High-Bandwidth Force Control

How to use Aerotech linear motors to servo on a force input/output signal from a force gage.

By Matt Davis, Aerotech, Inc.

Traditionally linear stages are used with encoders to position to precise points in a motion profile. These encoders are used for both positioning and, in the case of brushless servomotors, for motor commutation as well. However, in certain applications like material press-bonding, position is not as important as force. When binding two materials together, if a position is the target, the force applied while bonding can vary significantly. Part thickness, amount of epoxy, and stage backlash can combine to produce more or less force applied than is necessary for a complete and successful bonding process. If a force can be used in place of "position" at the servo-loop level, we can be assured of the same amount of force being applied cycle after cycle, regardless of part tolerances, to create a more uniform and successful bonding result.

The Challenge

In the following example the UUT can be moved to some load/unload position under normal positioning control mode. When the part is in position we switch to force control to either test or process the unit. During the force control portion, an exact constant force, or in some cases a precisely varying force, is needed to complete the process. We want this transition from position to force control to be smooth, and we want to be able to program the force as if it were a position.

The Solution

There are two basic methods to implement force control: an autofocus loop and a traditional PID loop. One may work better than the other depending on the application. The method used when the force loop bandwidth required is high, the traditional PID method, will be described here. The autofocus method usually has a bandwidth of about 1/5 of the position loop bandwidth, which makes the autofocus method more appropriate at a 3 Hz to 10 Hz bandwidth.

System Setup

The system is run in dual-loop mode. An encoder is used for the velocity loop and to commutate the motor. The output of a force gage is used as the positioning encoder. Aerotech's Ensemble controller can accept up to a ±10 V analog signal. This signal is read by a 16-bit ADC so we have a resolution of ±32768 counts for the system. The force gage being used in these tests outputs a ±1 V signal so the max counts we will see in the Ensemble are ±3276 counts.

When force is not being controlled, and simple motion is needed for a load/unload operation, the controller can be put back into single-loop mode programmatically by changing the position encoder to read the velocity encoder (linear encoder).

The equipment used in the corresponding tests consisted of:

- Ensemble HPe10
- Linear actuator (LMA-300) consisting of a BLM-142 linear motor with a 1 µm resolution encoder
- Omega 51-50 force gage

Parameter setup

The following parameters (Tables 1 and 2) need to be set-up to run this system.

Table 1. Force Control Settings

Parameter	Value	Explanation
PositionFeedbackChannel	0 or 1	Value equals the analog input number where the sensor is connected
PositionFeedbackType	3	Uses analog input as the feedback type
VelocityFeedbackChannel	-1	Default channel; assumes encoder plugged into feedback connector
VelocityFeedbackType	1 or 2	1 for square-wave encoder and 2 for sine-wave encoder (with MXH/MXU option)
GainKv	0	Set to 0 to divorce the position and velocity loops since they are in different units
CountsPerUnit[1,3]	calculated	32768/peak force output
FaultMask		The feedback scaling fault bit needs to be unchecked for this mode

Aerotech, Inc.
Website: www.aerotech.com
Email: sales@aerotech.com
Phone: 412-963-7470

Table 2. Position Mode Settings

Parameter	Value	Explanation
PositionFeedbackChannel	-1	Value equals the VelocityFeedbackChannel
PositionFeedbackType	1 or 2	Value equals the VelocityFeedbackType
VelocityFeedbackChannel	-1	No change from force mode
VelocityFeedbackType	1 or 2	No change from force mode
GainKv	1	Only one encoder for both velocity and position so value is 1
CountsPerUnit[2,3]	calculated	Resolution of encoder/unit of measure
FaultMask		The feedback scaling fault bit can remain unchecked as only one encoder is used

[1] CountsPerUnit is based on the ADC of the Aerotech drive. For a 16-bit analog input there is a range of ±10 V; -10 V equates to 0 counts of the ADC and +10 V equates to 65536 counts; 0 V equates to 32768 counts. The number of ADC counts per pound is calculated for this parameter. For example, a load cell that can measure up to 350 pounds at a peak output of 10 V would have a CountsPerUnit of 32768/350= 93.6229 counts per pound of force.

