5

Mode 30 – Pulse Train Output

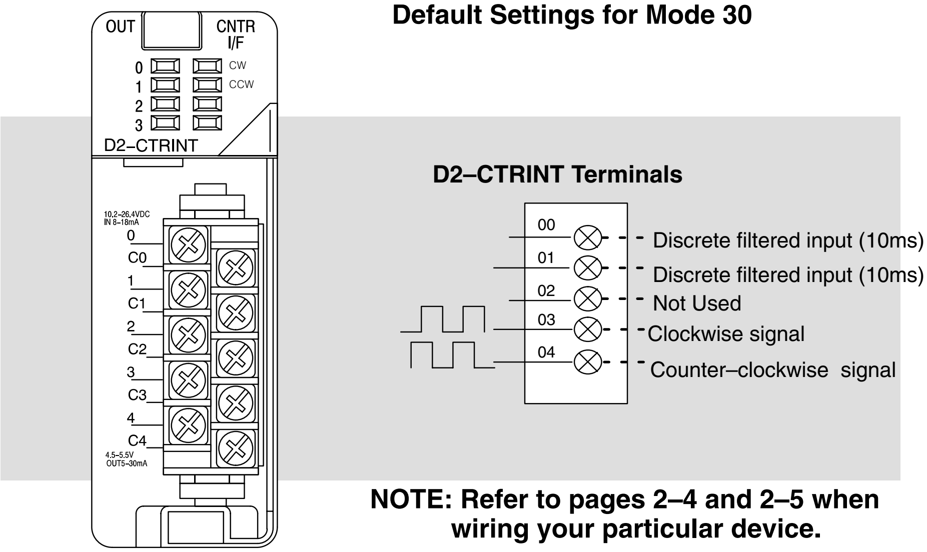
In This Chapter. . . .

Using the Pulse Train Output, Mode 30 (DL240/250–1/260 only)

It is recommended that you read Chapter 1, Getting Started, which introduces the six different modes of operation of the D2–CTRINT module, before selecting a mode. Even though several features can be mixed from several modes, *you must select one of the modes as your primary mode.* Mode 30, Pulse Train Output will be the only mode covered in this chapter.

It is also important to read Chapter 2, concerning the general guidelines for field wiring your device to the module. You may want to refer to Chapter 2 as you learn about the D2-CTRINT's high speed UP counting function.

Default Settings for Mode 30



NOTE: Refer to pages 2–4 and 2–5 when wiring your particular device.

The above diagram shows points 00 and 01 which default to discrete filtered (10ms) inputs while points 03 and 04 are the respective outputs. Refer to pages 2–4 and 2–5 when connecting your device.

D2–CTRINT Limitations

The D2–CTRINT, Counter Interface Module, is a powerful tool for providing solutions for stepper motor applications, however, there is a limitation. **The D2–CTRINT can only be used in open loop stepper motor applications.** The Counter Interface Module can not simultaneous read encoded information while providing a pulse train output.

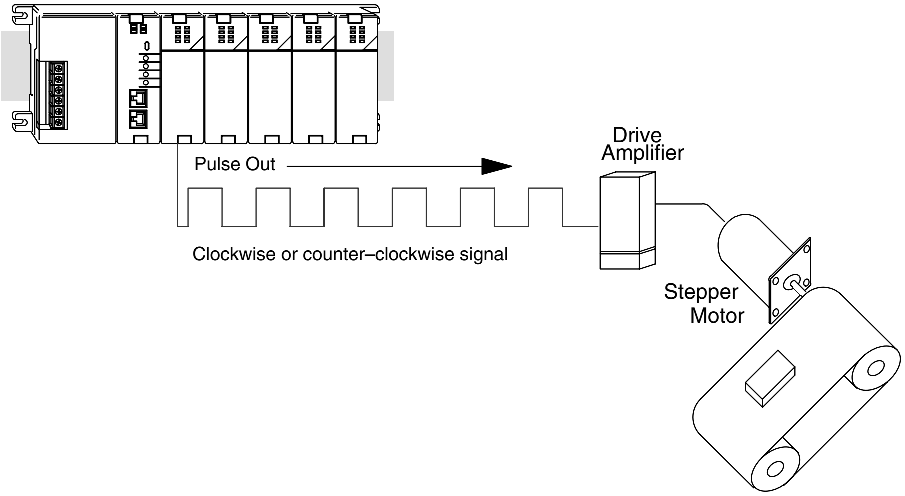
The D2–CTRINT module can not be used for multiple indexing. Such tasks require multiple sets of pulse trains all working off a single CPU or a multiple processor system. These applications are better suited for using *multiple indexers* that are specially designed to handle such tasks.

Less than 25% of all stepper motor applications in use today require closed loop control or multiple indexing. The D2–CTRINT is a cost-effective solution for most open loop stepper motor applications.

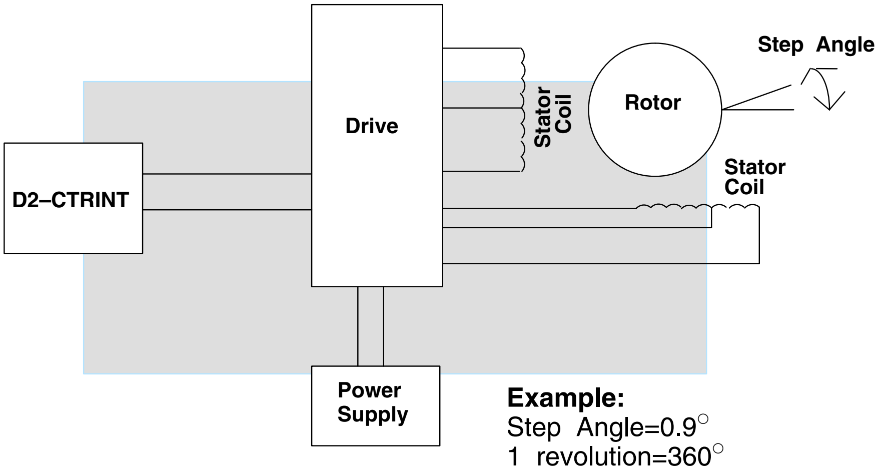
Open Loop Stepper Motor System

In the open loop system, a clockwise signal or counter-clockwise signal (determined by the PLC program) is sent from the interface module to the stepper motor drive amplifier. This amplifier translates the single pulse train signal sent from the interface module to the required number of signals for the stator windings of the motor. When applied to the stator windings, the pulses from the drive amplifier are converted into discrete rotor movements, or steps.

Pulse Output



Rotary stepper motors are typically designed for 0.9, 1.8, 7.5, 15, 45 and 90 degrees of revolution per pulse. The resolution of the stepper motor (steps per revolution) can be determined by dividing the stepper angle into 360 degrees. For example, the 0.9–degree motor will provide 400 steps per revolution, and the 15–degree motor is will provide 24 steps per revolution. The larger the number of steps the motor provides per revolution, the better the resolution it will provide. Also note that the higher the resolution, the slower the speed of the shaft.



Example:
Step Angle=0.9°
1 revolution=360°
steps/rev.=360°/.9° =400 steps

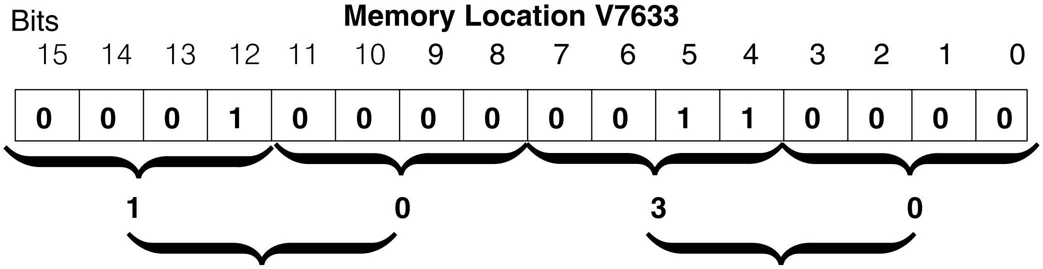
Since the rotation of the stepper motor's shaft is predictable for a given number of pulses, open loop systems are quite reliable. A typical system will use a “home position” which can be determined with a limit switch to orient the load to a start position. After the home position is determined, the controller can send the motor a specific number of pulses to cause it to move to a predetermined location.

The open loop system assumes that if a pulse is received, the motor will increment a step, which will move the load the designated distance. *Inaccuracy of this system may occur if the stepper motor develops a fault or if the load is obstructed in any way.* If your application is prone to such occurrences, the closed loop system with a shaft encoder providing feedback data of the actual position of the load may be the best choice. However, even with open loop systems, the normal safety features built into the drive amplifier system often afford a way to cope with unexpected faults or obstructions. Most stepper motor drive systems provide circuitry that will detect when the shaft is stopped for no apparent reason and is not allowed to step the correct number of times. In such systems, the PLC can be programmed to sense the condition through the system interface, trigger an alarm, and shut down the system.

Understanding V-Memory Setup Locations

The Pulse Train Output, Mode 30, associated with the D2-CTRINT requires V-memory configuration before it can be used.

V-memory location V7633 is the most important of all the reserved memory areas because it stores the numeric value which lets the CPU know which mode has been selected. The following diagram shows the 16-bit word and the various information it stores—including the values used for the Counter Interface Module. The example shown here uses Pulse Train Output, Mode 30. The lower bits are set to 30 and the upper bits set to 10 so the backup battery is enabled. Together they form the number 1030.



Memory Location V7633

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**1**	**0**	**0**	**0**	**0**

Miscellaneous Setup
Binary Coded Decimal:

00 = Not Used (default)
10 = Battery Enabled (DL230/240/250–1/260)
20 = Power Up in Run (DL230 only)
30 = Selects both Battery Enable and Power Up in Run (DL230 only)
40 = Mode Change Enable in K–sequence (DL240 only)
50 = Battery Enable and Mode Change Enable in K–sequence (DL240 only)

D2-CTRINT Mode Setup
Binary Coded Decimal:

00 = Not Used
10 = UP Counting Mode
20 = UP/DOWN Counting Mode
30 = Pulse Output Train
40 = High Speed Interrupts
50 = Pulse Catching
60 = Discrete Filtered Inputs

NOTE: It is important to look at the entire 16 bits at V7633. If the RLL program only sets the bits in the lower byte when entering the mode value, the upper bits will be overwritten with zeros (0's). Always enter a 4-digit BCD value when writing to V-memory. This way, the proper value will be written into the upper bits.

There are also other V-memory locations which contain High Speed Counter Interface Module setup information for each I/O point. The CPU will automatically configure them with default values for the mode which has been selected.

Default Settings When, xx30 is placed in V7633, the default configuration places the following values in V-memory:

Configuration	Point 00	Point 01	Point 02	Points 03 & 04
Default Settings	Discrete Input w/10 sec. filter	Discrete Input w/10 sec. filter	Not available	Clockwise and Counter-Clockwise Outputs
V–Memory Value Location	V7634	V7635	V7636	V7637
Value (hexadecimal)	1006	1006	0000	0003

The values shown above have the following meaning:

1006 = Discrete input with 10 ms filter (explained in Chapter 8.)

0003 = Assumes you are connecting to a drive unit that has separate channels for CW and CCW signals.

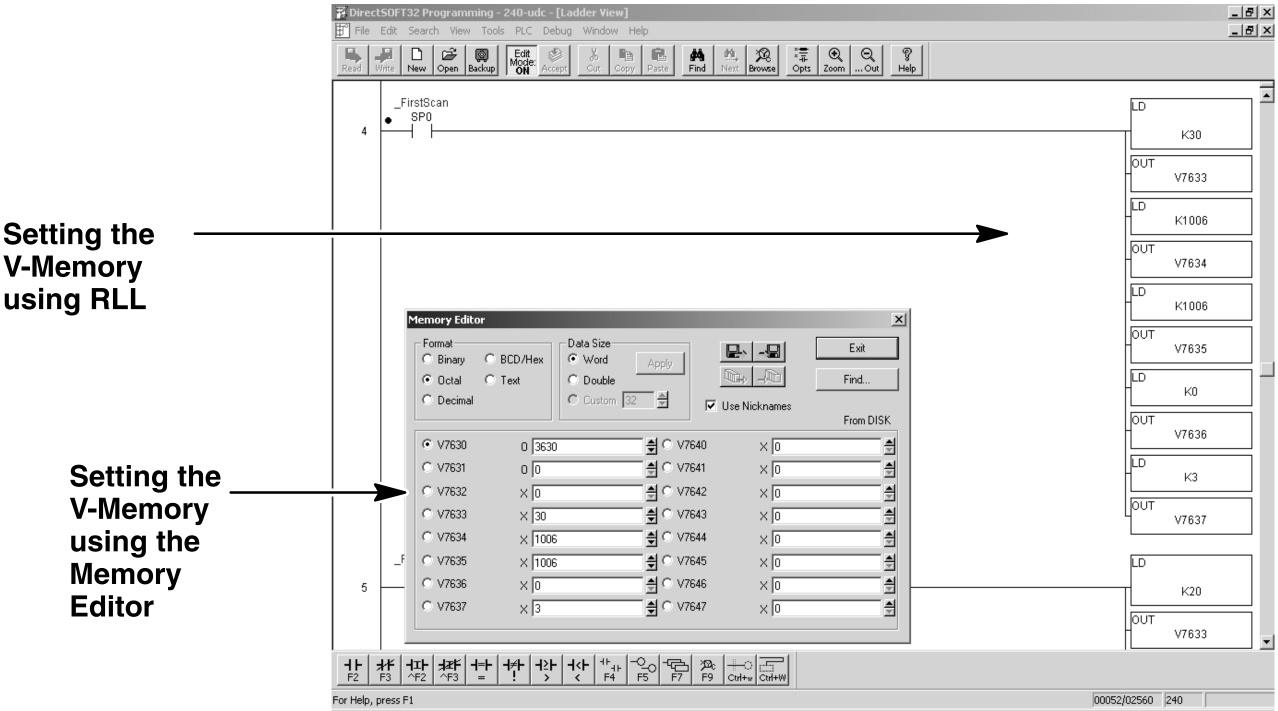
Refer to the diagram on page 5–2.

