SERVO**STAR**® S600 Electronic Gearing and Software User Interface

This Product Note describes how gearing works on the SERVO**STAR**® S600. It begins with a general discussion of gearing and continues with details on hardware and software set up. This document only describes the software setup for gearing using the Software User Interface for the SERVO**STAR** S600.

How Electronic Gearing Works

Electronic gearing provides a position command to a drive that depends on the motion of another device. That device is typically an encoder (or its equivalent) connected to a motor. However, it can also be a hand-driven encoder used to accept input from an operator or an output from a stepper motor controller. This is different from internal profile generation where position commands are calculated inside the drive based on motion parameters, such as distance, acceleration, and velocity.

The external encoder signal provides an input to the gearing software. In many applications, the motor follows the input, step for step where one revolution of the input device should create one revolution in the electronically-geared motor. This is called one-to-one gearing.

Other times, it is necessary to have a non-unity scale factor. The gear ratio may be a non-integer, such as 1:3. It is important that the gearing software not introduce round-off errors when the gear ratio is non-integer because over long periods of time, the round-off error accumulates and causes the electronically-geared motor to drift with respect to the input. The SERVO**STAR** S600 supports non-integer gearing ratios while remaining free of round-off errors.

Gearing Applications

There are several ways to use electronic gearing in applications. They vary depending on the source of the input signal and how the gear ratio is set. However, they all share the structure where the electronically-geared drive controls the motor to follow the input.

Following a Non-Driven Encoder

Some gearing applications require that a servomotor follow a non-servo-driven encoder. The encoder may be on an idle roller of a conveyor belt or in a line-driven induction motor. The encoder is connected to an axis that is not under the direct control of another drive.

DANAHER
MOTION
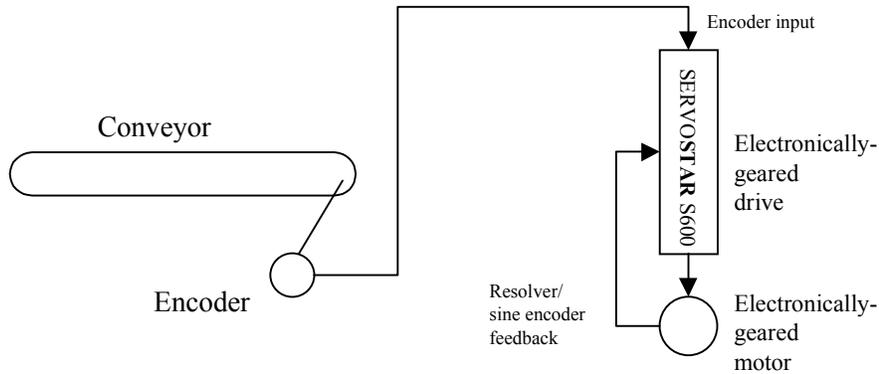
Figure 1: Following a Non-driven Encoder

Following a Servomotor

Some gearing applications need one servomotor to follow another. Here, the encoder from the first (master) motor becomes the gearing input to the second (slave) motor. In this case, the master encoder must connect as both the feedback for the master drive and input for the slave drive.

For cases where the feedback device of the master servomotor is something other than a physical encoder, the SERVO**STAR** S600 provides encoder emulation. For example, neither a sine encoder nor a resolver directly produce signals compatible with encoders. The SERVO**STAR** S600 encoder emulation circuit produces output signals equivalent to an encoder from any feedback device that is used as the main feedback device. This allows any SERVO**STAR** S600 to serve as the master for any other electronically-geared SERVO**STAR** S600.
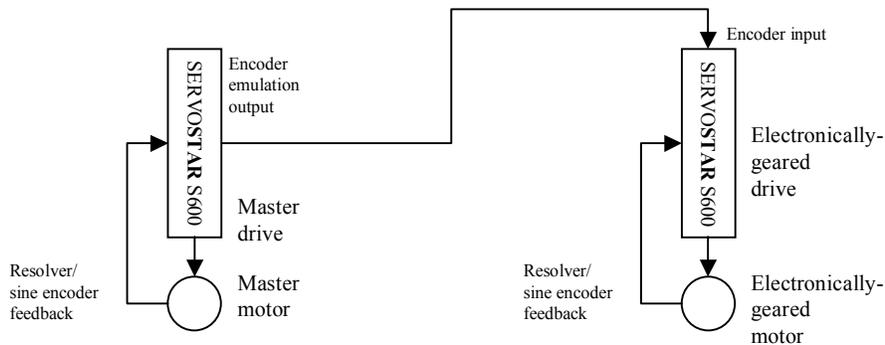
Figure 2: Following a Servo-driven Encoder

Following a Multi-Pulse Generator (MPG)

Some applications require a motor to follow the input from an operator via a multi-purpose generator (MPG), such as an encoder handwheel. In many cases, a traditional contact-based potentiometer is used to take input from the operator. However, such a device is not appropriate for some environments because the contact surface wears over time and becomes unreliable. By using electronic gearing, an optical encoder is used to replace a contact potentiometer. The SERVO**STAR** S600 accepts the input from an encoder as either a velocity or position command.
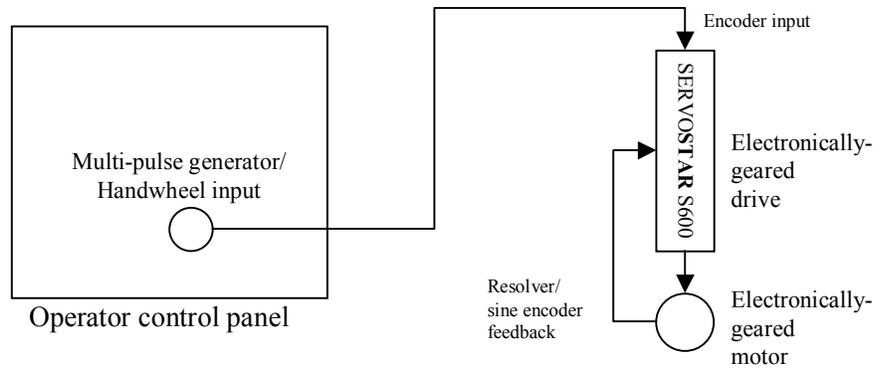
Figure 3: Following a Potentiometer

Following a Pulse/Direction Signal

For some applications, it is useful for a servomotor to follow a pulse or direction input. For example, pulse and direction inputs are generated by some stepper motor controllers. By using electronic gearing, a SERVO**STAR** S600 is controlled by a stepper motor or similar controller.
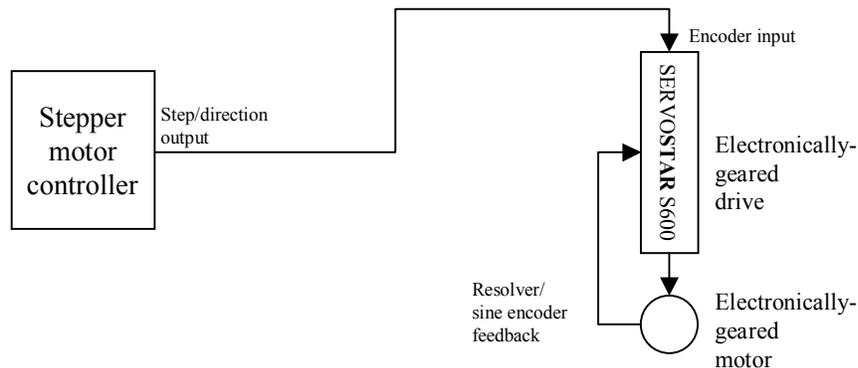
Figure 4: Following a Stepper Motor Controller

Adjusting the Gear Ratio in Realtime

Some applications require that the gear ratio be controlled in realtime. An example is a web-tensioning application processing soft material. The material is stretched across two rolls. The faster the outtake roll spins, the greater the tension held across the web. The SERVO**STAR** S600 provides several ways to adjust this gear ratio. Figure 5 shows a control scheme where one analog input commands speed of the master axis and the slave axis follows that command according to the gear ratio. The ratio is adjusted in realtime by the slave's analog input.

