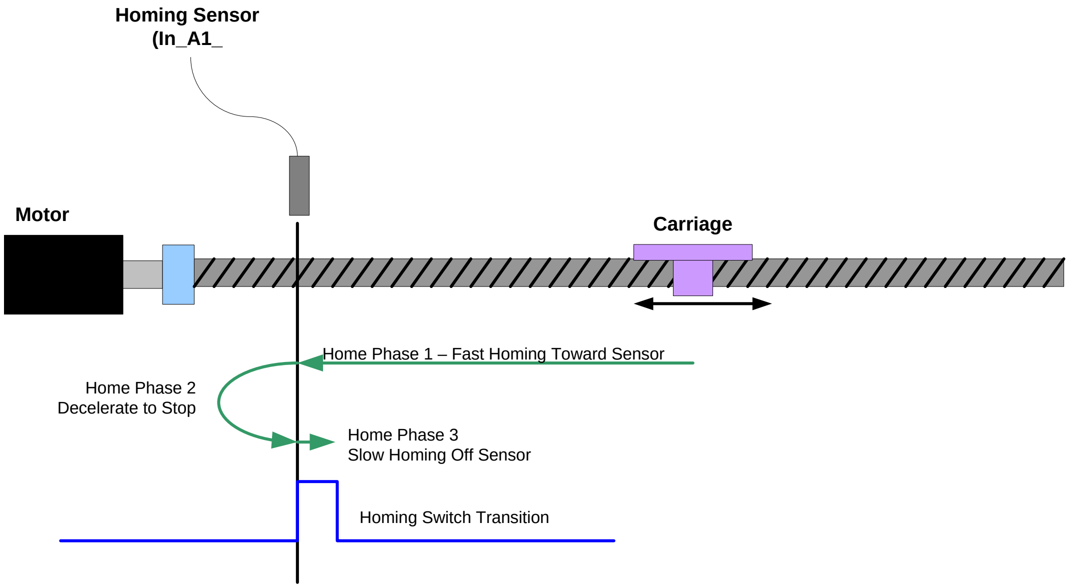
# PositionServo Sample Program: Homing Example (Basic)

**Concept:**
This sample program shows how to manually create a basic homing routine from within the User Program. The code executes homing on start up, and then when it is requested by the 'Request homing' Input. Following a successful home sequence the program carries out a pre-defined sequence of Motion Commands, until either the enable is removed or homing is requested again.

The program uses the Enable Input on A3 as a safety enable for safety devices on the system.
Input A4 is used to request the drive re-home the system again to the homing sensor position.
Input A1 is the input from the Homing Sensor.
Output 1 is used to signal to the outside world that homing was completed successfully.
The PositionServo is placed in Position Mode with internal reference selected through the relevant variable at the start of the program.

At initialisation (following power up or reset) the drive will first run the homing routine.
The homing routine will perform a continuous backwards move at a 'Fast' homing speed (toward the homing sensor) until the homing sensor is triggered. When the sensor is activated the drive will end this move (stop), select a 'Slow' homing speed and move forward until the homing sensor deactivates.
The Actual Position Register is then set to zero and velocity set to the value required for normal operation.
The homing complete indicator (Out1) is switched off when the homing routine is started and switched on when the routine is completed successfully.
The Homing routine motion Profile is Shown below.



An event is used to detect the input request to carry out the homing procedure again. This event is turned off while homing is actually taking place.

No subroutines are used in this example.

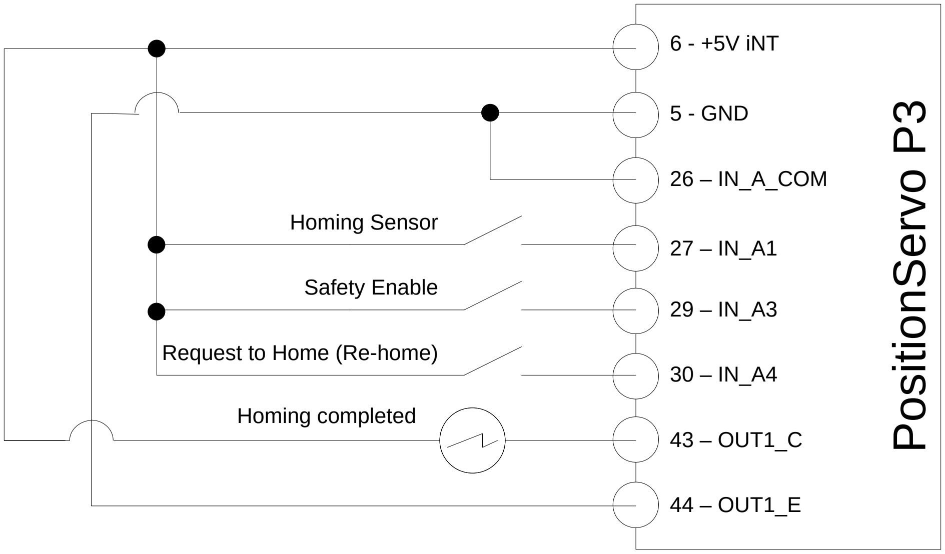
Examples are provided as an aid in system design. Proper safety considerations remain the responsibility of the system designer.

