

D2-CTRINT
Counter Interface Module

Manual Number D2-CTRIF-M



WARNING

Thank you for purchasing automation equipment from **Automationdirect.com™**, doing business as, **AutomationDirect**. We want your new **DirectLOGIC™** automation equipment to operate safely. Anyone who installs or uses this equipment should read this publication (and any other relevant publications) before installing or operating the equipment.

To minimize the risk of potential safety problems, you should follow all applicable local and national codes that regulate the installation and operation of your equipment. These codes vary from area to area and usually change with time. It is your responsibility to determine which codes should be followed, and to verify that the equipment, installation, and operation are in compliance with the latest revision of these codes.

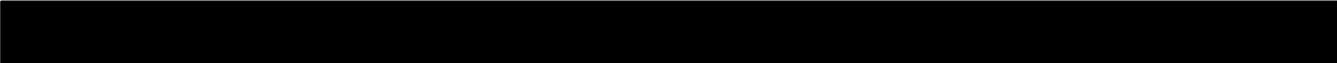
At a minimum, you should follow all applicable sections of the National Fire Code, National Electrical Code, and the codes of the National Electrical Manufacturer's Association (NEMA). There may be local regulatory or government offices that can also help determine which codes and standards are necessary for safe installation and operation.

Equipment damage or serious injury to personnel can result from the failure to follow all applicable codes and standards. We do not guarantee the products described in this publication are suitable for your particular application, nor do we assume any responsibility for your product design, installation, or operation.

*Our products are not fault-tolerant and are not designed, manufactured or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the product could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). **AutomationDirect** specifically disclaims any expressed or implied warranty of fitness for High Risk Activities.*

For additional warranty and safety information, see the Terms and Conditions section of our Desk Reference. If you have any questions concerning the installation or operation of this equipment, or if you need additional information, please call us at 770-844-4200.

This publication is based on information that was available at the time it was printed. At **AutomationDirect** we constantly strive to improve our products and services, so we reserve the right to make changes to the products and/or publications at any time without notice and without any obligation. This publication may also discuss features that may not be available in certain revisions of the product.

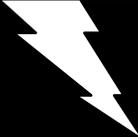


Trademarks

This publication may contain references to products produced and/or offered by other companies. The product and company names may be trademarked and are the sole property of their respective owners. **AutomationDirect** disclaims any proprietary interest in the marks and names of others.

**Copyright 2003, Automationdirect.com™ Incorporated
All Rights Reserved**

No part of this manual shall be copied, reproduced, or transmitted in any way without the prior, written consent of **Automationdirect.com™** Incorporated. **AutomationDirect** retains the exclusive rights to all information included in this document.



AVERTISSEMENT



Nous vous remercions d'avoir acheté l'équipement d'automatisation de **Automationdirect.com**™, en faisant des affaires comme, **AutomationDirect**. Nous tenons à ce que votre nouvel équipement d'automatisation **DirectLOGIC**™ fonctionne en toute sécurité. Toute personne qui installe ou utilise cet équipement doit lire la présente publication (et toutes les autres publications pertinentes) avant de l'installer ou de l'utiliser.

Afin de réduire au minimum le risque d'éventuels problèmes de sécurité, vous devez respecter tous les codes locaux et nationaux applicables régissant l'installation et le fonctionnement de votre équipement. Ces codes diffèrent d'une région à l'autre et, habituellement, évoluent au fil du temps. Il vous incombe de déterminer les codes à respecter et de vous assurer que l'équipement, l'installation et le fonctionnement sont conformes aux exigences de la version la plus récente de ces codes.

Vous devez, à tout le moins, respecter toutes les sections applicables du Code national de prévention des incendies, du Code national de l'électricité et des codes de la National Electrical Manufacturer's Association (NEMA). Des organismes de réglementation ou des services gouvernementaux locaux peuvent également vous aider à déterminer les codes ainsi que les normes à respecter pour assurer une installation et un fonctionnement sûrs.

L'omission de respecter la totalité des codes et des normes applicables peut entraîner des dommages à l'équipement ou causer de graves blessures au personnel. Nous ne garantissons pas que les produits décrits dans cette publication conviennent à votre application particulière et nous n'assumons aucune responsabilité à l'égard de la conception, de l'installation ou du fonctionnement de votre produit.

Nos produits ne sont pas insensibles aux défaillances et ne sont ni conçus ni fabriqués pour l'utilisation ou la revente en tant qu'équipement de commande en ligne dans des environnements dangereux nécessitant une sécurité absolue, par exemple, l'exploitation d'installations nucléaires, les systèmes de navigation aérienne ou de communication, le contrôle de la circulation aérienne, les équipements de survie ou les systèmes d'armes, pour lesquels la défaillance du produit peut provoquer la mort, des blessures corporelles ou de graves dommages matériels ou environnementaux ("activités à risque élevé"). La société **AutomationDirect** nie toute garantie expresse ou implicite d'aptitude à l'emploi en ce qui a trait aux activités à risque élevé.

Pour des renseignements additionnels touchant la garantie et la sécurité, veuillez consulter la section Modalités et conditions de notre documentation. Si vous avez des questions au sujet de l'installation ou du fonctionnement de cet équipement, ou encore si vous avez besoin de renseignements supplémentaires, n'hésitez pas à nous téléphoner au 770-844-4200.

Cette publication s'appuie sur l'information qui était disponible au moment de l'impression. À la société **AutomationDirect**, nous nous efforçons constamment d'améliorer nos produits et services. C'est pourquoi nous nous réservons le droit d'apporter des modifications aux produits ou aux publications en tout temps, sans préavis ni quelque obligation que ce soit. La présente publication peut aussi porter sur des caractéristiques susceptibles de ne pas être offertes dans certaines versions révisées du produit.

Marques de commerce

La présente publication peut contenir des références à des produits fabriqués ou offerts par d'autres entreprises. Les désignations des produits et des entreprises peuvent être des marques de commerce et appartiennent exclusivement à leurs propriétaires respectifs. **AutomationDirect**™ nie tout intérêt dans les autres marques et désignations.

Copyright 2003, **Automationdirect.com**™ Incorporated

Tous droits réservés

Nulle partie de ce manuel ne doit être copiée, reproduite ou transmise de quelque façon que ce soit sans le consentement préalable écrit de la société **Automationdirect.com**™ Incorporated. **AutomationDirect** conserve les droits exclusifs à l'égard de tous les renseignements contenus dans le présent document.

Manual Revisions



If you contact us in reference to this manual, be sure and include the revision number.

Title: DL205 High Speed Counter Interface Manual, Rev B

Manual Number: D2-CTRIF-M

Issue	Date	Description of Changes
Original	10/94	Original Issue
Rev A	6/98	Minor corrections
2nd Edition	1/03	Upgrade for DL250-1/260 and rewritten for clarity
Rev A	3/03	Minor corrections

Table of Contents

Chapter 1: Getting Started

Manual Introduction	1-2
The Purpose of this Manual	1-2
Supplemental Manuals	1-2
Who Should Read This Manual	1-2
Technical Support	1-2
Conventions Used	1-3
Key Topics for Each Chapter	1-3
Module Overview	1-4
Low-Cost Motion Control Solution	1-4
Typical Applications	1-4
Modes of Operation	1-4
Physical Characteristics	1-5
Input and Output Terminals	1-5
Module Indicators	1-5
Specifications	1-6
General	1-6
Input Specifications	1-6
Output Specifications	1-6
Modes of Operation	1-7
Mode 10: Up Counter	1-7
Mode 20: Standard or Quadrature Up/Down Counter (DL240/250-1/260 Only)	1-9
Mode 30: Pulse Output (DL240/250-1/260 Only)	1-11
Mode 40: External Interrupts	1-13
Mode 50: Pulse Catch Inputs	1-14
Mode 60: Discrete Inputs with Filter	1-15

Chapter 2: Installation & Field Wiring Guidelines

How to Install the Module	2-2
Insert the Module	2-2
General Guidelines for Field Wiring	2-3
Wiring Diagram for Modes 10, 20, 40, 50 and 60	2-4
Wiring Diagram for Mode 30	2-4
Solid State Field Device Wiring	2-5
NPN Field Device Example	2-5
PNP Field Device Example	2-5

Chapter 3: Mode 10 –High Speed UP Counters

Using the UP Counters, Mode 10	3-2
DL240/250-1/260 Applications	3-2
DL230 Applications	3-3
Understanding V-Memory Setup Locations	3-4
Setting Up the CPU	3-6
Configuring the V-Memory	3-6
Step 1: Entering the Selected Mode	3-6
Step 2: Select the Preset Mode (DL240/250-1/260 Only)	3-7
Step 3: Using Presets	3-9
Triggering Presets to Outside Events	3-10

Custom Configurations	3-12
DL230 Custom Configuration	3-12
DL240/250–1/260 Custom Configuration	3-13
Writing the Control Program	3-14
Example 1: UP Counters Without Presets	3-15
Example 2: UP Counters with Multiple Presets	3-17
Example 3: Speed Control for an AC Motor	3-19
Example 4: Counting Pulses at a High Frequency Where Accuracy is Critical	3-22
Troubleshooting	3-24
Things to check	3-24
Counter Doesn't Count	3-24
No Input Signal	3-24
LED's Do Not Light	3-26
Non-Synchronous Pulsing	3-26
No Reset	3-26
Not Jumping to Interrupt	3-26
Not Returning from Interrupt	3-26
Chapter 4: Mode 20 –UP/DOWN Counter	
Using the UP/DOWN Counter, Mode 20 (DL240/250–1/260 Only)	4-2
UP/DOWN Counting Overview	4-3
Encoders In General	4-3
Standard UP/DOWN Counting	4-3
Quadrature UP/DOWN Counting	4-4
How a Quadrature Encoder Works	4-5
Understanding V-Memory Setup Locations	4-6
Setting Up the CPU	4-8
Configuring the V-Memory	4-8
Step 1: Entering the Mode Selected	4-8
Step 2: Select the Preset Mode	4-9
Step 3: The Presets	4-11
Loading the Presets	4-11
Using Negative Presets	4-11
Triggering Presets to Outside Events	4-11
The Other Channels	4-14
Filtered Input–Point 03	4-14
Discrete Filtered Inputs	4-14
Point 02 – Reset w/o an External Interrupt	4-15
External Reset with Interrupt	4-15
Custom Configurations	4-16
Mode 20 Custom Configuring	4-16
Writing the Control Program for the UP/DOWN Counter	4-17
Example 1: UP/DOWN Counting with an Interrupt	4-18
Example 2: An UP/DOWN Counter with Standard Inputs	4-19
Example 3: Quadrature Counting	4-21
Troubleshooting	4-24
What Can Go Wrong?	4-24
Counter Doesn't Count	4-25
LED's Do Not Light	4-26
Non-Synchronous Pulsing	4-26
No Reset	4-26
Not Jumping to Interrupt	4-26
Not Returning from Interrupt	4-26
Rotary Encoders	4-26

Chapter 5: Mode 30 –Pulse Train Outputs

Using the Pulse Train Outputs, Mode 30 (DL240/250–1/260 only)	5–2
D2–CTRINT Limitations	5–3
Open Loop Stepper Motor System	5–3
Understanding V-Memory Setup Locations	5–5
Default Settings	5–6
Setting Up the CPU	5–7
Configuring the V–Memory	5–7
Assigning Module Functions	5–8
Custom Configurations	5–9
Custom Options for Mode 30	5–9
Determine the Positioning Profile	5–10
Automatic Accel/Decel Trapezoid Profile	5–10
Understanding the Positioning Profile of Setup Locations	5–12
Y4 use in the Absolute Mode: Bit 15 of V3630 set to 0	5–12
Y4 use in the Incremental Mode: Bit 15 of V3630 set to 1	5–12
Y4 use with an Interrupt –Bit 12 of V3630 set to 1	5–13
Actual Count V1175/V1174	5–13
The Automatic Accel/Decel Profile	5–14
Configuring V-Memory for the Method of Your Choice	5–14
Step 1: Setup V3630	5–15
Step 2: Set the Target Pulse Count and Shaft Direction	5–15
Step 3: Set the Starting Velocity	5–16
Step 4: Entering the Acceleration Time	5–16
Step 5: Entering the Deceleration Time	5–16
The Step Trapezoid Profile	5–18
Configuring V-Memory	5–18
Step Trapezoidal Profile Example	5–19
Step 1: Setup V3630	5–20
Step 2: Setup the Target Pulse Count	5–20
Step 3: Select the Height and Width	5–21
Step 4: Enter the Step Information Into Memory	5–22
Ladder Logic Example	5–23
The Straight Velocity Profile	5–25
Configuring V-Memory	5–25
Step 1: Setup V3630	5–25
Step 2: Set the Direction for Shaft Rotation	5–25
Step 3: Set Shaft Velocity	5–26
Velocity Control Used to Find the Home Position	5–26
Straight Velocity Profile Example	5–26
What Happens If the Ramping and Target Are Mismatched?	5–28
Activating the Positioning Profile	5–30
Putting It All Together	5–31
A Complete Program	5–31
Graphical Diagram of Positioning Profile	5–36
The External Interrupt at X1	5–37
How It Works	5–38
Memory When Using the Interrupt	5–39
Troubleshooting	5–40
Is The Module Working Properly?	5–40
CW and CCW Indicators	5–40
I/O Indicators 00 through 01	5–41
Verify Stepper Motor Rotation	5–41
Verify Calculations	5–41
Motor Overshoot and Stalls	5–41

Chapter 6: Mode 40 –High Speed Interrupts

Using the Interrupt Inputs, Mode 40	6-2
High Speed Interrupts	6-3
High Speed Interrupts Process	6-4
Single Input Signal Processing	6-4
Multiple Signals from Same Input	6-4
Simultaneous Skeans from Different Inputs	6-5
Leading Edge Triggering	6-5
Understanding V-Memory Setup Locations	6-6
Default Settings	6-7
Explanation the Values	6-7
Custom Configuration	6-8
Setting Up the CPU for the Interrupts	6-9
Configuring the V-Memory	6-9
Step 1: Enter the Mode	6-9
Step 2: How Many Interrupts	6-10
Step 3: Configure the V-Memory	6-10
Two Types of Interrupts	6-11
Troubleshooting	6-13
Interrupt Device Working But No Interrupt	6-13
Interrupt Occurs, But No Normal Scan Afterwards	6-13

Chapter 7: Mode 50 –Pulse Catch Inputs

Using the Pulse Catch Inputs, Mode 50	7-2
Pulse Catching Explained	7-3
Understanding V-Memory Setup Locations	7-4
Default Settings	7-5
Custom Configuration	7-6
Setting UP the CPU for the Pulse Catch Inputs	7-7
Configuring the V-Memory	7-7
Step 1: Enter the Mode Selected	7-7
Step 2: How Many Pulse Catch Inputs	7-8
Step 3: Configure the V-Memory	7-8
Troubleshooting	7-10

Chapter 8: Mode 60 –Discrete/Filter Inputs

Using the Discrete Filtered Inputs, Mode 60	8-2
Discrete Filtered Inputs	8-3
Understanding V-Memory Setup Locations	8-4
Default Settings	8-5
Custom Configuration	8-6
Setting Up the CPU for Discrete Filtered Inputs	8-7
Configuring the V-Memory	8-7
Step 1: Enter the Mode	8-7
Step 2: How many Channels	8-8
Step 3: Configure the V-Memory	8-8
Troubleshooting	8-10

Getting Started

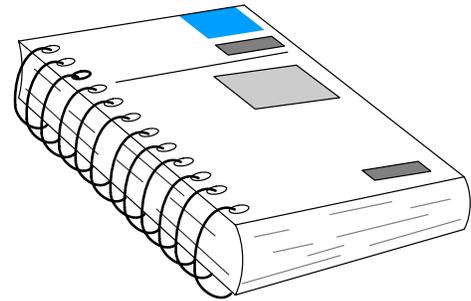
In This Chapter. . . .