Setting Up the CPU

Configuring the V–Memory

The DL240/250–1/260 CPUs check the V-memory to see if there is a D2–CTRINT module present. The number 30 will be in V7633 if a module has been properly configured. When the CPU finds that a Counter Interface Module is present, it checks other V-memory locations to see how each point of the module has been configured.

The values can be entered into memory by using either a handheld programmer or by editing them into a control program using ***Direct***SOFT32. The following examples will show how to use ***Direct***SOFT32 to configure the D2–CTRINT module for Pulse Output operation.



Setting the V-Memory using RLL

Setting the V-Memory using the Memory Editor

DirectSOFT32 Programming - 240-udc - [Ladder View]
File Edit Search View Tools PLC Debug Window Help
Read Write New Open Backup Edit Mode: ON Accept Cut Copy Paste Find Next Browse Opts Zoom ... Out Help
F2 F3 ^F2 ^F3 = ! > < F4 F5 F7 F9 Ctrl+w Ctrl+W
For Help, press F1
00052/02560 240

```
4   _FirstScan
    SP0
    ─┤ ├──────────  LD   K30
                    OUT  V7633
                    LD   K1006
                    OUT  V7634
                    LD   K1006
                    OUT  V7635
                    LD   K0
                    OUT  V7636
                    LD   K3
                    OUT  V7637
5                   LD   K20
                    OUT  V7633
```

Memory Editor

Format: ○ Binary ○ BCD/Hex ● Octal ○ Text ○ Decimal

Data Size: ● Word ○ Double ○ Custom 32 — Apply

☑ Use Nicknames — Exit — Find... — From DISK

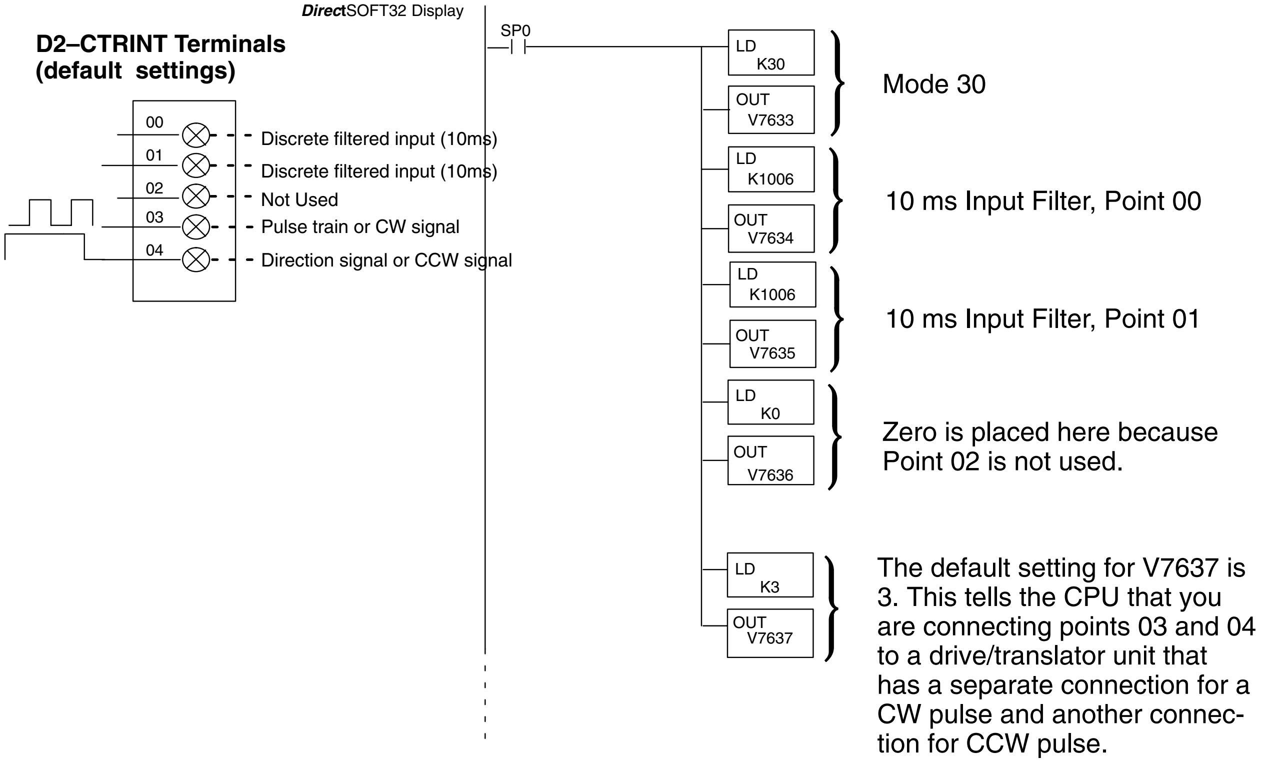
● V7630	O	3630	○ V7640	X	0
○ V7631	O	0	○ V7641	X	0
○ V7632	X	0	○ V7642	X	0
○ V7633	X	30	○ V7643	X	0
○ V7634	X	1006	○ V7644	X	0
○ V7635	X	1006	○ V7645	X	0
○ V7636	X	0	○ V7646	X	0
○ V7637	X	3	○ V7647	X	0

Editing the D2–CTRINT setup at the beginning of the user program is the most efficient method for setting up the counter mode. Should there be a need to change any of the counter setup values after the PLC has been put in the RUN Mode, use the Memory Editor to change the values. These values will only be temporary. They should be put into the program if they are to be used permanently.

Assigning Module Functions

One of the first things to do is to enter the Mode 30 value and assign functions to all of the points on the module.

Below is a sample RLL program for entering Mode 30, Pulse Train Output, in V-memory location V7633. The default V–memory locations are also setup even though the defaults are accepted.


*Direct*SOFT32 Display
D2–CTRINT Terminals
(default settings)
00
01
02
03
04
Discrete filtered input (10ms)
Discrete filtered input (10ms)
Not Used
Pulse train or CW signal
Direction signal or CCW signal
SP0
LD K30
OUT V7633
Mode 30
LD K1006
OUT V7634
10 ms Input Filter, Point 00
LD K1006
OUT V7635
10 ms Input Filter, Point 01
LD K0
OUT V7636
Zero is placed here because Point 02 is not used.
LD K3
OUT V7637
The default setting for V7637 is 3. This tells the CPU that you are connecting points 03 and 04 to a drive/translator unit that has a separate connection for a CW pulse and another connection for CCW pulse.

These commands are needed to put the values into V-memory. The value must first be loaded into the accumulator of the CPU, then the CPU must transfer the value to the memory location. In this case, 30 is to be placed in V7633. This value is loaded into the accumulator, LD K30. The CPU then writes this data to the memory location, V7633, once it reads the OUT command, OUT V7633. Notice that an SP0 contact is used in this rung. This relay is on for the first scan only. This will load the values into memory initially, thereby keeping the scan time to a minimum.

This is all that is needed to setup the CPU for Mode 30. If all of the defaults are accepted, the CPU will automatically assign discrete input functions with 10 ms filters to points 00 and 01. It will leave Point 02 unused. The configuration can be changed if all of the defaults are not acceptable.

Custom Configurations

Up to this point, only Mode 20 default settings have been discussed. The default settings are suitable for many applications, they will not require further configuration. However, for those applications needing the defaults changed so the D2–CTRINT will work for the applications, use the following table which contains the options available.

Custom Options for Mode 30

Mode 30 Options

Point Number	V-Memory Location	Possibility (One per point)	Hex Value
point 00	V7634	Discrete Filtered Input (default)	xx06 (xx=filter time) (default)
		Pulse Catch Input	0005
point 01	V7635	Discrete Filtered Input (default)	xx06 (xx=filter time) (default)
		Pulse Catch Input	0005
point 02	V7636	Not used	0000
points 03 & 04	V7637	Pulse Train	0003 (default) when drive has two lines: One marked CW and the other marked CCW. Point 3 – CW Point 4 – CCW
			0103 when drive has one line for the step pulse and another line to determine direction by either a HIGH or LOW pulse. High=CW, Low=CCW Point 3 – step Point 4 – direction

NOTE: See Chapter 8 for a detailed description of discrete filtered inputs, and see Chapter 7 for a detailed description of the pulse catch inputs.

Determine the Positioning Profile

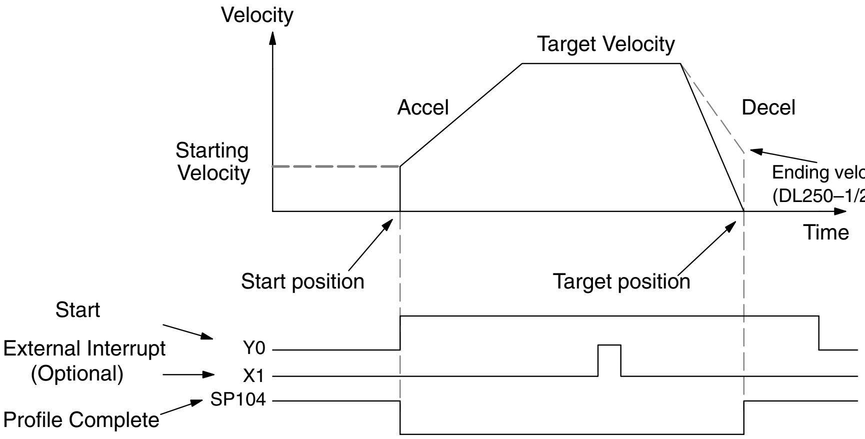
There are three different types of profiles available for the D2–CTRINT module:

(1) **Automatic acceleration/deceleration**

Once the starting velocity, acceleration/deceleration times and the total number of pulses are determined, the CPU will automatically compute the balance of the positioning profile. Starting velocities must be within the range of 40 pps to 1k pps. The remainder of the profile parameters are in the profile parameter table.

Automatic Accel/Decel Trapezoid Profile

Automatic Trapezoidal Profile

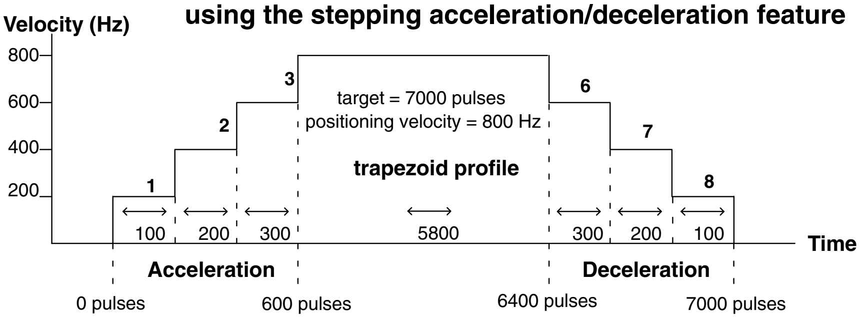


(2) **Stepped trapezoid**

Define 1 to 4 steps of gradual acceleration, 1 to 4 steps of gradual deceleration and define target pulse values. This type of profile is used with large stepper motors and/or large inertia loads. It can be used with any load or motor requiring slow gradual ramping.

Step Trapezoid Profile

Example positioning operation using the stepping acceleration/deceleration feature



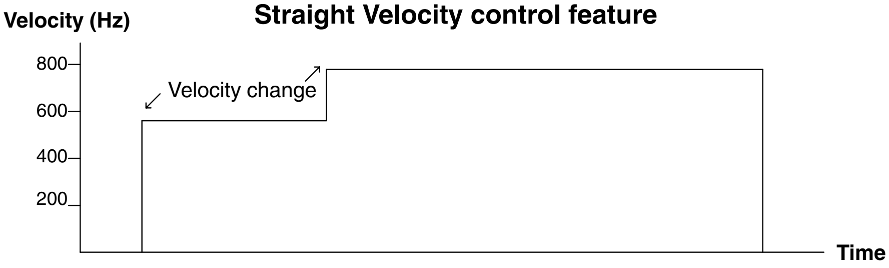
NOTE: Only 3 steps are shown here for both acceleration and deceleration. Up to 4 steps for each and 1 target can be used.

(3) **Straight velocity**

This profile only controls pulses per second with no automatic ramping; however, the pulse output frequency can be modified by adding additional ladder logic, therefore, changing the ramping characteristics.

Straight Velocity

Straight Velocity control feature



Understanding the Positioning Profile of Setup Locations

Choose the positioning profile then activate it by placing special values in consecutive addresses starting at the default address of V3630. Then, use the logical output, Y4, to start the pulse output.

NOTE: Pointers for the beginning of these addresses are stored at V7630. If there is a conflict of addresses because of pre-existing values written to these locations, change the beginning address by having the RLL place a different pointer value in V7630. For example, to change the beginning address to V2500, have the program write octal 2500 to V7630. Then, the CPU would reserve consecutive addresses beginning at V2500 for the chosen positioning profile value.

When using the straight velocity profile, there isn't a target pulse count to reach and the current pulse count will not be constantly compared to a target pulse count. The opposite is true when using the other positioning profiles.

Y4 use in the Absolute Mode: Bit 15 of V3630 set to 0

When using either the automatic acceleration/deceleration profile or the step trapezoid profile, a target must be achieved. The process is such that when the actual pulse count equals the target, a logical contact (always designated in the RLL program as X4) goes HIGH signaling that the profile is complete. If the target is changed to either a higher value (CW) or a lower value (CCW), the pulse count is no longer equal to the target. The system also uses a logical output, Y4, to activate the configuration. When Y4 goes HIGH, the difference between the target and the actual pulse count is equal to the number of output pulses.

