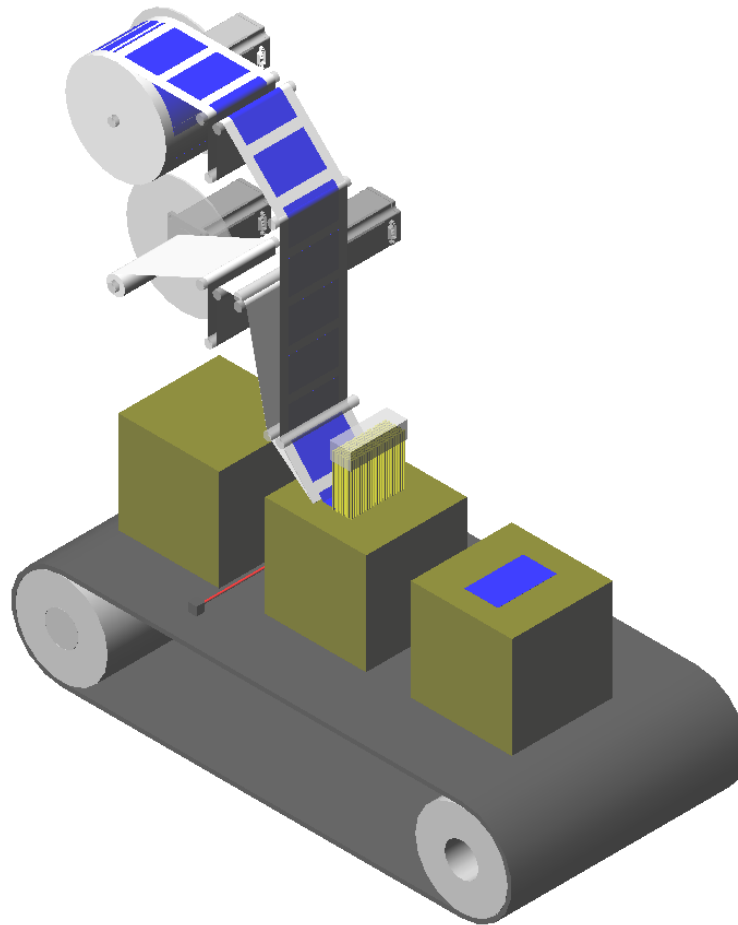


Label Applicator



Associated QuickControl® programs included:
 LabelApplicatorSD.qcp
 LabelApplicatorSN.qcp

This application note describes how to implement a label applicator (a.k.a. label head) using electronic gearing between two SilverLode servos or a master encoder and a SilverLode servo.

In these examples, the “Master” servo is driving the conveyor belt and is setup to output encoder signals to the “Slave” servo which is applying the labels.

Prerequisites

It is assumed that the user is already familiar with the following:
 Electronic Gearing (See App Note QCI-AN019 Electronic Gearing)

Hardware

Master Encoder

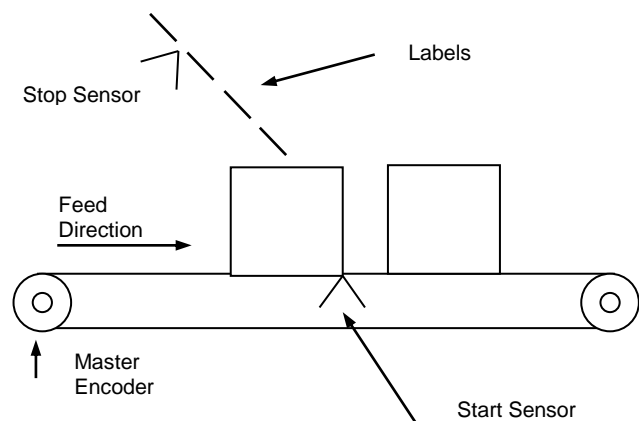
In general, the encoder output of the Master must be wired to the External (or secondary) Encoder input of the Slave. The details of which pins to use depends on the controller being used. See Application Note “QCI-AN019 Electronic Gearing” for details.

Start Sensor

In this example, the Start sensor detects the front edge of the box.

Stop Sensor

In this example, the Stop sensor detects the gaps between labels.



Simplified Label Head

Algorithm

- 1) The Slave servo waits for a box (wait for Start sensor to go LOW).
- 2) The Slave ramps up to synchronous speed with Master encoder to apply the label.
- 3) The Slave stops when it sees a full label applied (wait for Stop sensor to go LOW).

