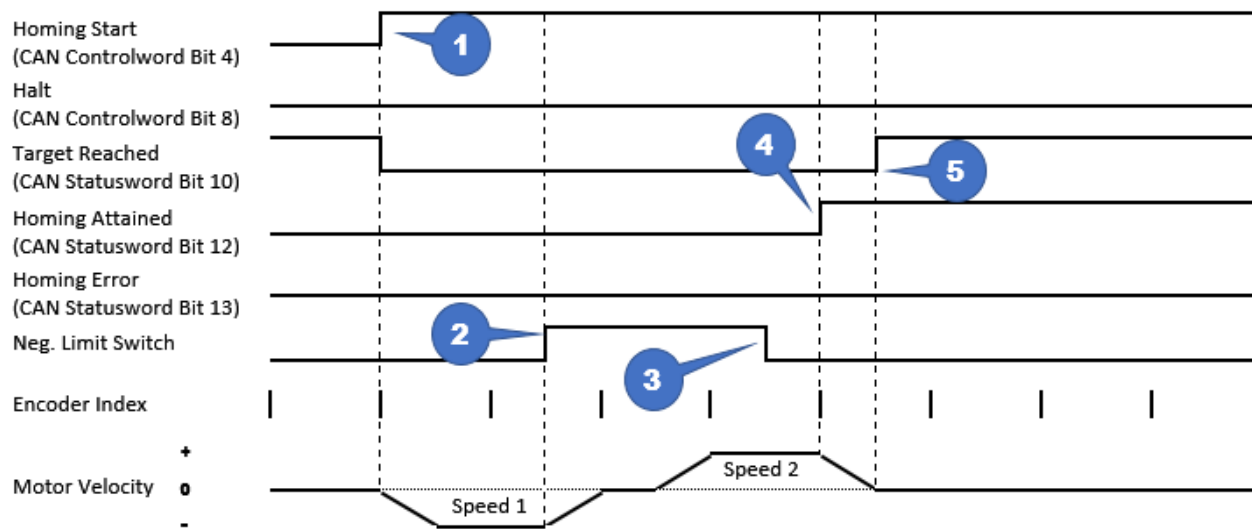
The G command starts the homing operation in both the Animatics and CAN Interruptible Homing styles using the following steps:

1. Clear the Homing CAN Controlword Homing Start bit if necessary.
2. Set the CAN Controlword Homing Start bit.

Assuming HM_MTHD is set to method 1 (homing on the negative limit switch):

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
0	0	0	0	1	Before the homing operation starts:
1	0	0	0	0	<ol style="list-style-type: none"> 1. A G command starts the operation. 2. The motor moves in the negative direction, at Homing Speed 1, until the negative limit switch is detected and then ramps to a stop. 3. The motor then moves in a positive direction, at Homing Speed 2, while looking for the negative limit switch to release and then for the next encoder index mark.
1	0	0	1	0	<ol style="list-style-type: none"> 4. The index mark is detected. <ul style="list-style-type: none"> • The home position is set to the encoder index position. • The zero position is calculated using the home position and home offset. • The motor begins ramping to a stop.
1	0	0	1	1	<ol style="list-style-type: none"> 5. The motor comes to a halt; the homing operation is complete.

The event timing is as follows:

