

Nanomotion Tech Note 105

Using AC and DC Modes with Nanomotion AB2 Driver in Closed-Loop for Nanometer Level Positioning



Rev A

March 28, 2006

1 Comac Loop, Suite 14B2
Ronkonkoma, NY 11779
Tel (800)821-6266
Fax (631)585-1947

1. Introduction - Ultra- High Resolution in DC Mode

Nanomotion motors normally operate in resonance, or AC mode. However, they can also operate in non-resonant, or DC mode. When operating in DC mode, the motor no longer drives by dynamic friction but rather firmly holds position by its static friction, benefiting from the built-in preload which firmly attaches the motor to the application mass. By applying a non-resonant driving voltage the core piezo-element of the motor would bend, like an actuator, dragging with it the application mass. As long as the acceleration is kept below a threshold value, there will be no slip and the application will move according to the deflection of the motor. The sign of the applied voltage determines the direction of motion. Nanomotion has developed a methodology which combines the standard resonant mode for long travel with the non-resonant motion. The AC mode is used to move inside the target radius. Once within the target radius the DC mode is activated, thus enabling travel of up to ± 150 nm with single nm resolution. The AB2 drivers were developed to facilitate both the AC and DC modes. This Tech Note describes how to use the AB2 in conjunction with any servo controller (with the SpiiPlus controller as an example) for combined AC and DC modes.

2. DC Mode Characteristics

Figure 1 shows a typical position versus command transfer function of a Nanomotion motor operating in the DC mode, being driven by an AB2 driver. As can be seen in Figure 1, varying the command to the AB2 from +10V to -10V yields a displacement of +550nm to -350nm. However, as a conservative design rule, *a maximum swing of ± 150 nm should be assumed*. The position is a linear function of the command voltage, with some hysteresis. Some degree of asymmetry is normal and expected.

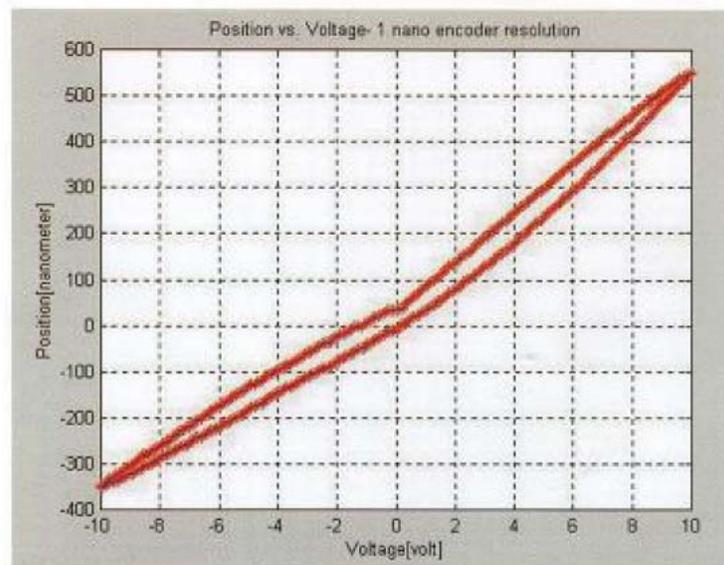


Figure 1 - DC Mode Position versus Command Transfer Function

1 Comac Loop, Suite 14B2
Ronkonkoma, NY 11779
Tel (800)821-6266
Fax (631)585-1947

Figure 2 below depicts near-perfect open-loop tracking of a triangular wave, with a 1 nm encoder. The sensitivity is typically 0.8 nm/volt.