- Introduction
 - Physical Characteristics
 - Specifications
 - Modes of Operation
 - Motion Control Made Easy - Four Simple Steps
-

Manual Introduction

The Purpose of this Manual

This manual describes the installation of the D2-CTRINT Counter Interface Module, and how to select the operating modes of the high-speed counting features. It also shows several ways to use the data in a PLC program.



Supplemental Manuals

The following manuals are essential for the proper use of your DL05 DeviceNet Slave Module.

- DL205 PLC User Manual part number **D2-USER-M**
This manual contains detailed descriptions of the instructions used to setup and control the counter module. It also contains the complete I/O Memory Map which will be helpful.
- The **Direct**SOFT32 Programmer Software Users Manual

Who Should Read This Manual

If you understand the DL205 instruction set and system setup requirements, this manual will provide the information you need to install and use the Counter Interface Module. This manual *is not* intended to be a tutorial on motion control theory, but rather a user reference manual for the D2-CTRINT Counter Interface Module.

Technical Support

We strive to make our manuals the best in the industry and rely on your feedback in reaching our goal. If you cannot find the solution to your particular application, or, if for any reason you need additional technical assistance, please call us at

770-844-4200.

Our technical support team is glad to work with you in answering your questions. They are available **weekdays from 9:00 a.m. to 6:00 p.m. Eastern Time**. We also encourage you to visit our website where you can find technical and nontechnical information about our products and our company.

www.automationdirect.com

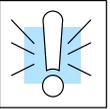
Conventions Used



The “light bulb” icon in the left-hand margin indicates a **tip** or **shortcut**.



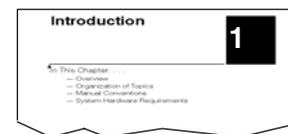
The “note pad” icon in the left-hand margin indicates a **special note**.



The “exclamation mark” icon in the left-hand margin indicates a **warning** or **caution**. These are very important because the information may help you prevent serious personal injury or equipment damage.

Key Topics for Each Chapter

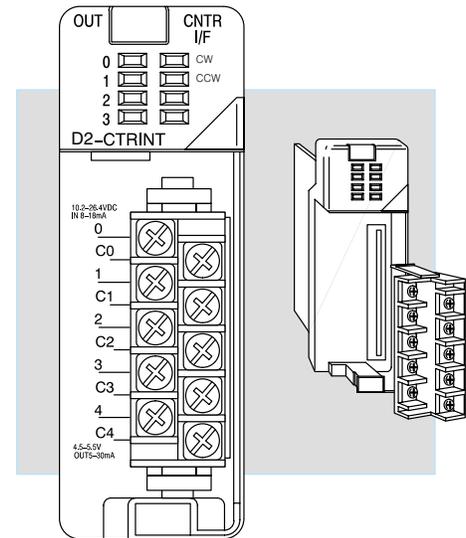
The beginning of each chapter will list the key topics that can be found in that chapter.



Module Overview

Low-Cost Motion Control Solution

Many machine control applications require various types of simple high-speed control. These applications usually involve some type of motion control, or high-speed interrupts for time-critical events. The DL205 product family solves this traditionally expensive problem with built-in CPU features which are accessed by the D2-CTRINT, Counter Interface Module.



Typical Applications

The Counter Interface Module is well suited for monitoring and controlling various types of high-speed input signals (pulses) which cannot be measured with standard discrete input modules. The Counter Interface Module works with a special part of the DL205 CPU which operates independently of the CPU program scan. This provides the accurate measurement and capturing of high-speed pulses, short duration discrete inputs, etc. The pulses can be provided from several different devices, such as, pulse encoders, sensors and limit switches.

A typical application is an encoder which is connected to a rotating device. The encoders produce a given number of pulses for each rotation of the shaft, so control decisions can be made based on the number of pulses counted.

Modes of Operation

The D2-CTRINT module operates at up to 5 KHz and allows you to access several different counting options. These counter features are built into the CPU, but the Counter Interface Module must be used to connect these features to a particular process. There are several options available:

- High Speed counter with up to 24 presets and built-in interrupt subroutines
- Quadrature encoder input to measure counts and clockwise or counter clockwise direction
- Programmable pulse output with external interrupts and separate acceleration and deceleration profiles for positioning and velocity control
- External interrupt inputs for immediate response to critical or time sensitive tasks
- Pulse catch feature to read up to 4 inputs, with each input having a pulse width as small as 0.1ms
- Programmable discrete filtering (both on and off delay up to 99ms) to ensure input signal integrity

Physical Characteristics

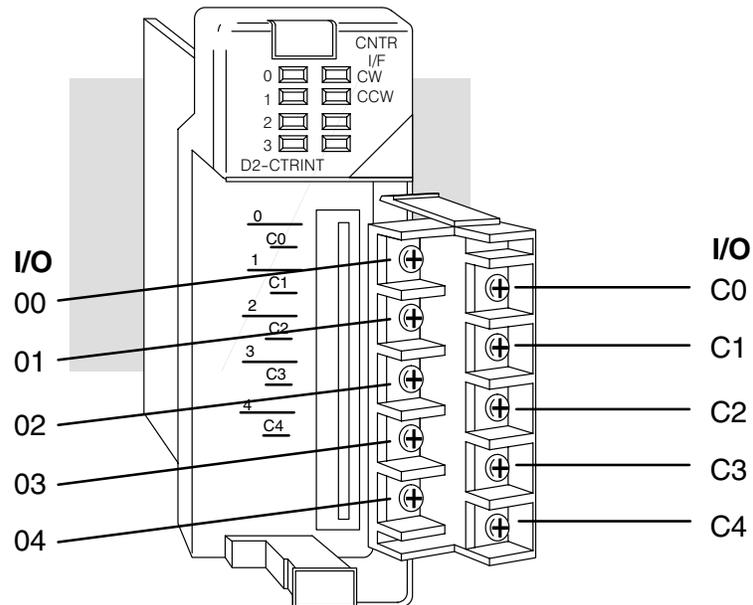
Input and Output Terminals

The D2-CTRINT, Counter Interface Module, is an interface for several counting methods. It also provides a means to use embedded features of the CPU for interrupt, pulse catching, pulse output and discrete input. As a result, the module has both input and output terminals. Sometimes the same terminals are used as inputs in one mode and outputs in a different mode. The terminal block can be removed for ease of wiring.

There are pinch tabs on the top and bottom of the terminal block. To remove the terminal block, squeeze the pinch tabs and pull the terminal block from the module.

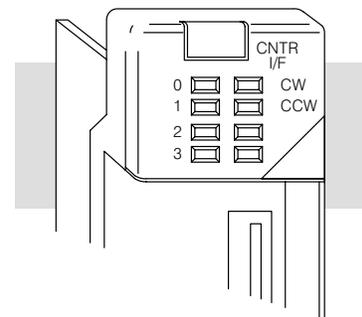


WARNING: Field device power may still be present on the terminal block even though the power has been disconnected from the PLC system. To minimize the risk of electrical shock, check all field device power before removing the connector.



Module Indicators

The Counter Interface Module also has LEDs to indicate the module operation. The module has four indicators: one for each input, one for clockwise and one for counter clockwise operation (for those modes which use direction.)



Specifications

The following tables provide specifications for the Counter Interface Module.

General

Modules per System	1 only
Module Type	Discrete Input/Pulse Output
Installation Requirements	Must install in slot 0, next to CPU
Power Budget Requirement	200mA, 5 VDC (from base)
Digital I/O Consumed Input Points Output Points	8 input points (X0–X7) 8 output points (Y0–Y7)
Field Wiring	Standard 8pt. removable terminal block
Operating Temperature	32 to 131° F (0 to 55° C)
Storage Temperature	–4 to 158° F (–20 to 70° C)
Relative Humidity	30 to 95% (non-condensing)
Environmental air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

Input Specifications

Input	4 points, current sink/source, 5KHz maximum
Minimum Pulse Width	100 μ s (0.1ms)
Input Voltage Range	12 or 24 VDC \pm 15%
Maximum Voltage	30 VDC
Rated Input Current	10mA typical, 13mA maximum
Minimum ON Voltage	8.0 VDC
Maximum OFF Voltage	1.0 VDC
Minimum ON Current	8.0mA
Maximum OFF Current	1.0mA
OFF to ON Response	Less than 30 μ s
ON to OFF Response	Less than 30 μ s

Output Specifications

Output	2 points, current sinking, 5KHz maximum
External Power Supply Requirements	5.0 VDC \pm 10%
Output Voltage Range	5.0 VDC \pm 15%
Maximum Voltage	5.5 VDC
Maximum Load Current	30mA
Minimum Load Voltage	4.5 VDC
Leakage Current	Less than 0.1mA at 5.5 VDC
Inrush Current	0.5A (for 10ms)
OFF to ON Response	Less than 30 μ s
ON to OFF Response	Less than 30 μ s

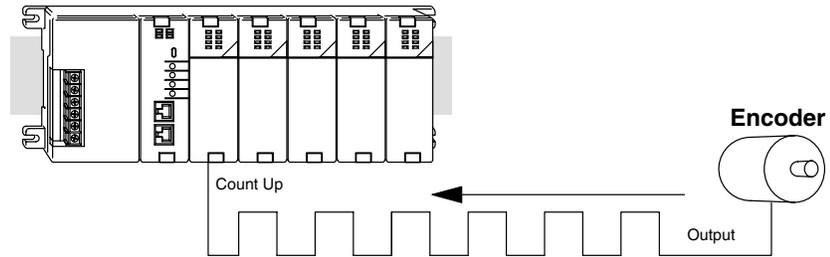
Modes of Operation

Mode 10: Up Counter

The D2-CTRINT module provides easy access to six modes of operation for the DL240/250-1/260 and four modes for the DL230.

There are two high-speed UP counters (5KHz) embedded internally in the DL240/250-1/260 CPUs and one in the DL230 CPU. These counters work independently of the CPU scan. When the counter reaches a preset value (up to 24 per counter), the CPU stops normal operations and executes an interrupt subroutine which is associated with the UP counter (one interrupt subroutine per UP counter).

Counting Input Pulses



Each of the 24 presets has a special relay which is used to trigger events whenever the preset equals the current value. Absolute presets can be used to compare the preset directly to the current count, or incremental presets can be used to compare the current count to the current preset plus the accumulated value of the previous presets.

The subroutine can be programmed to perform whatever tasks are necessary. For example, Immediate I/O instructions can be used in the interrupt subroutine to provide a very fast response. Once the interrupt execution is complete, the CPU resumes normal program execution from the point where the interrupt occurred.

Turning the ENABLE of the counter ON and OFF will start and halt the counting.

There are three ways to reset the counters, (1) signal sent from an input module to the CPU where the input is read during the normal scan update, (2) signal sent directly into the module via point 02 and/or point 03, and (3) using an internal relay of the PLC to perform the reset via the program. The second choice is the quickest reset.



This module remains an OPEN LOOP counting solution for your application and depends on the system designer to close the loop properly.

The following diagram provides a quick overview of the specifications and operations sequence for the Up Counters.