Gear Ratio (GR)

The GR of Slave to Master is expressed as follows:

$$\frac{\text{< Change in Slave Encoder (CS) >}{\text{<CS|CM>}} = \frac{\text{< Change in Master Encoder (CM) >}{\text{<CM>}}$$

For example a 1 to 2 GR would be expressed as 1:2 and would mean for every 2 encoder counts of the Master, the Slave would move 1 encoder count.

Therefore a Gear Ratio of CS:CM would be

$$\begin{aligned} \text{Gear Ratio (GR)} &= \text{CS/CM} = \text{CS:CM} \\ \text{Slave Position (S)} & \\ \text{Master Position (P)} & \end{aligned}$$

$$\begin{aligned} S &= \text{GR} * M \\ S &= (\text{CS/CM}) * M \end{aligned}$$

For 1:2 Gear Ratio example

$$\begin{aligned} \text{CS} &= 1 \\ \text{CM} &= 2 \\ \text{GR} &= \text{CS/CM} = \frac{1}{2} \\ S &= \text{GR} * M = \frac{1}{2} * M \end{aligned}$$

There are several ways of programming GR depending on what controller is being used. See below for details.

Software

Gear Ratio Commands

Registered Step and Direction (RSD) and Electronic Gearing Mode (EGM) can both be used to specify a GR and start electronic gearing.

RSD	EGM
Simple	Advanced
Available on both SilverDust and SilverNugget	Available on SilverDust Only
Specify GR as interger ratios i.e. 1:3	Specify GR as decimal i.e. 1:1.234567
	Optional: Acceleration Factor An acceleration ramp can be specified to change between GR.

Electronic Gearing Mode (EGM) - SilverDust Rev 34

EGM provides high-resolution electronic gearing capability including the ability to smoothly transition between different gear ratios.

The basic EGM command requires two registers.

Scale Factor (SF) Register

The Scale Factor (SF) register is Gear Ratio multiplied by 10,000,000 to provide a range of +/- 200 with 7 places behind the decimal point. The upper limit will depend upon the motor encoder resolution and the allowable output shaft motion for each Master count.

$$\text{SF} = \text{GR} * 10,000,000$$

Acceleration Factor (AF) Register

The Acceleration Factor (AF) register is used to limit the rate of change of the Internal Scale Factor (ISF). ISF is the internal or actual scale factor used by EGM while SF is the desired or target scale factor. Normal usage has a positive value for AF.

When first initialized, the ISF starts at 0. For each count change in the "Master" position, ISF is moved in the direction of SF by AF counts. Once running, when the SF register is changed, the ISF smoothly ramps from its previous value to the new requested value over a number of Master counts.

In the example "SlaveSD EGM.qcp"

SF=10,000,000 (1:1)

AF=10,000

Then ISF will ramp from 0 (stopped) up to SF=10,000,000 (1:1) over 1000 Master counts.

Note: if the input is oscillating back and forth, these movements count as excursions and the ISF will grow without regard to the direction of the motion.

Setting AF to a negative value will act as if a Stop Condition was met, causing the EGM to ramp to a stop using the absolute value of AF. This will also cause the EGM command to end regardless of the "End Command When Stopped" option setting. If the continuous operation is desired, then set the SF register to zero, which will cause ISF to ramp to zero, but will not cause the motion to end.

A value of zero for the Acceleration Factor is flagged as an error, and will cause the EGM to ramp down fairly quickly and to end.

If the user does not want the command to smoothly ramp between values, but to rather operate as just a fixed scale factor, set the Acceleration Factor to larger than the largest expected Scale Factor (i.e. 100,000,000) and the ramping will all occur with a single count of motion.

See below for description of Options.

LabelApplicatorSD.qcp

SEE configures the Slave to receive the Master encoder into I/O lines 4&5.

Register SF[31] is set to 10,000,000 here which correlates to a 1:1 gear ratio between Master and Slave. Change SF[31] for your gear ratio.

Register AF[30] sets the acceleration ramp. Increase this value for a faster ramp or reduce it for a slower ramp. See above for more details setting AF.

The WBE here causes the Slave to wait for a box arrive.

The EGM causes the Slave to ramp up to synchronous speed with the Master using the given gear ratio. The servo will stop when a label gap is detected (Stop sensor goes LOW).

Jump back to WAIT FOR START to wait for the next box.