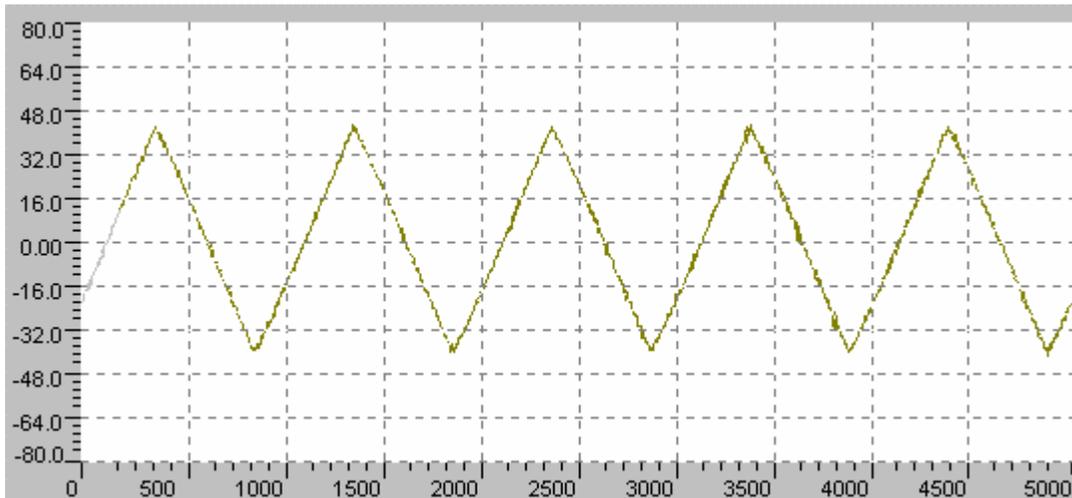


Figure 2 - Triangular Wave in DC mode, 1 Hz , 90 V peak-to-peak (at motor terminals)

Note: The Y-axis is displacement in nm. The X-axis is time in mSec.

3. AC and DC Modes Operation with the AB2 Driver

3.1 Electrical

The DC mode is enabled by shorting pin 21 at the AB2 I/O connector to ground. Leaving the pin open or pulling it high selects the AC mode. The delay between changing the state of pin 21 and when the AB2 driver changes its mode from AC to DC or vice versa is about 1 milliseconds.

Warning: when switching between AC and DC modes, the command voltage should be less than 5V. If this condition is violated, the AB2 might trip a protection circuit, disable itself and issue a FAULT output.

If possible, it is preferred to force the command zero volts while switching between modes. With some controllers this can be done disabling the amplifier. If the command is at a non-zero value during the switch, there could be a position jump of up to 300 nm in either direction.

3.2 Interfacing to a Controller

Figure 3 below shows a typical interface between a servo controller and the AB2 driver. The controller shown in the example is ACS-Tech80 SpiiPlus PCI-4/8. However, any standard servo controller can be interfaced to the AB2. All that is needed in addition to the standard interface is a digital output which is connected to pin 21 of the AB2 I/O connector. A logic high selects AC mode, and should be the default initial state upon power up. A logic low selects the DC mode.