For example, a RLL program such that whenever Y4 goes HIGH, the actual pulse count will stop once 2000 is reached and X4 will go HIGH. In this case, if Y4 goes HIGH again, nothing will happen. If the target is changed to 5000 and Y4 goes HIGH, the number of output pulses will be 3000 clockwise in order to achieve the new target. At this point, the actual pulse count is 5000. If Y4 goes HIGH again, no pulses will be generated. If the target is changed from 5000 to 1000, and Y4 is turned on, the module will generate 4000 pulses (CCW) to achieve its target.

Y4 use in the Incremental Mode: Bit 15 of V3630 set to 1

In this mode, each time Y4 goes HIGH, the actual pulse count is incremented by the target amount.

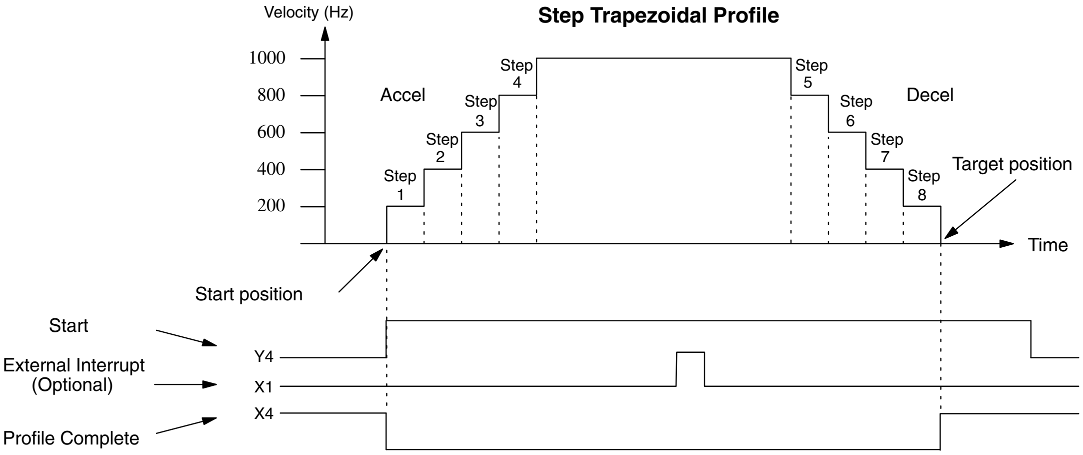
For example, suppose the target is 2000 and Y4 goes HIGH. The actual pulse count goes to 2000 and reaches the target. Then, if Y4 goes HIGH again and the pulse train produces 2000 pulses, the actual pulse count will be 4000 (2000+2000=4000).

Y4 use with an Interrupt – Bit 12 of V3630 set to 1

It is possible to configure the V-memory with a value which will allow a field device to be attached to point 01 of the module which will have the role of providing an interrupt signal. If this option is used, Y4 will start the pulse counting procedure and step a motor, but it will not look for the target pulse count until it receives the interrupt signal. If it does not receive an interrupt, it will continue to rotate the motor shaft indefinitely. Keep track of the pulses by using either the absolute or the incremental modes.

Step Trapezoidal Profile Operation

The acceleration and deceleration slopes are controlled with the step trapezoidal profile.



The time line signal traces below the profile indicates the order of events. The D2–CTRINT uses logical output Y4 as the start input, which starts the profile. The D2–CTRINT turns off the Profile Complete signal, X4, immediately, so the ladder program can monitor the progress of the move. Typically, a ladder program will monitor this bit so it knows when to initiate the next profile move. The External Interrupt, X1, can also be used. Once the External Interrupt feature is selected for the profile, the PLC will continue to output the pulses until X1 turns on. Then, the PLC outputs the pulses defined as the target position.

Each acceleration and deceleration slope consists of four steps. The velocity and distance (number of pulses) can be set up for each step. It is not necessary to use all four steps of each slope. For instance, it is possible to use only two steps, just set the velocity to zero (0), then set the distance for Step 3 and Step 4. If the acceleration slope and deceleration slopes are identical, all of the velocity and distance parameters can be set to zero (0) for the deceleration slope.

Actual Count V1175/V1174

The actual pulse count is always being counted and stored in V1175/V1174. This is the running total of the module output pulses. The count values increment in V1175/V1174 for clockwise output pulses and decrement for counter–clockwise output pulses.

The Automatic Accel/Decel Profile

Configuring V-Memory for the Method of Your Choice

The CPU looks at V3630 (by default) to see how the Pulse Output is to be used. If the Automatic Accel/Decel Profile has been chosen, the 16 bits at that address will contain the following information:

1. Bit 15 will select the pulses to be stored and used either in the Incremental mode or the Absolute mode
2. Bit 14 is used to select the profile type
3. Bits 12 & 13 are used to select to use an interrupt with the profile
4. Bits 0 – 11 contains the positioning velocity in pulses per second

Refer to the chart below to configure V3630

Automatic Accel/Decel Trapezoidal Profile

V–Memory	Function	Range	Units

V3630, bits 12–15	Automatic Trapezoidal Profile without Ending Velocity (Ending Velocity is fixed to 0)	4 = absolute w/o interrupt
		5 = absolute with interrupt*
		C = relative w/o interrupt
		D = relative with interrupt*
			—

	Automatic Trapezoidal Profile with Ending Velocity (Use V3637 to setup Ending Velocity)	4 = absolute w/o interrupt
		5 = absolute with interrupt*
		C = relative w/o interrupt
		D = relative with interrupt*
			—

V3630, bits 0–11	Target Velocity	4 to 500	x 10 pps
V3631/3632	Target Position**	–8388608 to 8388607	Pulses
V3633	Starting Velocity	4 to 100	x 10 pps
V3634	Acceleration Time	1 to 100	x 100 ms
V3635	Deceleration Time	1 to 100	x 100 ms
V3637	Ending Velocity	4 to 100	x 10 pps

*If interrupt is selected, the PLC will not start looking for the target count until the interrupt X4 is on.

**To set a negative number, put an 8 in the most significant digit. For example: –8388608 would be written as 883388608 in V3631 and V3632.

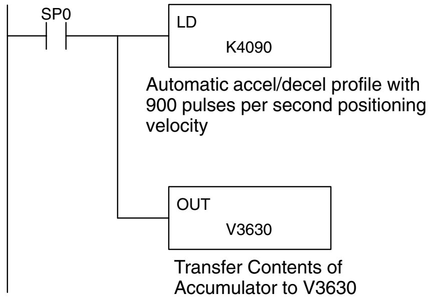
When choosing this profile, a starting velocity and target pulse count must be specified. You must also specify the time requirements for acceleration and deceleration.

Implementing ***Direct***SOFT32 to setup the Automatic acceleration/deceleration profile.

Step 1: Setup V3630

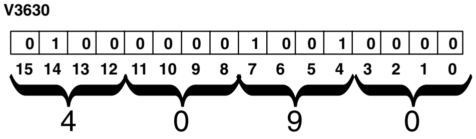
The CPU needs to know that the module is to be used with the Automatic acceleration/deceleration profile. The RLL diagram below shows what needs to be done to setup V3630.

DirectSOFT32 Display



Automatic accel/decel profile with 900 pulses per second positioning velocity

Transfer Contents of Accumulator to V3630



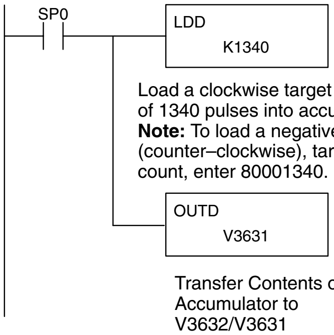
Bit 15=0 (Absolute Counting)
Bit 14=1 (Auto Accel/Decel)
Bits 12 & 13=00 (No Interrupt)
Bits 0 - 11 = 090

For this example, 900 pps will be used for the Positioning Velocity. The value of 090 (900 pps) will be written to bits 0 – 11. Interrupts will not be used, so bits 12 and 13 will be set to zero (0). Bit 14 is set to one (1) to let the CPU know that the Acceleration/Deceleration Mode is being used by the module. Bit 15 is set to zero (0) indicating that the pulses are to be stored and used in V1175/1174 in the Absolute mode. The value to be stored to V3630 will be 4090. The Incremental mode could have been selected by setting bit 15 to a one (1), a hex C in this example, i.e. C090.

Step 2: Set the Target Pulse Count and Shaft Direction

The target pulse count needs to be set next. This will be the end of the deceleration ramp where the stepper motor will stop and some event is usually triggered. The load is to be moved six (6) inches. It has been determined that it will take 1340 pulses to move the load the six (6) inches. This will need to be stored in V3632/V3631.

DirectSOFT32 Display



Load a clockwise target pulse count of 1340 pulses into accumulator. **Note:** To load a negative (counter–clockwise), target pulse count, enter 80001340.

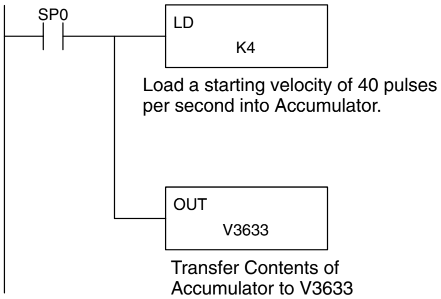
Transfer Contents of Accumulator to V3632/V3631

A positive target pulse count was used in the above example. The positive value means the shaft will rotate clockwise. A negative target pulse count could have been used to rotate the shaft counter–clockwise. Whenever a negative target number is used, it will be in the form 8xxxxxxx, where the 8 signifies that it is a negative number. Negative pulses are output at point 04 (CCW) and positive pulses are output at point 03 (CW) when the system is configured to output separate CW and CCW pulses.

**Step 3:
Set the Starting Velocity**

The starting velocity must be a number in the range of $x \geq 40$ but ≤ 1000. The information must be stored in V3633. Since there is a multiplier of 10 built in, 4 would be entered for a starting velocity of 40 pps. Be sure to verify the motor specifications to determine a proper base speed from which to start the acceleration.

DirectSOFT32 Display



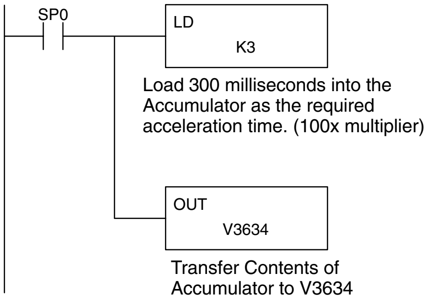
Load a starting velocity of 40 pulses per second into Accumulator.

Transfer Contents of Accumulator to V3633

**Step 4:
Entering the Acceleration Time**

Next, the acceleration time and the deceleration time is to be entered. Assume that the calculated acceleration time is to be 300 milliseconds and the deceleration time is to be 400 milliseconds. The acceleration time will be entered in memory location V3634 first. The number that goes into this location must be an integer multiplied times a weight factor of 100 milliseconds, therefore, 3 will be entered.

DirectSOFT32 Display



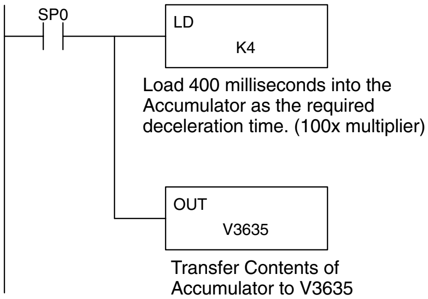
Load 300 milliseconds into the Accumulator as the required acceleration time. (100x multiplier)

Transfer Contents of Accumulator to V3634

**Step 5:
Entering the Deceleration Time**

The deceleration time is entered into V3635. The value to be entered into this memory location has the same weight factor as for acceleration. The above example calculated the deceleration time to be 400 milliseconds, therefore, 4 will be entered in this location.

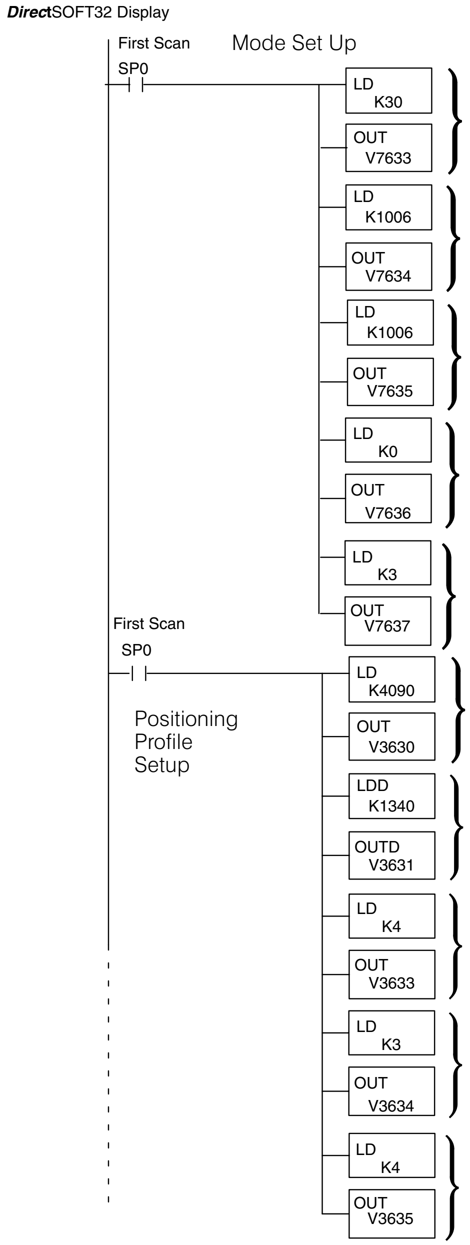
DirectSOFT32 Display



Load 400 milliseconds into the Accumulator as the required deceleration time. (100x multiplier)

Transfer Contents of Accumulator to V3635

Here is the full RLL for the Acceleration/Deceleration profile.