Document Number: P-SS-004-09 Rev 5   Old Numbers (AS6007H, A-SS-004-20208)
Revised 11/20/2003   www.DanaherMotion.com
Page 3
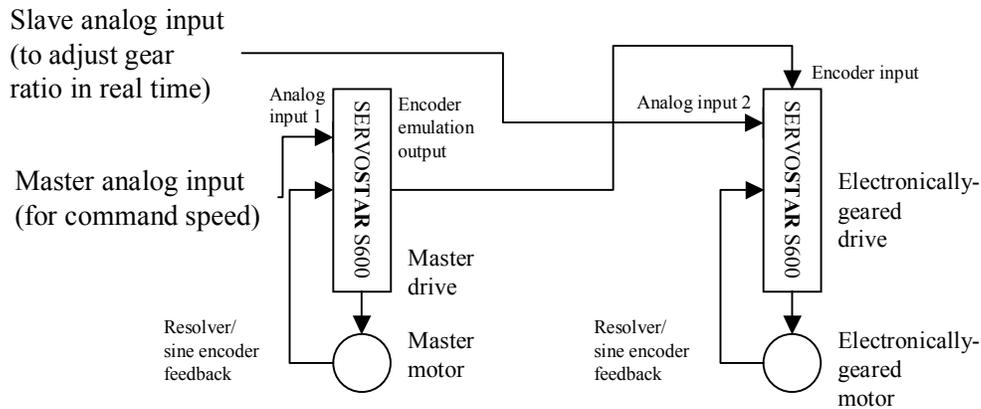
Figure 5: Realtime Adjustable Gear Ratio for Web-Handling

Following Error

The electronically-geared motor follows the input as well as the servo tuning allows. If a servo loop is poorly tuned so that the geared motor is sluggish, the motor will not follow the input signal well. It is important to tune the geared motor with servo gains as high as possible to minimize this problem. High values of velocity feed-forward (GPFFV) are also helpful in reducing following error. In fact, if GPFFV is set to 1.0, all following error is eliminated during steady-state operation. However, such large values of GPFFV can cause overshoot of rapidly-accelerating commands, so you may not be able to set the value that high.

The distance between the command generated from the encoder input and the electronically-geared motor is called position error (PE) or following error. You can view the value of this variable using the **Monitor** dialog box. The higher the servo gains, the less following error. In addition, the closer GPFFV is to 1, the lower the following error. If the following error exceeds the value of PEMAX, the drive generates warning N03.

Hardware Setup

Hardware setup is primarily selecting and wiring the pulse input signal. If you need to modify the gear ratio in realtime, you need to determine the appropriate hardware and software inputs.

Determine the Encoder Type

Encoder Voltage

The first step to selecting the encoder is determining the voltages. The SERVO**STAR** S600 accepts signals with three voltage levels:

- 0 to 5 VDC via Connector X5
- 0 to 24 VDC via Connector X3
- 0 to 1 volt via Connector X1 (not available when primary feedback sensor is a sine encoder)
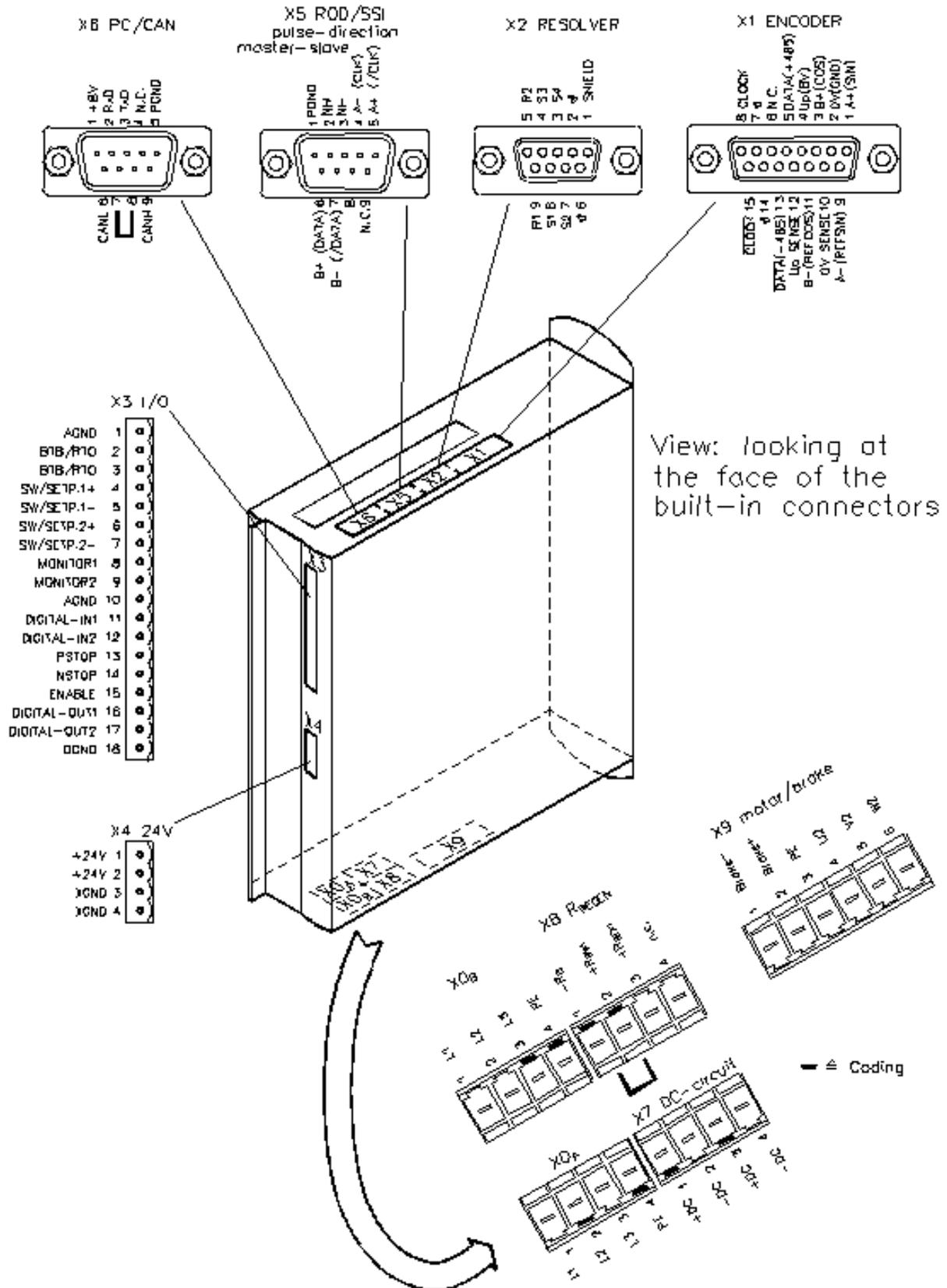
Figure 6: Connector Locations and Pin-outs

Most optical encoders provide 0 to 5 VDC signals. The encoder emulation output from another (master) SERVO**STAR** S600 are also 0 to 5 VDC signals and should be brought into Connector X5.