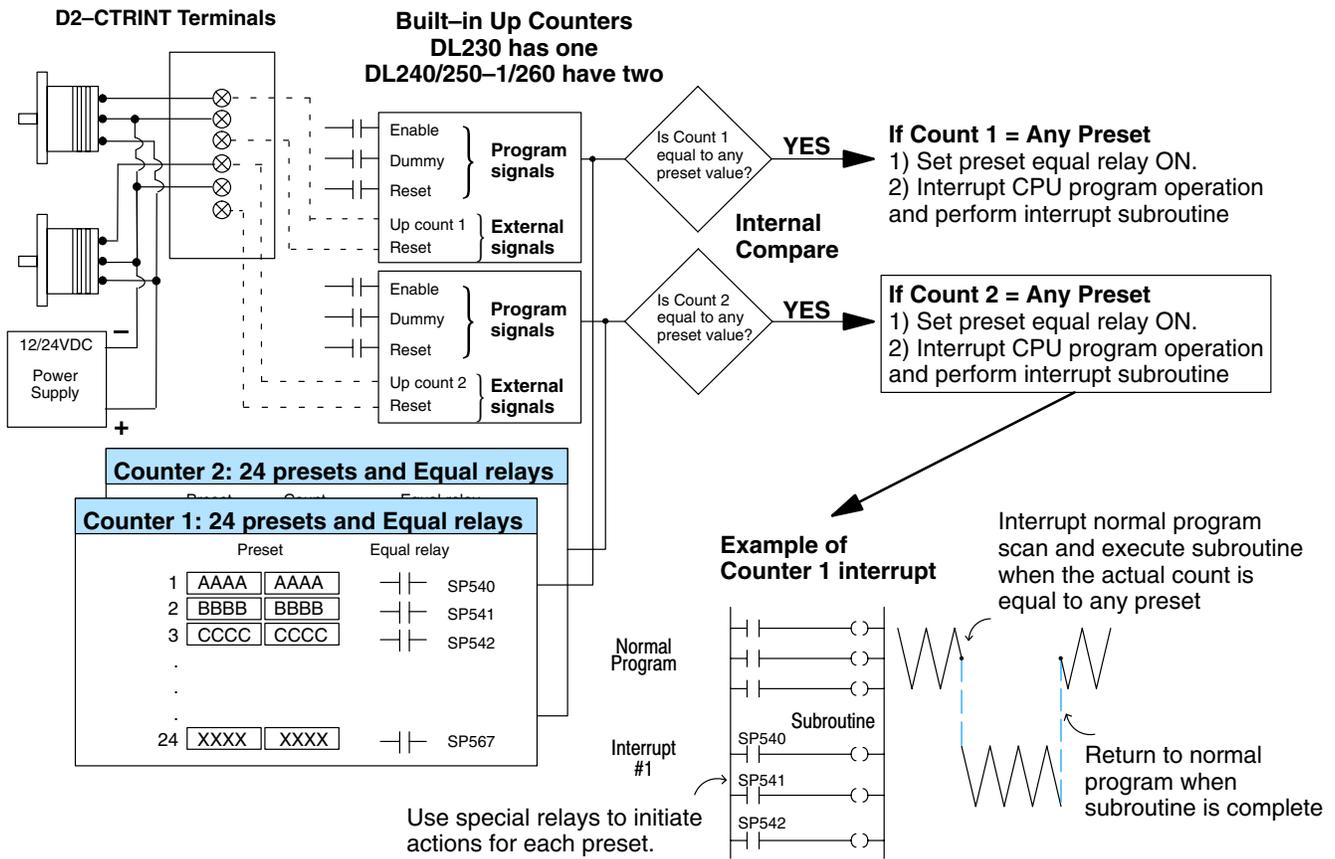
Specifications for the Up counters

Input voltage	12 or 24VDC
Frequency	5KHz maximum
Minimum pulse width	100 μSec
Maximum count	99,999,999
Preset types	Absolute or Incremental
Number of presets	24 per counter
Interrupt priority	Counter 1 over Counter 2

Hardware Connections

(DL240/250-1/260)

Input 1:	Up count of UP counter 1 (pt. 00)
Input 2:	Up count of UP counter 2 (pt. 01)
Input 3:	External Counter 1 reset (pt. 02)
Input 4:	External Counter 2 reset (pt. 03)



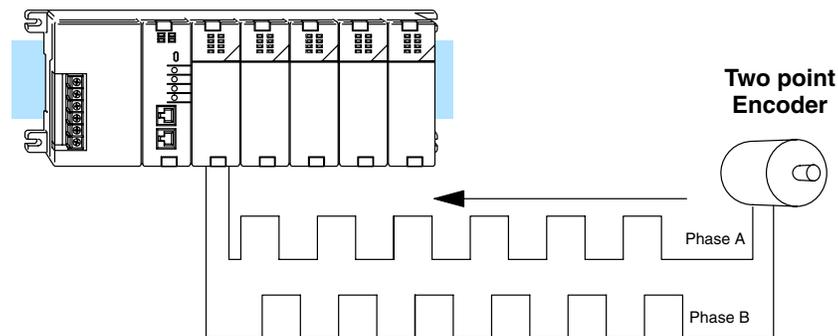
Note: Refer to page 2-4 for actual wiring diagram.

**Mode 20:
Up/Down
Counter
(DL240/250-1/260
Only)**

The internal counters can be configured to operate either as a standard Up/Down Counter or as a single Quadrature Up/Down Counter with a DL240/250-1/260 CPU. As a standard Up/Down Counter, the input signal for up counting to a preset is wired to point 00 on the module. The signal that is to count down from a preset is connected to point 01. Typically, these signals are being processed from two separate *single point* encoders.

In the Quadrature mode, the signals are typically received from a two-point quadrature encoder interfaced to a motor. The two signals are 90 electrical degrees out of phase with each other. By counting the signals and comparing them, the CPU can determine both shaft velocity and direction of motor rotation.

Quadrature Signal Processing



The counter operates independently of the CPU scan. Ladder logic can be written so that when the counter reaches a preset value (up to 24 presets), the CPU stops normal operations and executes an interrupt subroutine which is associated with the counter .

There are 24 presets, each of them having a special relay that can be used to trigger events when the preset equals the actual value. Absolute presets can be used to compare the preset directly to the actual count. Incremental presets can be used to compare the actual count to the current preset plus the accumulated value of the previous presets.

A subroutine can be programmed to perform whatever tasks are necessary. For example, Immediate I/O instructions can be used in the interrupt subroutine to provide a very fast response. Once the interrupt execution is complete, the CPU resumes normal program execution from the point where the interrupt occurred.

Turning the enable of the counter ON and OFF will start and halt the counting. Counters can be reset either by an external signal (point 02) or by special internal relays which can be activated by the program.

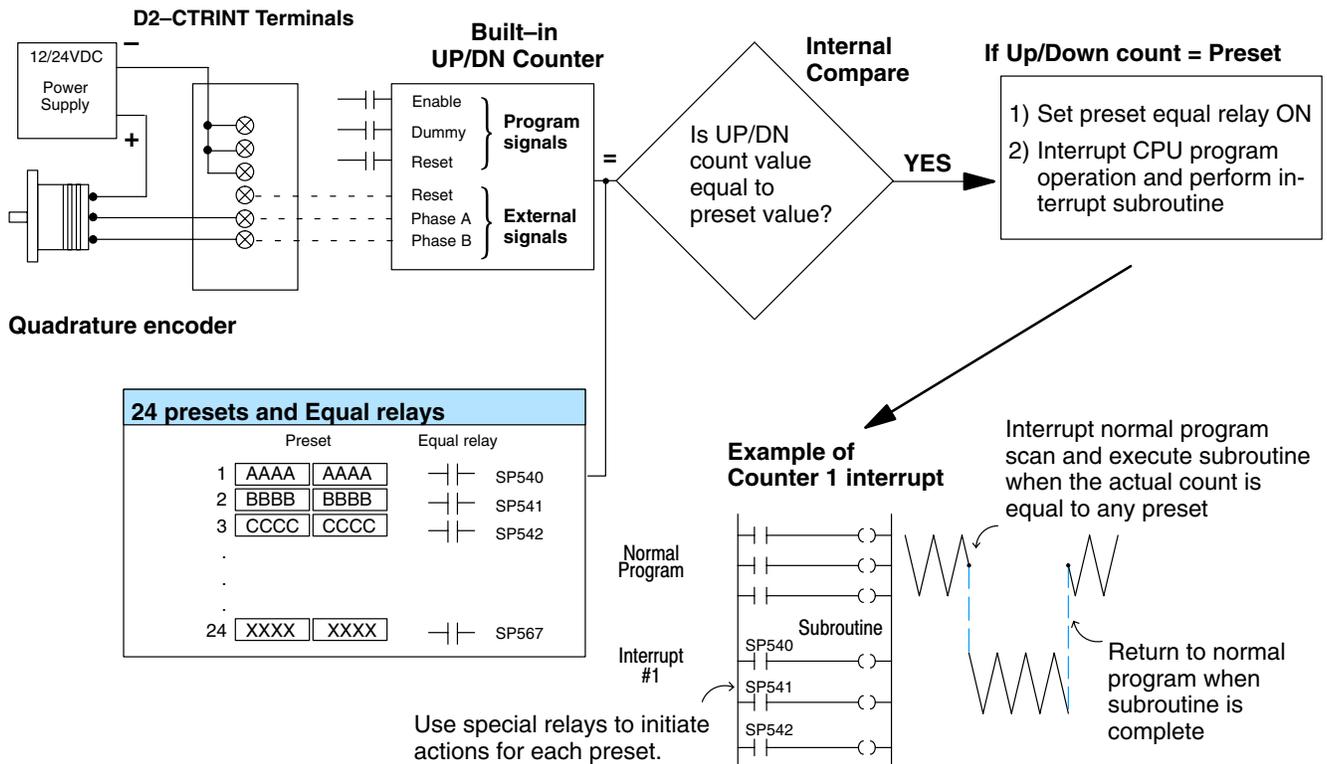
The following diagram provides a quick overview of the specifications and operations sequence for the Up/Down Counter.

Specifications for Up/Down Counter

Input voltage	12 or 24VDC
Frequency	5KHz maximum
Minimum pulse width	100µs
Count Range	-8,388,608 to 8,388,607
Preset types	Absolute or Incremental
Number of presets	24 (2 words per preset)

Hardware Connections

Input 1:	Phase A (pt. 00)
Input 2:	Phase B (pt. 01)
Input 3:	External counter reset (pt. 02)

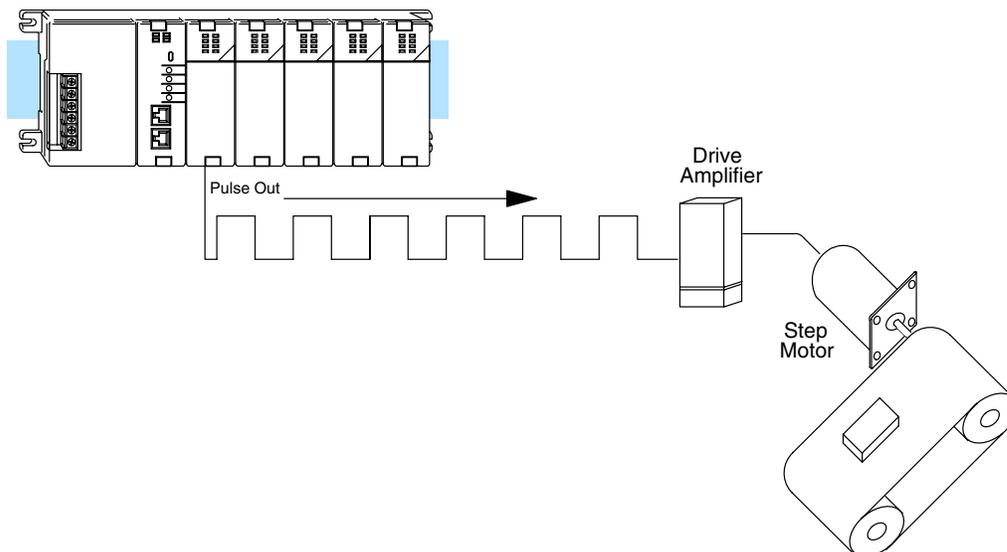


Note: Refer to page 2-4 for actual wiring diagram.

Mode 30: Pulse Output (DL240/250-1/260 Only)

This feature is only available for the DL240/250-1/260. The CPU has embedded pulse output circuitry which can be used to build simple motion and positioning control systems for transfer and indexing tables which are common applications. The pulse output is typically used with stepper motors which translate the pulses into an amount and direction of rotation.

Pulse Output

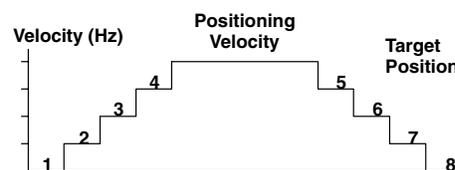


This mode of operation has two basic pulse output options; positioning control and velocity control.

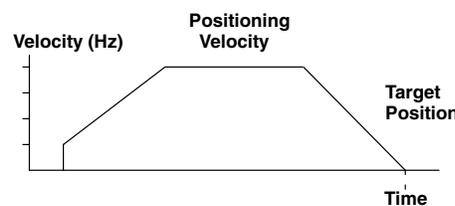
Positioning Control

This option uses a trapezoid profile to achieve a target position.

One approach provides preset acceleration control by defining four (4) acceleration steps to reach the positioning velocity and four deceleration steps as an approach to the target position.

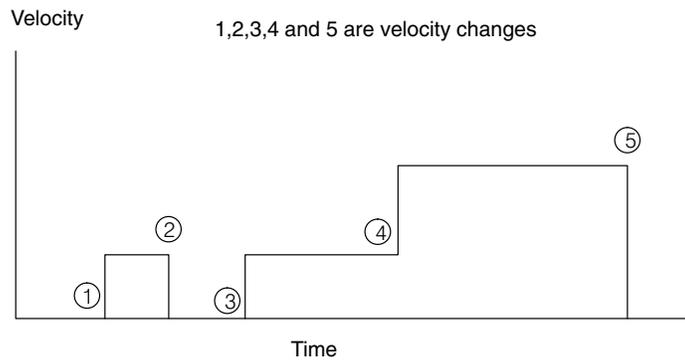


Another method provides automatic acceleration. With this method the starting velocity, the positioning velocity, and the amount of time required (100ms to 10 sec) are specified to accelerate from the start to positioning velocity. The CPU then automatically calculates the acceleration and deceleration.



Velocity Control

The velocity control option has clockwise (CW) and counter-clockwise (CCW) output capability, two separate outputs generating pulses. Unlike the other two pulse output functions mentioned on the previous page, velocity control does not seek a target pulse count for positioning. It only has two parameters, velocity and direction. During the velocity control operation, the direction and velocity can be set at anytime. By setting the upper bit in a reserved V-memory location the CPU will either send a CW or CCW signal to the motor drive. Another reserved V-memory location configuration will control the velocity in the same manner. Both of these V-memory locations are configured in ladder logic. The details of these V-memory locations are covered in the chapter that specifically discusses the pulse output mode in depth.

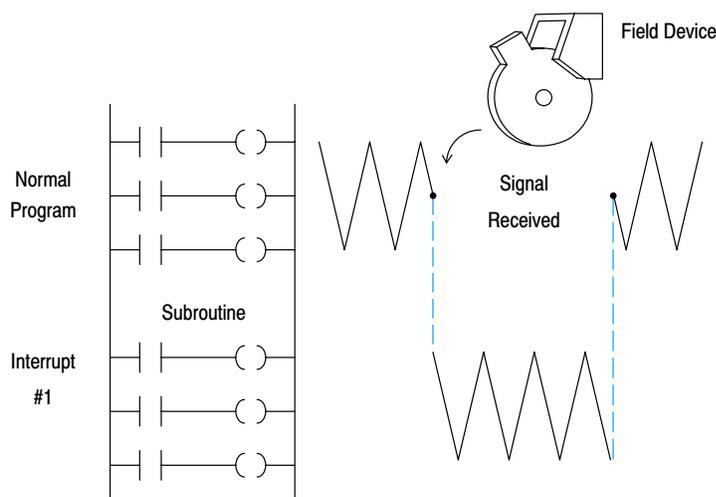


Mode 40: External Interrupts

Any one of the four points on the D2-CTRINT module can be used as external interrupts. One to four points can be configured to be external interrupt inputs. This is accomplished through the V-memory setup location in the DL230, DL240, DL250-1 and DL260 CPUs.