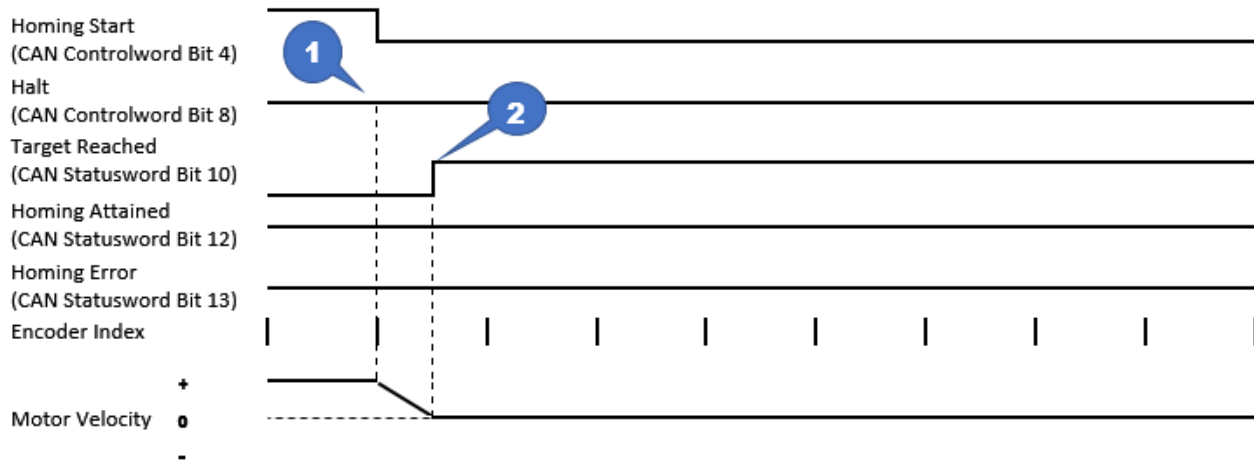


Halt a Homing Operation - X command (Animatics Style)

The following X-command procedure stops an in-progress Animatics Style homing operation. For information on halting a CAN Interruptible Style homing operation, see the next section.

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	0	0	0	0	A homing operation is in progress.
0	0	0	0	0	1. Issue an X command, the motor begins decelerating.
0	0	0	0	1	2. The motor comes to a halt; the motor velocity is zero.

The event timing is as follows:

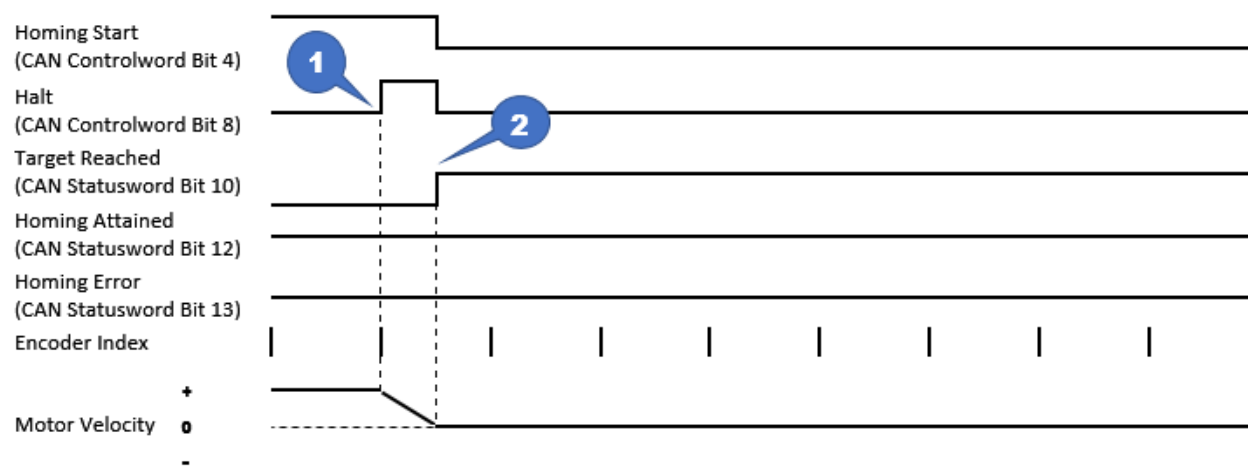


Halt a Homing Operation - X command (CAN Interruptible Style)

The following X-command procedure stops an in-progress CAN Interruptible Style homing operation. For information on halting an Animatics Style homing operation, see the previous section.

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	0	0	0	0	A homing operation is in progress.
1	1	0	0	0	1. Issue an X command, the motor begins decelerating.
0	0	0	0	1	2. The motor comes to a halt; the motor velocity is zero.