Mode 30 (Pulse Output Mode)

Discrete Filtered Input at (00)
10 ms. See Chapter 8

Discrete Filtered Input at (01)
10 ms. See Chapter 8

Zero is placed here because Point 02 is not being used

If a 3 is used here, the system provides CW or CCW pulse output. If a 103 is used here, the system provides CW or CCW using high and low signals for signal direction.

Automatic accel/decel Profile Positioning
Velocity=900 pps

Target Pulse Count (1340 pulses)

Starting Velocity (40 pps)

Acceleration Time (300 ms)

Deceleration Time (400 ms)

The Step Trapezoid Profile

Configuring V-Memory

The CPU looks at V3630 (by default) to see how the pulse output is being used. If the Step Trapezoid Profile has been chosen, the 16 bits at V3630 will contain the following information:

1. Bit 15 will select the pulses to be stored and used either in the Incremental mode or the Absolute mode
2. Bit 14 is used to select the profile type
3. Bits 12 & 13 are used to select to use an interrupt with the profile
4. Bits 0 – 11 contains the positioning velocity in pulses per second

Refer to the chart below to configure V3630.

Step Trapezoidal Profile

V–Memory	Function	Range	Units
V3630, bits 12–15	Step Trapezoidal Profile without)	0=absolute w/o interrupt 1=absolute with interrupt* 8=relative w/o interrupt 9=relative with interrupt*	—
V3630, bits 0–11	Target Velocity	4 to 500	x 10 pps
V3631/3632	Target Position**	–8388608 to 8388607	Pulses
V3633	Step 1 Acceleration	4 to 500	x 10 pps
V3634	Step 1 Distance	1 to 9999	Pulses
V3635	Step 2 Acceleration	4 to 500	x 10 pps
V3636	Step 2 Distance	1 to 9999	Pulses
V3637	Step 3 Acceleration	4 to 500	x 10 pps
V3640	Step 3 Distance	1 to 9999	Pulses
V3641	Step 4 Acceleration	4 to 500	x 10 pps
V3642	Step 4 Distance	1 to 9999	Pulses
V3643	Step 5 Acceleration	4 to 500	x 10 pps
V3644	Step 5 Distance	1 to 9999	Pulses
V3645	Step 6 Acceleration	4 to 500	x 10 pps
V3646	Step 6 Distance	1 to 9999	Pulses
V3647	Step 7 Acceleration	4 to 500	x 10 pps
V3650	Step 7 Distance	1 to 9999	Pulses
V3651	Step 8 Acceleration	4 to 500	x 10 pps
V3652	Step 8 Distance	1 to 9999	Pulses

* If interrupt is selected, the PLC will not begin to look for the target count until the interrupt X4 is on.
**To set a negative number, put an 8 in the most significant digit. For example: –8388608 would be written as 883388608 in V3631 and V3632.
When choosing this profile, you also must specify target pulse count, accel/decel rates and distance for each step in pulses.

Notice that deceleration starts counting in sequence backward from Steps 8 to 5.

Step Trapezoidal Profile Example

It is desired to move a punch head via a lead screw a distance of 1000 millimeters. The screw has a 7mm pitch and the stepper motor turning it has 7.5 degrees of revolution per pulse. The head is to move at a speed of 117 mm/sec and it is to accelerate to this speed in 1.5 seconds. The stepper motor has a maximum speed of 1200 rpm.

Calculations:

Begin by making some basic calculations:

Pulses per revolution = 360 ÷ 7.5 = 48 pulses

Movement per pulse = 7 mm per rev. ÷ 48 pulses per rev. = 0.146 mm per pulse

Target pulse value = 1000 mm ÷ 0.146 mm per pulse = 6849 pulses

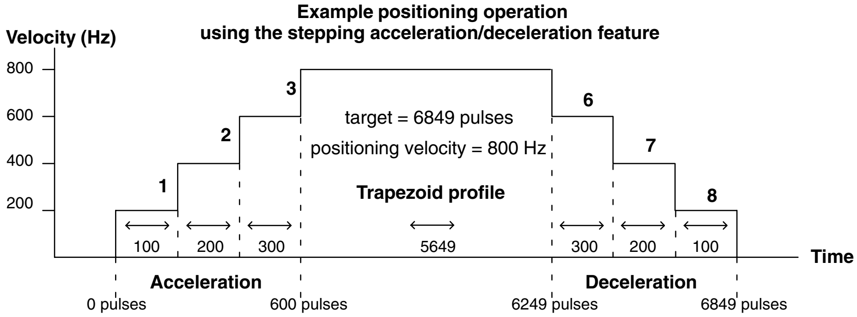
Positioning velocity = 117 mm per sec ÷ 0.146 mm per pulse = 800 pps

Steps used to reach positioning velocity = 4 (200 pps increments)

Slope of acceleration ramp = 2:1

Note: There are no clear–cut rules for selecting the slope of the acceleration ramp; therefore, it is often an arbitrary decision. One factor that does affect the slope, however, is the motor load. If it is heavily loaded, the ramp needs to be slow to avoid a stall. In this example, the motor is loaded heavily for its torque rating so a conservative slope ratio of 2:1 has been used. Since there are four steps of acceleration, they are each incremented 200 pps (800 ÷ 4 = 200).

The diagram below shows the characteristics of the example profile.

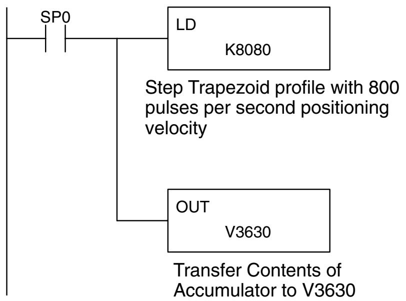


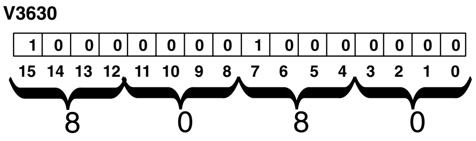
**Step 1:
Setup V3630**

The first memory location that should be configured is V3630. This will setup the Step Trapezoidal profile data for the CPU to operate the module. It will also set the type of numbers stored for each step change (i.e. incremental or absolute), and set the positioning velocity.

If you set bit 15 to 1 , then you are telling the CPU that you want to store and use the pulse counts in V1175/1174 incrementally. If bit 15 is 0, you are telling the CPU that you want to store and use the pulses absolutely. Bit 14 is set to 0 to indicate step trapezoid profile. In our example here, we have chosen to use the Incremental mode for our stepping (that is Bit 15=1 and Bit 14=0 or hex 8xxx):

*Direct*SOFT32 Display


SP0
LD
K8080
Step Trapezoid profile with 800 pulses per second positioning velocity
OUT
V3630
Transfer Contents of Accumulator to V3630



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bit 15=1 (Incremental Counting)
Bit 14=0 (Step Trapezoid Profile)
Bits 12 & 13=00 (No Interrupt)
Bits 0 – 11 = 080

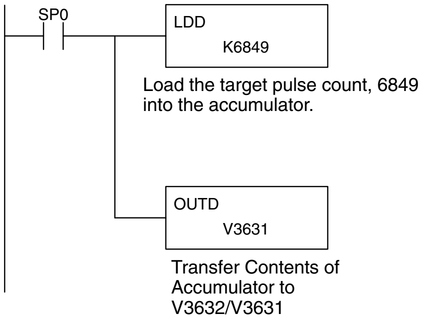
For this example, 800 pps will be used for the Positioning Velocity. The value of 080 (800 pps) will be written to bits 0 – 11. Interrupts will not be used, so bits 12 and 13 will be set to zero (0). Bit 14 is set to zero (0) to let the CPU know that the Step Trapezoid Mode is being used by the module. Bit 15 is set to zone (1) indicating that the pulses are to be stored and used in V1175/1174 in the Incremental mode. The value to be stored to V3630 will be 8080. The Absolute mode could have been selected by setting bit 15 to a zero 01).

NOTE: There is a 10 pps multiplier for the number placed in the lower 12 bits of this memory location. The code is therefore 080 for 800 pps positioning velocity.

**Step 2:
Setup the Target
Pulse Count**

The target pulse has been computed to be 6849. This value is stored in V3632/V3631. Referring to the RLL diagram below, the LDD instruction is a double word load, the K6849 is actually 00006849. When this is written to V3631 with the OUTD, the 0000 portion is stored in V3632, indicating clockwise (CW) shaft rotation.

*Direct*SOFT32 Display

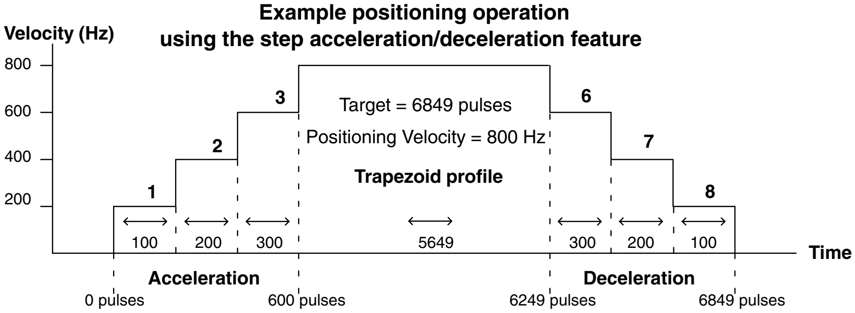

SP0
LDD
K6849
Load the target pulse count, 6849 into the accumulator.
OUTD
V3631
Transfer Contents of Accumulator to V3632/V3631

Step 3: Select the Height and Width

Three steps are used for the acceleration/deceleration and a 2:1 slope for this example. 200 pulses per second increments will be used for the height; this means the width will be half that amount.

Step One = 100 pulses at 200 pps = 0.5 secs
Step Two = 200 pulses at 400 pps = 0.5 secs
Step Three = 300 pulses at 600 pps =0.5 secs

At this point a total of 600 pulses have been moved, and the Positioning Velocity of 800 pps has been reached. Refer to the diagram below.



The CPU will calculate the Positioning Velocity from the acceleration/deceleration steps:

Step Six = 300 pulses at 600 pps. = 0.5 secs
Step Seven = 200 pulses at 400 pps = 0.5 secs
Step Eight = 100 pulses at 200 pps = 0.5 secs

The CPU also calculates the time it takes to move at the target Positioning Velocity:

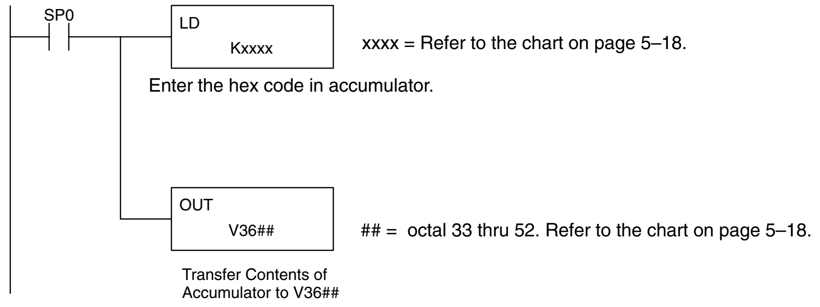
(6849–1200 = 5649)pulses at 800 pps = 7.06 secs

The total stepping cycle will take about 8.5 secs to reach the target distance to perform the punch operation. The values for the height (Positioning Velocity) and width (number of pulses) of each step are stored in V-memory locations V3633 through V3652. Refer to the chart on page 5–18.

Step 4: Enter the Step Information Into Memory

Except for the target value which uses LDD and OUTD, each step value is to be placed into the respective V-memory locations using the same RLL instructions, i.e. LD (load accumulator) and OUT (output to memory). The RLL diagram below can be repeated to enter each value.

DirectSOFT32 Display


SP0
LD
Kxxxx
xxxx = Refer to the chart on page 5–18.
Enter the hex code in accumulator.
OUT
V36##
## = octal 33 thru 52. Refer to the chart on page 5–18.
Transfer Contents of
Accumulator to V36##

Shortcuts: The data for each step does not have to be entered. For example, if only two steps are used, then enter the data for Step 1 and Step 2 and a zero (0) for each of the V–memory locations from V3643 to V3652.