This mode is intended for applications that have a high-priority event which requires special operations to be performed. When this high-priority event occurs, the D2-CTRINT module senses an ON input signal. The module automatically informs the CPU to interrupt its present operation. The CPU immediately suspends the scan cycle and jumps to a subroutine identified with that particular interrupt input signal point. The CPU executes this interrupt subroutine (subroutines can use immediate I/O to immediately read and write I/O points). When the subroutine is complete, the CPU automatically resumes the normal scan cycle starting at the exact location from where it was interrupted. The CPU continues the normal scan until another interrupt signal is sensed.

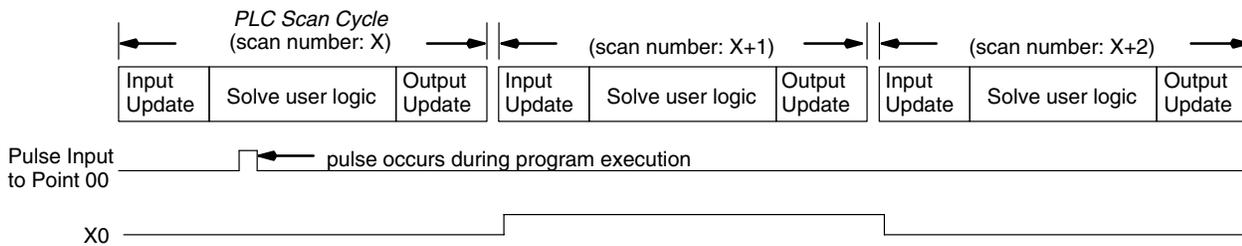
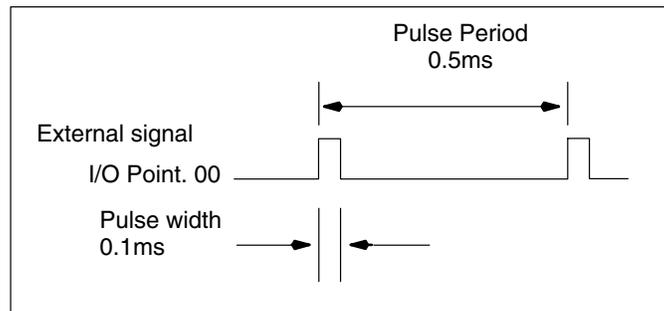
The interrupts can be pulses with a width as small as 100 μ s. The leading edge of the interrupt request pulse is what triggers the CPU. It takes 0.5ms to register an interrupt; if multiple pulses are sent to the same input point in a span of time less than 0.5ms, only the first signal will be acknowledged. If pulses are received simultaneously at two or more of the four input points, the priority of acknowledgement by the CPU will be point 00 first, point 01 second, point 02 third, and point 03 will be last.



Mode 50: Pulse Catch Inputs

Using the D2-CTRINT module with the DL240/250-1/260, any one or all four points of the module can be used as “pulse catchers”. The DL230 offers this feature for channel one (point 00) only. A pulse catching point can sense and latch a pulse width from 0.1ms to 0.5ms. This type of feature is needed when processing signals of a very short duration. Typically, these pulses are of such short duration that they cannot be detected during the normal program execution.

Instead of increasing the pulse length so that it can be processed, the D2-CTRINT and CPU combination makes use of special internal relays embedded in the CPU firmware which “trap” the pulses. When an external pulse is encountered, a special internal status relay (SP) is set to ON. The special purpose relay remains in the ON state for the next scan of the CPU, then gets set automatically to the OFF state. A pulse can be trapped at anytime regardless of the CPU scan.

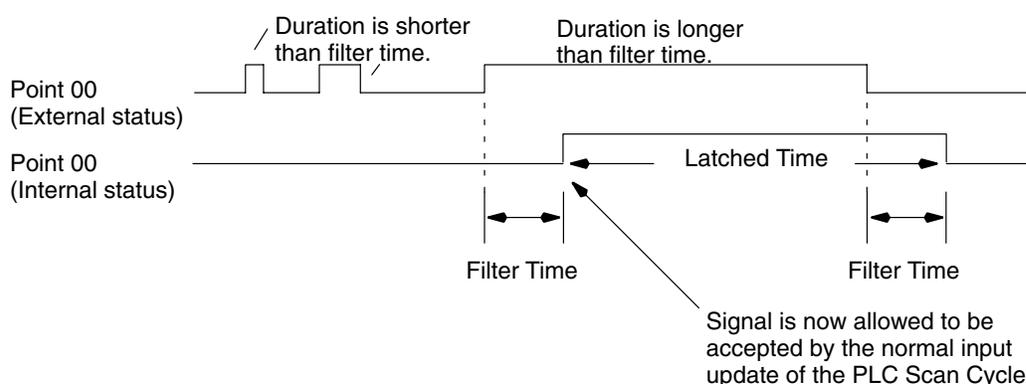


Mode 60: Discrete Inputs with Filter

This mode is selected when the primary need is to have four or less discrete inputs with very high integrity. Noise caused by switch bounce or other sources can be filtered so the inputs will be valid.

When an input signal is first detected at any one of the four points, a programmable filter is activated which begins a timed countdown. The ON status of the signal is temporarily prevented from being read by the input update of the CPU. The ON signal must stay present long enough for the filter to “time out”.

Once the signal has remained ON for the required time, it is latched and allowed to be accepted by the CPU during the normal input update of the PLC scan cycle. The signal is latched for the remaining duration of the ON signal plus an amount of time equal to the filter time. The filter time can be programmed for 0 to 99ms in 1ms increments.



Installation & Field Wiring Guidelines

In This Chapter. . . .

- Installing the Module in the Base
 - General Guidelines for Field Wiring
-

How to Install the Module

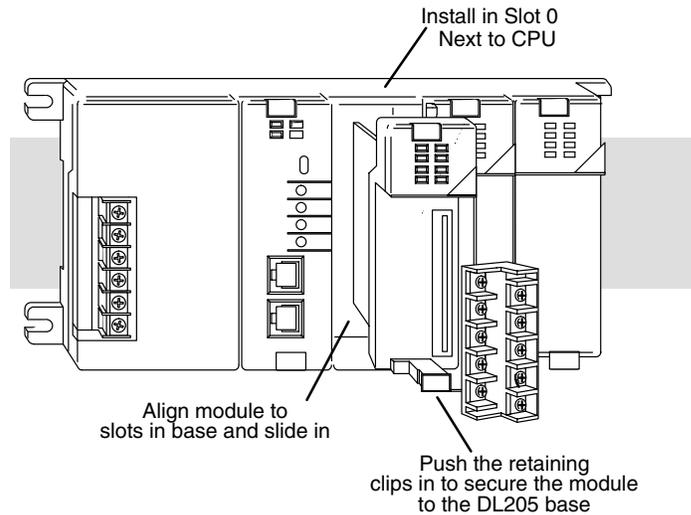
The D2-CTRINT module must be installed in slot 0 next to the CPU. It can only be used with the DL205 PLC family.



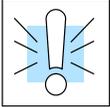
WARNING: Power to the PLC **must** be disconnected before inserting or removing the D2-CTRINT module. Failure to disconnect power could result in serious damage to the module, the PLC or both.

Insert the Module

Insert the D2-CTRINT module into the slot next to the CPU (slot 0). Locate the module so the printed information is oriented in the same direction as the markings on the PLC. Be careful to align the female connector on the printed circuit board of the module with the male connector on the PLC mother board. Press the module into the slot until the front of the module is flush with the front of the PLC.



General Guidelines for Field Wiring

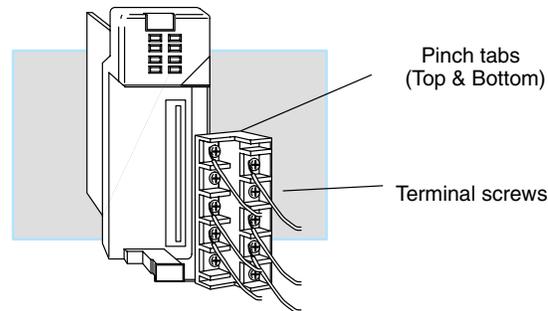


Wiring Guidelines

WARNING: To minimize the risk of personal injury or property damage, remove all power from the PLC and field devices before wiring the module.

The D2-CTRINT High Speed Counter Interface Module features a removable terminal block. It is held in place by pinch tabs located at the top and bottom of the terminal block.

Before removing the terminal block, be sure that the power to the module is disconnected. To remove the terminal block, squeeze the pinch tabs and pull the terminal block from the module.

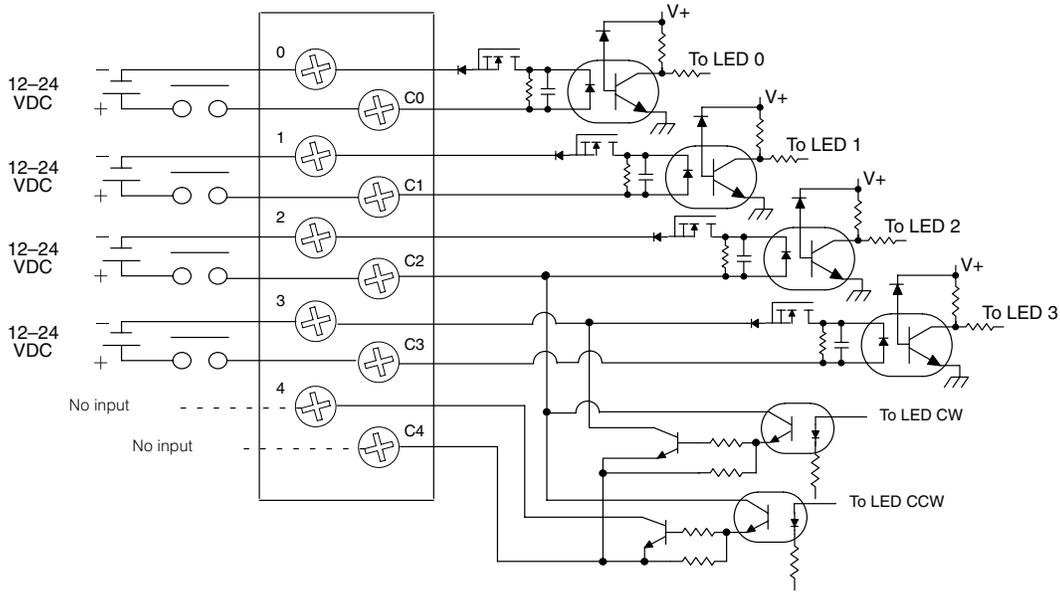


Consider the following wiring guidelines when wiring your system.

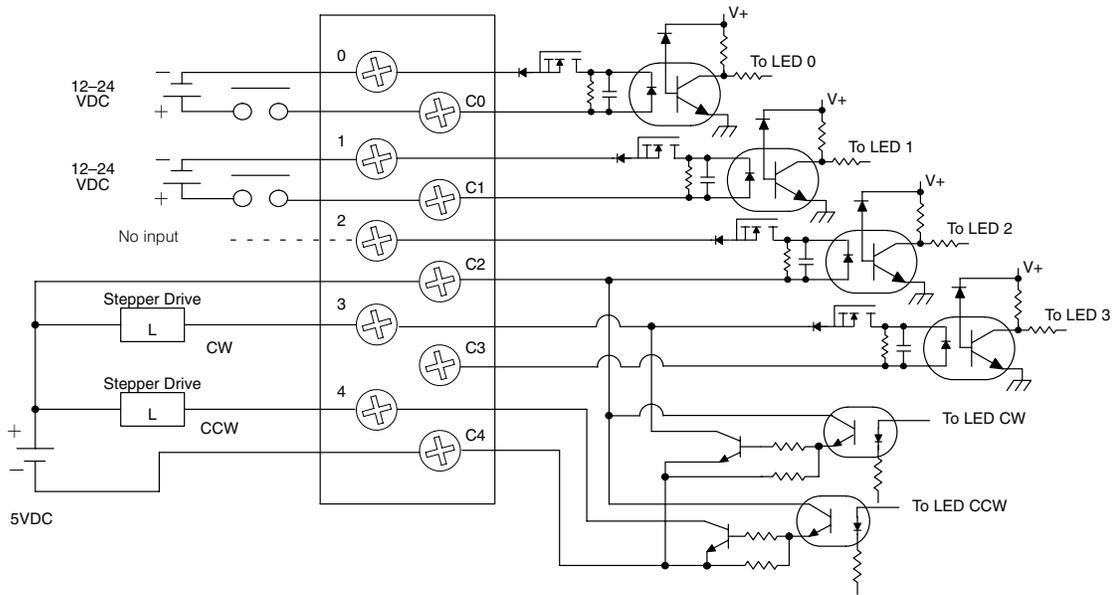
1. There is a limit to the size of wire the modules can accept. 16 AWG to 24 AWG is recommended for the Counter Interface Module wiring. Smaller AWG is acceptable.
2. Always use a continuous length of wire, do not combine wires to attain a needed length.
3. Use the shortest possible cable length.
4. Where possible, use wire trays for routing.
5. Avoid running wires near high energy wiring.
6. Avoid running input wiring in close proximity to output wiring where possible.
7. To minimize voltage drops when wires must run a long distance, consider using multiple wires for the return line.
8. Avoid running DC wiring in close proximity to AC wiring.
9. Avoid creating sharp bends in the wires.
10. Ground all shields to their respective signal source.

After the wiring is finished, return the terminal block to the module, making sure the terminal block is tightly seated.