> *A SERVOSTAR **S600** cannot simultaneously take 0 to 5 VDC connections into Connector X5 and provide encoder emulation outputs because both signals share the same pins on Connector X5. If you are using encoder emulation on a given axis, the encoder input for gearing will either be 0 to 24 VDC (Connector X3) or sine encoder levels (Connector X1).*

> *There are frequency limitations on each voltage level. As this time, 0 to 5 V signals are limited to 1 MHz. The 0 to 24 VDC signals are limited to 100 kHz. Older drives (those that have serial numbers below 220,000), are limited to 10 kHz. The sine encoder signals are limited to 125 kHz.*

Encoder Formats

The SERVO**STAR** S600 supports three formats for the electronic gearing input. If you are using a digital (non-sine) encoder, you can choose between digital quadrature format and step/direction format. If you are using a sine encoder, you must use the sine encoder format.

- **Digital Quadrature Format**
  Digital quadrature format is a 2-bit Gray code. There are two channels, identified as Channels A and B. Both channels produce square waves (one shifted 90° from the other). This is shown in Figure 7. The digital quadrature format can be used for 0 to 5 VDC inputs and for 0 to 24 VDC inputs.
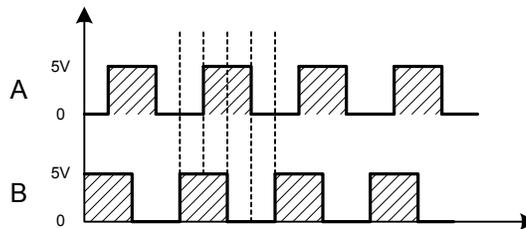
Figure 7: Digital Quadrature Encoder Format

- **Step/Direction Format**
  Step/direction format uses one channel for input pulses and the other to indicate direction. When the direction line changes state, the input command reverses direction. This is shown in Figure 8. The step/direction format is used for 0 to 5 VDC inputs and for 0 to 24 VDC inputs.
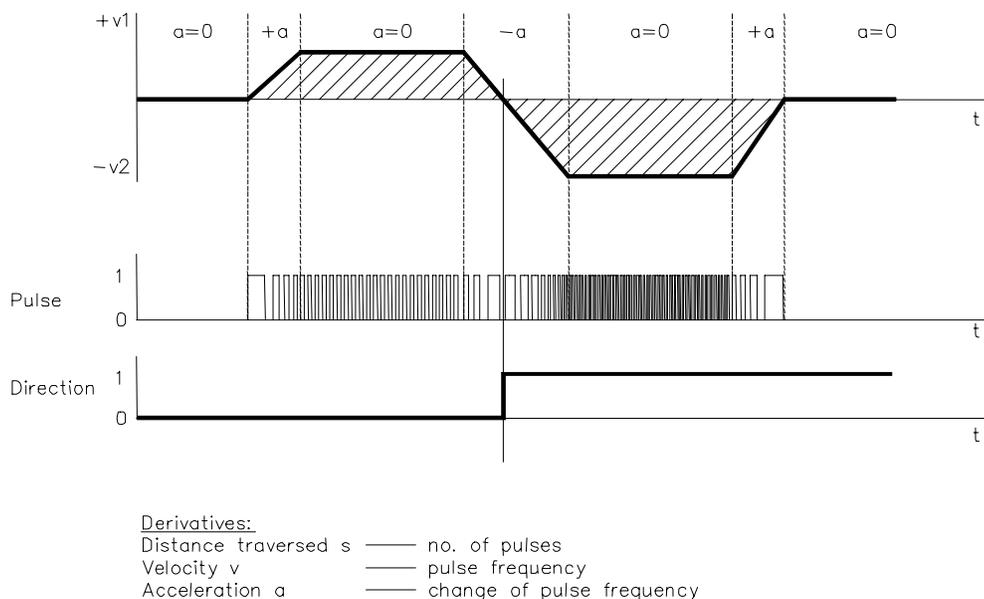
Figure 8: Step/Direction Encoder Format

Some encoders provide a single channel output. If you are using such an encoder, use the step/direction format. Connect the output channel to the step input and tie the direction channel either high or low, as is appropriate for the application requirements.

- **Sine Encoder Format**
  Sine encoders provide two channels in quadrature. This is similar to the digital-quadrature signals except the sine-encoder signals are continuous rather than being limited to two discrete values. This allows the sine-encoder format to provide much greater resolution, as shown in Figure 9. The sine encoder format is only used with the sine-encoder inputs. The sine encoder format for the gearing input is not available when the drive uses a sine encoder as the primary feedback device.
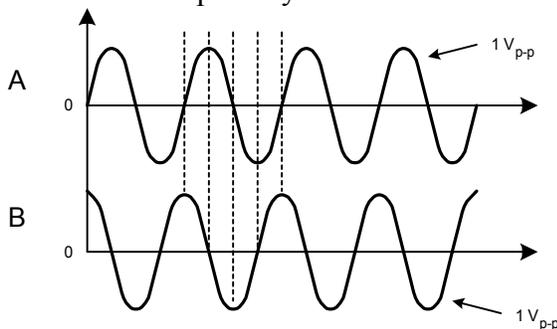
Figure 9: Sine Encoder Format

Wiring the Encoder

Consult the *SERVOSTAR® 600 Installation Manual* for details on connector pin-out and wiring practices.

Multiple Slaves

One master signal can drive many slaves. The limitations are that the master must have the current available to drive all inputs. Refer to the manufacturer's information on the encoder to ensure that sufficient current is available to supply multiple electronically geared drives. If a SERVO**STAR** S600 is being used as the master, it can drive up to 16 slave axes.

Encoder emulation output — Master drive — SERVO**STAR** S600

Encoder input — Slave drive #1 — SERVO**STAR** S600

Encoder input — Slave drive #2 — SERVO**STAR** S600

Encoder input — Slave drive #3 — SERVO**STAR** S600
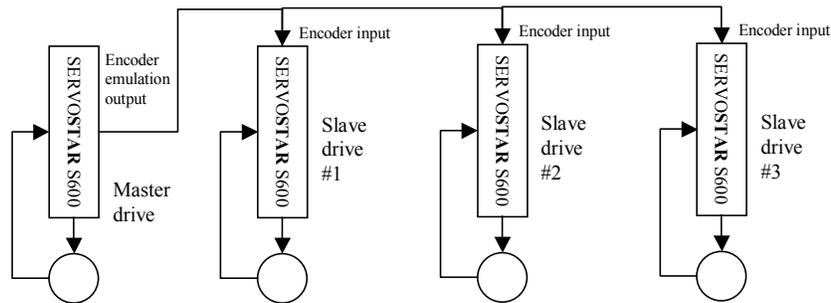
Figure 10: Multiple Slaves

Sometimes is it desirable to connect electronically-geared drives in a chain. In this configuration, a single SERVO**STAR** S600 is master to one axis and slave to another. This configuration is supported in the SERVO**STAR** S600 if the drives use sine encoders. Because a single SERVO**STAR** S600 cannot have encoder emulation and 0 to 5 VDC encoder input operating simultaneously, it is not possible to chain drives using the 0 to 5 VDC inputs. Contact customer support for more information regarding this configuration.