[2] CountsPerUnit is based on resolution and user units. For an encoder with a resolution of 0.25 µm and a metric programming unit in mm, the value of this parameter would be 1 mm/.00025 mm resolution = 4000 CountsPerUnit.

[3] CountsPerUnit changes do not take effect until a reset is performed in the Ensemble and Soloist controller. In these examples this is used to calculate out a conversion factor so that the motion commands can be scaled appropriately so that it is programmed in the new units, but using the old factor.

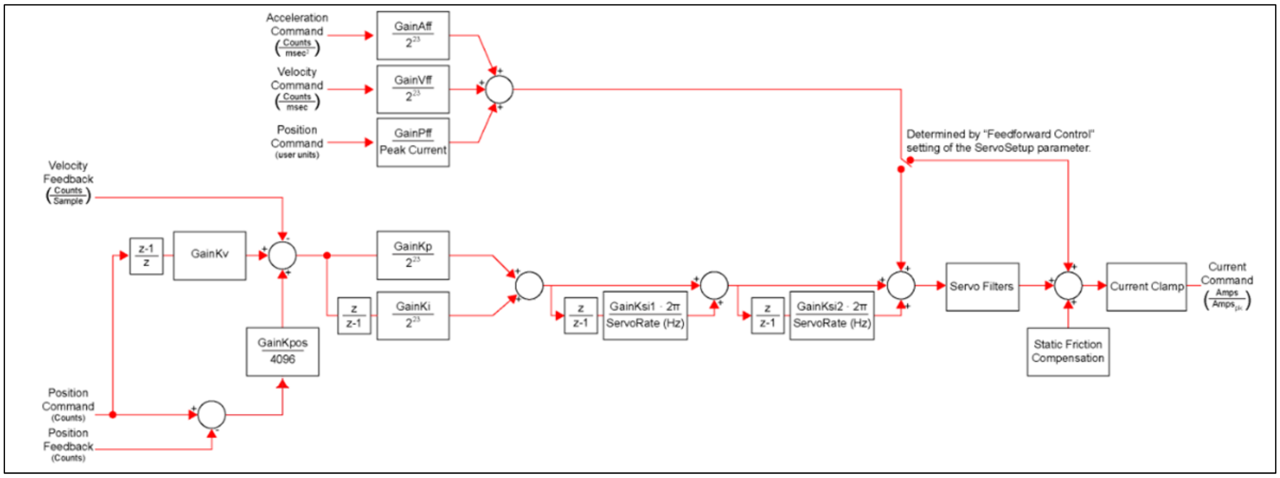

Figure 1. Servo-loop block diagram.

Tuning for each mode may be different, and can be adjusted based on force versus position control. The gains for each mode will need to be calculated separately. The Autotune can be used to set up the position control. For the force control Autotune can be used as well, or a manual step response can be used to tune and adjust the gains manually.