The event timing is as follows:



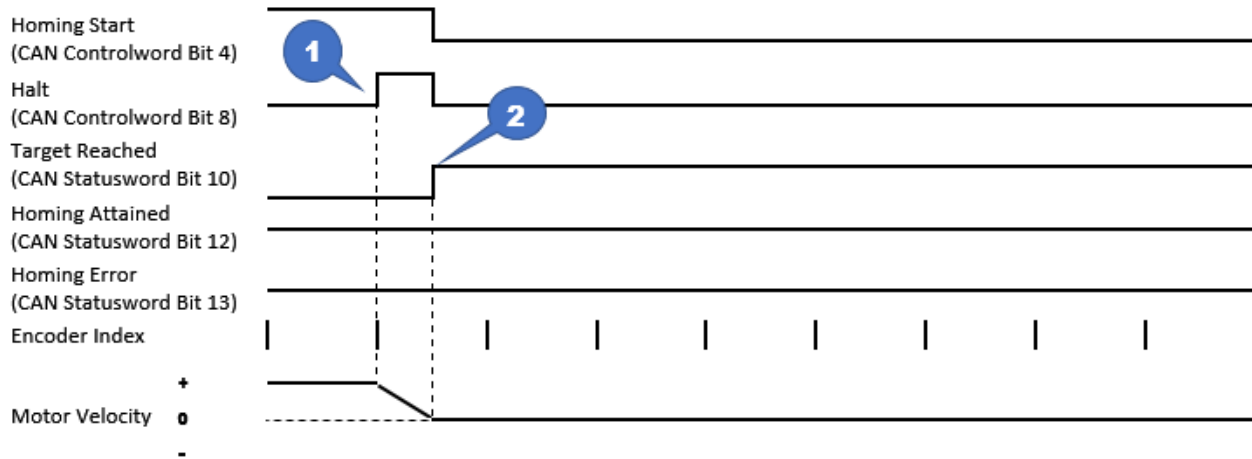
Halt a Homing Operation - S command (Animatics Style)

The following S-command procedure stops an in-progress Animatics Style homing operation. For information on halting a CAN Interruptible Style homing operation, see the next section.

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	0	0	0	0	A homing operation is in progress.
1	1	0	0	0	1. Issue an S command, the motor begins decelerating.
0	0	0	0	1	2. The motor comes to a halt; the motor velocity is zero.

NOTE: The interruptible mode was designed to use the Homing CAN Controlword bits to initiate, halt, and potentially resume operation. The G, X, and S commands can be used to start and stop operation but only consider doing this as a backup / emergency stop scenario. For example, an S command can immediately stop a homing operation, but an X command *is required* to clear the emergency stop operation.

The event timing is as follows:



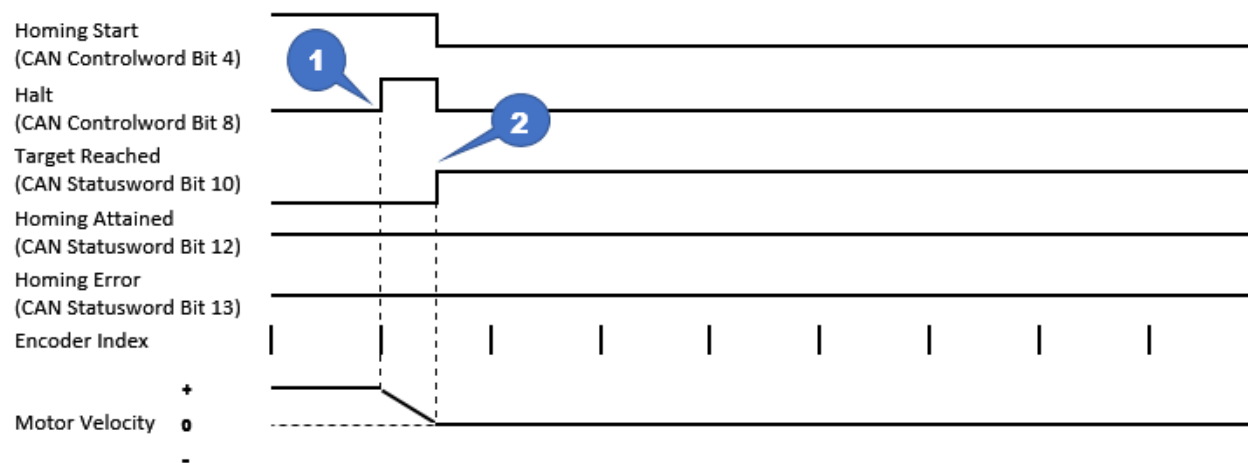
Halt a Homing Operation - S command (CAN Interruptible Style)