Hardware for Changing the Gear Ratio

If you plan to change the gear ratio in realtime, you must select the source for the input.

- **Analog Input (Connector X3)**
  Using the analog input, you can scale the gear ratio. Usually the range of the gear ratio is from zero to two times the nominal gear ratio. However, using the analog input offset, you can shift the range up or down. The analog input is the fastest way to change the gear ratio, allowing up to 4000 adjustments per second. However, analog inputs are subject to noise. This is avoided by using a field bus or serial port.

- **Field Bus Input**
  Using the field bus input (PROFIBUS, CAN bus, etc.), you can change the gear ratio settings to any value. Using a field bus, you can change the gear ratio approximately 50 to 100 times per second.

- **Serial Input**
  Using the serial port, you can change the gear ratio the way you can with a field bus. However, this method is slow compared to the other alternatives. Typically, you can change the gear ratio 20 to 30 times per second with the serial port.

Software Setup

The following section describes the software setup for gearing using the Software User Interface for the SERVO**STAR** S600. The Software User Interface provides a user-friendly graphical interface and terminal for the setup, monitoring, and troubleshooting of the SERVO**STAR** S600. When setup is complete, save your settings to the EEPROM. Otherwise, all settings are lost at power down or during a "COLDSTART."

Selecting the Operational Mode

From the **OPMODE** drop-down box in the **Amplifier** dialog box, select **4. electr. Gearing**. Alternatively, type **OPMODE = 4** in the terminal window.
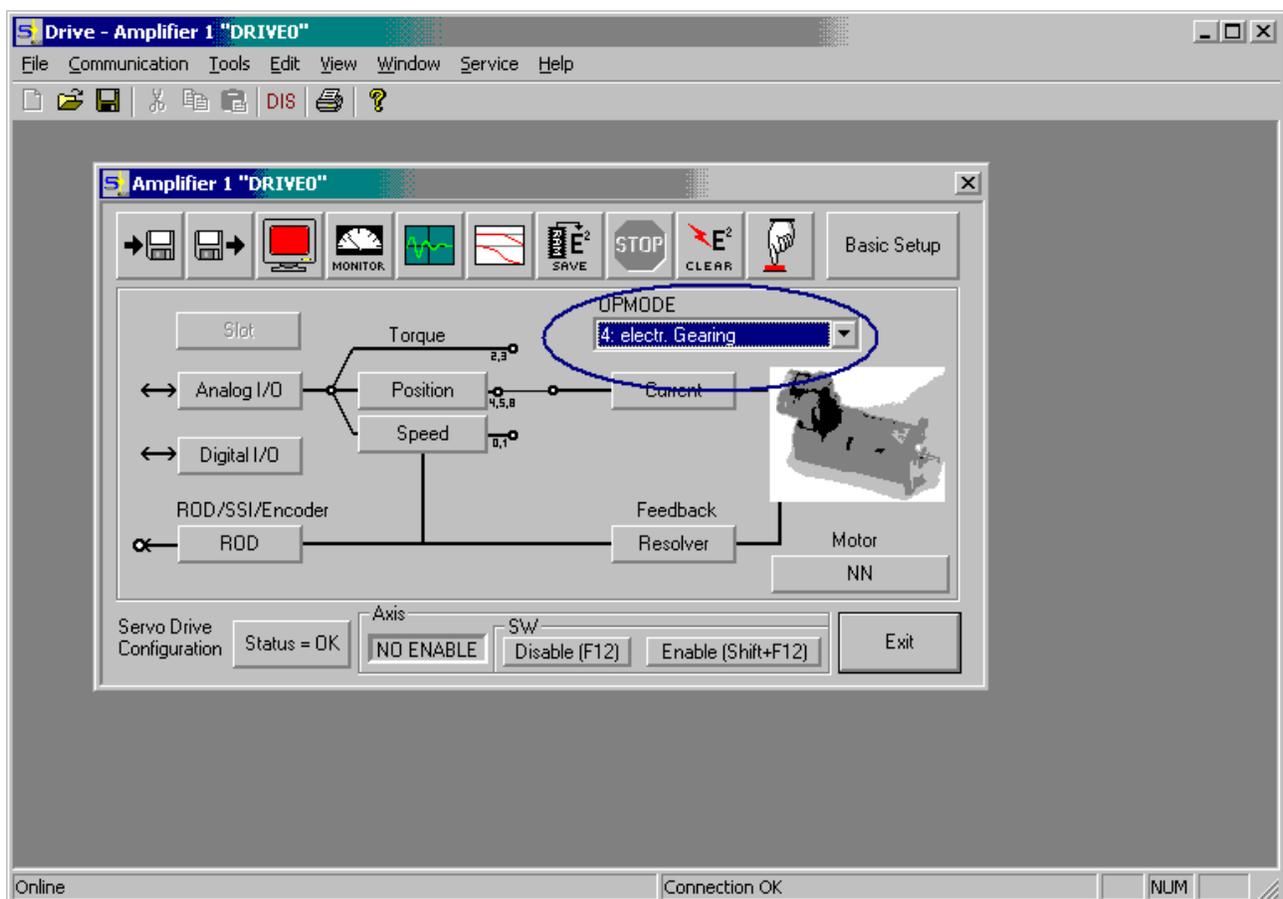
Figure 11: Amplifier Dialog Box: Selecting the Operational Mode, Electronic Gearing

Selecting the Gearing Mode (GEARMODE)

Select the gearing mode (GEARMODE) based on the hardware used for the encoder input. Currently, there are six choices:

GEARMODE	Encoder Input Voltage Levels and Format
0	Digital quadrature format with 0 to 24 VDC Inputs (Connector X3).
1	Step/direction format with 0 to 24 VDC Inputs (Connector X3).
2	Reserved
3	Digital quadrature format with 0 to 5 VDC Inputs (Connector X5). DO NOT USE if the drive outputs encoder emulation signals.
4	Step/direction format with 0 to 5 VDC inputs (Connector X5). DO NOT USE if the drive outputs encoder emulation signals.
5	Reserved
6	Sine encoder (Connector X1). DO NOT USE if a sine encoder is used for the encoder primary feedback device.
7	SSI format (Connector X5).