1 Comac Loop, Suite 14B2
Ronkonkoma, NY 11779
Tel (800)821-6266
Fax (631)585-1947

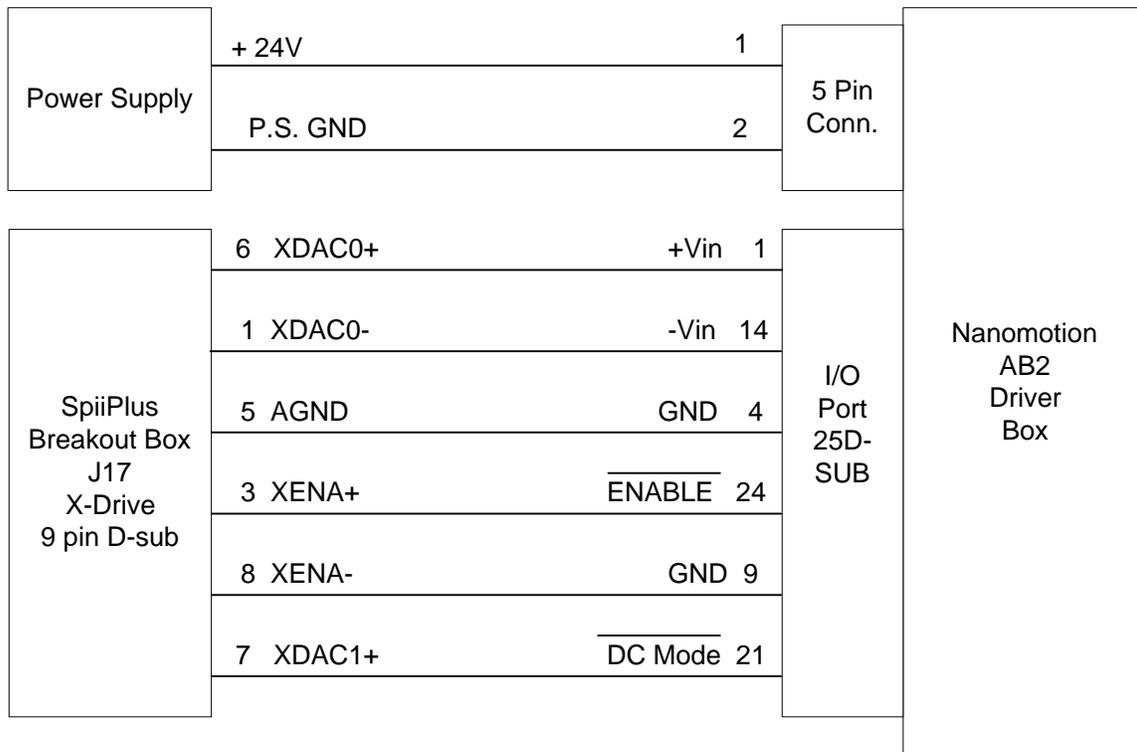


Figure 3 - Interfacing the AB2 to the SpiiPlus Controller

Note: Interface is shown for X-axis.

3.3 Control Methodology

In general, the AC mode should be used for long and relatively coarse moves. The DC mode should be used for fine positioning or for holding the current position very tightly (down to ± 1 count.) To illustrate the control strategy, let us take an example. A motion control system is composed of the following components:

- Motor: HR4
- Driver: AB2-HR-E2
- Stage: 50 mm travel, cross-rollers, with 0.5 Kg load
- Encoder resolution: 1nm
- Controller: SpiiPlus PCI-4/8

Let's assume the stage needs to move from zero position to 1 mm, hold position to within ± 5 nm and go back to zero position.

To achieve this performance, the following procedure is recommended:

- a) Tune the system in AC mode to achieve consistent settling to within the DC capture range (typically $\pm 150\text{nm}$). Record the AC servo gains.
- b) Since the AB2 driver exhibits dead-band behavior, the controller's firmware needs to include the appropriate Nanomotion Algorithm, i.e. it needs to support friction compensation (SpiiPlus SLFRC parameter) and the dead zone (SpiiPlus SLDZMIN and SLDZMAX parameters.)
- c) The Dead Zone Max parameter should be set to a value that is less than the DC mode capture range (typically $\pm 150\text{nm}$) and less than the system position error spec. This will ensure that the AC mode will bring the stage into a position band that is tight enough for the DC mode to "pull into" the final position band. In our example let us choose Dead Zone Max = 70nm.
- d) The Dead Zone Min parameter is set to some fraction of Dead Zone Max. Let us choose a value of 10nm.
- e) Tune for gentle deceleration curve. The idea is to move into the Dead Zone Min band such that the motor "falls into" the breaking mode at a low velocity. This will ensure minimal mechanical relaxation as the motor enters DC mode and will guarantee that the DC mode will be able to hold position without saturating the motor's command.
- f) Tune the system in DC mode. Different servo gains might be required than those obtained for AC mode. In general, the DC mode is mostly affected by Kp (SLPKP in the SpiiPlus) and Ki (SLVKI in the SpiiPlus). Record the DC servo gains.
- g) Recommended program sequence:
 - 1) Disable Axis
 - 2) Initialize variables and parameters
 - 3) Load AC servo gains and select AC mode
 - 4) Enable Axis
 - 5) Home the stage
 - 6) Disable Axis
 - 7) Load AC servo gains
 - 8) Enable Axis
 - 9) Perform move to desired location
 - 10) Wait until the end of actual motion and settling is complete
 - 11) Select DC mode
 - 12) Load DC servo gains
 - 13) Wait until axis has settled to within required position band
 - 14) Perform desired process at position
 - 15) Go to step 6

1 Comac Loop, Suite 14B2
 Ronkonkoma, NY 11779
 Tel (800)821-6266
 Fax (631)585-1947

3.4 Actual Test Results

Figure 4 below shows actual test results obtained on the system discussed in paragraph 3.3. As can be seen, a move of 1 mm (1 million counts, 1 count = 1nm) is followed by settling into ± 1 nm. The ACSPL program used to generate the move is documented in Appendix A.