The following S-command procedure stops an in-progress CAN Interruptible Style homing operation. For information on halting an Animatics Style homing operation, see the previous section.

CAN Controlword		CAN Statusword			Description
Bit 4	Bit 8	Bit 13	Bit 12	Bit 10	
Homing Start	Halt	Homing Error	Homing Attained	Target Reached	
1	0	0	0	0	A homing operation is in progress.
1	0	0	0	0	1. Issue an S command, the motor begins decelerating.
0	0	1	0	1	2. The motor comes to a halt; the motor velocity is zero.

NOTE: The interruptible mode was designed to use the Homing CAN Controlword bits to initiate, halt, and potentially resume operation. The G, X, and S commands can be used to start and stop operation but only consider doing this as a backup / emergency stop scenario. For example, an S command can immediately stop a homing operation, but an X command *is required* to clear the emergency stop operation.

The event timing is as follows:



Example: AniBasic (Class 6 Only)

This user program will:

- Start a homing operation
- Display the CAN Statusword changes while waiting for the operation to complete
- Go to the zero position after the homing operation completes

```
#define HM_CANCTL_Statusword      3      'CANCTL homing index values
#define HM_CANCTL_HomingInvert    60     '
#define HM_CANCTL_HomingStyle     61     '
#define HM_CANCTL_HaltOption      62     '
#define HM_CANCTL_QSDecl          63     '
#define HM_CANCTL_HomingStart     68     '
#define HM_CANCTL_HomingHalt      69     '-----
#define HM_CANSTS_HomeError       8192   'CAN Statusword bit masks
#define HM_CANSTS_HomeAttained    4096   '
#define HM_CANSTS_HomeReached     1024   '
#define HM_CANSTS_HomeComplete    5120   '
#define HM_CANSTS_HomeInProgress  0      '-----
#define HM_CAN_STATUSBITS 13312        'CAN Statusword bits mask

' Set up the homing variables
HM_VTZ=20000
HM_VTS=40000
HM_ADT=21
HM_OSET=3000
HM_MTHD=1
CANCTL(HM_CANCTL_HomingStyle,0)      'Use Animatics Style homing

GOSUB2                                ' Do homing operation and wait for it to complete
MP PT=0 VT=50000 G TWAIT              ' Go to position zero
END

' Start homing operation
' Wait for homing operation to complete, displaying CAN Statusword changes
C2
MH      ' Set motor to homing mode
G       ' Start the homing operation

' Wait for the homing operation to complete
s = HM_CANSTS_HomeInProgress
ss = HM_CANSTS_HomeReached
WHILE s != HM_CANSTS_HomeComplete | s!=ss
  ' Break from the loop if a homing error is detected
  IF s & HM_CANSTS_HomeError
    PRINT ("*")
    BREAK
  ENDIF

' Update the controller motor's homing status bit if changed
IF (MODE==6 & s!=ss)
  'Display the homing status bit value
  GOSUB (9)

  ' Save the homing status bit value
```

```

        SS=S
    ENDIF

    ' Get the current CAN Statusword value,
    ' extract only the homing status bit values.
    s = CAN(HM_CANCTL_Statusword) & HM_CAN_STATUSBITS

    LOOP
    PRINT(#13)
RETURN

' Display the CAN Statusword homing status bit value(s)
C9
    IF (s==0)
        a = HM_MTHD
        IF (CAN(HM_CANCTL_HomingStyle)==0)
            PRINT("Animatics: ",a," ")
        ELSE
            PRINT("Interruptible: ",a," ")
        ENDIF
    ENDIF
    PRINT(" s=",s," ")
    IF (s==0)
        PRINT("In-Progress")
    ELSE
        IF (s & HM_CANSTS_HomeError)
            PRINT("Error")
            ' Append a comma if more than one bit set
            IF (s != HM_CANSTS_HomeError) PRINT(",") ENDIF
        ENDIF
        IF (s & HM_CANSTS_HomeAttained)
            PRINT("Attained")
            ' Append a comma if HomeReached bit is set
            IF (s & HM_CANSTS_HomeReached) PRINT(",") ENDIF
        ENDIF
        IF (s & HM_CANSTS_HomeReached)
            PRINT("Reached")
        ENDIF
    ENDIF
    PRINT(" ")
RETURN