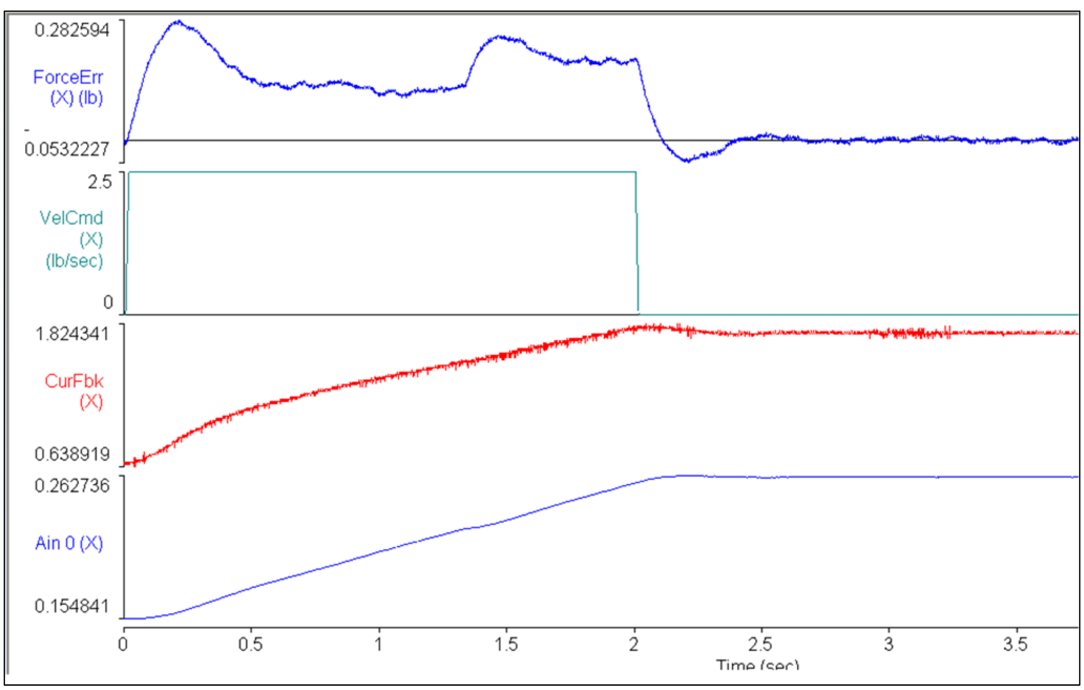

Figure 2. Step move response plot to view error during move.

Aerotech, Inc.
Website: www.aerotech.com
Email: sales@aerotech.com
Phone: 412-963-7470

Sample Code

In the program that follows the parameters are set-up in the position mode settings. This sample uses the ForceControl library. Velocity feedback type cannot be NONE in this configuration.

```
OPTION EXPLICIT
                                                        'system starts up in positioning mode with encoder control
                                                        'velocity feedback type cannot be NONE in this configuration
DECLARATIONS
                                                        'variables to record parameter file settings
    GLOBAL CountsPerMM as double                        'counts per mm from parameter file
    GLOBAL EncoderFeedbackChannel as integer            'original position feedback channel
    GLOBAL OriginalPositionFeedbackType as integer      'original position feedback type parameters used to set
                                                        force control settings
    GLOBAL CountsPerLb as double                        'A/D counts per unit (lb)
    GLOBAL ForceFeedbackChannel as integer = 0          'Analog Input used
    GLOBAL ForceFeedbackType as integer = 3             'Analog Input PosFeedbackType = 3
    GLOBAL ForceScaleFactor as double                   'scale needed to turn position units into force calculations
    GLOBAL ADCBits as integer = 1                       'A/D resolution of drive, MP AIN0 is a 12 bit input
    GLOBAL SensorRangeVolts as double = 10              'absolute sensor range in volts
    GLOBAL SensorRangeForce as double = 500             'absolute sensor range in force units
END DECLARATIONS
HEADER
    INCLUDE "ForceInclude.abi"
END HEADER
PROGRAM
    dim adscale as double
    dim sensorscale as double
                                                        'Setup Code
    CountsPerMM = GETPARM (X,CountsPerUnit)             'get original counts per unit
```