Wiring Diagram for Modes 10, 20, 40, 50 and 60



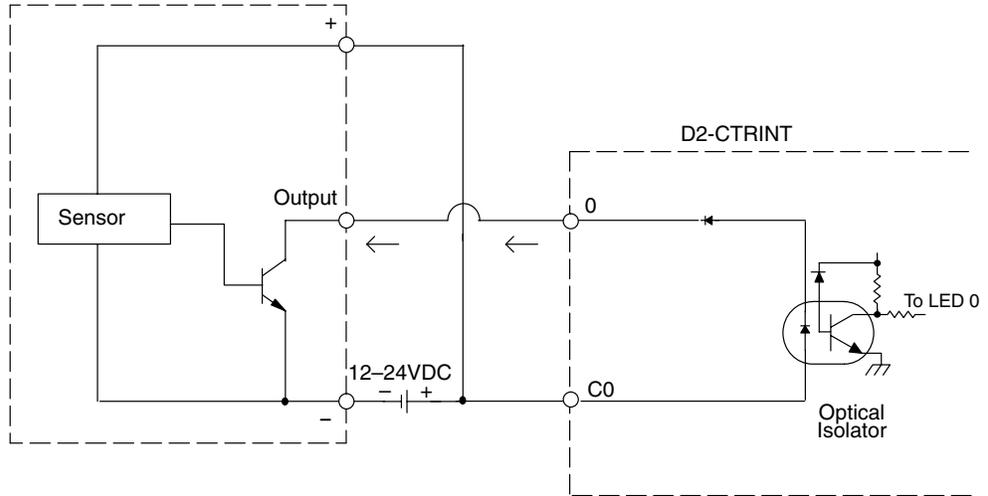
Wiring Diagram for Mode 30



Installation & Field Wiring Guidelines

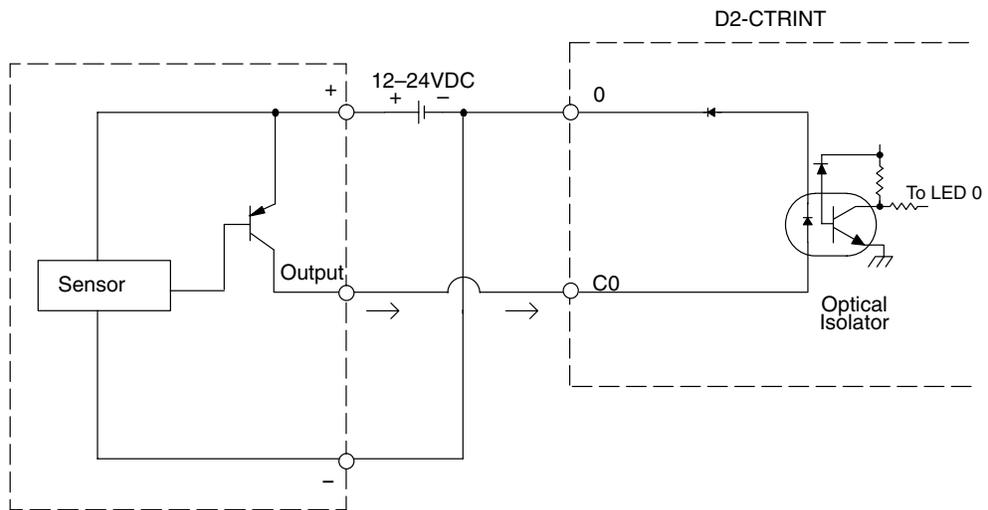
Solid State Field Device Wiring

NPN Field Device Example



(NPN) Current Sinking Field Device

PNP Field Device Example



(PNP) Current Sourcing Field Device

Mode 10 – High Speed UP Counters

In This Chapter. . . .

- Wiring the UP Counter Mode
 - Configuring the UP Counter
 - Writing the Control Program
 - Verification of Proper Operation
 - Troubleshooting
-

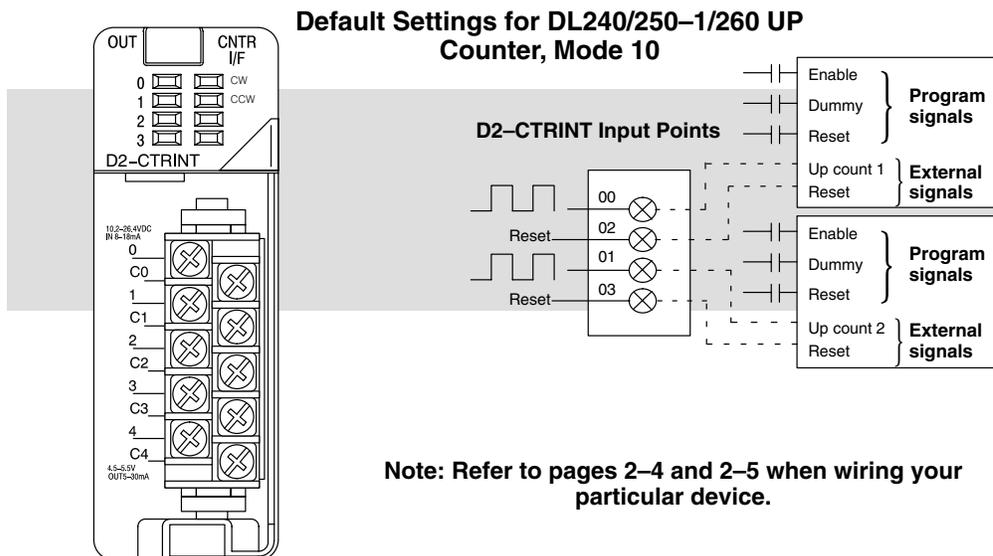
Using the UP Counters, Mode 10

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, one of the modes must be selected as the primary mode. Mode 10, High Speed UP Counter, will be the only mode covered in this chapter.

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

DL240/250-1/260 Applications

Mode 10
UP Counter

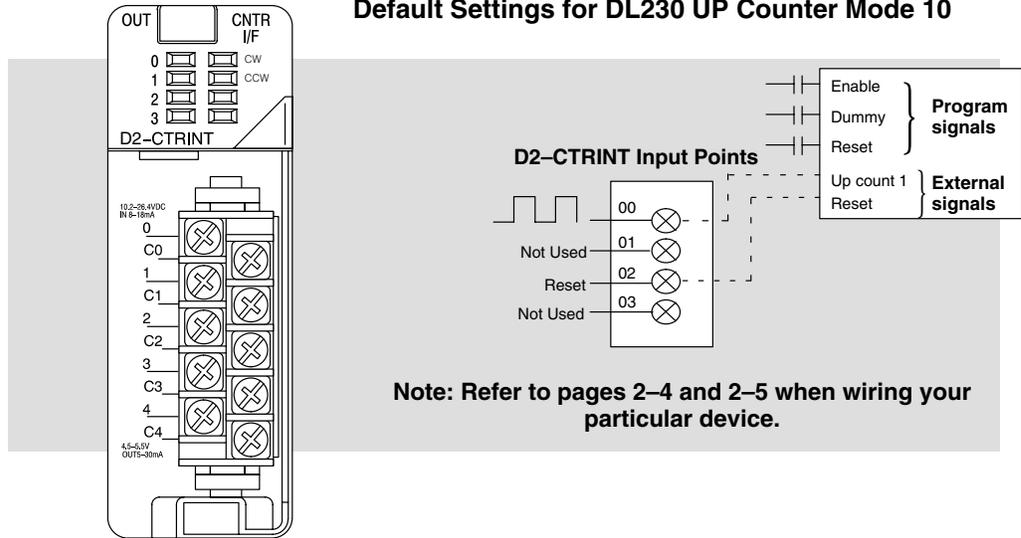


The above diagram shows points 00 and 01 which are the primary UP counter connecting points for field devices while points 02 and 03 are the respective reset inputs. The two counters can be reset either through the operating program using relay ladder logic or they can be reset by an external field device connected to points 02 and 03. Point 04 is for a pulse output signal, but is not used in this mode. Refer to pages 2-4 and 2-5 when connecting your device.

DL230 Applications

Unlike the DL240/250-1/260 CPU, the DL230 CPU has only one high speed counter embedded in its architecture. The diagram below shows the default settings for the UP Counter mode.

Default Settings for DL230 UP Counter Mode 10



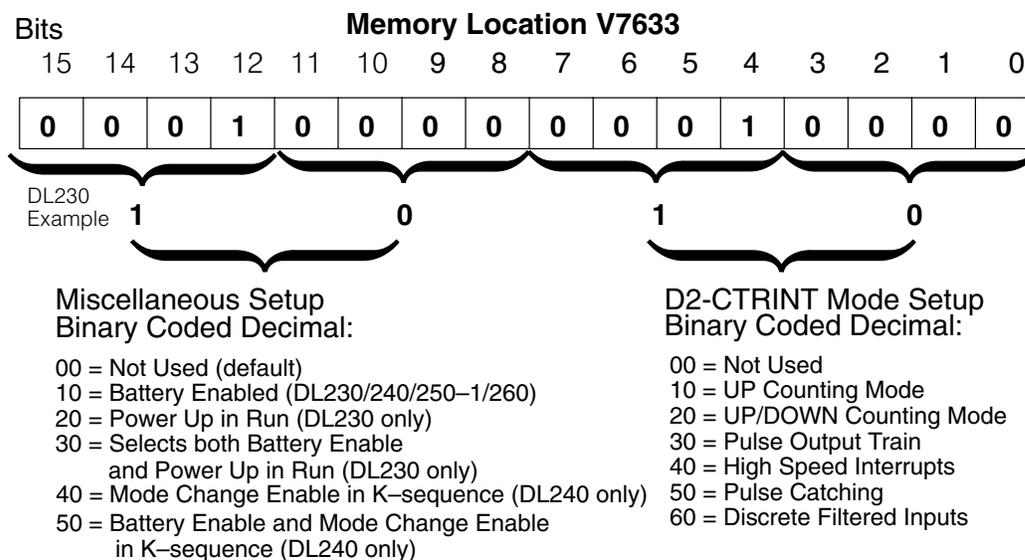
Note: Refer to pages 2-4 and 2-5 when wiring your particular device.

In this example, the first channel (point 00) receives the input from the device or equipment sending the DC level signals for counting. The third channel (point 02) is configured so that it can be wired to some device that will trigger an external reset. The other two points (points 01 and 03) are not used when the defaults are accepted. Refer to pages 2-4 and 2-5 when wiring your device to the D2-CTRINT module.

Understanding V-Memory Setup Locations

The D2-CTRINT needs to have the V-memory configured in order to use the UP Counter, Mode 10.

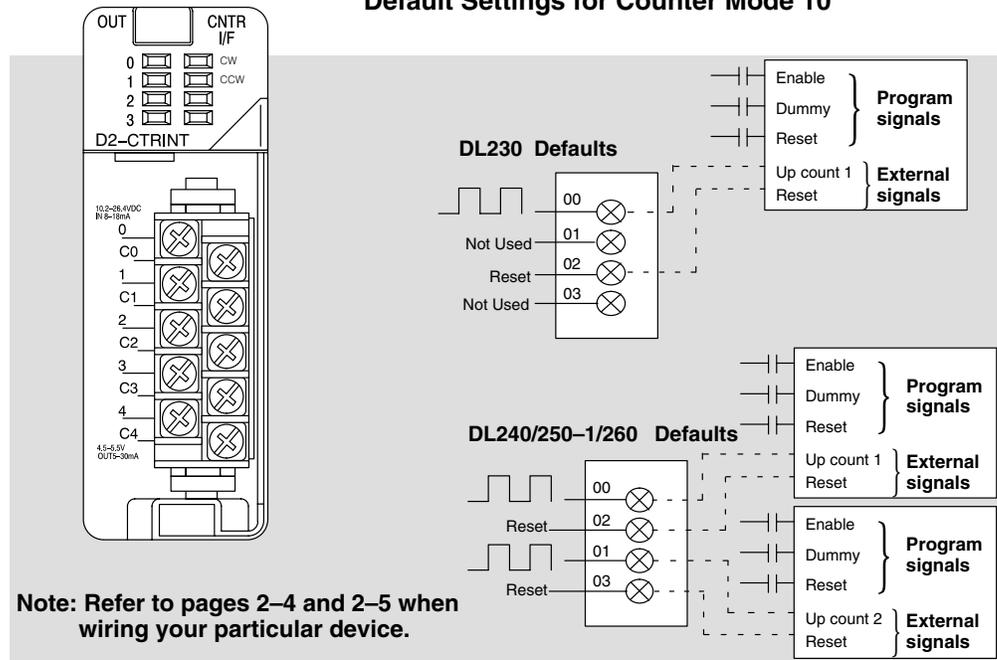
V-memory location V7633 is the most important of all the reserved memory areas because it stores the numeric value which lets the CPU know which mode has been selected. The following diagram shows the 16-bit word and the various information it stores, including the values used for the D2-CTRINT. The example shown here is for a DL230 and uses the UP counting mode. The lower bits are set to 10 for the mode, and the upper bits are set to 10 so the battery backup is enabled. Together they form the number 1010.



NOTE: It is important to look at the entire 16 bits of V7633. For simplification, zeros have been placed in bits 8 thru 15 for all ladder logic examples when loading the mode codes.

There are also other V-memory locations which contain High Speed Counter Interface 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 for Counter Mode 10



The above diagram shows the physical layout of the front of the module. The actual wire connecting points on the module can be visually related to the various functions that are associated through the default configuration.

There are programming examples in this chapter which show how to change some of these defaults.

Default settings with V7633 set to XX10:

V-Memory Location and Module I/O Points	Default Value in DL230	Default Value in DL240/250-1/260
V7634 (point 00)	0001 (Counter Input)	0001 (Counter 1 Input)
V7635 (point 01)	0000 (Not Used)	0001 (Counter 2 Input)
V7636 (point 02)	0007 (Reset Input)	0007 (Reset Input)
V7637 (point 03)	0000 (Not Used)	0007 (Reset Input)

The values shown above have the following meaning:

1 = absolute counting mode input

7= external reset input without an interrupt

While discussing memory configuration, it is good to know about the reserved memory for the presets. Below is a table with the default memory locations.

CPU	Channel 1 (pt.00)	Channel 2 (pt.01)
DL230	V2320 thru V2377	Not Applicable
DL240/250-1/260	V3630 thru V3707	V3710 thru V3767

Setting Up the CPU

Configuring the V-Memory

Each of the DL230, DL240, DL250-1 or DL260 CPUs will check V-memory to see if there is a High Speed Counter Interface Module present. The value XX10 will be read in V7633 if the module has been properly configured for UP counting. The XX represents the upper byte of V7633 value which handles such functions as enabling the backup battery and setting the PLC to start in the RUN mode. If the CPU finds that a D2-CTRINT module is present, it will check other V-memory locations to see how each point of the module has been configured.