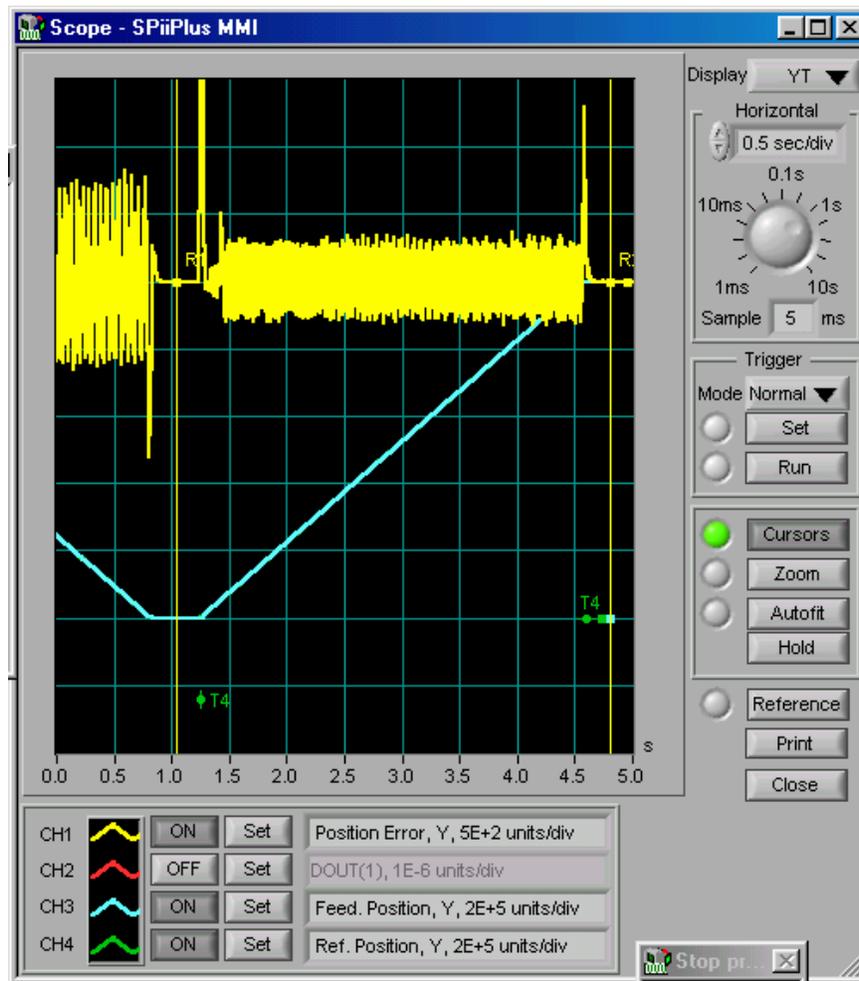


Figure 4 – Actual Test Results: 1 mm move and settle to ± 1 nm

1 Comac Loop, Suite 14B2
Ronkonkoma, NY 11779
Tel (800)821-6266
Fax (631)585-1947

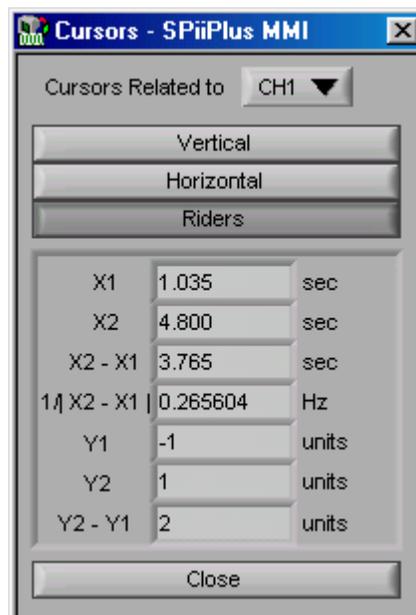


Figure 4 – Actual Test Results: 1 mm move and settle to ± 1 nm (cont.)