Aerotech, Inc.
Website: www.aerotech.com
Email: sales@aerotech.com
Phone: 412-963-7470

```
                                                        'get original feedback channel
EncoderFeedbackChannel = GETPARM (X,PositionFeedbackChannel)
                                                        'get original feedback type
OriginalPositionFeedbackType = GETPARM (X,PositionFeedbackType)
adscale=(2^ADCBits)/20                                  'A/D counts per Volt on ensemble
sensorscale=SensorRangeForce/SensorRangeVolts
CountsPerLb=adscale/sensorscale
ForceScaleFactor = CountsPerLb/CountsPerMM              'conversion factor to take mm to pounds
        WAIT MODE MOVEDONE
        ENABLE X
        HOME X
        DWELL .1
        SETEXTPOS X,0                                   'set ext pos register to 0 as well to correspond with home pos
        DWELL .1
        ABS
                                                        'Encoder Mode for Load position of part
        LINEAR X 133 F 50                               'Setpoint just before load is applied
        CALL ForceControlOn (ForceFeedbackChannel,ForceFeedbackType,CountsPerLb)
        DWELL 1
        CALL Force (0,1,2,ForceScaleFactor)             'axis number, force, speed in force/sec
        DWELL 2
        CALL Force (0,5,2,ForceScaleFactor)
        DWELL 2
        CALL Force (0,1,2,ForceScaleFactor)
        DWELL 2
        CALL Force (0,2,2,ForceScaleFactor)
        DWELL 2
```

```
                                                     'Switch back to Encoder Mode for unloading of part
        CALL ForceControlOff (EncoderFeedbackChannel, OriginalPositionFeedbackType,CountsPerMM)
        LINEAR X 0 F 50                              'back to home position
        SCOPETRIG STOP                               'stop data collection
END PROGRAM
```

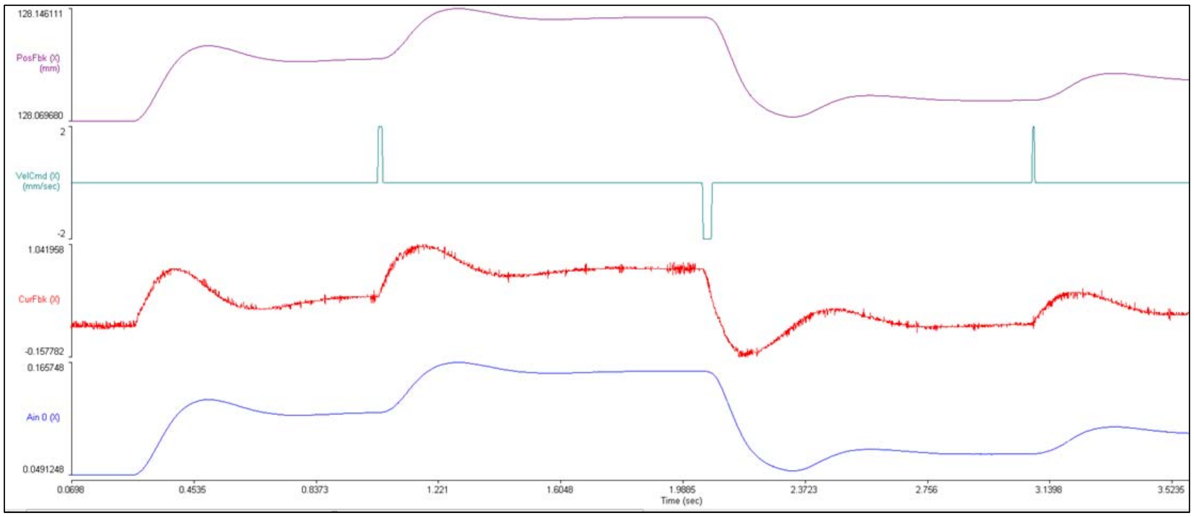

Figure 3. Force control portion of program.