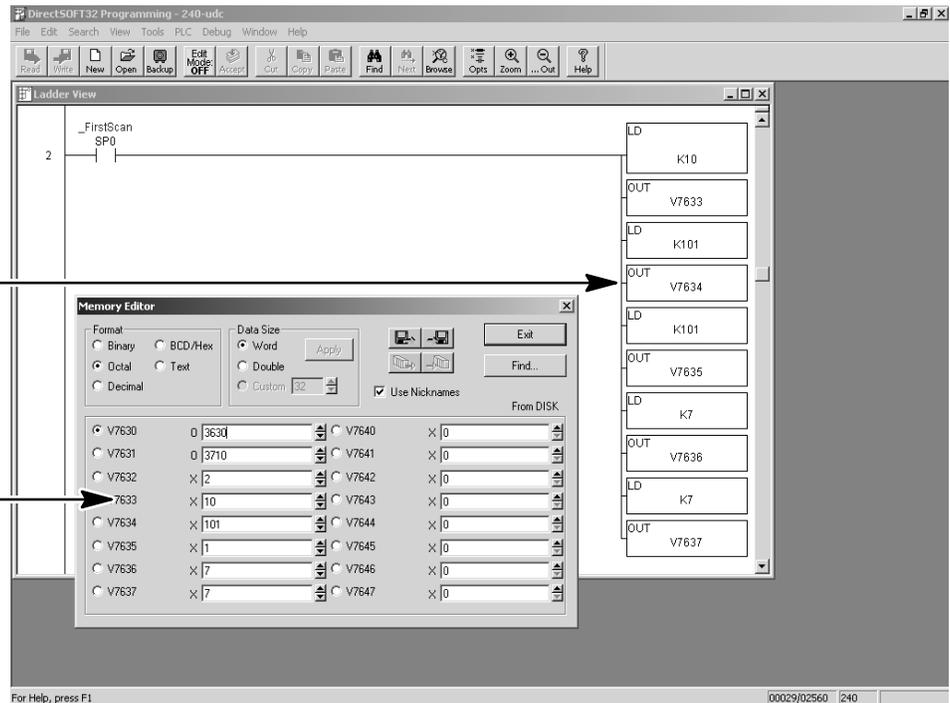
The values can be inserted into memory by using either a handheld programmer or a program edited with **DirectSOFT32**. There are program examples in this chapter for each counter function and mode. The following steps is a guide for configuring the V-memory for the UP counter function.

Step 1: Entering the Selected Mode

The UP Counter is Mode 10 which is the value to be set into V7633. The following **DirectSOFT32** diagram shows the setup procedures for communicating with your DL230, DL240, DL250-1 or DL260 CPU. Refer to **DirectSOFT32 Programmers User Manual** for more details.

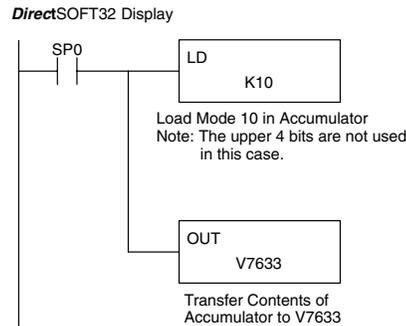
Setting the
V-Memory
using RLL

Setting the
V-Memory
using the
Memory
Editor



Editing the D2-CTRINT setup at the beginning of the user program is the most common 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 permanently used.

The following RLL example shows how to set the D2-CTRINT to Mode 10, UP Counter, in V-memory location V7633.



Two 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, 10 is to be placed in V7633. This value is loaded into the accumulator, LD K10. 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.

**Step 2:
Select the
Preset Mode
(DL240/250-
1/260 Only)**

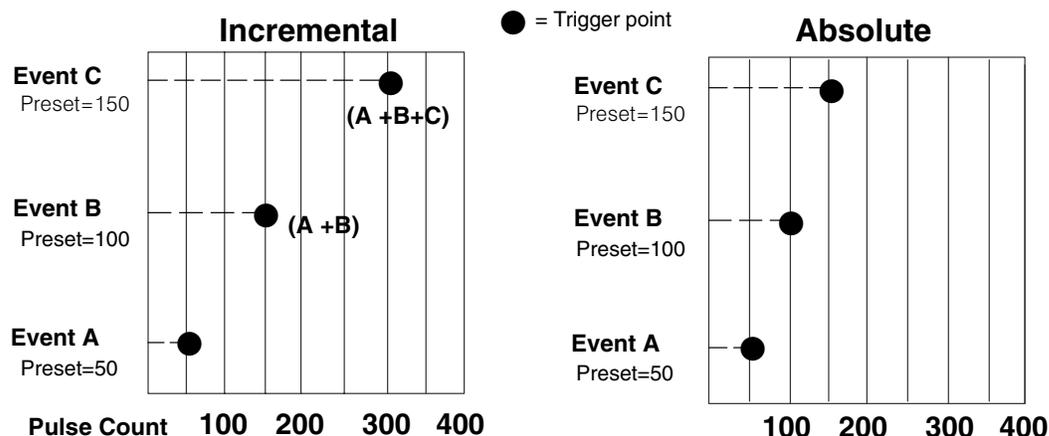
If a DL230 is being used or if the Absolute counting mode is accepted for a DL240, DL250-1 or a DL260, you can skip this step. The CPU automatically assumes that the absolute presets are being used and configures the V-memory accordingly.

There is only one preset mode for the DL230—*Absolute*. There are two different preset modes for the DL240/250-1/260,—*Absolute* and *Incremental*. The Absolute preset mode can be selected for one counter and the Incremental preset mode can be selected for the second counter, or both counters can be configured to use the same preset mode.

To understand the concepts of using the Absolute and Incremental preset modes, it is essential to know some basics about the counter's presets. Inside the PLC's memory are twenty-four (24) preset values which can be setup for each counter. A preset is the number of pulses which are set into V-memory to be counted before an event is to be initiated.

Presets are entered into successive areas of V-memory. The presets are all independent in the Absolute mode. That is, the counter compares the actual total count received from the D2-CTRINT module to a preset, when the two are equal, the event is triggered. With an Incremental preset mode, however, the presets are related to each other. In such case, the counter reaches preset A and triggers event A, then preset B is added to preset A and that becomes the number of pulses which must be counted before event B is triggered. Preset C is added to the sum of presets A and B — that is the number of pulses required for event C to be triggered. The process of adding all the presets continues until the CPU is notified that there are no more presets to satisfy.

Below is an example showing the difference between using the presets incrementally and absolutely.

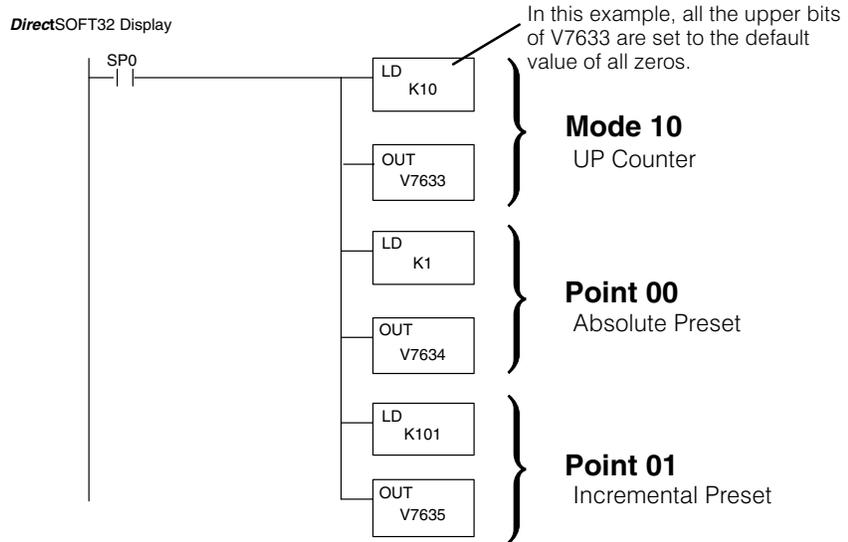


Assume that a limit switch (SW1) is being connected to one of the high speed counters. The counter is to initiate three different events at certain points in time determined by the pulse count received. Also, assume that the presets are stored (associated with each of the three events) in successive memory as 50, 100, and 150. In this example, these values are to be attended to in an incremental fashion. The counter would, in such case, trigger the 1st event when it counts 50 pulses, the 2nd event when it receives 150 pulses total (50 + 100) and the 3rd event when it receives 300 pulses total (50 + 100 + 150).

The result would have been different if the counter had been configured to count in the Absolute mode. In such case, the counter would trigger the 1st event when it counted 50 pulses, the 2nd event when it received 100 pulses and the 3rd event when it counted 150 pulses.

Configuring the Preset Mode

Select the Preset mode to be used by placing the proper value in memory location V7634 and/or V7635. If only one of the counters for a DL240, DL250-1 or DL260 is being used, place a counter preset value in one of these locations. The value for Absolute is 0001 and the value for Incremental is 0101. V7634 holds the value for Channel 1 (point 00), and V7635 holds the value for Channel 2 (point 01). The DL230 does not require either of these choices; because there is only one counter which is absolute.

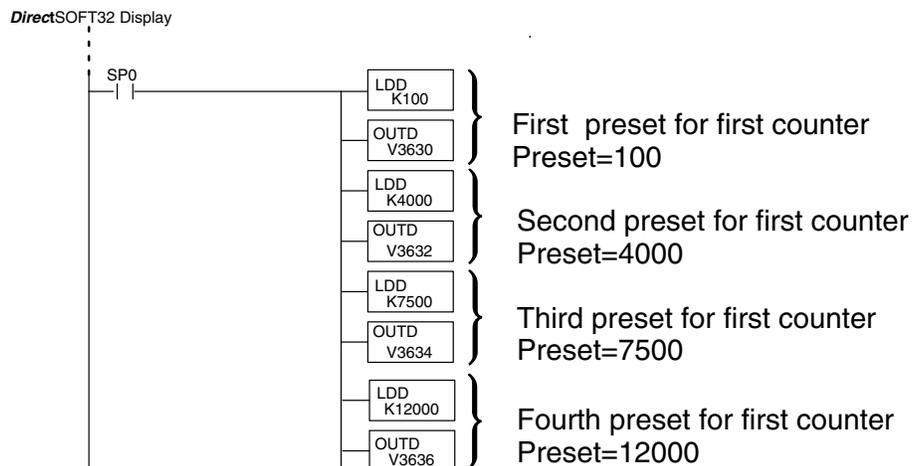


The previous RLL example which set the D2-CTRINT to Mode 10 now has the two preset values added to it. The CPU is set for the counter to use the Absolute preset mode at point 00 and to use the Incremental preset mode at point 01.

Step 3: Using Presets

The final step for setting up the CPU is to configure the presets. There are up to 24 presets per counter available, also internal equal relays are embedded in the CPU so that you can use the presets for triggering events inside the RLL program.

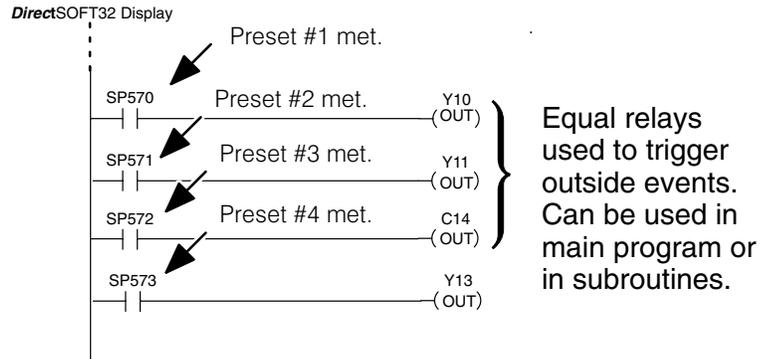
The following RLL diagram loads presets into consecutive V-memory locations, starting with the default memory location V3630. Use LDD and OUTD instructions as indicated in the example below:



Triggering Presets to Outside Events

Each of the presets are associated with special relays called equal relays. Each equal relay is assigned its corresponding preset value into two 16-bit V-memory addresses.

Below is an example of how to use the equal relays. They can be referenced in relay ladder logic like a standard RLL relay. Outside events are triggered whenever the preset assigned to a particular relay is satisfied by the pulse count, the relay closes.



DL240/250-1/260 Equal Relays		DL230 Equal Relays
Channel 1 (point 00)	Channel 2 (point 01)	Channel 1 (point 00)
SP540 (V3631/V3630)	SP570 (V3711/V3710)	SP540 (V2321/V2320)
SP541 (V3633/V3632)	SP571 (V3713/V3712)	SP541 (V2323/V2322)
SP542 (V3635/V3634)	SP572 (V3715/V3714)	SP542 (V2325/V2324)
SP543 (V3637/V3636)	SP573 (V3717/V3716)	SP543 (V2327/V2326)
SP544 (V3641/V3640)	SP574 (V3721/V3720)	SP544 (V2331/V2330)
SP545 (V3643/V3642)	SP575 (V3723/V3722)	SP545 (V2333/V2332)
SP546 (V3645/V3644)	SP576 (V3725/V3724)	SP546 (V2335/V2334)
SP547 (V3647/V3646)	SP577 (V3727/V3726)	SP547 (V2337/V2336)
SP550 (V3651/V3650)	SP600 (V3731/V3730)	SP550 (V2341/V2340)
SP551 (V3653/V3652)	SP601 (V3733/V3732)	SP551 (V2343/V2342)
SP552 (V3655/V3654)	SP602 (V3735/V3734)	SP552 (V2345/V2344)
SP553 (V3657/V3656)	SP603 (V3737/V3736)	SP553 (V2347/V2346)
SP554 (V3661/V3660)	SP604 (V3741/V3740)	SP554 (V2351/V2350)
SP555 (V3663/V3662)	SP605 (V3743/V3742)	SP555 (V2353/V2352)
SP556 (V3665/V3664)	SP606 (V3745/V3744)	SP556 (V2355/V2354)
SP557 (V3667/V3666)	SP607 (V3747/V3746)	SP557 (V2357/V2356)
SP560 (V3671/V3670)	SP610 (V3751/V3750)	SP560 (V2361/V2360)
SP561 (V3673/V3672)	SP611 (V3753/V3752)	SP561 (V2363/V2362)
SP562 (V3675/V3674)	SP612 (V3755/V3754)	SP562 (V2365/V2364)
SP563 (V3677/V3676)	SP613 (V3757/V3756)	SP563 (V2367/V2366)
SP564 (V3701/V3700)	SP614 (V3761/V3760)	SP564 (V2371/V2370)
SP565 (V3703/V3702)	SP615 (V3763/V3762)	SP565 (V2373/V2372)
SP566 (V3705/V3704)	SP616 (V3765/V3764)	SP566 (V2375/V2374)
SP567 (V3707/V3706)	SP617 (V3767/V3766)	SP567 (V2377/V2376)



NOTE: The consecutive addresses shown above for each relay are those assigned by the CPU as default addresses. Pointers for the start of these addresses are stored by the CPU at V7630 and V7631 for the DL240/250-1/260 and V7630 for the DL230 only. If there is a conflict of addresses because of pre-existing code written to these addresses, change the default block of addresses by placing a different pointer value in V7630 or V7631 in the relay ladder logic, depending on the CPU and preference. For example, to change the starting address for the relays belonging to Channel 1 to V2500, change the program to write an octal 2500 to V7630. The results are that the CPU will reserve 48 consecutive 16-bit memory addresses (32 bits per preset for the 24 presets available) for SP540 to SP567 equal relays.