```

AniBasic Parameters (Class 6 Only)

The commands listed in the following table are used for optional homing operations and fault checks on the Class 6 SmartMotor.

Command	Read Command	Description
HM_TIML=n	RHM_TIML	Homing Time Limit – an optional parameter that allows the operator to specify the maximum time (in seconds) that the homing method should complete by. If homing does not complete by this time limit, then homing will stop. The valid range of values are 0 to 65535. CAN Interruptible Homing style does allow a homing operation to Halt and Resume. If the Homing time limit checking is enabled and the CAN Interruptible Halt is commanded, the Homing time limit check is also halted.
HM_DSTL= n	RHM_DSTL	Homing Distance Limit – an optional parameter that allows the operator to specify the maximum distance (in encoder counts) that the homing method should complete by. If this check is enabled, the zero position is where homing was commanded and the maximum distance is the plus or minus encoder counts from this starting position. The valid range of values are 0 to 2147483647.
	RHM_FLT	Homing Fault Code(Read only) – in instances where the homing operation failed, this fault code can be queried by the user to assist in troubleshooting the cause of the failure. See the next section, Homing Fault Codes (Class 6 Only), for information on returned codes.
	RHM_STAT	Homing State(Read only) – in instances where the homing operation failed, this code can be queried by the user to assist in troubleshooting the cause of the failure.
HM_POST (0,n)	RHM_POST(0)	Homing Post Operation n = 0 – This Post Homing Operation allows the actuator to stay at home position found during the homing operation n = 1 – This Post Homing Operation allows the actuator to move to the zero position AFTER the homing operation has completed n = 2 – This Post Homing Operation allows the actuator to move to the desired offset AFTER the homing operation has completed NOTE: The post homing move uses the default values associated with the SmartMotor. For example, if the post homing value is set to move to the zero position but the VT variable is set to zero, the post homing move to the zero position will not occur.
HM_POST (1,n)	RHM_POST(1)	Homing Post Offset - Sets the offset to use when the Homing Post Operation is set to 2.
NOTE: All of these commands support Combitronic communications.		

Homing Fault Codes (Class 6 Only)

These are the valid error codes that the software currently supports for homing operations:

Code	Description
-31	Distance Limit Violation (directly related to Homing Distance Limit variable)
-30	Time Limit Violation (directly related to Homing Distance Limit variable)
-24	Homing Method is invalid
-23	Fault related to Homing methods 9, 10, 25, and 26 – Disarm limit errors on Positive Limit switch
-21	Fault related to Homing methods 2, 7, 8, 18, 23, and 24 – Disarm limit errors on Positive Limit switch
-20	Fault related to Homing methods 13, 14, 29, and 30 – Disarm limit errors on Negative Limit switch
-18	Fault related to Homing methods 1, 11, 12, 17, 27, and 28 – Disarm limit errors on Negative Limit switch
-17	An unexpected limit switch error occurred or homing parameter error occurred
-1	Unexpected limit switch was hit or an unexpected stop occurred
0	No Fault NOTE: The Homing Fault variable is automatically set to 0 at the beginning of a homing operation
The faults below are related to the state machine.	
1	HOMING_FAIL_CONFIG – Improper configuration led to an error state: relevant limit was disabled
2	HOMING_HALT_ABORT – Halt motor, abort homing
3	HOMING_HALT_NOABOR – Halt motor, no abort (allow homing operation to resume)
4	HOMING_BEGIN – Determine initial action
5	HOMING_SLOWON_L_PHME – Slow when left positive home edge is detected
6	HOMING_SLOWON_R_PHME – Slow when right positive home edge is detected
7	HOMING_ARMON_PLIM – Arm when positive limit is deasserted
8	HOMING_ARMON_NLIM – Arm when negative limit is deasserted
9	HOMING_ARMON_PHME – Arm when positive home edge is detected
10	HOMING_ARMON_NHME – Arm when negative home edge is detected
11	HOMING_CAPON_NLIM – Capture position and stop when falling edge of negative limit is detected
12	HOMING_CAPON_PLIM – Capture position and stop when falling edge of positive limit is detected.
13	HOMING_CAPON_NHME – Capture position and stop when falling edge of home switch is detected.
14	HOMING_CAPON_PHME – Capture position and stop when rising edge of home switch is detected
15	HOMING_STOPON_PHME – Stop when we see the positive edge of the home

Code	Description
16	HOMING_STOPON_NHME – Stop when we see the negative edge of the home
17	HOMING_STOPON_NLIM – Stop when we see the negative limit asserted
18	HOMING_STOPON_PLIM – Stop when we see the positive limit asserted
19	HOMING_STOPON_NENDSTOP – Stop on negative end stop - relevant for homing method -1 and -17
20	HOMING_HOLDON_NENDSTOP – Hold on negative end stop - relevant for homing method -1 and -17
21	HOMING_STOPON_PENDSTOP – Stop on positive end stop - relevant for homing method -2 and -18
22	HOMING_HOLDON_PENDSTOP – Hold on positive end stop - relevant for homing method -2 and -18
23	HOMING_STOPON_CAP – Stop when the position is captured by internal index.
24	HOMING_RETURN_POSITION – Return the present position as the Home Position
25	HOMING_WAIT_FOR_STOP2 – A stop has been discovered/commanded and we are monitoring for speed to come to a stop
26	HOMING_WAIT_FOR_FSTOP – A stop has been discovered/commanded and we are monitoring for speed to come to a stop

Using Mixed Mode Operations After Homing

There are applications where you may wish to use mixed modes of operation after homing. For example, you may want to use dual trajectory and gearing. In these cases, you need to ensure that the motor is at a known starting location. To accomplish that, issue the next command sequence after the homing operation.

```
' Issue these commands immediately after homing, while at position 0.
O=0      'Set the origin to 0
O(1)=0   'Set trajectory 1 to 0
PRT=0    'Set the relative target position to 0
```

That command sequence sets the applicable internal counters to zero, which ensures that counter values are as expected for mixed mode operations.

Homing to a Hard Stop (Class 6 Only)

The Class 6 motor is also capable of homing to a hard stop to facilitate finding the zero position of an application. Homing to a hard stop is a sensorless method for homing/referencing the axis to a known absolute position. Hard-stop homing is similar to switch-based homing but does not require any type of hardware limit switch. Instead, hard-stop homing detects the mechanical end of travel of the axis being homed. For details and diagrams on these and other homing methods available for Class 5 and Class 6 motors, see Homing Methods on page 18.

Homing to Hard Stop Control Variables

Homing to a hard stop detects the mechanical end of travel associated with the application and homes based on this end of travel.

The unique variables for homing to hard stop are defined below. In reality, homing to a hard stop uses many of the same variables that were defined earlier in this document.