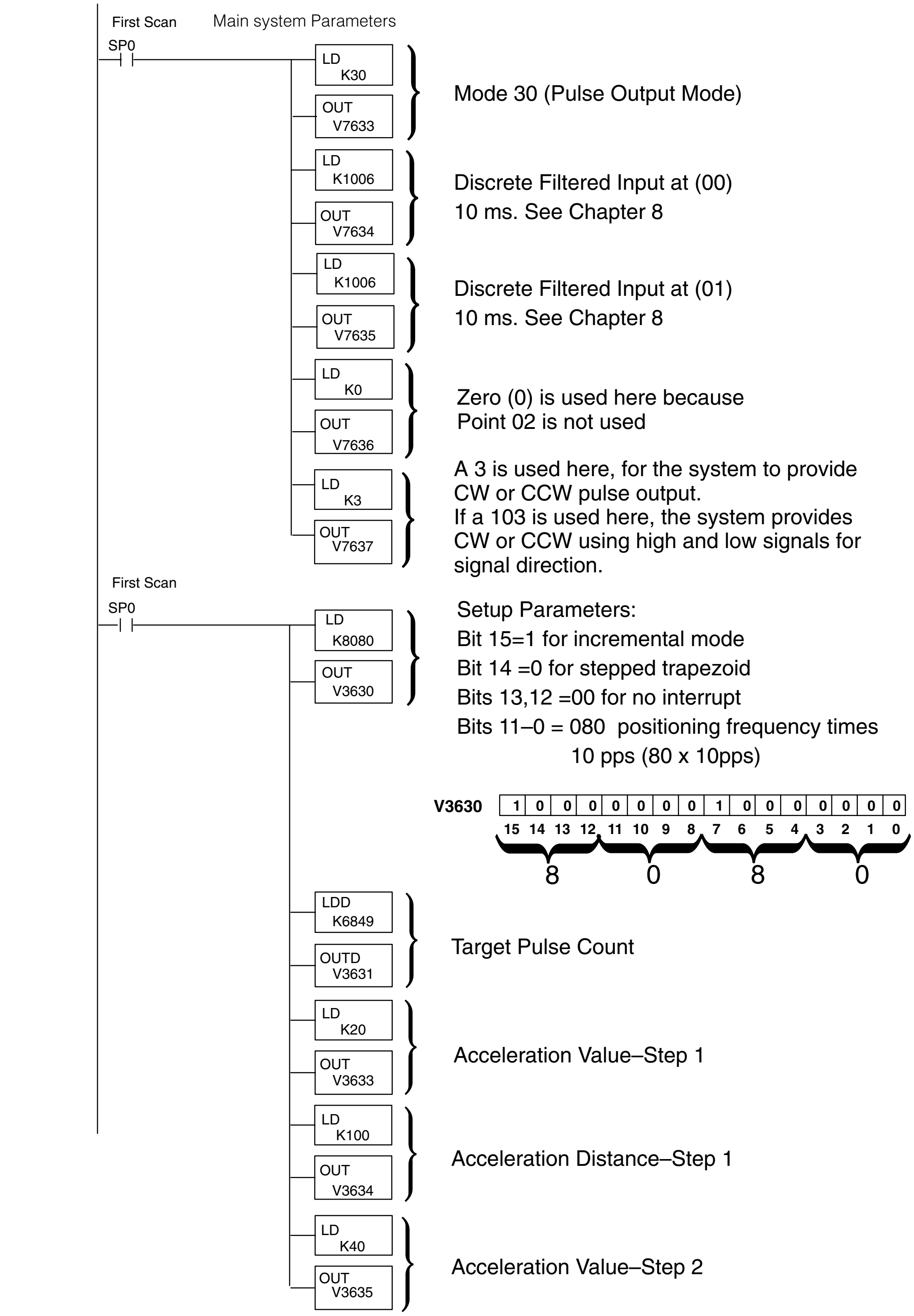
Also, if the acceleration and deceleration are identical, place a zero (0) in each V–memory location, V3643 through V3652, and the CPU will automatically set the acceleration and deceleration equal.

Ladder Logic Example

The following RLL program shows how to setup the memory configuration. It does not provide the logic to activate the configuration and perform a specific task. It does provide a starting point developing a control program for the Step Trapezoid Profile.

Positioning Velocity/Step Profile Method
Step Trapezoid, Incremental Mode No Interrupt

*Direct*SOFT32 Display



Mode 30 (Pulse Output Mode)

Discrete Filtered Input at (00)
10 ms. See Chapter 8

Discrete Filtered Input at (01)
10 ms. See Chapter 8

Zero (0) is used here because
Point 02 is not used

A 3 is used here, for the system to provide CW or CCW pulse output.
If a 103 is used here, the system provides CW or CCW using high and low signals for signal direction.

Setup Parameters:
Bit 15=1 for incremental mode
Bit 14 =0 for stepped trapezoid
Bits 13,12 =00 for no interrupt
Bits 11–0 = 080 positioning frequency times 10 pps (80 x 10pps)

Target Pulse Count

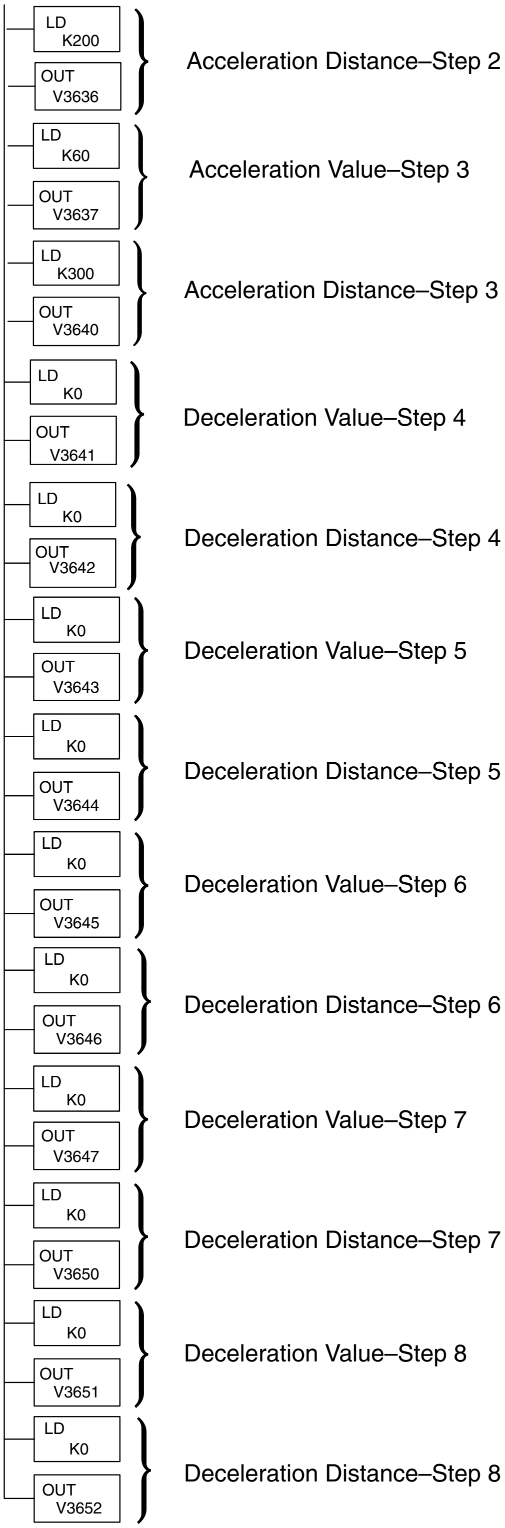
Acceleration Value–Step 1

Acceleration Distance–Step 1

Acceleration Value–Step 2

Continued on next page....

Continued from previous page....


LD
K200
OUT
V3636
Acceleration Distance–Step 2
LD
K60
OUT
V3637
Acceleration Value–Step 3
LD
K300
OUT
V3640
Acceleration Distance–Step 3
LD
K0
OUT
V3641
Deceleration Value–Step 4
LD
K0
OUT
V3642
Deceleration Distance–Step 4
LD
K0
OUT
V3643
Deceleration Value–Step 5
LD
K0
OUT
V3644
Deceleration Distance–Step 5
LD
K0
OUT
V3645
Deceleration Value–Step 6
LD
K0
OUT
V3646
Deceleration Distance–Step 6
LD
K0
OUT
V3647
Deceleration Value–Step 7
LD
K0
OUT
V3650
Deceleration Distance–Step 7
LD
K0
OUT
V3651
Deceleration Value–Step 8
LD
K0
OUT
V3652
Deceleration Distance–Step 8

The Straight Velocity Profile

Configuring V-Memory

The CPU looks at V3630 (by default) to see how the pulse output is being used. If the Straight Velocity Profile has been chosen, the 16 bits at that address will contain the the hex value of 2000. This profile uses the Velocity Control mode only; *it does not use a target*. The following table contains the information to configure the profile.

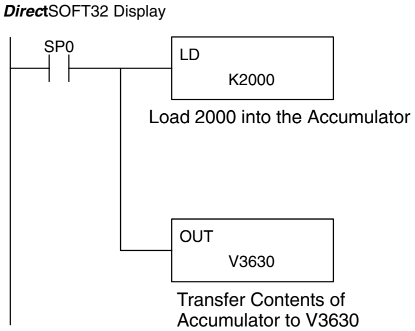
Straight Velocity

V–Memory	Function	Range	Units
V3630	Velocity Profile	2000 only	—
V3631/3632	Direction Select	80000000=CCW, 0= CW	Pulses
V3633	Velocity	0 to 500	x 10 pps

The following RLL diagrams can be used as a guide for setting up the Straight Velocity Profile.

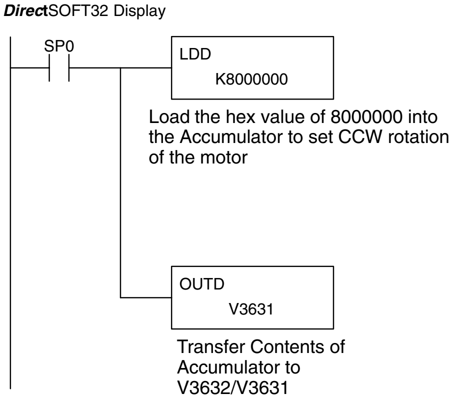
Step 1: Setup V3630

Place the hexadecimal value of 2000 in V3630 to set the Straight Velocity Profile.


*Direct*SOFT32 Display
SP0
LD
K2000
Load 2000 into the Accumulator
OUT
V3630
Transfer Contents of Accumulator to V3630

Step 2: Set the Direction for Shaft Rotation

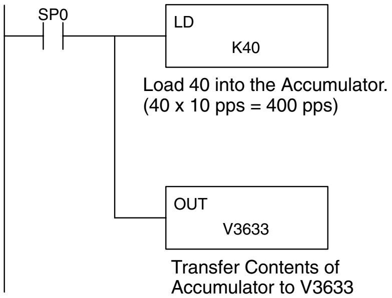
Set the direction the stepper motor shaft is to turn. Use the 32 bit value of 80000000 for counter–clockwise direction or zero (0) for clockwise direction in V3632/V3631.


*Direct*SOFT32 Display
SP0
LDD
K8000000
Load the hex value of 8000000 into the Accumulator to set CCW rotation of the motor
OUTD
V3631
Transfer Contents of Accumulator to V3632/V3631

Step 3: Set Shaft Velocity

Set the velocity the motor shaft is to turn.

***Direct*SOFT32 Display**



Load 40 into the Accumulator.
(40 x 10 pps = 400 pps)

Transfer Contents of
Accumulator to V3633

Velocity Control Used to Find the Home Position

Straight velocity control is often used to find a “home” position. Home position is a reference point (usually the starting point) where the object being moved can return on command at any time during or after a positioning process has begun. Whenever a home position is used there also is a “near” home position used as an intermediate reference point to slow down the velocity to avoid overshoot.

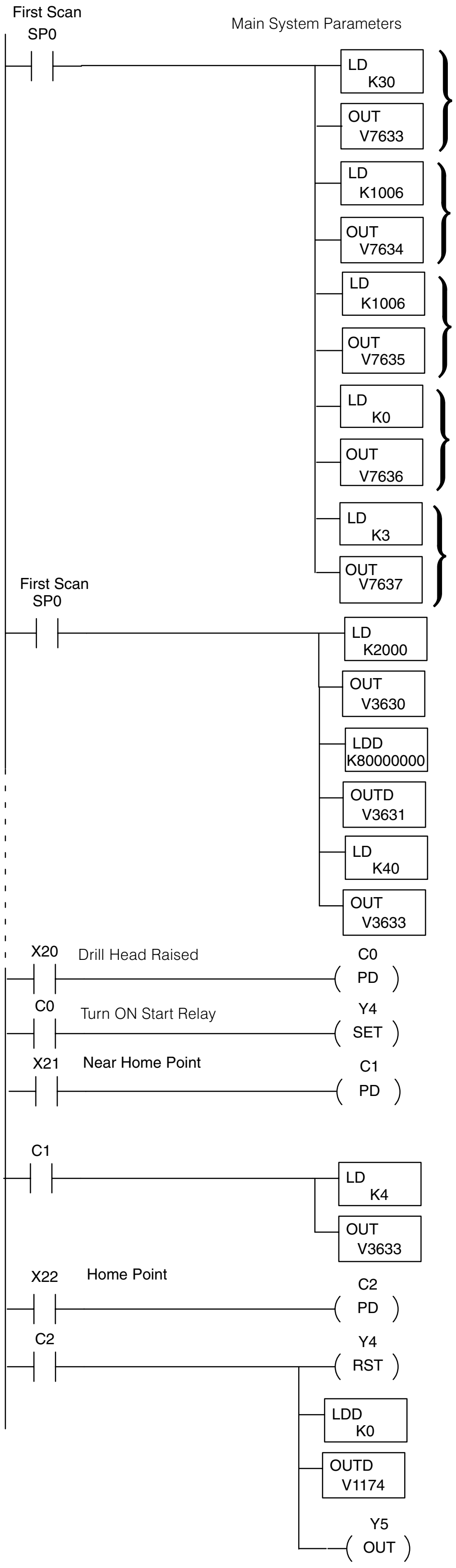
Straight Velocity Profile Example

The following example will explain a process where a drill head locates the home position after completing its last task. Some events established are:

1. Switch (X20) is tripped after a drill operation has been completed and the drill has been raised from the table at the target position. The drill head is to be moved to the home position.
2. A high speed sensor (X21) has been placed at a reasonable point to sense the returning drill head. It has been placed there to reduce the velocity to a very slow speed to avoid overshoot when it reaches the home position.
3. A sensor (X22) is located at the home position to sense the drill head to indicate that the process is to stop.

The RLL for this example is on the next page.

DirectSOFT32 Display



Mode 30 (Pulse Output Mode)

Discrete Filtered Input at (00)
10 ms. See Chapter 8

Discrete Filtered Input at (01)
10 ms. See Chapter 8

Zero is used here because Point 02
is not being used.

A 3 is used here, for the system to provide CW or CCW pulse output. If a 103 is used here, the system provides CW or CCW using high and low signals for signal direction.

