www.triomotion.com

Trio Motion Technology Ltd.
Shannon Way, Tewkesbury,
Gloucestershire. GL20 8ND
United Kingdom
Tel: +44 (0)1684 292333
Fax: +44 (0)1684 297929

1000 Gamma Drive
Suite 206
Pittsburgh, PA 15238
United States of America
Ph: +1 412.968.9744
Fx: +1 412.968.9746

Doc No.: TN20-100
Date: 16 July 2007
Subject: Using dual servo control loops

Application Note

Introduction:

Sometimes it is desirable to have an axis' position controlled by a remote encoder, while maintaining a "Local" feedback to help with loop stability and to prevent the axis "running away" if the remote encoder becomes disconnected. To enable this kind of operation, it is possible to set up 2 position control servos to work together controlling one motor.

An example application would be where a web is pulled through rollers, but there is not a precise grip on the material so it can slip a little and so loose position. This is solved by an external encoder touching the web and providing the accurate feedback of the web position.

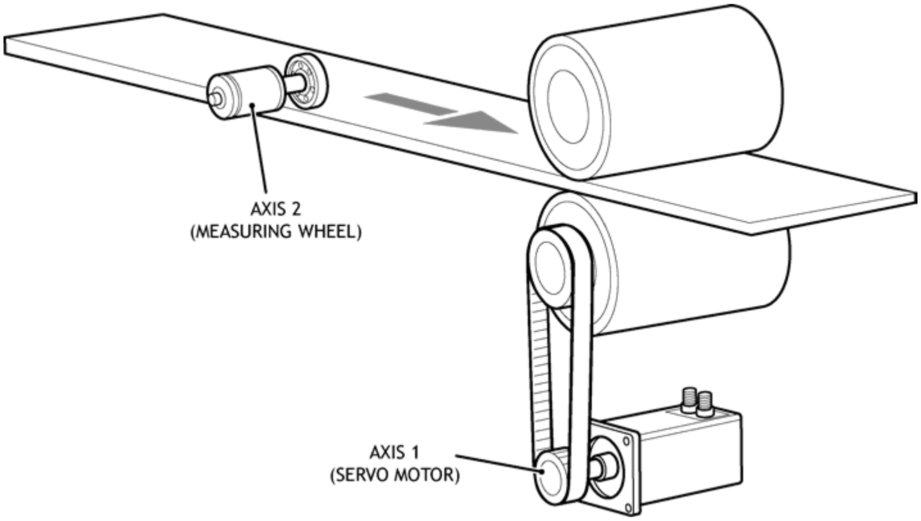

Figure 1: Web feed machine with remote encoder

Axis requirements:

The main axis must be a full servo axis. Either a Servo daughter board or a Servo Feature Code in a feature-enabled Motion Coordinator.

The second axis can be a reference encoder daughter board but the Axis Type (ATYPE) must be set to "servo". (ATYPE=2) In a feature-enabled Motion Coordinator like the MC206X, the second axis must be a full servo axis to allow the use of ATYPE=2. The reference encoder axis of the MC206X, axis 4, can be enabled for this use by the addition of the correct Feature Code.

Both the main axis and the second axis must have the same calibration. i.e. the number of encoder counts per unit length (mm or inches) MUST be the same. If this is not possible by selecting a suitable encoder, then the ENCODER_RATIO command must be used to set one of the axes so that they can have the same UNITS value.

Example axis initialisation:

```
BASE(1) 'main axis has 320 counts per mm
UNITS = 320

BASE(2) 'second axis (encoder) has only 255.25 counts per mm
ENCODER_RATIO(255.25, 320)
UNITS = 320
```

Connecting the 2 loops:

The loops are connected together inside the Motion Coordinator by use of the ADD_DAC and ADDAX commands. ADD_DAC routes the output of the second axis to be added to the DAC_OUT value of the main axis. This means that the speed command sent to the servodrive is the sum of the control signal from the main axis and the DAC_OUT from the second axis.

ADDAX causes the second axis to follow the same position and velocity profile demands as the main axis. Here is an example where axis 1 is the main axis and axis 2 is the second axis.

```
BASE(1)
ADD_DAC(2) 'axis 2 output will be summed with axis 1 signal to servodrive

BASE(2)
ADDAX(1) 'axis 2 position demand will follow the axis 1 demand
value
```

After the connection is made, the 5 servo control terms from each axis are available for tuning the dual loop system. See figure 2 for details of the control loop structure.