Variable	Description
Homing Amps	Indicates the motor amperage that will be used during homing to hard stop operation. The valid range of values are 0 to 1023 with 1023 being the maximum current.
Homing Error Limit	Indicates the position error limit that will be used to stop motion when the actuator is against the hard stop. Generally, this position error limit should be a low value. The valid range of values are 0 to 262143.
Homing Torque	The four homing to hard stop methods support leaving the homing operation with the actuator held against the hard stop in torque mode (for a given time). This variable indicates the torque value. The valid range of values are 0 to 32767.
Homing Torque Time	The four homing to hard stop methods support leaving the homing operation with the actuator held against the hard stop in torque mode (for a given time). This variable indicates the time (in msec). The valid range of values are 0 to 32767.

NOTE: When homing to hard stop is initiated, the KI variable is saved and temporarily brought to 0 during all homing to hard stop operations. Similarly, the AMPS variable is saved and temporarily adjusted to the Homing Amps variable defined above. At the conclusion of the homing operation, these saved KI and AMPS variables are restored. Therefore, it is recommended to perform an "end of homing check" before adjusting KI, AMPS, or initiating a new mode (such as a velocity move or a position move).

Homing to Hard Stop - Communicating with the Class 6 SmartMotor

Description	CANopen					EtherNet/IP Position Control Supervisor	AniBasic Command (Class 6 Only)
	Object Index / Sub- Index	Data Type	Access	Required	Default	Attribute (Class 6 Only)	
Homing Amps	2231h, 0	Unsigned 16- bit	rw	Yes, Hom- ing to HS	0	130	HM_AMPS
Homing Position Error	2232h, 0	Signed 32-bit	rw	Yes, Homing to HS	0	131	HM_EL
Homing Torque	2233h, 0	Unsigned 16- bit	rw	Yes, if Torque timer > 0	0	132	HM_T
Homing Torque Timer	2234h, 0	Unsigned 16- bit	rw	No	0	133	HM_HTMR

rw = read/write access, ro = read only access

NOTE: When homing to hard stop is initiated, the KI variable is saved and temporarily brought to 0 during all homing to hard stop operations. Similarly, the AMPS variable is saved and temporarily adjusted to the Homing Amps variable defined above. At the conclusion of the homing operation, these saved KI and AMPS variables are restored. Therefore, it is recommended to perform an "end of homing check" before adjusting KI, AMPS, or initiating a new mode (such as a velocity move or a position move).

Homing to Hard Stop - AniBasic Homing

The commands listed in the following table are used for the optional homing to hard stop through AniBasic on the Moog Animatics SmartMotor.

NOTE: When homing to hard stop is initiated, the KI variable is saved and temporarily brought to 0 during all homing to hard stop operations. Similarly, the AMPS variable is saved and temporarily adjusted to the Homing Amps variable defined above. At the conclusion of the homing operation, these saved KI and AMPS variables are restored. Therefore, it is recommended to perform an "end of homing check" before adjusting KI, AMPS, or initiating a new mode (such as a velocity move or a position move).