Straight Velocity Control

CCW rotation

Velocity = 400 pps

Start CCW output pulse

Slow Drill Head to 40 pps

Stop Movement

Reset actual pulse count

Y5 goes HIGH to validate
the new pulse count

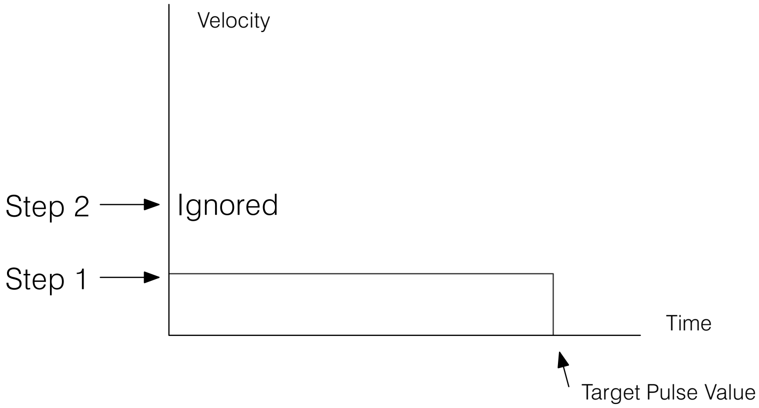
What Happens If the Ramping and Target Are Mismatched?

Any one of the positioning profiles may have a situation where the CPU detects the target pulse value in terms of distance to be less than what would be required for the the configured acceleration and deceleration distance. Whenever this occurs, the CPU will make adjustments in order to reach the target pulse distance.

The following four cases depict situations where the target value is less than the acceleration/deceleration distance. An explanation of the adjustments made by the CPU is included:

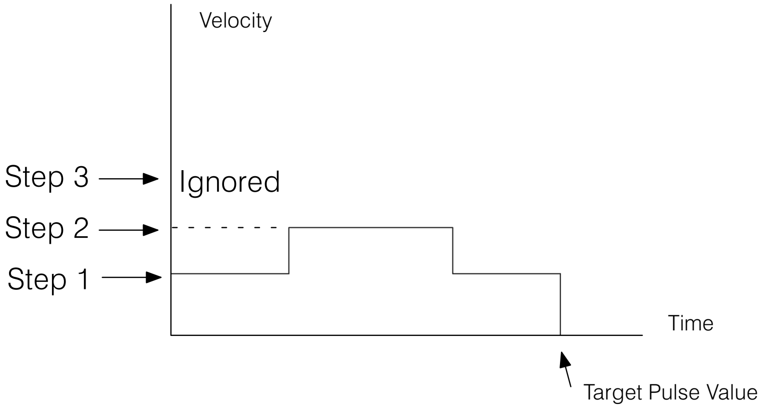
Case 1: The target value is less than the total distance required for acceleration/deceleration Steps 1 and 2.

Solution: Step 2 will be ignored and move to the target position with acceleration and deceleration at the velocity achieved by Step 1 only.



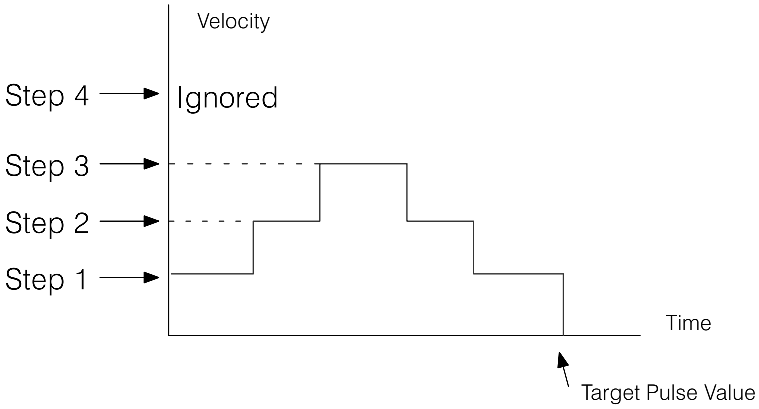
Case 2: The target value is less than the total distance required for acceleration/deceleration Steps 1, 2 and 3.

Solution: The CPU will ignore Step 3 and move to the target position with acceleration and deceleration at the velocity achieved by Steps 1 & 2 only.



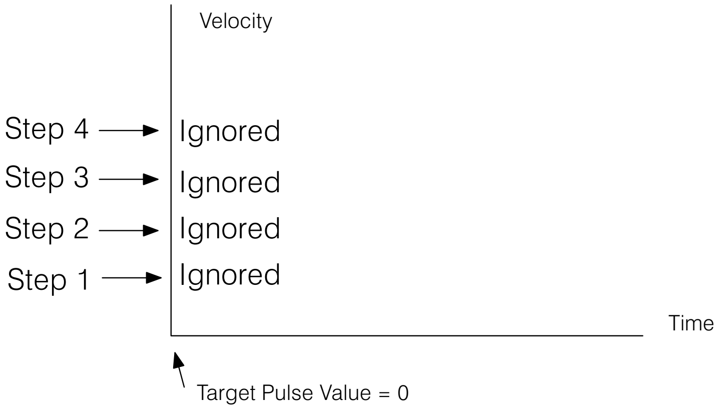
Case 3: The target value is less than the total distance required for acceleration/deceleration Steps 1 2, 3 and 4.

Solution: the CPU will ignore Step 4 and move to the target position with acceleration and deceleration at the velocity achieved by Steps 1, 2 and 3.



Case 4: The target value is set to zero.

Solution: X4 will turn off at the end of the first PLC scan to indicate that positioning is complete, but in reality nothing will happen. That is, the motor will not be activated. If the absolute mode has been selected and space equals the actual current count in V1175/V1174, the motion will also not be activated.



Activating the Positioning Profile

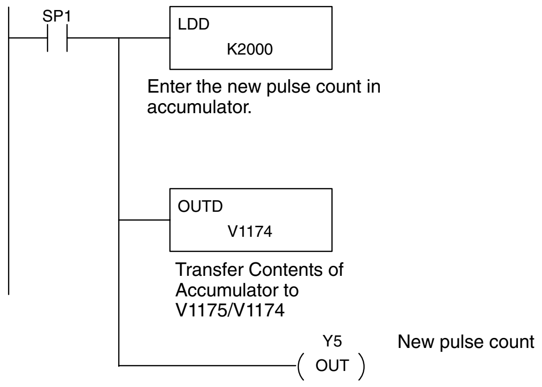
Once the pulse train output has been properly configured, additional relay ladder logic must be added in order to activate the positioning process.

The D2-CTRINT module consumes the first 16 I/O points, X0 through X7 and Y0 through Y7. This means that the I/O addresses for the other I/O modules in the chassis must begin with X10 for inputs, and Y10 for outputs. Listed below are descriptions of four (4) I/O points which can be used in the ladder logic whenever a positioning application is being programmed.

Start Relay – Y4
The positioning process is activated whenever Y4 is turned ON in the RLL program. The positioning stops when Y4 is turned OFF. If the Incremental mode is being used and Y4 turns ON again, the pulse train will be activated again. However, if the Absolute mode is being used, Y4 turns ON again and the target is equal to the actual count, the pulse train will not be activated.

Actual Value Relay – Y5
Anytime a pulse train is initiated, the actual pulse count is stored in V1175/1174. If it is necessary to change this value, Y5 must be used in a RLL program to make this change. When Y5 is used in the RLL, the value which is stored in V1175/V1174 can be changed whenever Y5 is turned ON. For example: If the pulse is started three times in Incremental mode with a target of 5000 pulses, the value in V1175/1174 will be 15000 after the third pulse train. If the actual pulse count is to be changed to a new value, say 2000, use the 32 bit instructions and add an additional rung that will turn ON Y5. When Y5 goes HIGH, the actual count of 15000 will be replaced with 2000. Refer to the RLL diagram below.

DirectSOFT32 Display


SP1
LDD
K2000
Enter the new pulse count in accumulator.
OUTD
V1174
Transfer Contents of Accumulator to V1175/V1174
Y5
OUT
New pulse count

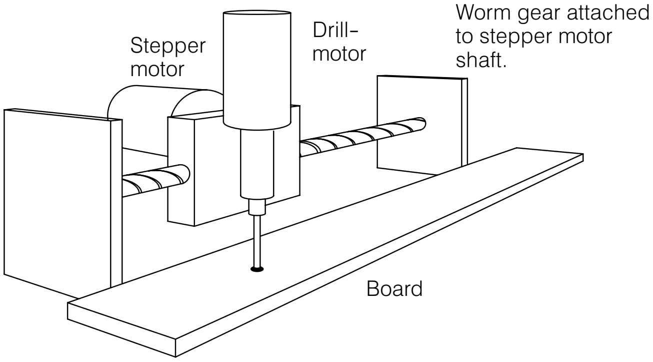
External Interrupt (X1)
This input interrupts the velocity control of the Trapezoid profile or Automatic accel/decel profile and starts the positioning operation to reach its target value.

Positioning Complete Relay (X4)
This input turns ON when it completes the positioning operation.

Putting It All Together

A Complete Program

Many items have been covered in this chapter. The best way to show how they work together is by using them in a computer program. The following example will put it all together. A Step Trapezoid profile will be configured in a completed stepper motor application. The diagram below shows the general setup.



A stepper motor is controlling the positioning of a table carrying a drill press. Clockwise motion of the motor shaft will move the table forward and counter-clockwise will move it backward. Wooden boards are being loaded in turn to a fixed position for drilling. Three holes will be drilled in each board measured from one end of the board, at 10-inches, 25-inches and 50-inches.

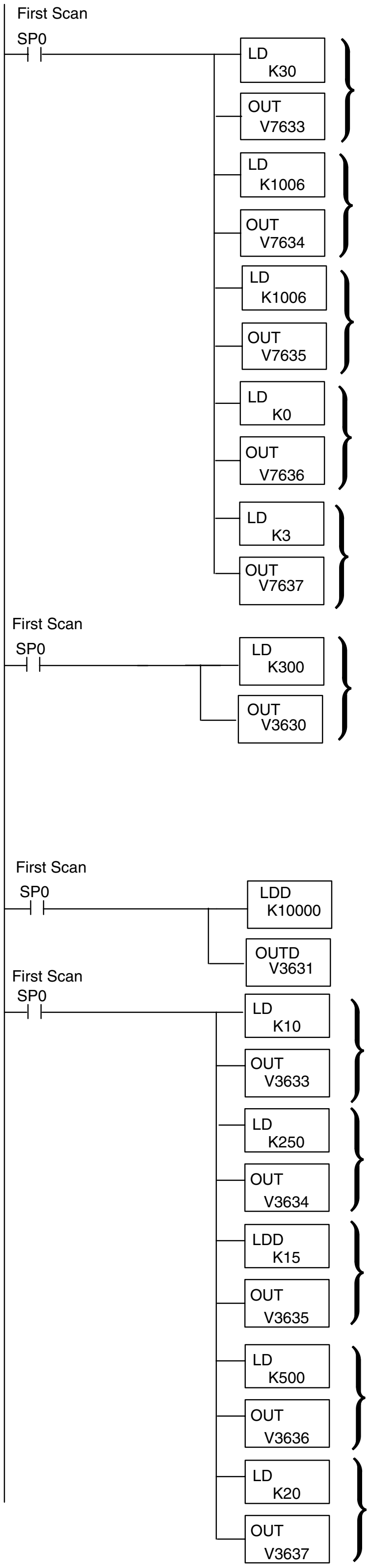
The stepper motor has a resolution of 1000 pulses per revolution. It has been determined through testing that one revolution of the stepper motor (1000 ppr) will move the drill head 1-inch. It has been calculated that it will take 10000, 25000, and 50000 pulses (in absolute terms) to reach each of the three drill positions respectively. The home position is at 0 pulses which will be the beginning of the process.

A start button, X10, will be used to initiate the process. Once this pushbutton is pressed, the drill head will accelerate to a traveling speed, then decelerate to the target position. It will do this for each of the three target positions. Anytime it reaches one of these three targets, the table will stop, and the drilling will proceed automatically. After the drill has finished drilling the third hole, the table will automatically reverse to achieve its new target, which will be the home position.

Drill Task	Pulse Count (1000 pulses=1 inch)
First Hole Location	10,000 pulses
Second Hole Location	25,000 pulses
Third Hole Location	50,000 pulses
Home Position	0 pulses

The ladder logic for this application begins on the next page.

Direct**SOFT32 Display**



Mode 30 (Pulse Output Mode)

Discrete Filtered Input at (00)
10 ms. See Chapter 8

Discrete Filtered Input at (01)
10 ms. See Chapter 8

Must be zero.

If a 3 is used here, the system provides CW or CCW pulse output. If a 103 is used here, the system provides CW or CCW using high and low signals for signal direction.

Setup Parameters:
Bit 15=0 to indicate absolute mode
Bit 14 =0 to indicate stepped trapezoid
Bits 13,12 =00 to indicate no interrupt
Bits 11–0 = positioning frequency times 10 pps
(300 x 10pps).

0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

$\underbrace{15\ 14\ 13\ 12}_{0}\ \underbrace{11\ \ldots\ 0}_{300}$

32–bit target value loaded into V3632/3631 where the Trapezoid profiles use this value to determine the number of pulses to output.

K10=1st preset acceleration (value is multiplied times 10 pps).

K250 = Number of pulses in this step.

K15=2nd preset acceleration (value is multiplied times 10 pps).