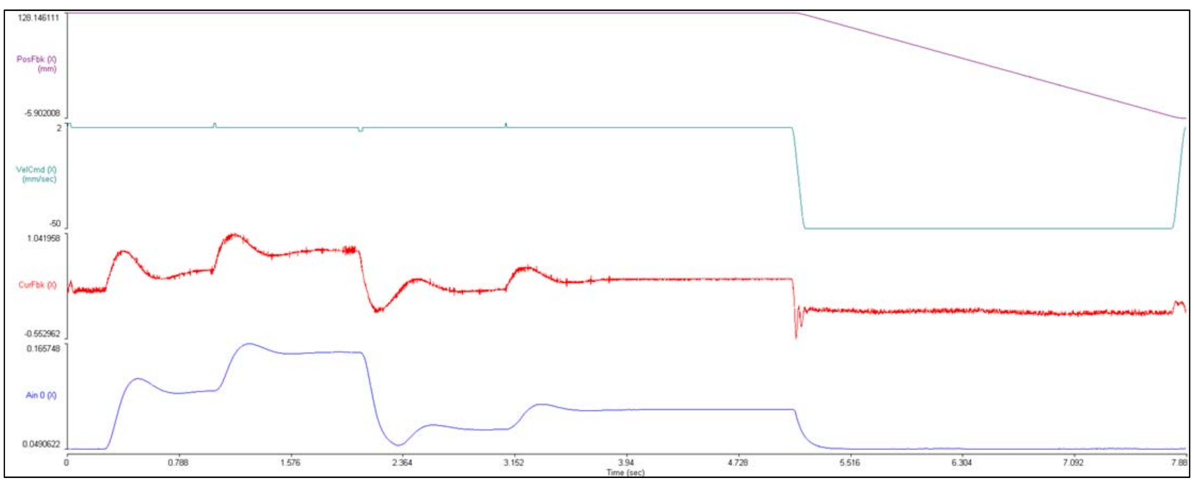

Figure 4. Force and unload moves on the same graph.