**Motor mechanics:**
For the purpose of the demonstration a motor should be used with a disc fitted to the motor shaft so that the user can see the Positioning / Velocity output executed by the PositionServo.

**Fault Handling:**
In the event of a fault, the code will be restarted. The operator must switch the drive enable input off and on again for the program loop to restart. Homing will be performing again immediately following reset from a fault condition.

**I/O:**
IN_A1: Homing Sensor
IN_A3: Safety Enable / stop - Connected to machine safety Guards / Devices
IN_A4: Request Homing Input
OUT1: Homing Completed Indicator

## Homing Example (Basic) Connection Diagram



Examples are provided as an aid in system design. Proper safety considerations remain the responsibility of the system designer.

## Indexing Code:

```
;********************************  PositionServo User Indexing Program  ********************************
;********************************************* Header *******************************************
;Title              :       Sample Homing Program From PositionServo Training Exercises
;Author             :       AC Technology International Ltd
;Description        :       Program Performs Simple Homing Routine followed by Preset Sequence of Moves
;                   :       Input A4 used to Re-Initalise Homing Routine during Main Program Execution
;                   :       Homing Sensor on Input B1
;                   :       Homing Complete Output on Out 1
;Version Number :           V1.0.1
;Date               :       22/11/06
;
;********************************************  I/O List  *********************************************
;       Input A1-           Homing Sensor
;       Input A2-           not used
;       Input A3-           Enable Input / Safety stop
;       Input A4-           Request Homing Input
;       Input B1-           not used
;       Input B2-           not used
;       Input B3-           not used
;       Input B4-           not used
;       Input C1-           not used
;       Input C2-           not used
;       Input C3-           not used
;       Input C4-           not used
;
;       Output 1-           Homing Complete
;       Output 2-           not used
;       Output 3-           not used
;       Output 4-           not used
;
;       Analog In 1         -       not used
;       Analog In 2         -       not used
;       Analog Out          -       not used
;
;       Encoder Out         -       not used
;


;******************************** Initialize and Set Variables ***************************************
; Define Constants and Variables. Assign I/O and Initialize Variable Values

 UNITS = 1                                   ; Units in RPS
Accel = 50                                   ; Accelerate 50 RPS/S
Decel = 50                                   ; Decelerate 50 RPS/s
Define Homing_Sensor In_A1           ; Define Variable for Homing Sensor Input
Define System_Enable In_A3                   ; Define Variable for Enable Input
Define Homing_Request In_A4          ; Define Variable for Homing Request Input
Define Home_Completed Out1           ; Define Variable for Homing complete Indicator

VAR_REFERENCE = 1                            ; Set Reference to Internal
VAR_DRIVEMODE = 2                            ; Set Operating mode to Position mode
VAR_ENABLE_SWITCH_TYPE = 0                   ; Enable switch function set to "Inhibit"
```

```
;************************************************** Events ********************************************
Event homing_call input Homing_Request rise          ; Events looks for Homing Request Input and Jumps to Homing Code
            jump Homing
Endevent


;********************************************** Main Program ********************************************

PROGRAM_START:
Wait While System_Enable == 0               ; Wait for drive safety enable to be active - Safe Condition
Enable                                      ; Enable drive

Homing:                                     ; Homing Routine, Performed at Start Up and after Homing Routine Request inout
goes high
Home_Completed = 0                          ; Turn off homing completed indicator
event homing_call off                       ; Switch off event to check for homing request input
maxv = 10                                   ; set slow velocity for homing
move back until Homing_Sensor               ; move back at slow velocity until homing sensor is activate
maxv = 2                                    ; set very slow velocity
move until homing_sensor == 0               ; move forward off homing sensor
APOS = 0                                    ; Set actual position register to 0
maxv = 50                                   ; Set velocity back to normal speed
wait while Homing_Request == 1              ; wait for the homing request signal to be removed
event homing_call on                        ; Switch on event to check for homing request input
Home_Completed = 1                          ; Turn on homing completed indicator



Move_Profile:                                        ; Basic position loop
Moved 5                                     ; move distance 5
wait time 200                               ; wait 200 ms
moved 10                                    ; move distance 10
wait time 500                               ; wait 500 ms
moved -15                                   ; move back distance 15
wait time 1500                              ; wait 1500 ms
goto move_profile                           ; restart positioning loop

end

fault_section:
            wait while System_Enable == 1   ; wait until enable input is removed
            goto program_start              ; restart program


;********************************************** Sub-Routines ***********************************************
;           Enter Sub-Routine code here



;****************************************** Fault Handler Routine *****************************************
;           Enter Fault Handler code here
ON FAULT
            resume fault_section
ENDFAULT
```

Examples are provided as an aid in system design. Proper safety considerations remain the responsibility of the system designer.