Line# Oper	Label	Command
1:REM		LabelApplicatorSD.qcp This file is used in Application Note QCI-AN065 Label Applicator Requires: QuickControl Rev 4.63 SilverDust Firmware Rev 34
2:REM		Read ini master encoder
3:SEE		Select External Encoder: Index Source I/O #6 Encoder Style: A/B Quad on I/O #4 & 5 Scale Factor (SF)
4:REM		SF= GR * 10,000,000 GR = SF/10,000,000 Where GR is Gear Ratio i.e. SF=10,000,000, GR=1 (1:1)
5:WRP		Write 10000000 to "SF[31]" Register
6:REM		AF= 10,000 Ramp Internal Scale Factor (ISF) from its current value to SF (desired Scale Factor) AF counts for every Master count. Note ISF starts at zero when EGM is first executed. Example AF=10,000 SF=10,000,000 (1:1) ISF will ramp from 0 to 10,000,000 in 1000 Master counts. Or in other words, GR will ramp from 1 to 1:1 in 1000 Master counts.
7:WRP		Write 10000 to "AF[30]" Register
8:REM	WAIT FOR START	Wait for Start sensor to go LOW.
9:WBE		Wait On Bit Edge Until "I(1) - Start" is LOW/FALSE
10:REM		Apply label Stop when Stop sensor goes low
11:EGM		Electronic Gearing Mode: Acc Factor in "AF[30]" Scale Factor in "SF[31]" Stop when "I(3) - Stop" is LOW/FALSE
12:REM		Next Label
13:JMP		Jump to "WAIT FOR START"
14:REM		

Registered Step And Direction (RSD)- SilverNugget

An older implementation of RSD defines the SF register as follows. Note this implementation is the only option for the older SilverNugget controllers.

$$SF = GR * SF1$$

SF1 is the Scale Factor for a 1:1 gear ratio. SF1 depends on encoder resolution as follows:

$$4000 \text{ CPR; } SF1 = 1024$$

$$8000 \text{ CPR; } SF1 = 512$$

$$16000 \text{ CPR; } SF1 = 256$$

Example:

If the Slave has an 8000 counts per revolution (CPR) encoder and the desired GR is 1:2, what should be stored in the SF register.

Step 1. Determine the SF1.

The Slave's encoder is 8000 CPR so SF1=512.

Step 2. Calculate SF Value.

$$SF = GR * SF1$$

$$SF = 1/2 * 512$$

$$SF = 256$$

LabelApplicatorSN.qcp

SEE configures the Slave to receive the Master encoder into I/O lines 4&5.

Register SF[11] is set to 256 here which correlates to a 1:1 gear ratio between Master and Slave if the Slave has a 16000 CPR encoder. Change SF[11] for your gear ratio.

The WBE here causes the Slave to wait for a box arrive.

The RSD causes the Slave to ramp up to synchronous speed with the Master using the given gear ratio.

The EMT command on line 6 causes RSD to launch in the background. The WBE on line 12 waits for the Stop sensor (end of label) to go LOW. The VMP on line 14 will then stop the servo.

Jump back to WAIT FOR START to wait for the next box.

Line# Oper	Label	Command
1:REM		LabelApplicatorSN.qcp This file is used in Application Note QCI-AN065 Label Applicator Requires: QuickControl Rev 4.63 SilverDust Firmware Rev 34
2:REM		Read ini master encoder
3:SEE		Select External Encoder: Index Source I/O #6 Encoder Style: A/B Quad on I/O #4 & 5
4:REM		Set Scale Factor (SF) for a 1:2 Gear Ratio (GR) assuming an 8000 CPR encoder. Scale Factors for 1:1 Ratio (SF1) CPR SF1 4000 1024 8000 512 16000 256 SF = GR * SF1 For our example: SF1=512 GR =1/2 SF = GR * SF1 = 256
5:WRP		Write 256 to "SF[11]" Register
6:EMT		Enable Multi-Tasking
7:REM	WAIT FOR START	Wait for Start sensor to go LOW.
8:WBE		Wait On Bit Edge Until "I(1) - Start" is LOW/FALSE
9:REM		Apply label Stop when Stop sensor goes low
10:RSD		Registered Step and Direction: Scale Factor = "SF[11]" Register
11:REM		Wait for Stop sensor to go LOW
12:WBE		Wait On Bit Edge Until "I(3) - Stop" is LOW/FALSE
13:REM		Command servo to ramp down to zero velocity and wait for it to get there.
14:VMP		Velocity Mode: acc=80000.61 cps/s, vel=0 cps
15:WBS		Wait On Bit State Until "Trajectory Active" is LOW/FALSE
16:REM		Next Label
17:JMP		Jump to "WAIT FOR START"