You may enter the setting for GEARMODE directly in the **Gearing** dialog box (Figure 12). See ascii.chm for details on more advanced settings of GEARMODE.
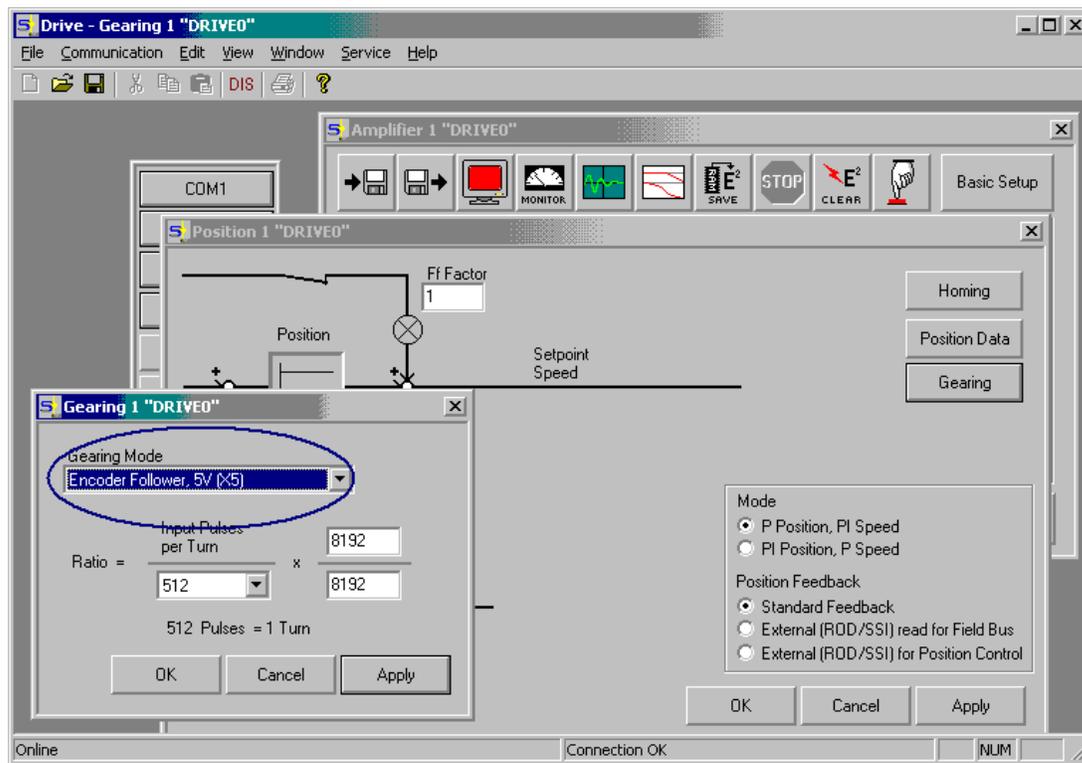
Figure 12: Selecting the Gearing Mode

To access the **Gearing** dialog box, click **Position** in the **Amplifier** dialog box (Figure 11). Click **Gearing**, located near the bottom-right of the displayed **Position** dialog box (Figure 12). From the **Gearing** dialog box, select the appropriate value of GEARMODE from the **Gearing Mode** drop-down box at the top of the window. Alternatively, type a GEARMODE value (**GEARMODE = 3**, for example) in the terminal window.

Selecting Encoder Emulation Mode (ENCMODE)

Select the encoder emulation mode (ENCMODE) based on the encoder input voltage level. If you are using 0 to 5 VDC signal levels (Connector X5), you must select ENCMODE = 0. If you are not using 0 to 5 VDC signal levels, this setting does not effect gearing. To set the encoder emulation mode, click **"ROD/SSI/Encoder"** on the **Amplifier** dialog box (Figure 11). From the displayed **Encoder** dialog box (Figure 13), select **Input** from the **Encoder Emulation** drop-down box. Alternatively, type **ENCMODE = 0** in the terminal window.
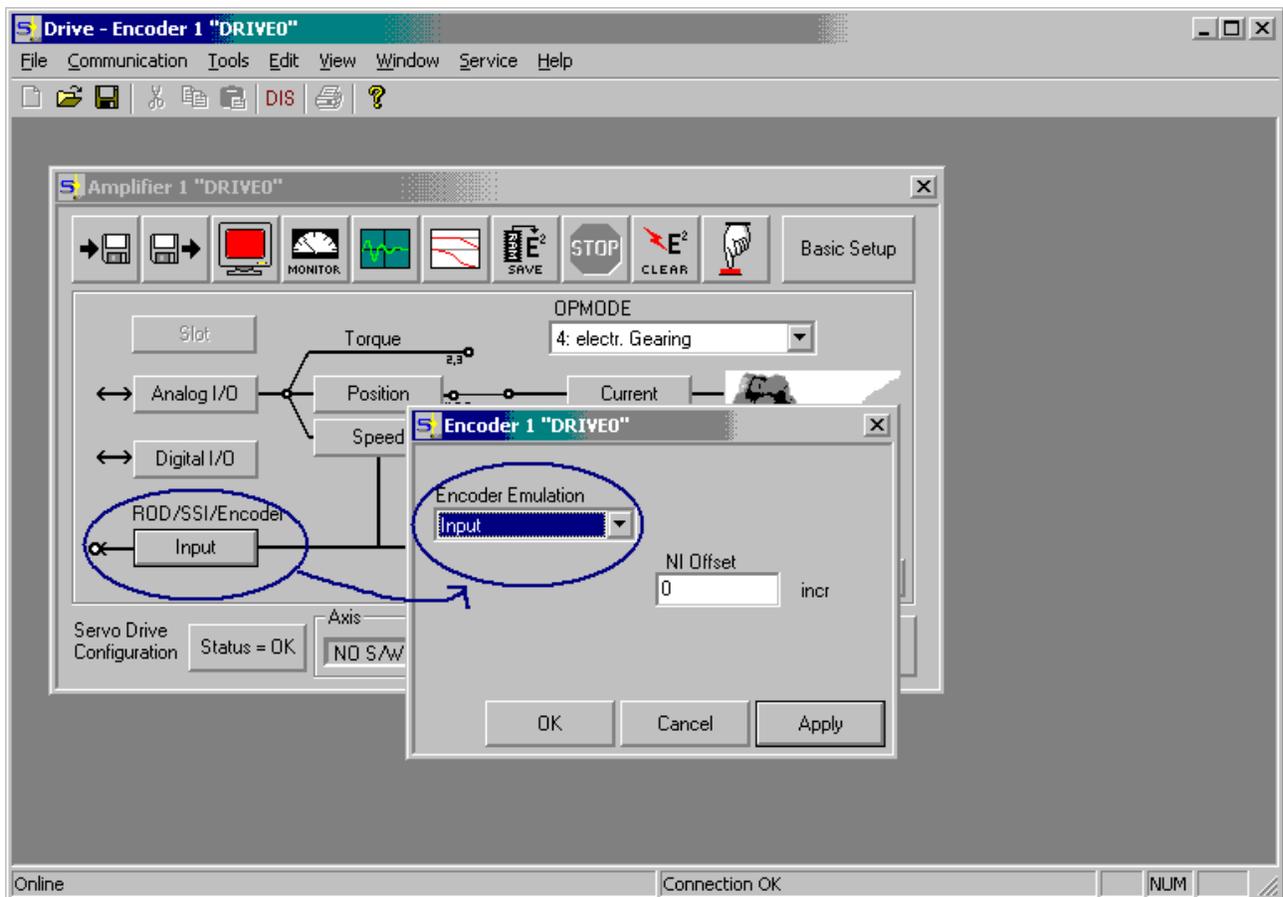
Figure 13: Emulation Mode (ENCMODE)

Selecting Velocity or Position Control (EXTPOS)

Select whether you want the encoder input to represent a position command or a velocity command. For most applications, the setting is a position command. In some applications, the encoder input may represent a velocity command. The most common example of this is when a multi-pulse generator (MPG) is used by a machine operator to command speed.