Custom Configurations

There may be a time when the factory defaults will not satisfy an application. In this case the defaults can be changed so the D2-CTRINT can work for almost any application.

The following tables are provided with the values needed for custom configurations.

DL230 Custom Configuration

Mode 10 Options for the DL230

Point Number	V-Memory Location	Definition	Hex Value
point 00	V7634	UP Counter #1 (default)	0001 (absolute) default
point 01	V7635	Not Used (default)	0000 default
		Discrete Input	xx06 *
point 02	V7636	Reset UP Counter (default)	0007 (no interrupt) default
			0107 (interrupt)
		Discrete Input	xx06 *
point 03	V7637	Not Used (default)	0000 default
		Discrete Input	xx06 *
point 04	Not Used	Pulse Output (CCW)	Not Used

Note: The lower byte of V7633 is set to 10.

* xx = filter time. Filter time range = 0 to 99ms.

For the filtered inputs, the xx signifies the filter time in milliseconds, i.e. 1006 is a 10ms filter.

For the DL230, either place zeros (0) in V7635, V7636 and V7637 or enter the proper hex value for the discrete input point.

The DL240/250-1/260 has more options to chose for custom configuring. The chart on the next page shows you those options.

DL240/250-1/260
Custom
Configuration

Point Number	V-Memory Location	Definition	Hex Value
point 00	V7634	UP Counter #1	0001 (absolute) default
			0101 (incremental)
point 01	V7635	UP Counter #2	0001 (absolute) default
			0101 (incremental)
		Pulse Catcher	0005
		Discrete Input	xx06 *
point 02	V7636	Reset UP Counter #1	0007** (no interrupt) default no Z pulse or index marker recognition
			0107 (interrupt) can recognize Z pulse or index marker
			0207 (interrupt) no Z pulse or index marker recognition
			0307 uses Z pulse recognition in the interrupt
		Pulse Catch	0005
		Discrete Input	xx06 *
point 03	V7637	Reset UP Counter #2	0007** (no interrupt) default no Z pulse or index marker recognition
			0107 (interrupt) can recognize Z pulse or index marker
			0207 (interrupt) no Z pulse or index marker recognition
			0307 uses Z pulse recognition in the interrupt
		Pulse Catch	0005
		Discrete Input	xx06 *
point 04 (Mode 30)	Not Used	Pulse Output (CCW)	Not Used

Note: The lower byte of V7633 is set to 10.

* xx = filter time. Filter time range = 0 to 99ms.

** In a high speed application using a high resolution encoder, with the CPU programmed with presets that span the resolution of the encoder, you **may lose** pulses in the higher range of presets.

When using constants **K7** or **K107** in V7636 or V7637, the counter module will read the preset V-memory each time the counter is reset or at the index marker location.

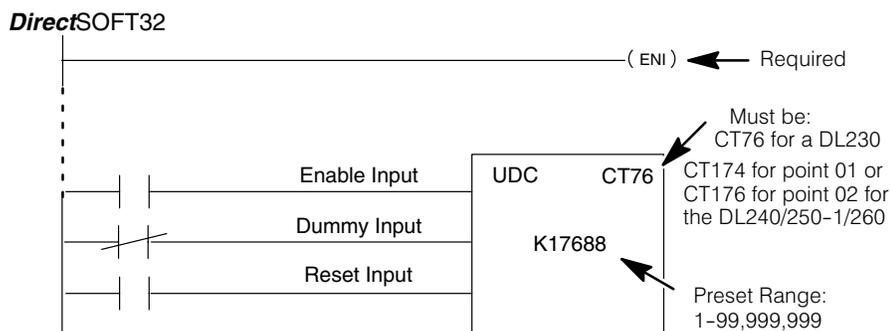
When using **K207** or **K307**, the counter module reads the preset V-memory only at power up, or when a CPU mode change occurs (i.e. PROGRAM to RUN transition).

NOTE: If presets are written to V-memory for the D2-CTRINT module during program execution the counter may not be able to run at high speed, high resolution applications, since reading these locations during a reset may result in lost pulses.



Writing the Control Program

The control program is ready to be written after the High Speed UP Counter parameters have been configured. Writing the RLL program for the High Speed Counter is much the same as for a regular counter. The embedded high speed counters which are activated through the D2-CTRINT module, have three inputs. The first input is to enable the counter, the second one is a dummy and the third is to reset the counter. The dummy has no function other than it is required by **DirectSOFT32**. The three inputs are required for **DirectSOFT32** to accept the program.



The mnemonic for the counter is UDC. It is found in the **DirectSOFT32** instruction browser that pulls down using the F7 hotkey while in the Edit Mode. After selecting the UDC, a box appears asking for the counter address and preset value. The counter address in the above example is CT76. This is the address which must be used for the DL230 CPU. The preset value of 17688 has been used in the example. This is the number of pulses to be received and counted before the output of the counter goes high. Any value between 1 and 99,999,999 can be programmed. The DL240/250-1/260 CPUs require addresses of CT174 for Channel 1 (point 00) and CT176 for Channel 2 (point 01). If a different address is used, the CPU will think that a standard, non-high speed, counter has been programmed, and the D2-CTRINT will not be enabled.

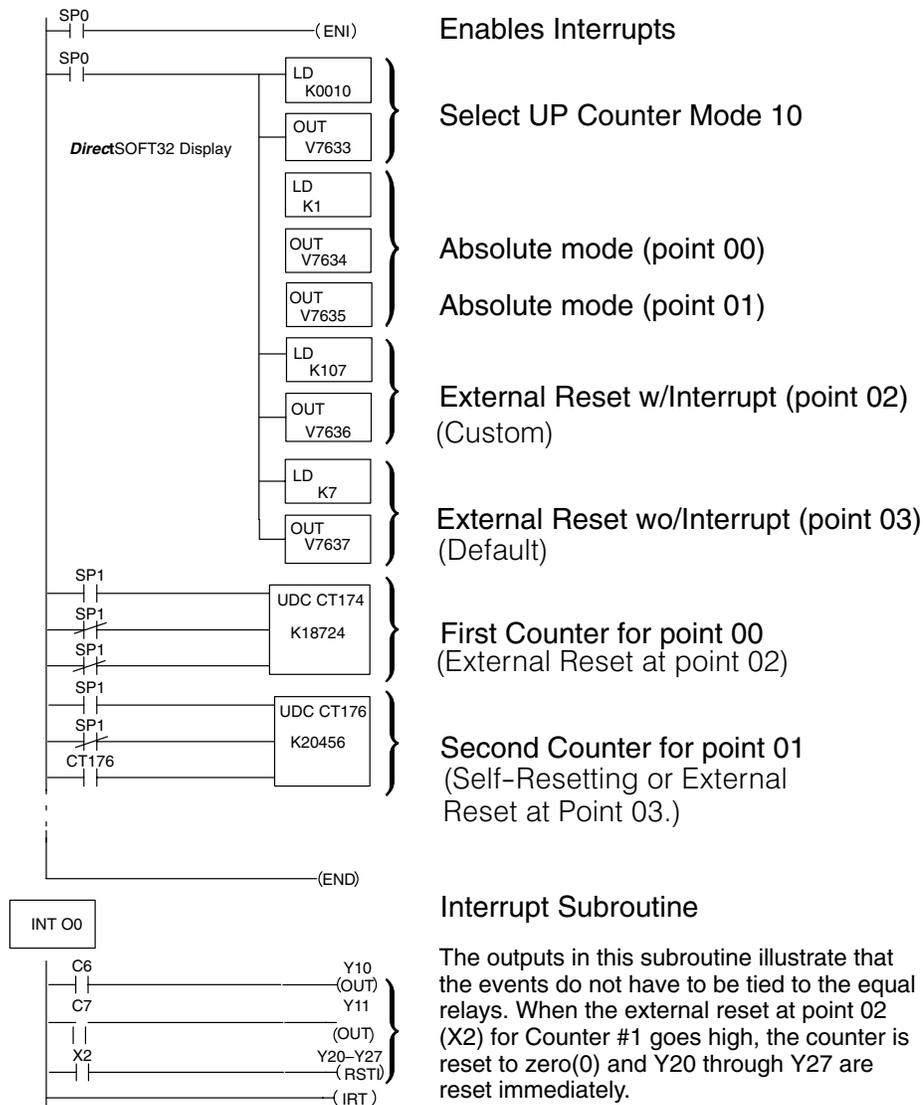
After entering the counter address and preset, enter the contact addresses for the contacts which are automatically drawn in the RLL program.

High Speed Counter data

CPU	Counter No. for Channel 1 (pt 00)	Counter No. for Channel 2 (pt 01)	Preset Limits
DL230	76	N/A	1-99,999,999
DL240/250-1/260	174	176	1-99,999,999

Example 1: UP Counters Without Presets

The following example uses either a DL240, a DL250-1 or a DL260 CPU. The example also uses both UP counters at points 00 and 01, and the resets at points 02 and 03. The counter attached to point 00 is to trigger a jump to an interrupt subroutine when it is externally reset by point 02. There is no interrupt subroutine written for the reset for point 03. The second counter resets after receiving 20,456 pulses.



Notice the counters have been properly labeled CT174 and CT176. The presets have been set to the respective numbers specified. The load accumulator instructions have setup the V-memory as required, i.e. 10 in V7633 for the mode, 1 in V7634 and V7635 for absolute mode and 0107 in V7636 to trigger a jump to an interrupt subroutine. Also in the setup is the reset of the counter CT174 when point 02 goes high and 0007 in V7637 to indicate no jump to an interrupt subroutine when CT176 is externally reset.

Reset with an Interrupt

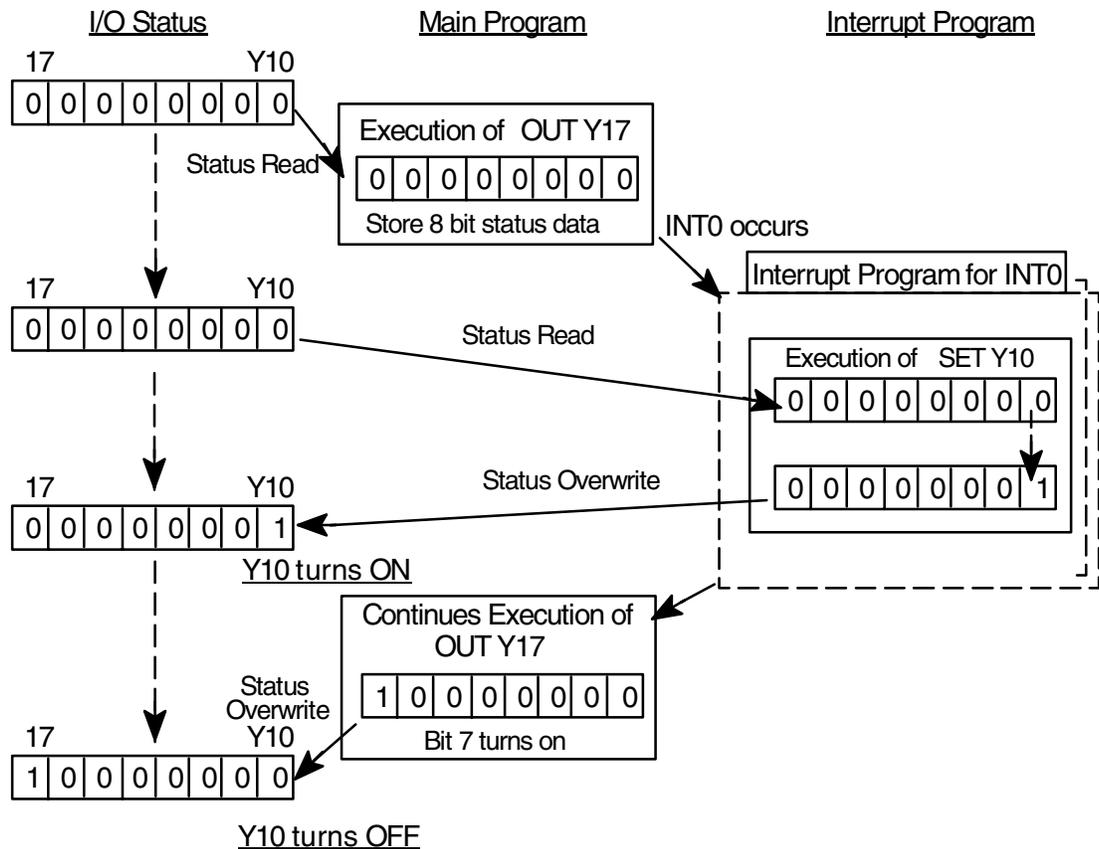
In the previous example, whenever point 02 is enabled, CT174 is reset to zero (0). At the same time, the CPU suspends its normal main program scan and execution, going to the interrupt subroutine. When it finishes executing the subroutine and encounters the return statement, the execution returns to the exact point in the main program where it left off. Notice the interrupt subroutine must be labeled to match the input point to which its corresponding counter is wired, i.e. 00 is used.

Interrupt Operation in Ladder Logic

The programmer should be aware of the sequence of events during an interrupt in order to avoid unexpected I/O operations. When an interrupt occurs, the CPU halts execution of the current instruction. It then executes the interrupt program, then returns to complete the suspended instruction. If the suspended instruction involves the same memory byte as an operation in the interrupt program, the CPU may not retain the correct status of that byte. This is because I/O status is updated one byte at a time.