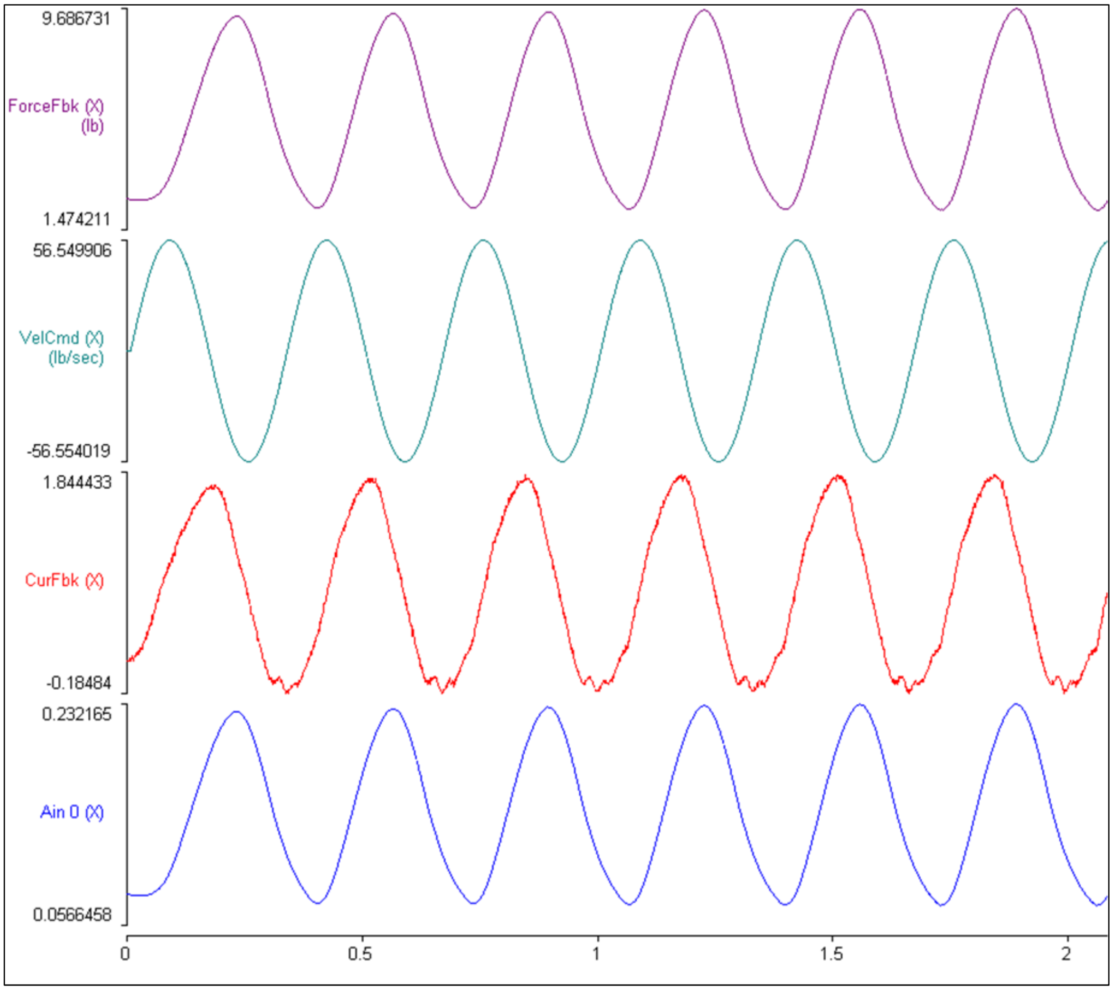

Figure 5. Plot of a 7 pound oscillation at 3 Hz.

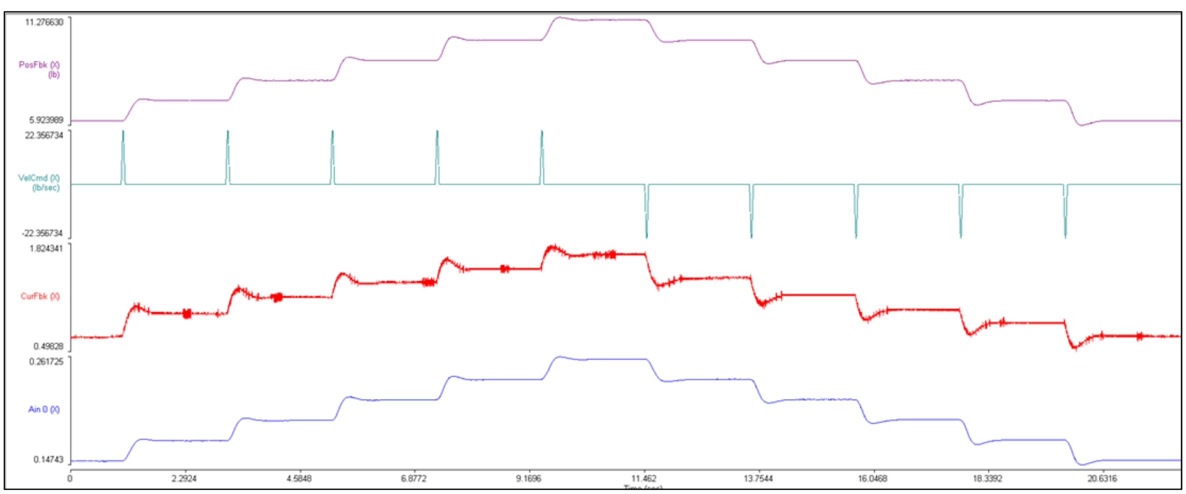

Figure 6. Plot of a 1 pound step sequence incrementing up and then back down.