Most applications require position control. Select this by typing **EXTPOS = 0** or **EXTPOS = 4** in the terminal window. EXTPOS = 0, the default value, configures the system for a PI position loop and EXTPOS = 4 configures the system for a P position loop. If your application requires velocity control, enter **EXTPOS = 1** in the terminal window. Setting EXTPOS to 2 (to allow recording of the external position under the variable, **PFB0)** causes electronic gearing to operate incorrectly.

Setting Input Encoder Resolution

You must enter the resolution of the encoder in lines per revolution. If the resolution of the encoder is binary or an even power of 2, you can enter the value directly in the **Gearing** dialog box (Figure 12). Select the resolution of your encoder from the "**Input Pulses per Turn**" drop-down box at the center of the window. See Figure 14. Alternatively, type the input encoder resolution (**ENCIN = 512**) in the terminal window. After changing ENCIN, you must save all settings to the EEPROM and perform a "COLDSTART."
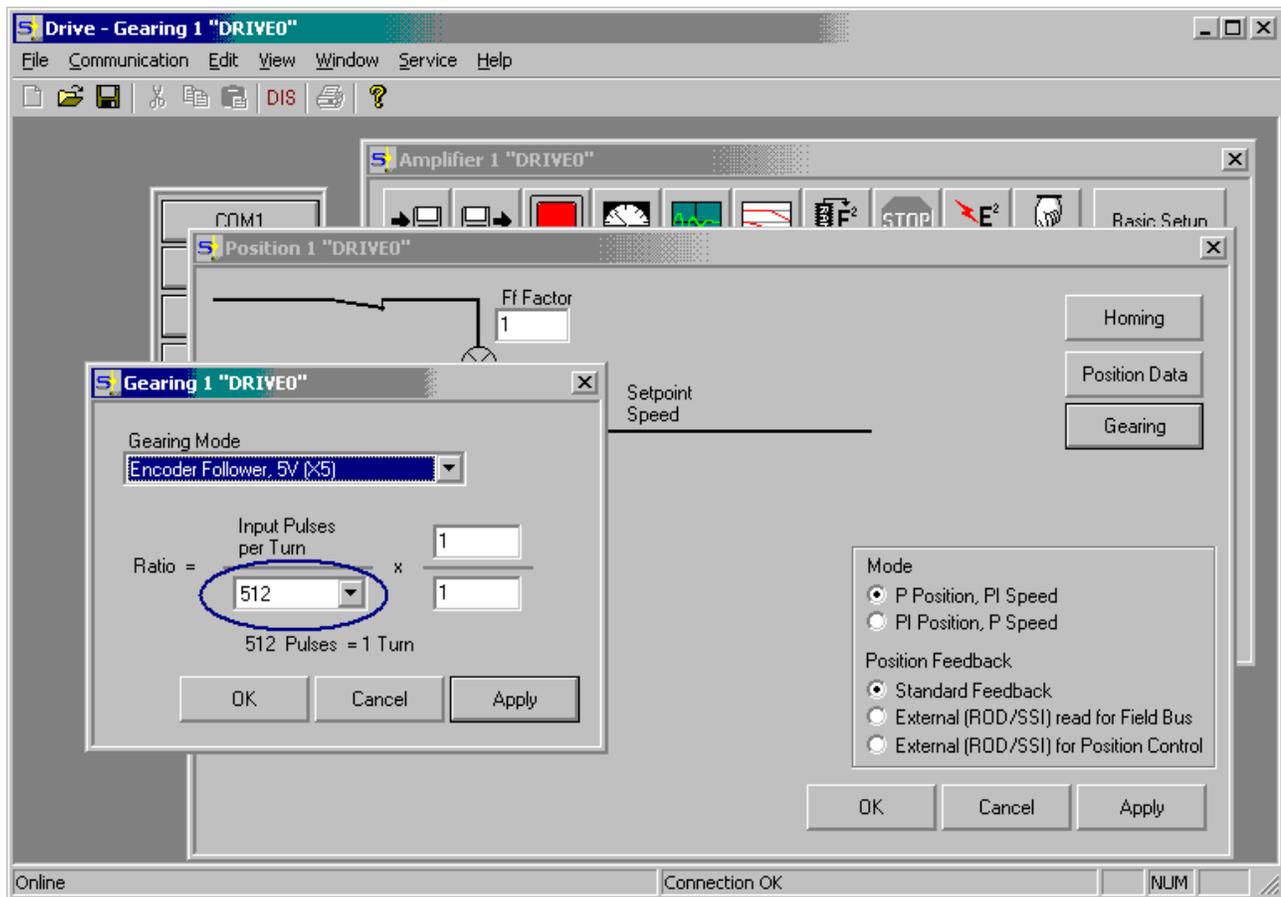
Figure 14: Setting input encoder resolution

Whether you use the **Gearing** dialog box or the terminal window, you can set the encoder resolution only to one of eight values: 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, and 65536. If your encoder has non-binary resolution (for example, 1000 lines/revolution), select the resolution closest to your encoder. You can adjust the gear ratio to correct for the difference of the selected value and the actual resolution.

Selecting the Gear Ratio

The gear ratio is defined by the ratio of two variables, GEARI and GEARO. This allows the accurate entry of non-integer ratios, such as 1:3. To understand how GEARI and GEARO set the ratio, imagine a pair of gears with the number of teeth on each gear defined by these variables. The gear ratio is:

$$VELOCITY_{SLAVE} = VELOCITY_{MASTER} * (GEARI / GEARO.)$$

The units of velocity are RPM. The values of GEARI and GEARO are somewhat arbitrary because the gear ratio is set by the ratio of these two variables. For example, setting GEARI and GEARO to the same values generates 1:1 gear ratio.