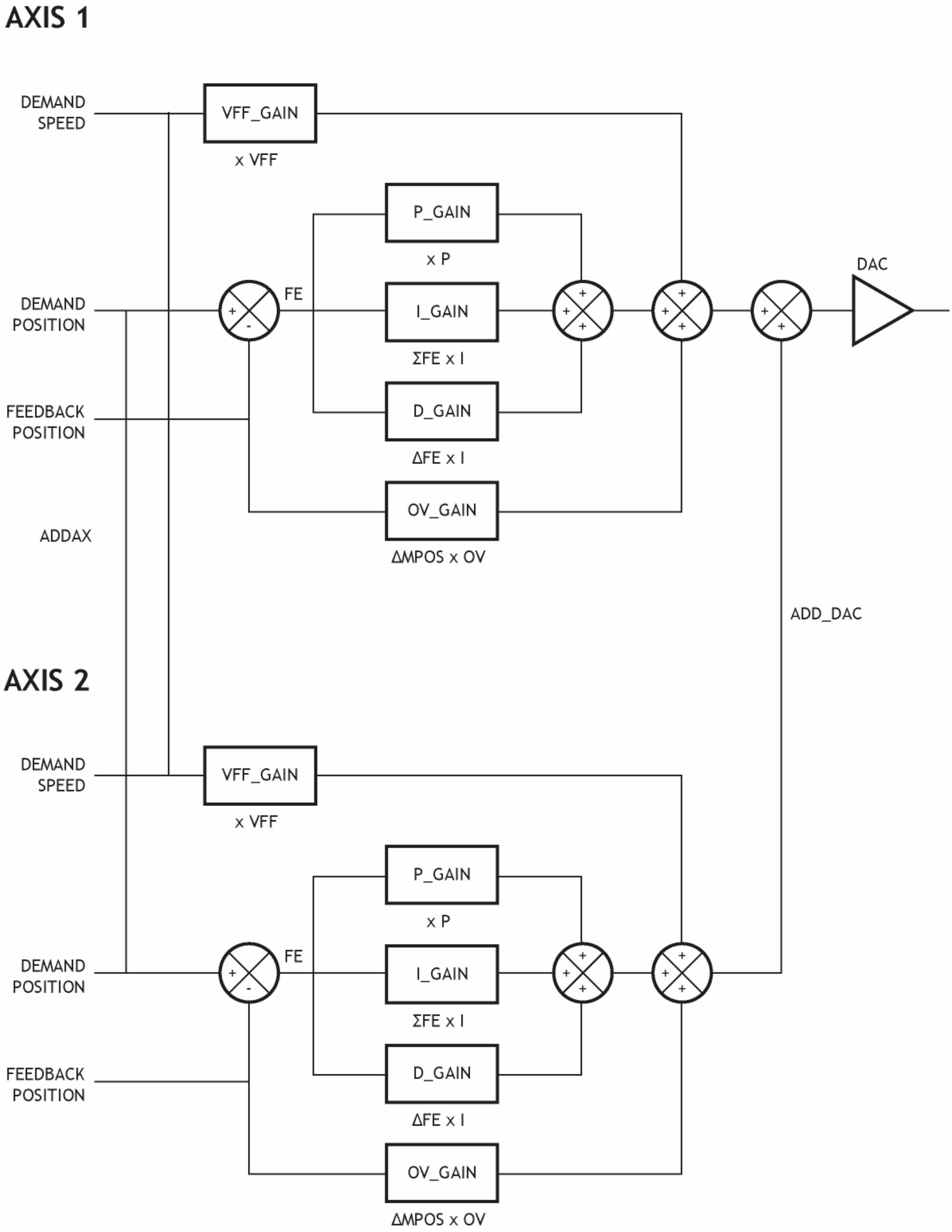

Figure 2: Dual servo loop structure

Tuning the control loops:

Tuning the dual loop is of course more complex than tuning just a single control loop. The starting point is to consider only the Servomotor axis, axis 1 in the examples above. Set all the gains for axis 2 to zero, including the P_GAIN, and tune axis 1 using P_GAIN only. Now set the VFF_GAIN to minimise the following error. (FE)

Once the main axis is tuned, there are some options to consider before deciding on the strategy for the secondary axis.

1. Is the motion continuous with very few stops during a day's operation?
2. Is the motion start-stop but by operator control so that it could stop in any part of the machine cycle. It is stopped maybe 2 or 3 times an hour.
3. Is the motion start-stop and defined in a repeating cycle so that the BASIC program has full control of the places where it stops?

The reason that we need to consider these, is that the 2 control loops will tend to get out of synchronisation and over a long period there could be very different values in the feedback position of the two axes. The different feedback values will lead to bigger and bigger FE values on one of the axes. In a cyclic machine it is easy to include some commands to remove these differences and maintain the 2 axes in sync with each other. On a continuous machine, a different strategy must be evolved.

Axis 2 with P_GAIN only.