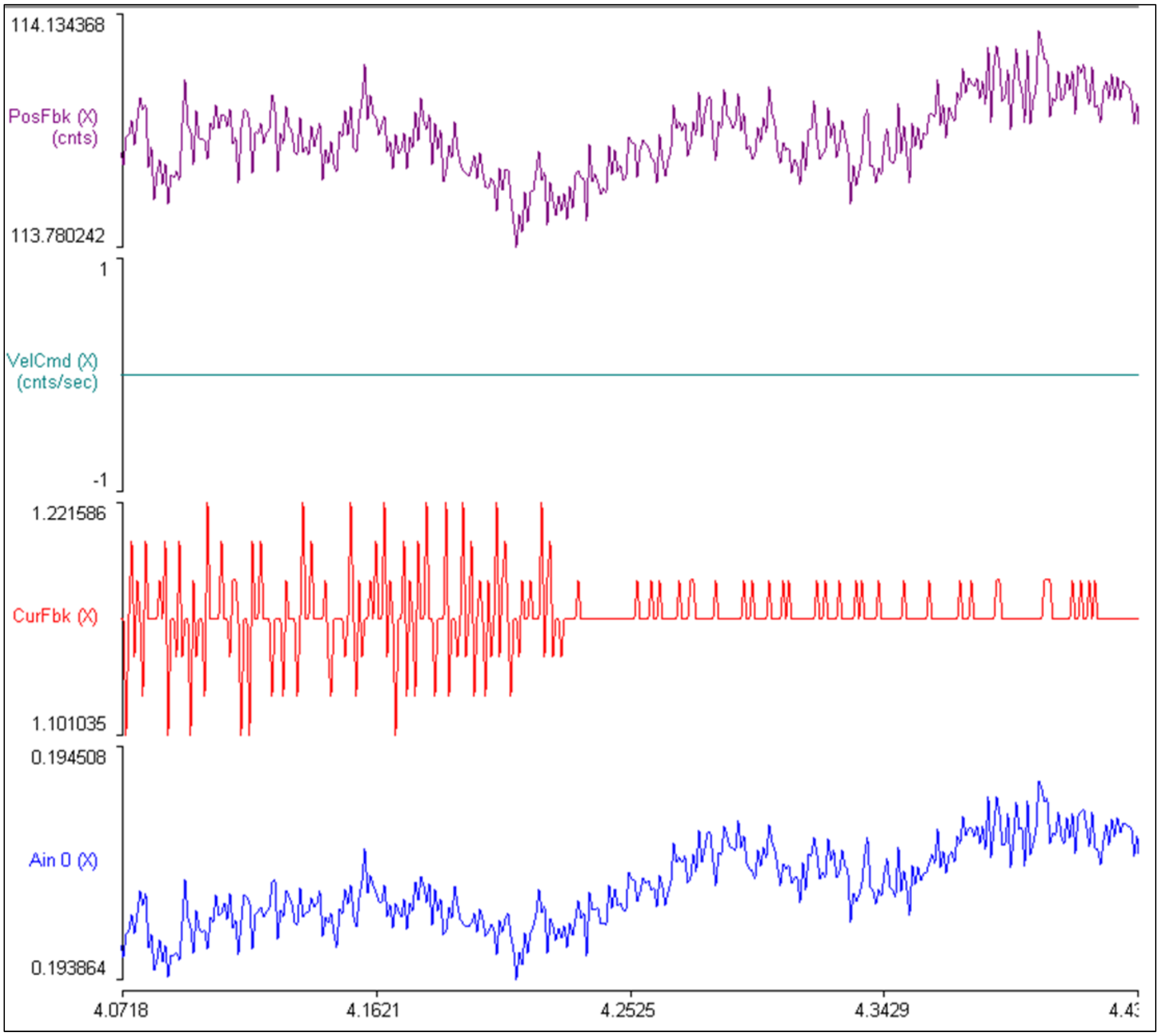

Figure 7. In-position stability of one of the steps in Figure 6, zoomed in, in AD counts.