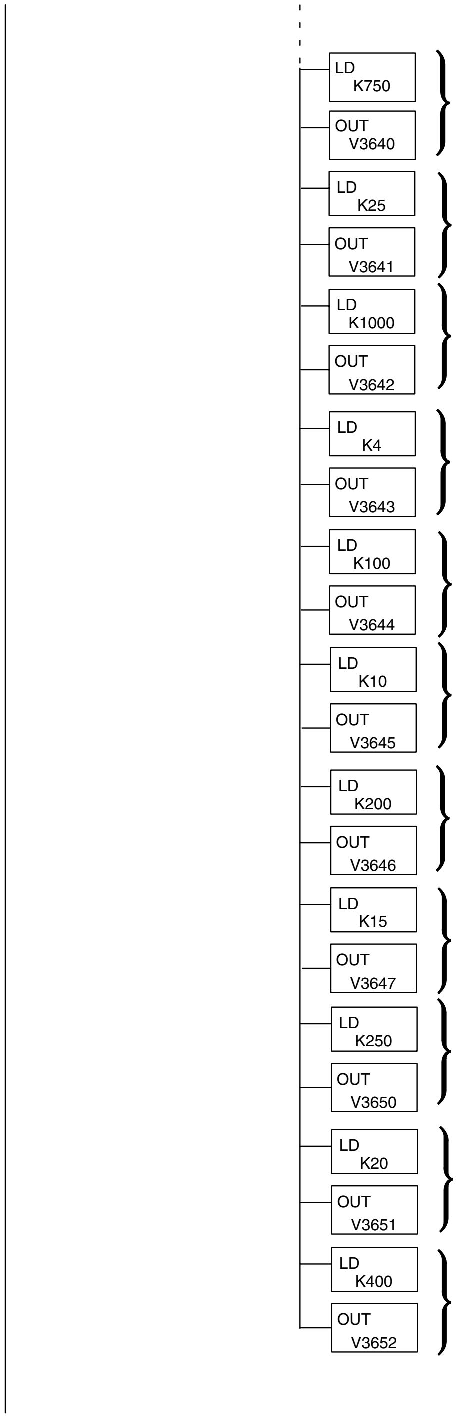
K500 = Number of pulses in this step.

K20=3rd preset acceleration (value is multiplied times 10 pps).

Continued on next page

Continued from previous page.

***Direct*SOFT32 Display**



K750 = Number of pulses in this step.

K25=4th preset acceleration (value is multiplied times 10 pps).

K1000 = Number of pulses in this step.

Note: Presets for deceleration are entered in memory in reverse order.

K4=8th preset deceleration (value is multiplied times 10 pps).

K100 = Number of pulses in this step.

K10=7th preset deceleration (value is multiplied times 10 pps).

K200 = Number of pulses in this step.

K15=6th preset deceleration (value is multiplied times 10 pps).

K250 = Number of pulses in this step.

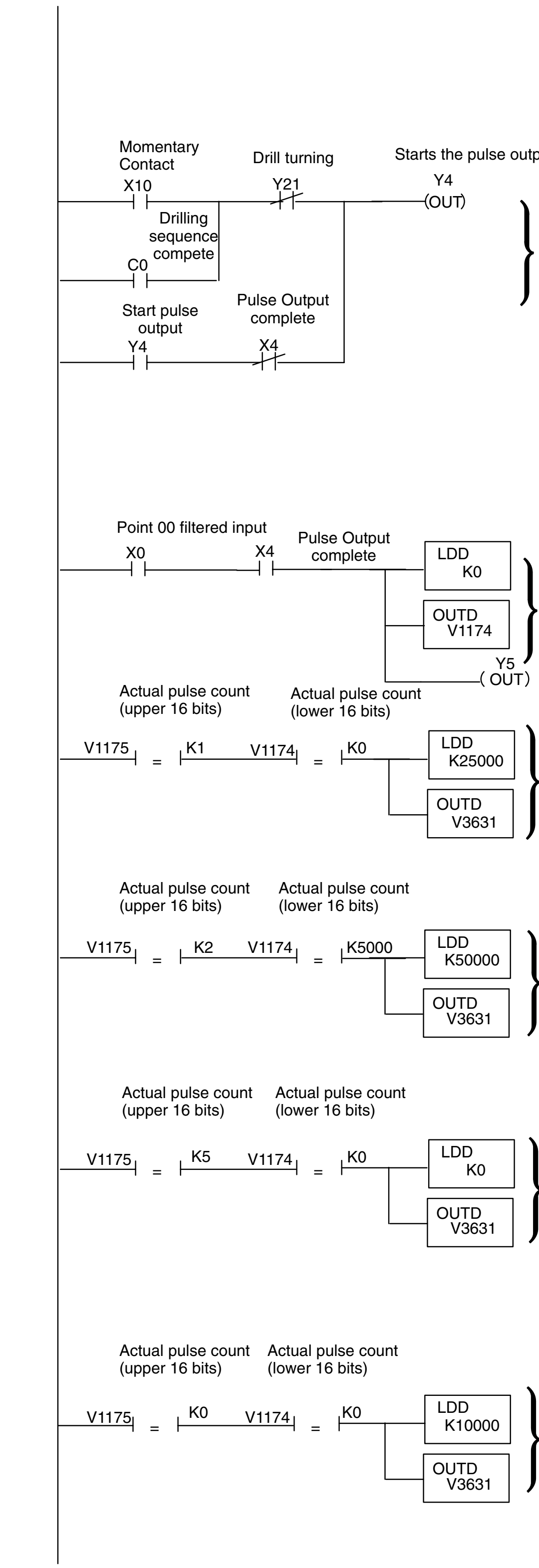
K20=5th preset deceleration (value is multiplied times 10 pps).

K400 = Number of pulses in this step.

Continued on next page

Continued from previous page.

*Direct*SOFT32 Display



X10 input latches Y4 ON, starting the positioning sequence; when the positioning is complete, Y4 is unlatched by X4 turning ON (positioning complete).

Resets the actual pulse count to zero. Zero is loaded into V1175/1174 and the control bit Y5 is turned ON to accept the transfer.

When the positioning table reaches the first positioning target (the first hole to be drilled), the pulse output count is 10,000. A new positioning target of 25,000 is entered for the second hole to be drilled. The pulse output count will be 15,000 pulses (25000–10000) since the actual pulse count is 10,000 at the first position.

When the positioning table reaches the second positioning target (the second hole to be drilled), the pulse output count is 15,000. A new positioning target of 50,000 is entered for the third hole to be drilled. The pulse output count will be 25,000 pulses (50000–25000) since the actual pulse count is 25,000 at the second position.

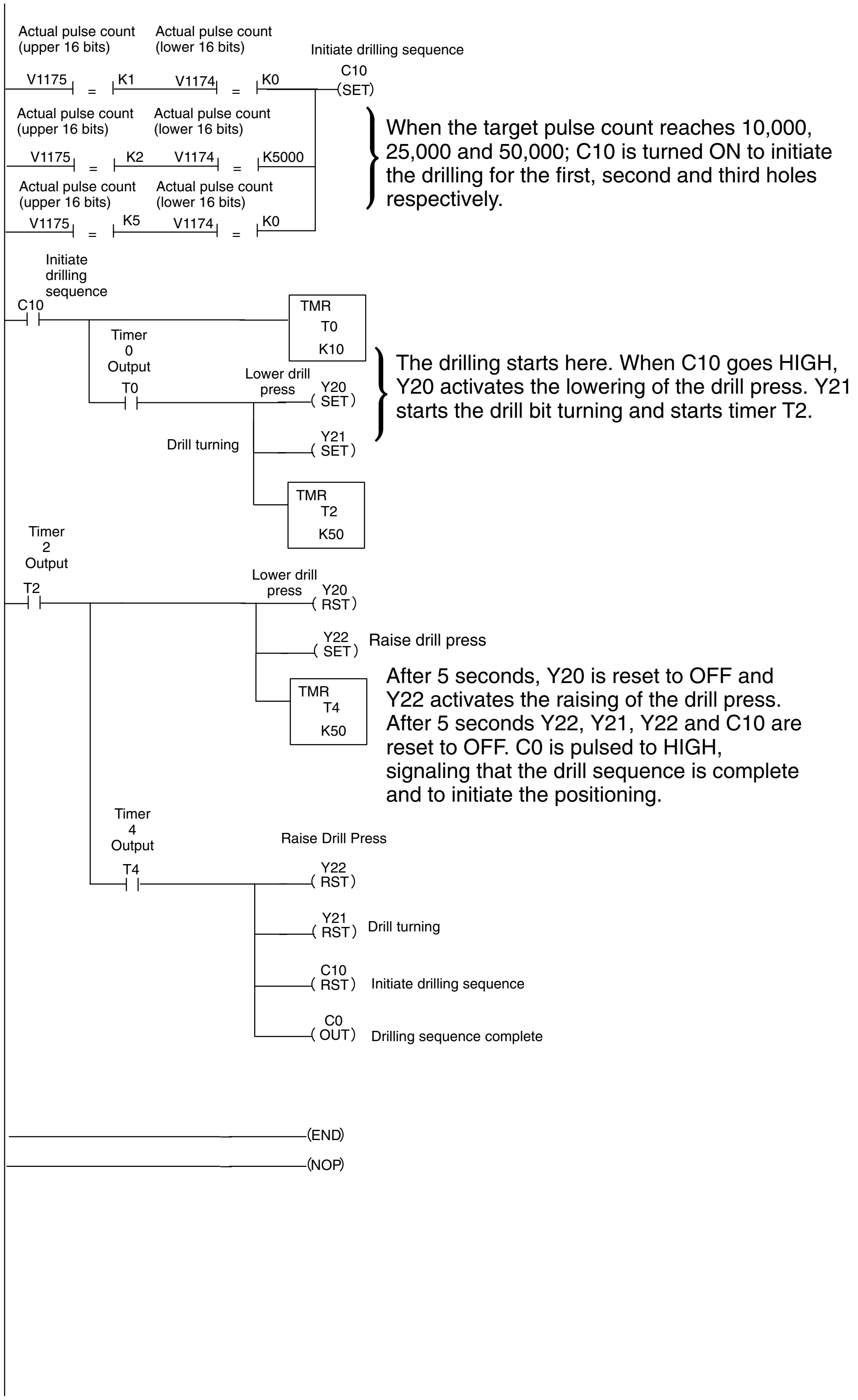
When the positioning table reaches the third positioning target (the third hole to be drilled), the pulse output count is 25,000. A new positioning target of 0 is entered for the return to home. The pulse output count will be 50,000 pulses CCW (0–50000) since the actual pulse count is 50,000 at the third position.

When the positioning table is at its home position, a pulse count of 10,000 is entered into the target. This sets up the operation to do the next board.

Continued on next page

Continued from previous page.

DirectSOFT32 Display


Actual pulse count (upper 16 bits)
Actual pulse count (lower 16 bits)
Initiate drilling sequence
V1175
=
K1
V1174
=
K0
C10
(SET)
Actual pulse count (upper 16 bits)
Actual pulse count (lower 16 bits)
V1175
=
K2
V1174
=
K5000
Actual pulse count (upper 16 bits)
Actual pulse count (lower 16 bits)
V1175
=
K5
V1174
=
K0
Initiate drilling sequence
C10
TMR
T0
K10
Timer 0 Output
T0
Lower drill press
Y20
SET
Drill turning
Y21
SET
TMR
T2
K50
Timer 2 Output
T2
Lower drill press
Y20
RST
Y22
SET
Raise drill press
TMR
T4
K50
Timer 4 Output
T4
Raise Drill Press
Y22
RST
Y21
RST
Drill turning
C10
RST
Initiate drilling sequence
C0
OUT
Drilling sequence complete
(END)
(NOP)

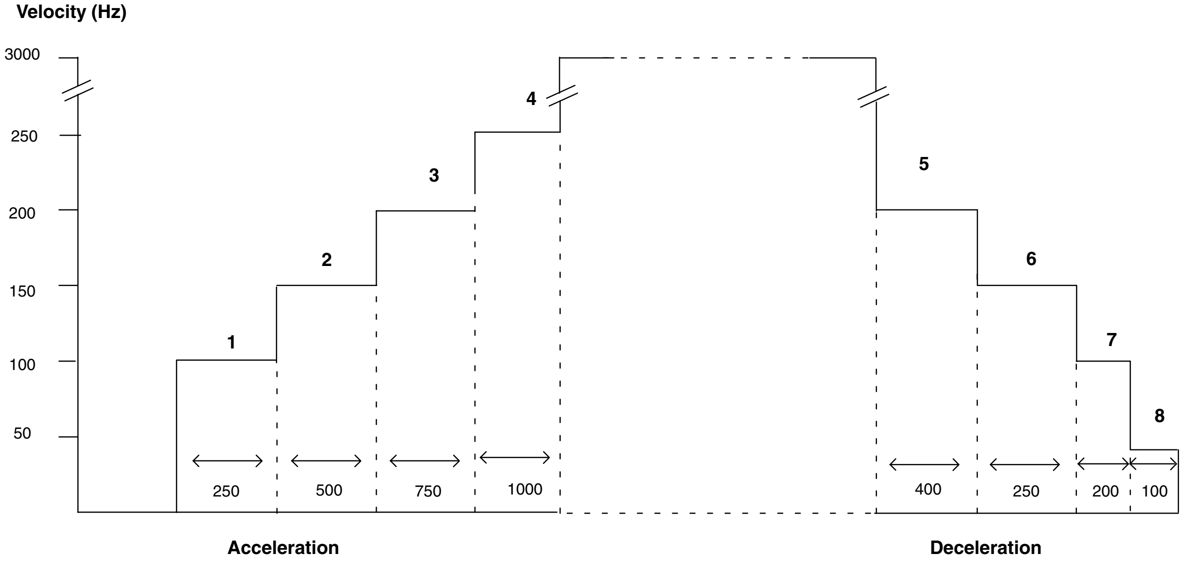
When the target pulse count reaches 10,000, 25,000 and 50,000; C10 is turned ON to initiate the drilling for the first, second and third holes respectively.