To set the gear ratio, type the ratio in the two text boxes at the right part of the **Gearing** dialog box (Figure 15). The example in Figure 15 shows the settings for a gear ratio of 1:2 (**GEARI = 1** and **GEARO = 2**), assuming the input encoder resolution (ENCIN) is **512** lines/revolution. In Figure 15, the window displays the gear ratio in counts by calculating it from ENCIN, GEARI, and GEARO. In this example, **512 Pulses** from the input encoder generates a command equal to one revolution, as shown at the bottom of the **Gearing** dialog box.
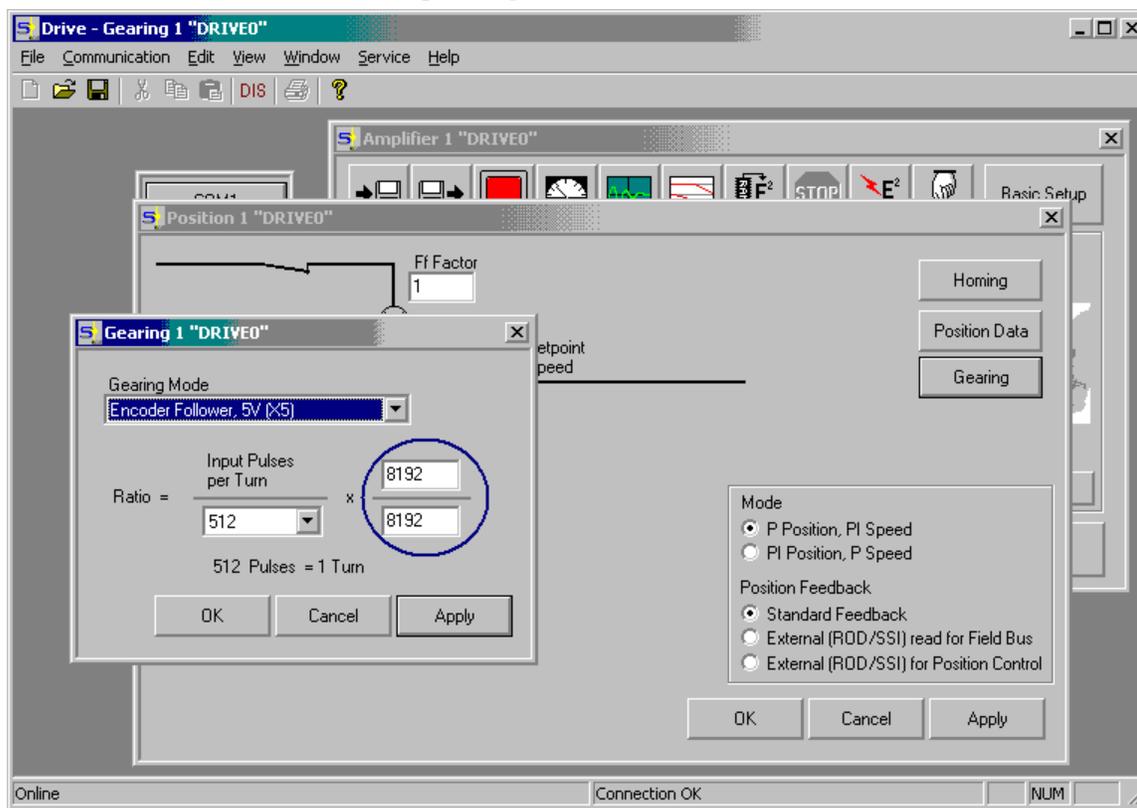
Figure 15: Setting the Gear Ratio

This example assumes that the input encoder resolution (ENCIN) accurately represents the input encoder resolution. If not, the formula must be modified:

$$\text{VELOCITY}_{\text{SLAVE}} = \text{VELOCITY}_{\text{MASTER}} * (\text{GEARI/GEARO}) * (\text{Encoder resolution/ENCIN})$$

For example, suppose you have a 500-line encoder which you want to electronically gear at 1:1. You would select ENCIN as 512, the closest allowable setting to 500 and adjust the gear ratio to correct for the difference between 500 and 512. For example, you could set GEARI = 8192 and GEARO = 8000, as in Figure 15. The programming window shows the input pulses per revolution as the desired 500 (just above the **OK** button).

The SERVO**STAR** S600 does not accumulate round-off error, even when the gear ratio is non-integer. It uses exact integer math, keeping track of the remainder. This eliminates round-off error from accumulating, even after many revolutions.

The use of a large electronic gear ratio usually produces noisy operation because a single pulse from the master produces a burst equivalent to multiple pulses on the electronically-geared drive. The higher the servo gains of the geared drive, the more the system demonstrates this problem. Avoid gear ratios much greater than one, especially on highly responsive systems. If you are using a multi-pulse generator (MPG), be sure the resolution of the generator is fine enough that the gear ratio is not much more than one.

Realtime Adjustments

You can adjust the gear ratio of the SERVO**STAR** S600 in realtime. Use caution when changing the gear ratio on-the-fly as this could produce unexpected results.

Realtime gear ratio variation is accomplished by modifying the constants GEARI or GEARO while the machine is in operation. The commands to change these constants are communicated via the serial port or via a field bus connection, such as CANopen or PROFIBUS. You cannot use gearing via SERCOS interface$^{\text{TM}}$.

You can adjust the gear ratio using the analog input to modify the gear ratio. In this case, GEARI/GEARO defines the nominal gear ratio and the analog input acts as a modifier to provide the effective gear ratio. The analog input can raise or lower the gear ratio

Select the Analog Configuration (ANCNFG = 6)

To use analog gearing, set the analog configuration to 6. To do this, click **Analog I/O** on the **Amplifier** dialog box (Figure 11). From the displayed **Analog I/O** dialog box (Figure 16), select **6. Analog Gearing** from the **Setp. Function** drop-down box. Alternatively, set ANCFG to 6 in the terminal window. Changing ANCNFG requires that you save to the EEPROM and perform a "COLDSTART" or cycle power.
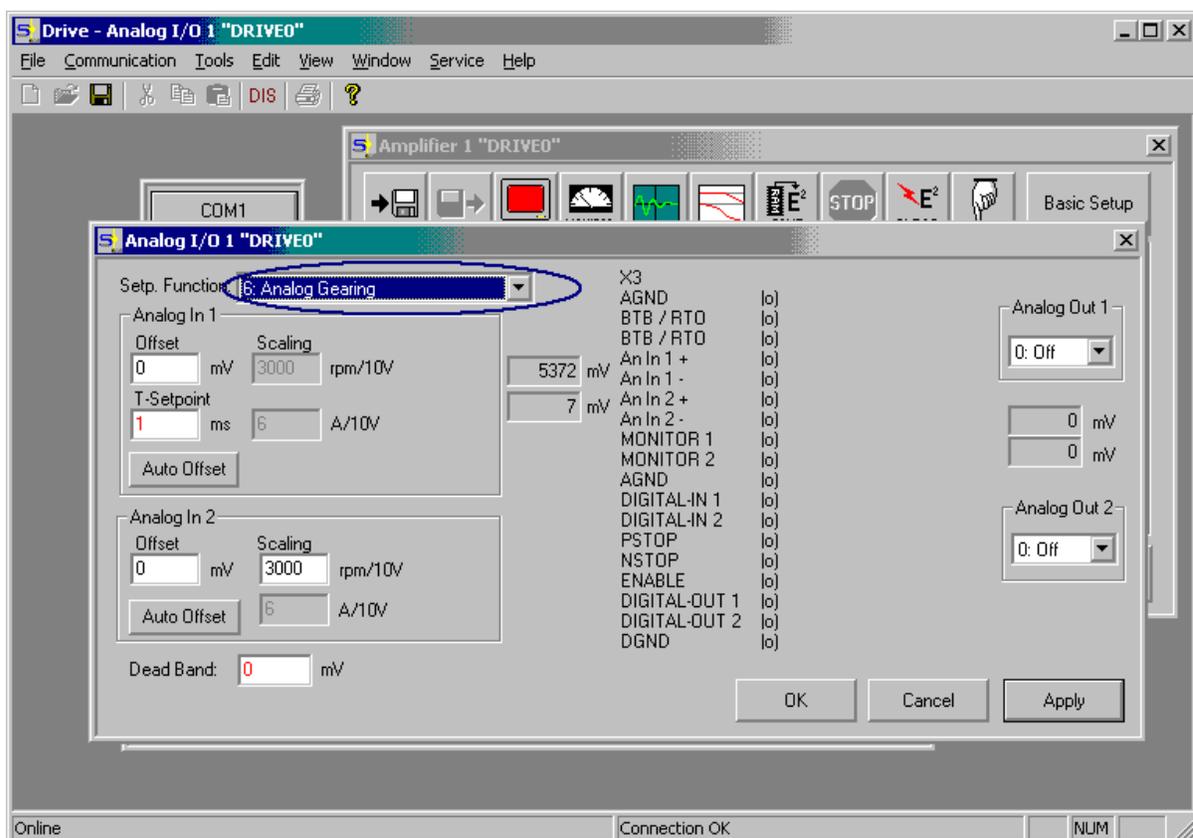
Figure 16: Analog I/O Setup