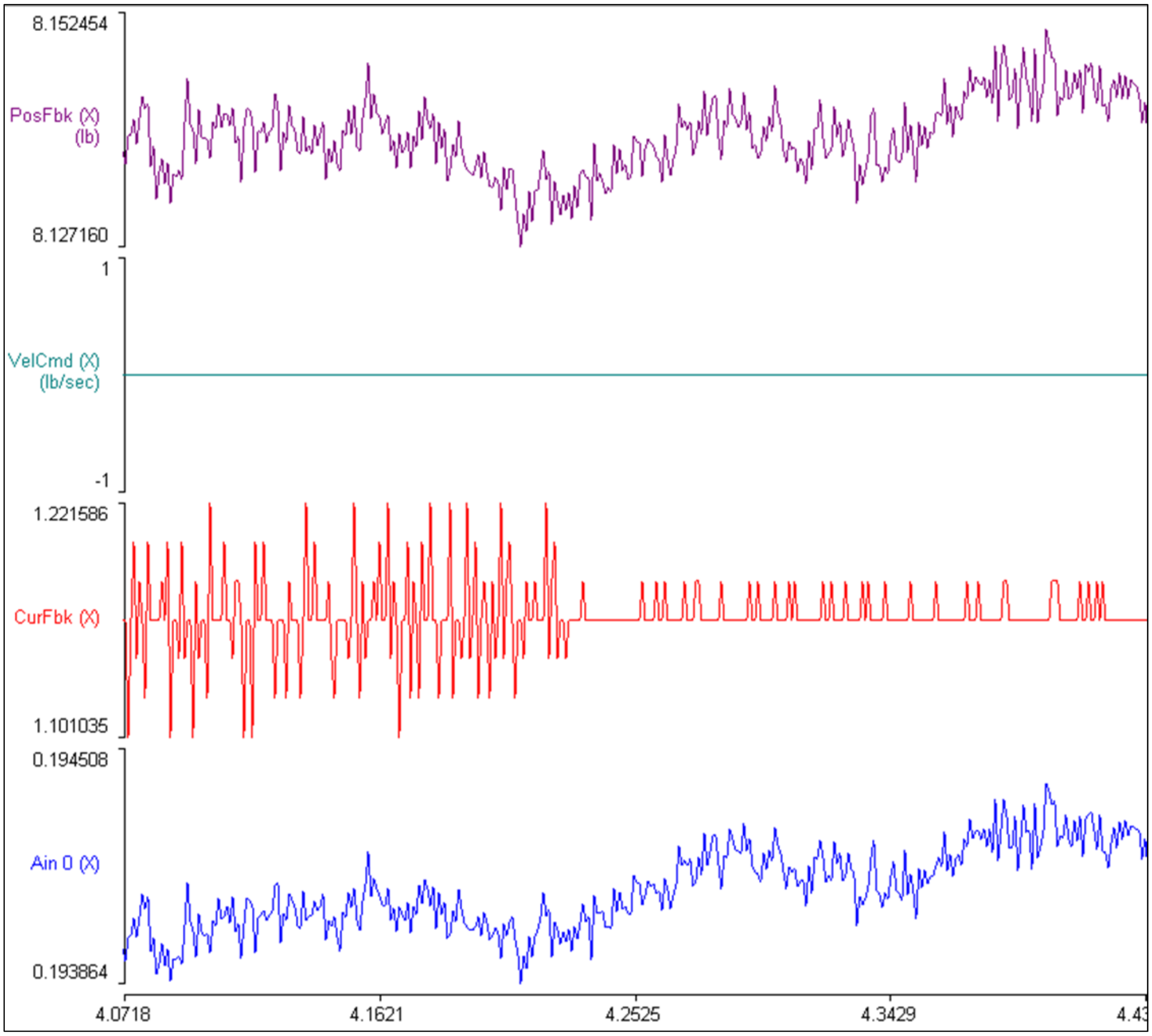

Figure 8. Same plot as Figure 7, in pounds.

As can be seen from Figure 8, the analog input is changing by 644 µV. Each AD count is about 305 µV, so we are dithering by about two counts. The resolution of the sensor will dictate how tightly you can track a given force.

The Digital Scope signal names can be changed to reflect user units. This allows the plots to better represent what is actually being measured, instead of always being in terms of volts and position units.