The drilling starts here. When C10 goes HIGH, Y20 activates the lowering of the drill press. Y21 starts the drill bit turning and starts timer T2.

After 5 seconds, Y20 is reset to OFF and Y22 activates the raising of the drill press. After 5 seconds Y22, Y21, Y22 and C10 are reset to OFF. C0 is pulsed to HIGH, signaling that the drill sequence is complete and to initiate the positioning.

Graphical Diagram of Positioning Profile

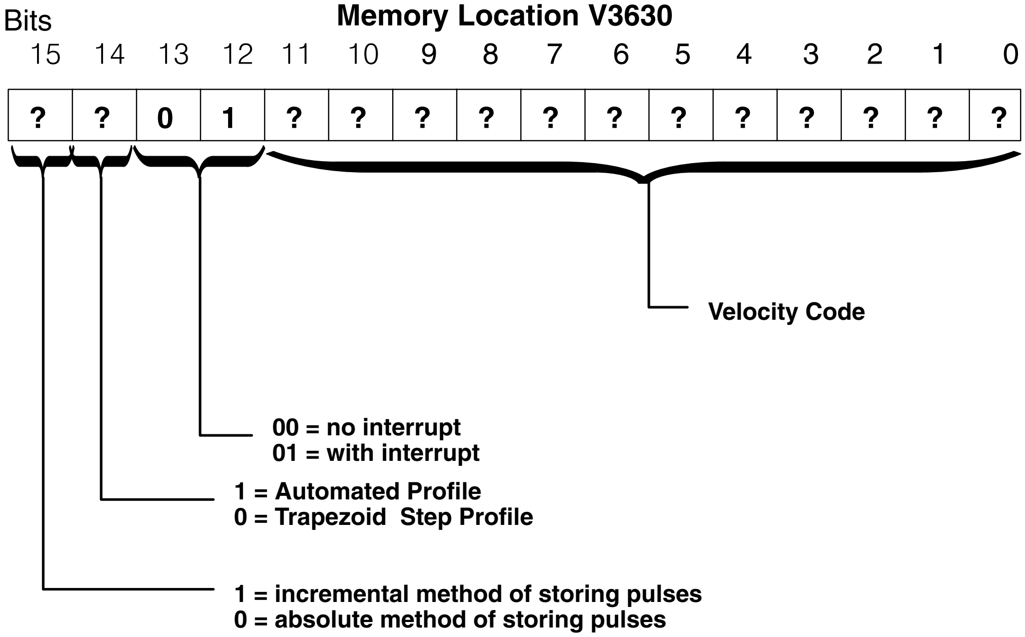
The diagram for the drilling operation just described is shown below:



All 8 presets available for acceleration and deceleration are used in the example. Notice that the acceleration and deceleration sides of the profile do not need to be the same.

The External Interrupt at X1

One option which was not shown in the completed example is the use of the external interrupt at X1. In order to make use of this option, the memory must be configured so the CPU knows to look for an interrupt at point 01, X1. This is selected in bits 12 and 13 of the 4-digit stored in V3630. This is the V–memory location where the pulse counts are specified for the way they are related to each other (incremental or absolute) and the type of positioning profile.



Memory Location V3630

Bits 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
?	?	0	1	?	?	?	?	?	?	?	?	?	?	?	?

Velocity Code
00 = no interrupt
01 = with interrupt
1 = Automated Profile
0 = Trapezoid Step Profile
1 = incremental method of storing pulses
0 = absolute method of storing pulses

Here are the upper 4-bit hexadecimal values for the possible interrupt configurations used with the positioning profile. The XXX are the digit positions for the velocity:

9xxx= Step profile w/incremental steps and interrupt at point 01 (X1).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	1	?	?	?	?	?	?	?	?	?	?	?	?

1xxx = Step profile w/absolute steps and interrupt at point 01 (X1).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	?	?	?	?	?	?	?	?	?	?	?	?

5xxx = Automated accel/decel profile with absolute current count mode and an interrupt at point 01 (X1).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	**1**	**0**	**1**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**

Dxxx = Automated accel/decel profile with incremental current count mode and an interrupt at point 01 (X1)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	**1**	**0**	**1**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**	**?**

9500 = Step profile with incremental current count mode and interrupt at point 01 (X1). Positioning velocity is 5000 pps.

9080 = Same as above but positioning velocity is 800 pps.

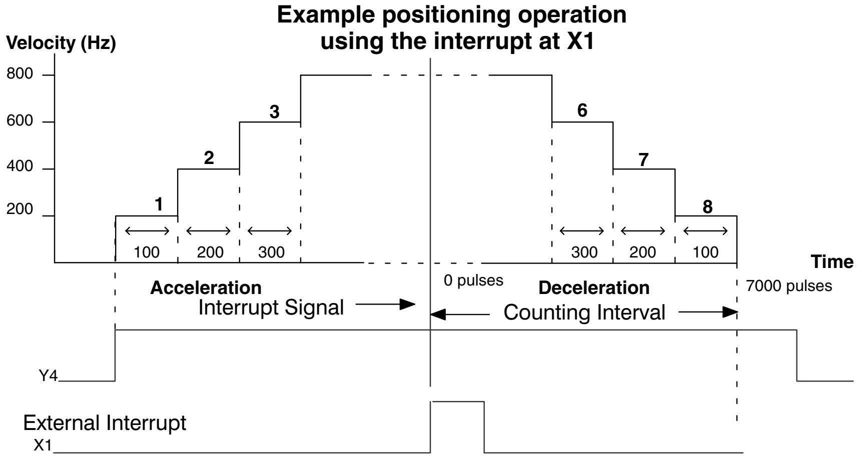
1060 = Step profile with absolute current count mode and interrupt at X1. Positioning velocity is 600 pps.

How It Works

With a standard Stepped Trapezoid Positioning profile, the module starts counting pulses and adding them to the actual count to achieve the target pulse from the time it makes the initial acceleration. However, with the Interrupt Stepped Trapezoid Positioning profile, ***the counting still occurs but the positioning to the target does not begin until the interrupt occurs***.

For the example on the facing page, the target pulse value has been set at 7000. The balance of the parameters are carried over from the previous stepped trapezoid example. As can be seen from the diagram the interrupt does not occur until the motor is far into the positioning velocity. Once it receives the interrupt, it will then start to count an additional 7000 pulses toward the target. The CPU will examine the deceleration information stored in memory, and move to the final position using the designated slope for deceleration.

The example diagram shows a Step Trapezoid profile with certain presets and the necessary values for the external interrupt option. Zeros have been placed in locations V3641 through V3652 so that the deceleration mirrors the same slope and steps as the acceleration. The table below the diagram shows the memory configuration requirements. The logic steps below the table explains what happens when the start relay Y4 is turned ON.



Memory When Using the Interrupt

Shown below is a table of the V-memory of the configuration for this particular example when using the interrupt feature at X1.

V3630	9080 (Incremental, interrupt, trapezoid,800 pps)
V3632/V3631	7000 (Target pulse value)
V3633	0020 (1st Step Acceleration=200 pps)
V3634	0100 (1st Step Distance=100 pulses)
V3635	0040 (2nd Step Acceleration=400 pps)
V3636	0200 (2nd Step Distance=200 pulses)
V3637	0060 (3rd Step Acceleration=600 pps)
V3640	0300 (3rd Step Distance=300 pulses)
V3641	0000
V3642	0000
V3643	0000
V3644	0000
V3645	0000
V3646	0000
V3647	0000
V3650	0000
V3651	0000
V3652	0000

1. When Y4 turns ON, the object moves 100 pulses at 200 pps.
2. It moves for 200 pulses at 400 pps.
3. It moves for 300 pulses at 600 pps.
4. It moves at 800 pps.
5. When interrupt occurs (X1 is ON), it moves for 7000 pulses from the interrupt position using the deceleration slope that matches that of the acceleration.
6. X4 turns on to indicate that positioning is complete.

Troubleshooting

Is The Module Working Properly?

When Mode 30 is used with the D2–CTRINT module, there are a number of areas which can be checked for proper operation. **Experience has shown that most problems occur because of improper configuration. Always re-check the CPU setup before anything else.** The questions below may help to identify any problems which may arise during the setup of the D2–CTRINT module.

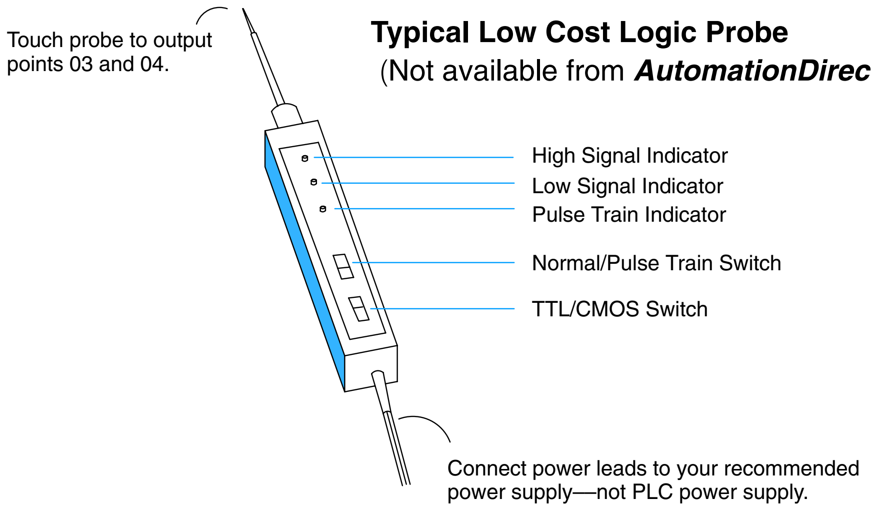
1. Do the CW and CCW indicators on the module blink when the respective pulse trains are output?
2. Do the I/O indicators turn ON when signals are sent to Points 00 to 03?
3. Is the stepper motor rotating when a signal is sent to the drive unit?
4. Is the stepper motor rotating in the proper direction?
5. Is the positioned load moving the proper distance for the corresponding number of pulses?
6. Is the load overshooting the target?
7. Is the stepper motor losing torque and/or stalling?

CW and CCW Indicators

There are two LED indicators located on the module face. These indicators are labeled CW and CCW for clockwise and counter-clockwise signals respectively. The CW indicator will blink when a CW signal is present at point 03. The CCW indicator will blink a CCW signal is present at point 04. If this is not true, follow this procedure:

1. Verify that there is a signal present at point 03 or 04. If the motor drive is connected to the module, observe the stepper motor to verify the motor shaft rotation. If the drive is not connected to the module, instrumentation can be used to check for outputs; either an oscilloscope or a logic test probe. If the logic test probe is used, it should be the type that has the option to select either TTL or CMOS settings, it should also be able to switch it to the pulse train mode. The peak signal level being output from points 03 and 04 is 5-volts with 15% tolerance. If the signals cannot be read with the TTL setting, use the CMOS setting. The probe may not be able to read the TTL because of the 15% tolerance specification.

Typical Low Cost Logic Probe
(Not available from ***AutomationDirect)***


Touch probe to output points 03 and 04.
High Signal Indicator
Low Signal Indicator
Pulse Train Indicator
Normal/Pulse Train Switch
TTL/CMOS Switch
Connect power leads to your recommended power supply—not PLC power supply.

2. If there isn't an output signal, check to make sure the configuration has the proper values in memory. Also check to be sure that Y4 is being turned ON.
3. If there still is no response, check the mode being used. The Absolute mode will output pulses until the target is achieved, i.e. Target (V3632/V3631) = Actual Pulse Count (V1175/V1174). If the target has changed up or down from the actual pulse count, then when Y4 turns ON, the pulse train will output the difference between the target and the actual count whether it is CW or CCW.
4. If there are pulse train signals at points 03 and 04, and the CW and CCW indicators are not blinking, the module is faulty.

I/O Indicators 00 through 01

The LED indicators 00 through 01 should turn ON when a signal is present at each input. Normally this signal will be sent from a field device which is connected to these points. For test purposes, these points can also be turned on using the Change Value feature in ***Direct***SOFT32 to force the input ON. Refer to the ***Direct***SOFT32 **Software Programmers Manual** to use this feature. If after forcing the input ON, and the appropriate LED does not light, the module is defective.

Verify Stepper Motor Rotation

If the stepper motor rotates in the wrong direction, the problem is an improper value that has been placed in V3631. A positive number placed there causes a CW signal output at point 03. A negative number placed there, causes a CCW signal output at point 04. Also, be sure the connection to the motor drive is correct. Be sure that point 03 of the interface module is connected to the drive's CW input terminal, and point 04 is connected to the drive's CCW input terminal. If not, reverse the connections.

Verify Calculations

If the stepper motor is turning in the proper direction, but the positioning load is not moving the correct number of pulses and distance, then check to make sure that the proper target pulse count has been entered in V3632/V3631. The motor rotations and its specifications determine how many pulses are needed for a given distance. Make sure the proper number of pulses required per revolution have been computed correctly. Also, that the proper number of steps or revolutions it will take to move the load the proper distance have the correct values computed. Also, don't forget to look at the pitch of lead screws and other drive train mechanisms which affect the distance moved. Adjust the calculations where necessary.

Motor Overshoot and Stalls

The whole idea behind a trapezoid shaped positioning profile is to get to the target as quickly as possible or as required, but allowing for the best acceleration and deceleration slopes to be sure the motor does not overshoot the target or stall.

Determining the slopes is largely dependent on the inertia characteristics of the load which is being moved. High inertia loads will have a tendency to keep moving after they have reached the target position unless the speed is slowed to overcome the inertia problem. Use a conservative gradual slope where possible.

As for stalling problems check the specifications of the motor. Is it being operated at the proper velocity when going through the base speed area of it's torque curve? Are pulses being sent to the motor too fast for the motor's size and for the load being moved? Make the height of each acceleration step shorter if necessary.