To set up axis 2 with P_GAIN tuning only, first decide what proportion of the control will be assigned to the main axis and how much is to be assigned to the secondary, encoder, axis. For example, it could be that the main axis should have only a weak influence on the motion, just enough to maintain control if the encoder comes away from the web and the secondary feedback is lost. For example, the main axis has 10% of the power to control position and the encoder axis has 90%.

```
‘ Tuning of P_GAIN on axis 1 on its own was found to be 1.26
‘ and the VFF_GAIN was found to be 15.5.

BASE(1) ‘ main axis
P_GAIN = 0.126 ‘ 10% of tuned gain
VFF_GAIN = 15.5
BASE(2) ‘ Encoder axis
P_GAIN = 1.134 ‘ 90% of tuned gain
VFF_GAIN=0.0
```

With such a P_GAIN only approach, the FE of the main axis will grow bigger and bigger as the motor and the encoder positions disagree. This FE must be "zeroed" from time to time, otherwise it will affect the overall position accuracy. Zeroing the FE on the motor axis can be done with the SERVO command:

```
BASE(1)
SERVO=OFF
WA(2)
SERVO=ON
```

This short stopping of the servo operation will cause a small movement of the axis and is best done during a pause in the motion of the machine.

Axis 2 with I_GAIN.

If I_GAIN is used on the secondary axis, it has the effect of removing the FE completely, even if the main axis P_GAIN is kept at the maximum. However, the use of I_GAIN has all the dynamic side effects that make PID tuning in general quite a tricky thing to do. D_GAIN would need to be introduced to dampen the overshoots and instability caused by the integrator. For example, a dual loop system might be set up like this:

```
BASE(1) ‘ main axis
P_GAIN = 1.26 ‘ 100% of tuned p gain
D_GAIN = 9.78 ‘ damping applied to the main axis
VFF_GAIN = 15.5
BASE(2) ‘ Encoder axis
P_GAIN = 0.0
I_GAIN = 0.0246 ‘ integral gain applied to secondary axis
D_GAIN = 0.0
VFF_GAIN = 0.0
```

The same problem applies to this combination of gain as we have with P_GAIN only. Over time the feedbacks will disagree and the Integrator will wind up to a bigger and bigger value. However, the advantage this time is that the true position will be maintained; i.e. it will hold the position given by the secondary encoder. The axes will still need to be reset from time to time, but perhaps less often. When the reset is needed, BOTH axes must be reset like this:

```
SERVO AXIS(1)=OFF ‘ disable the main axis
SERVO AXIS(2)=OFF ‘ disable the secondary axis to zero the integrator
WA(2)
SERVO AXIS(1)=ON ‘ restart the main axis
SERVO AXIS(2)=ON ‘ restart the encoder axis
```

This is likely to cause a much bigger disturbance than the reset process done with P_GAIN only. The I_GAIN will have a disturbing effect unless the feedbacks are perfectly aligned.

Axis 2 with full authority (zero P and I in main axis)

In machines that run continuously, there will not be a convenient point to reset the axes’ synchronisations. So only one axis can have full control of the servo loop and that axis must be the encoder wheel, secondary axis. However, we need to provide stability for the motor in case the wheel comes away from the web and the feedback is temporarily lost. This can be done by adding D_GAIN to the main axis:

```
BASE(1) ‘ main axis
P_GAIN = 0
D_GAIN = 9.78 ‘ damping applied to the main axis
VFF_GAIN = 15.5
BASE(2) ‘ Encoder axis
P_GAIN = 1.26 ‘ 100% of p gain on secondary axis
I_GAIN = 0.0246 ‘ integral gain applied to secondary axis (optional)
D_GAIN = 0.0
VFF_GAIN = 0.0
```

As there are no P or I gains in the main axis, the absolute FE value of this axis has no effect on the control. Therefore there is no need to stop and reset this kind of loop.

Example configuration program:

This program is an example of the full set up and tuning of a dual axis servo system as described above. The machine works continuously, so the option with P_GAIN in the secondary loop and D_GAIN in the main axis has been chosen.

```
BASE(1) ‘main axis has 490 counts per mm
UNITS = 490
ADD_DAC(2)

BASE(2) ‘second axis (encoder) has 370.915 counts per mm
ENCODER_RATIO(370.915, 490)
UNITS = 490
ADDAX(1)

BASE(1) ‘ main axis
P_GAIN = 0.0
I_GAIN = 0.0
D_GAIN = 14.55 ‘ damping applied to the main axis
VFF_GAIN = 23.178

BASE(2) ‘ Encoder axis
P_GAIN = 0.985 ‘ 100% of p gain on secondary axis
I_GAIN = 0.0 ‘ no integral gain
D_GAIN = 0.0
VFF_GAIN = 0.0
```