When using analog gearing, the gear ratio is set to:

VELOCITY$_{SLAVE}$ = VELOCITY$_{MASTER}$ * (GEARI/GEARO) * (Encoder resolution/ENCIN)
* (1 + (VSCALE2/100) * (ANIN2 – ANOFF2)/10.0V).

Where ANIN2 is analog input two, ANOFF2 is the programmed offset of analog input 2, and VSCALE2 is the velocity scaling for analog input two. All three variables are accessible from the bottom left of Analog I/O Setup screen (Figure 16). For example, assuming a zero offset (ANOFF2=0) and nominal VSCALE2 (100), the analog input can vary the gear ratio from the nominal GEARI/GEARO to a value as high as 2 * GEAR/GEARO (ANIN2 = +10.0 V) and as low as 0 (ANIN2 = -10 V). An analog input of 0 produces the nominal gear ratio.

> ***The current scaling of analog input 2 (ISCALE2) has no effect on the effective gear ratio. The maximum value of VSCALE2 is limited because the value of VSCALE2 x (GEARO/100) must remain less than 8191. High values of GEARO amplify this problem. If you need to extend the range of analog gearing, try reducing GEARI and GEARO by the same ratio. This allows for larger values of VSCALE2 without changing the nominal gear ratio.***

Set the Offset of Analog Input 2 (ANOFF2)

By using the analog offset (ANOFF2), you can shift the range across which the gear ratio is modified. For example, setting ANOFF2 = 5000 mV, an input of –5V generates the nominal gear ratio, +10 V generates 1.5 * GEARI/GEARO, and –10 V produces a ratio of 0.5 * GEARI/GEARO. By setting the offset to a negative value, cause the gear ratio to reverse directions (i.e., changes in the analog input causes motor direction to reverse).

Synchronization (Simple PLL)

The SERVO**STAR** S600 supports synchronization between drives. This function acts like a phase-locked loop (PLL) that is sampled every 250 microseconds. The function relies on any one of the digital inputs to indicate the command needs to be adjusted by a fixed amount. If the input is high, a fixed offset is added to the drive command; if it is low, the offset is not added.

To configure a drive for this synchronization, select the input (1 through 4). Using the terminal screen, set the input mode for that input to 36 (for example, "IN3MODE = 36"). Place the desired offset in the input's trigger variable (for example, IN3TRIG = 208). Once every 250 microseconds, this input is checked. Each time it is high, the value of the input trigger variable is used to increment the command position. The scaling for the synchronization input is set so that 2^{20} = 1 revolution of the motor if PRBASE = 20 and 2^{16} = 1 revolution if PRBASE = 16.

In general, the value of the trigger is set as:

INxTRIG = n x 250 / 60

Where n is the desired maximum offset speed in rpm. For example, if you wanted a maximum offset of 50 rpm, set INxTRIG = 50 x (250/60) = 208.

PRBASE

PRBASE must always be either 16 or 20. You can check the value of PRBASE in the terminal window. Normally, PRBASE is 20. The only exception is if your system needs to keep track of absolute motor position for more than 2048 revolutions. If the value of PRBASE is 16, electronic gearing may produce a rough output.

FPGA

FPGA selects different functions upon power up. The default value is FPGA = 0. This variable must be set to FPGA = 1, the digital encoder input (X5). Use the terminal screen to check and set FPGA.

Document Number: P-SS-004-09 Rev 5   Old Numbers (AS6007H, A-SS-004-20208)
Revised 11/20/2003   www.DanaherMotion.com
Page 16

Warning N03: Contouring Error

When gearing is enabled you may encounter Error N03. This is a "contouring" error, indicating that the position error is too large. This causes the drive to stop and remain enabled. In this case, reset the position error with either the command CLRFAULT or by using any digital input. To use a digital input, set the input mode of that input (INxMODE) to 14.

If you want this condition to disable the power stage, redefine the warning with the terminal command, WMASK = 4. If the error occurs, the power stage is disabled and the error messages F24 and N03 appear on the display. Contact the factory for more detail.

The following are several solutions for this condition:

- Increase the maximum following error, PEMAX. See the *Setup Software Manual* and the *SERVOSTAR® 600 VarCom* for more information.

- Increase servo gains.

- Reduce the acceleration rate or top speed of the input encoder.

- For motor-driven encoders, properly tune the master drive. Overshoot and ringing from a poorly-tuned master can create large excursions on the command, which the electronically-geared motor may not be able to follow.

Rollover

When the input encoder rotates enough that the position word that holds the encoder input overflows, a condition called "rollover" occurs. Because some applications require that the input encoder rotate the same direction indefinitely, even very large position words are subject to rollover over for long periods of time. The SERVO**STAR** S600 avoids rollover problems by using the encoder counts on a cycle-by-cycle basis. The only limitation is that the number of counts over a 250 µs period must be less than 8192.

Troubleshooting

If gearing is not working properly, the problem could be in one of two places: 1) getting the master signal into the drive and 2) calculating the geared command. Most problems are getting the master signal into the drive. The internal variable DF3MASTER allows you to monitor the gearing master signal. Using the terminal screen in "DRIVE.EXE," type "M DF3MASTER;" this prints the gearing master signal in hex. If the value changes by more than 1 count, then a master signal is being received.

```
                        ┌─────────────┐
                        │Gearing works?│ ──yes──▶  ( )  END
                        └─────────────┘
                              │ no
                              ▼
  Not changing or changing        ┌─────────────┐      Changing by more
  by one count        ───────────▶│ M DF3MASTER │─────  than 1 count
                                  └─────────────┘
              │                                          │
              ▼                                          ▼
      ┌──────────────┐                          ┌──────────────┐
      │    Check     │                          │Ensure drive is│
      │  GEARMODE    │                          │   enabled    │
      └──────────────┘                          └──────────────┘
              │                                          │
              ▼                                          ▼
      ┌──────────────┐                          ┌──────────────┐
      │    Check     │                          │Ensure gear ratio is│
      │  ENCMODE     │                          │     > 0      │
      └──────────────┘                          └──────────────┘
              │
              ▼
      ┌──────────────┐
      │Check FPGA = 1│
      │(if using X5 as│
      │master signal)│
      └──────────────┘
              │
              ▼
      ┌──────────────┐
      │Check encoder │
      │signal emulator,│
      │i.e. master signal│
      └──────────────┘
```