Command	Read Command	Description
HM_AMPS=n	RHM_AMPS	<p>Homing Amps - This parameter is used for Homing to Hard Stop to limit how hard the actuator will meet the end stop. Encourage the use of low values to start out and incrementally increasing until a proper value is found.</p> <p>Value... 1023 is max amperage</p> <p>Range is 0 to 1023</p> <p>Before using this homing amps value, the normal amps value is saved by the software and restored upon the completion of the homing. Moog strongly suggests that the normal amps value is not updated during the homing to hard stop operation because this could compromise the integrity of the normal amps value that is present after homing is complete. Please do not use "AMPS=" from the command line or the equivalent command from a communication interface while homing is taking place.</p>
HM_EL=n	RHM_EL	<p>Homing Position Error - This parameter defines how great a position error must exist before stopping the driving into the hard stop.</p> <p>Range is 0 to 262143</p> <p>Encourage the use of low values to start out and incrementally increasing until a proper value is found.</p> <p>Before using this homing position error value, the normal position error value is saved and restored (by the software) upon the completion of the homing. Moog strongly suggests that the normal position error value is not updated during the homing to hard stop operation because this could compromise the integrity of the normal position error value that is present after homing is complete. Please do not use "EL=" from the command line or the equivalent command from a communication interface while homing is taking place.</p>
HM_T=n	RHM_T	<p>Homing Torque</p> <p>This optional object is used for Homing to Hard Stop to use torque mode to drive into the hard stop (for a time period... see "Homing Torque Timer")</p> <p>Value is in increments of torque where 32767 is maximum torque</p> <p>Range is 0 to 32767</p> <p>Before using this homing torque value, the normal torque value is saved and restored upon the completion of the homing. Moog strongly suggests that the normal torque value is not updated during the homing to hard stop operation because this could compromise the integrity of the normal torque value that is present after homing is complete. Please do not use "T=" from the command line or the equivalent command from a communication interface while homing is taking place.</p>
HM_HTMR=n	RHM_HTMR	<p>Homing Torque Timer</p> <p>This object is used for Homing to Hard Stop determine the time that torque mode will be enabled (to drive the actuator into the hard stop).</p> <p>Value is in mS</p> <p>Range is 0 to 32767</p>
NOTE: All of these commands support Combitronic communications.		

This document contains technology subject to the U.S. Export Administration Regulations (EAR) and may not be exported or transferred to any unlicensed person, country, or company, by any means, or for end-uses requiring a license, without prior written approval from the U.S. Department of Commerce, Bureau of Industry and Security (BIS). Diversion contrary to U.S. law is prohibited.

AniBasic Example (Homing Method -1)

The code snippet below provides an example of homing to a hard stop with the homing method set to -1. Note that this isn't a full program – it's only the homing section and won't run on its own. For SmartMotor programming details, see the [SmartMotor™ Developer's Guide](#).

```
'Set slow homing speeds
HM_VTZ=1000
HM_VTS=1000

'Set conservative acceleration / deceleration values
HM_ADT=21

'Set a homing offset of 3000
HM_OSET=-3000

' Set homing method to -1 (minus 1)
HM_MTHD=-1

'Set homing to hard stop specific parameter - Homing Amps
HM_AMPS=100

'Set homing to hard stop specific parameter - Homing Position Error Limit
HM_EL=27

'Set homing to hard stop specific parameter - Homing Torque
HM_T=600

'Set homing to hard stop specific parameter - Homing Torque Time
HM_HTMR=2000

'Use Animatics style homing
CANCTL(61,0)

'Set the motor mode to "Homing"
MH

'Start the Homing to Hard Stop Operation
G
```

Revision History

March 2022

- Initial release

March 2023

The following content was added or expanded:

- Homing to hard stop
- Time checking for all homing operations
- Distance checking for all homing operations
- Post homing
- Error feedback for failed homing operations

February 2026 - Rev A

- Updated contact information.

This document contains technology subject to the U.S. Export Administration Regulations (EAR) and may not be exported or transferred to any unlicensed person, country, or company, by any means, or for end-uses requiring a license, without prior written approval from the U.S. Department of Commerce, Bureau of Industry and Security (BIS). Diversion contrary to U.S. law is prohibited.

TAKE A CLOSER LOOK

Moog Animatics, a sub-brand of Moog Inc. since 2011, is a global leader in integrated automation solutions. With over 30 years of experience in the motion control industry, the company has U.S. operations and international offices as well as a network of Automation Solution Providers worldwide.

Moog Animatics
1995 NC Hwy 141
Murphy, NC 28906
United States

Email: animatics_sales@moog.com

For Animatics product information, visit www.animatics.com

For more information or to find the office nearest you, email animatics_sales@moog.com

Moog is a registered trademark of Moog Inc. and its subsidiaries.
All trademarks as indicated herein are the property of Moog Inc. and its subsidiaries.
©2022 - 2026 Moog Inc. All rights reserved. All changes are reserved.

Moog AnimaticsSmartMotor™ Homing Procedures and Methods Application Note, Rev. A
SC80100023-001

www.animatics.com