1 Comac Loop, Suite 14B2
Ronkonkoma, NY 11779
Tel (800)821-6266
Fax (631)585-1947

Appendix A: AC and DC Modes Demo Program (in SpiiPlus ACSPL language)

```
#!/ Controller version = 4.20.01
#!/ Date = 01/06/2004 17:19
#0
! Test program with Firmware version 4.16
! HR4 motor, AB2 driver and 1 nm encoder moving mass 0.5 kg
! Example of PTP move with switching between AC and DC modes and settling to 1nm
!
! ***** Initialize variables *****
INT Axis
Axis = 0          ! Select X axis
CERRI(Axis)=1E4  ! Set critical position error to 10 micron
CERRV(Axis)=1E4
CERRA(Axis)=1E4
ENTIME(Axis)=1
VEL(Axis)=300000 ! Speed = 0.3 mm/sec
ACC(Axis)=1E7    ! Acc = 10 mm/sec^2
DEC(Axis)=1E7    ! Dec = 10 mm/sec^2
JERK(Axis)=1E9   ! Jerk = 1 m/sec^3
CALL AC_MOTOR_PAR ! Load AC mode
SET FPOS(Axis)=0 ! Zero current position
MFLAGS(Axis).13 =0 ! Select loop polarity (stage-dependent)
MFLAGS(Axis).7 =1 ! Select Nanomotion mode, as we are using the AB2 driver
! ***** Motion loop *****
FIRST_POINT:
ENABLE(Axis);PTP(Axis), 1E6 ! move 1 mm
TILL ^MST(Axis).#MOVE      ! wait until actual move is complete
CALL DC_MOTOR_PAR         ! Load DC mode
WAIT 50
DISP "Axis Position = ",FPOS(Axis)
WAIT 300
! The motor must be disabled when switching from DC to AC mode
DISABLE(Axis);CALL AC_MOTOR_PAR ! Load AC Servo parameters
ENABLE(Axis);PTP(Axis), 0      ! move to zero position
TILL ^MST(Axis).#MOVE         ! wait until actual move is complete
CALL DC_MOTOR_PAR            ! Load DC servo parameters
WAIT 50
DISP "Axis Position = ",FPOS(Axis)
WAIT 300
DISABLE(Axis);CALL AC_MOTOR_PAR !Load AC Servo parameters
GOTO FIRST_POINT
! ***** Load AC Mode *****
AC_MOTOR_PAR:
MFLAGS(Axis).30 = 0 ! Select AC
XVEL(Axis)=2E9 ! Do not change this number even if encoder resolution is 0.5nm
SLPKP(Axis)=200
SLVKP(Axis)=500
SLVKI(Axis)=2000
SLVLI(Axis)= 50
SLFRC(Axis)= 10 ! SET friction to 1v
XCURI(Axis)=50
```



1 Comac Loop, Suite 14B2
Ronkonkoma, NY 11779
Tel (800)821-6266
Fax (631)585-1947

```
XCURV(Axis)=100
XRMS(Axis) = 60 ! Limit the RMS of the driver input to 6V
SLDZMIN(Axis)=10 ! D_ZONE_MIN=10nm
SLDZMAX(Axis)=70 ! D_ZONE_MAX=70nm
SLZFF(Axis) = 5000 ! Zero velocity feed-forward 5 to 15 microns before target
TARGRAD(Axis) = SLDZMAX(Axis)
RET
! ***** Load DC mode *****
DC_MOTOR_PAR:
MFLAGS(Axis).30 = 1 ! Select DC mode
XVEL(Axis)=2E8 ! Do not change this number even if encoder resolution is 0.5nm
SLVKP(Axis)=5000
SLVLI(Axis)= 100
XCURI(Axis)=100
XCURV(Axis)=100
XRMS(Axis) = 100! Limit the RMS of the driver input to 10V FULL COMMAND
TARGRAD(Axis) = 5
RET
```