EXAMPLE: An interrupt occurs during execution of the "OUT Y17" instruction, which has read the current status of the I/O byte which contains Y17. The interrupt program in this example sets Y10, a bit which is in the same byte as Y17. When control returns to the suspended instruction, it uses the status it read from the I/O status table **before** the interrupt to complete its operation. It turns Y17 ON, but still has the old value for Y10 (OFF). When the byte is written to the I/O status table, the CPU has lost the result of the interrupt operation. To avoid this problem, the programmer should use control relays (C memory) in bytes unused elsewhere for SET or OUT operations in the interrupt programs.

Mode 10
UP Counter



Example 2: UP Counters with Multiple Presets

The next DL240/250-1/260 example will use some features from other modes with multiple presets for a single counter. The example will use only one counter for point 00 configuring the preset mode as absolute and having a reset with an interrupt at point 02.

Point 01 will be made a pulse catcher and point 03 will be used as a discrete filtered input. Referring to page 3-13 the V-memory locations for control points 01 and 03 are V7635 and V7637 respectively. This is true for any mode or mix that is used.

Place 0005 in V7635 for the pulse catch input, and xx06 in V7637 for the filtered discrete input. This example will use 10 ms for the filtered time constant. The value to put in V7637 is 1006. The relay ladder logic to accomplish this is on the next page.

This example also uses multiple presets of 5000, 10000 and 15000 for the counter.

Memory Locations for Multiple Presets

Counter No.	Memory Range/Default Preset Addresses	Equal Relays
76 (DL230)	V2320 to V2377	SP540-SP567
174 (DL240/250-1/260)	V3630 to V3707	SP540-SP567
176 (DL240/250-1/260)	V3710 to V3767	SP570-SP617



NOTE: The above table indicates the default V-memory locations for setting the presets and the corresponding special equal relays assigned to each. Be aware that the default addresses can be changed to begin elsewhere merely by changing the pointers at V7630 (Channel 1) and V7631 (Channel 2) to the octal value of the new addresses.

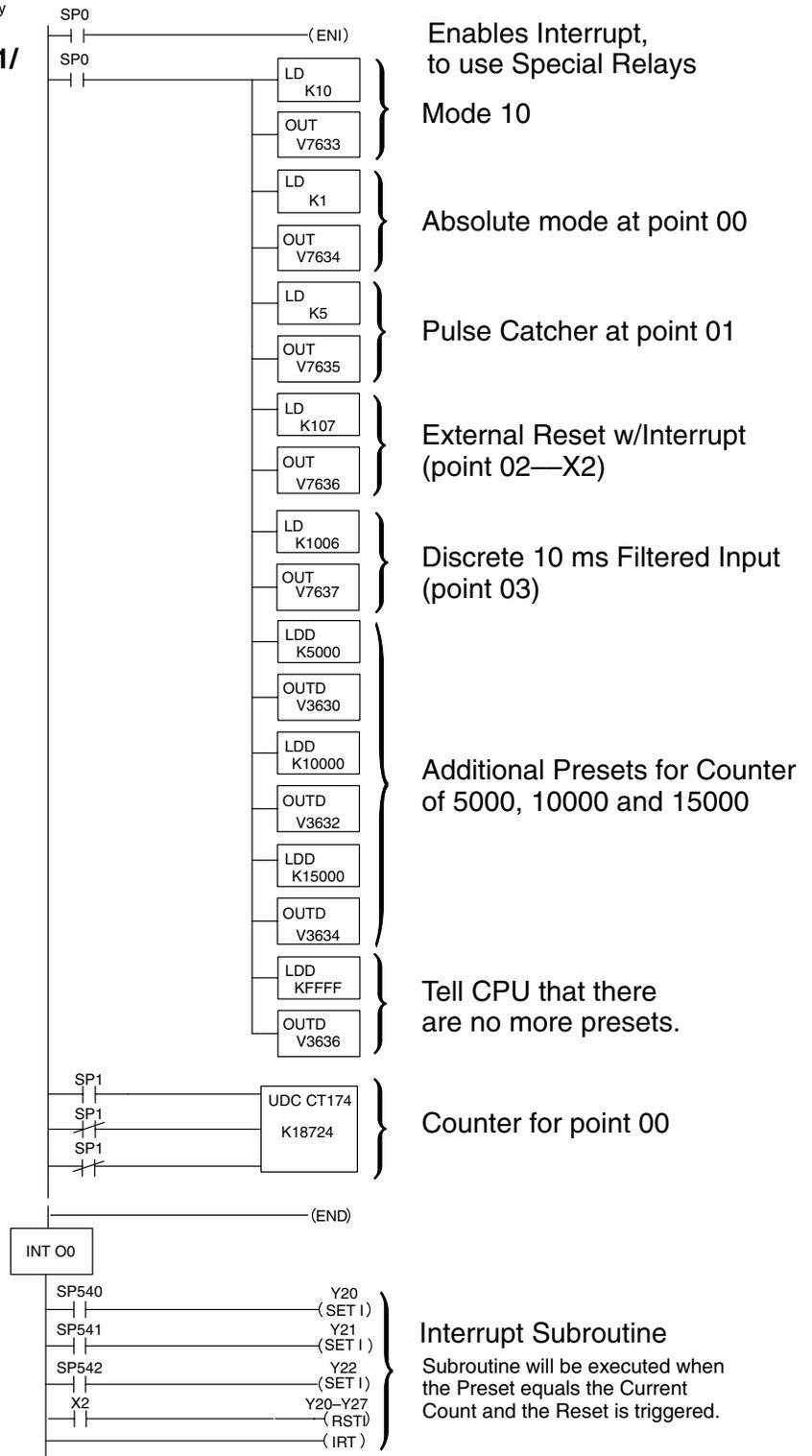
Each preset requires two V-memory locations (32-bits), i.e. V2320 and V2321 are the first preset, V2322 and V2323 are the second, etc. Each of these presets has a special equal relay assigned to it. The numbering assignment for these are one digit apart, i.e. SP540 assigned to the first preset, SP541 assigned to the second preset, etc.

So that the CPU knows that it has seen all the presets when in the Absolute mode, place an FFFF in the next available V-memory preset location following the final preset (if not using all 24 presets available). In the Incremental mode, place FFFF as in the Absolute mode or place FF00 in the last preset location to reset the counter, set the accumulated pulse count to zero (0) and continue counting. In the example, this has been placed in V3636 since the last preset was in V3634 and only using three presets. Place 00FF in the last preset location to reset the counter and not reset the accumulated pulse count.



NOTE: When using two counters with the SP570 contacts, etc., INT01 must be used.

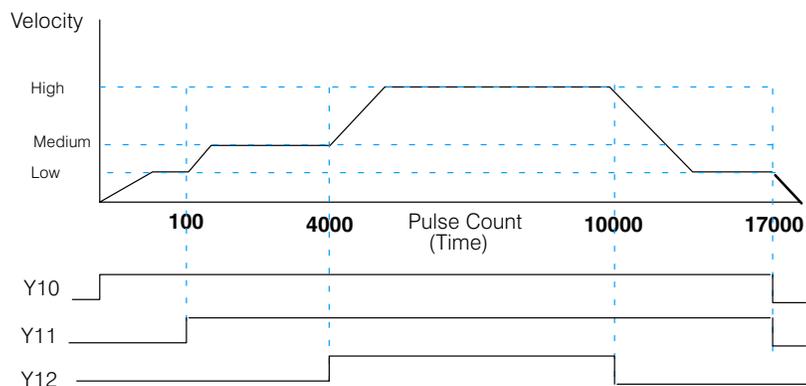
DirectSOFT32 Display
**DL240/250-1/
260
Example**



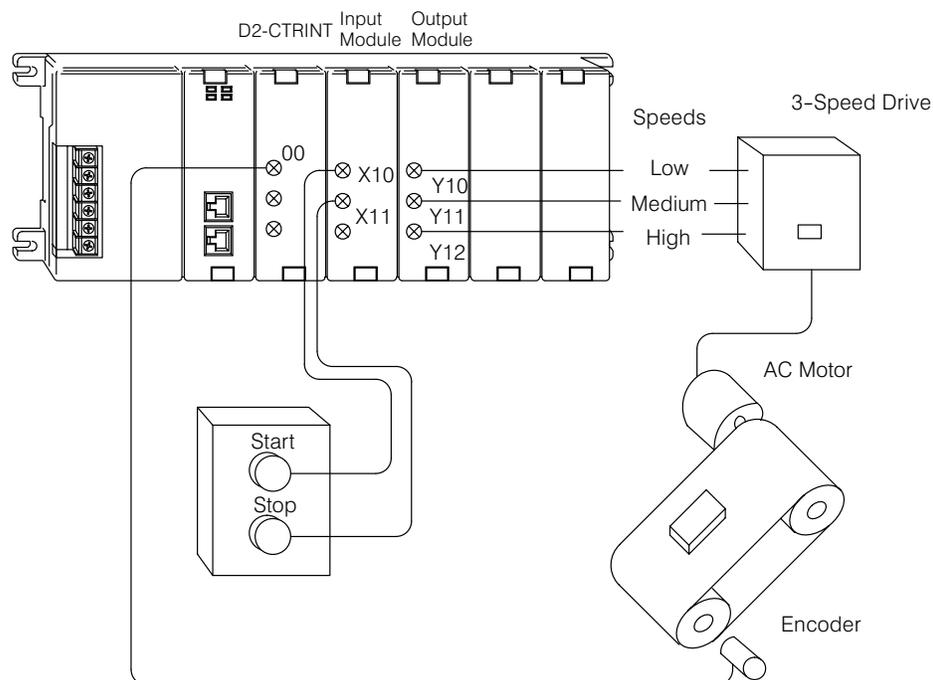
NOTE: To trigger outside events with the three presets, assign SP540,541 and 542 respectively to V3630, V3632, and V3634. (See Table on Page 3-11.)

Example 3: Speed Control for an AC Motor

The next example will use one UP counter for a DL240, DL250-1 or DL260 to control the timing for a small AC motor inverter. Unlike the positioning applications discussed later in Chapter 5 for the pulse train function (Mode 30) where the interface module does all the ramping and speed control of a stepper motor, this application will rely on a separate motor drive to provide all of the intelligence for proper acceleration/deceleration ramping and motor speed control for an AC induction type motor.

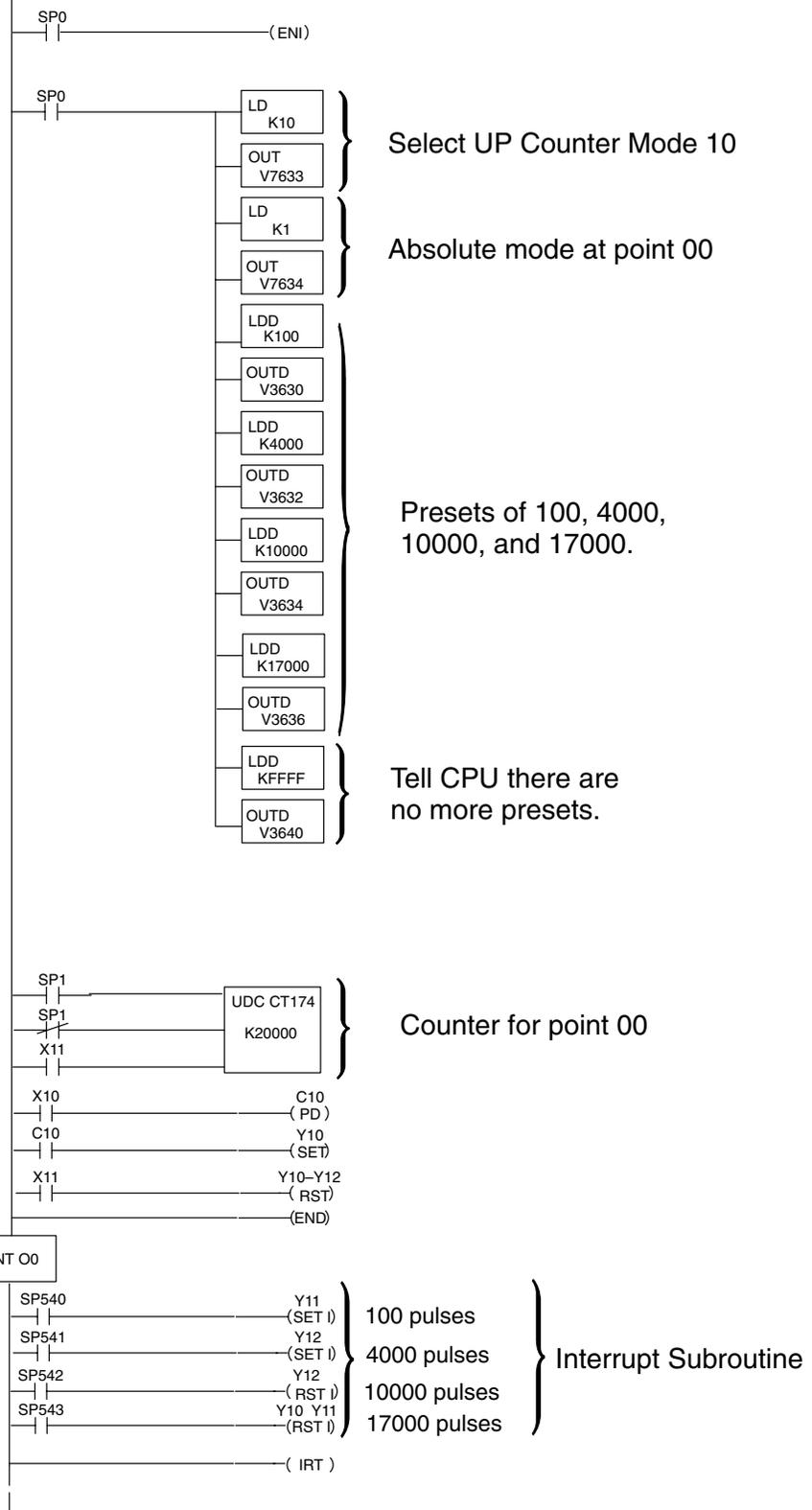


Assume that there are input and output modules in a DL205 base and that X10 and X11 of the input module are wired to a START/STOP switch. The inverter being used in this example has programmable input relays that accept low, medium and high speed ranges that have been programmed into the drive. The drive automatically powers up at the low speed setting. Low, medium and high speed input relays are wired to Y10, Y11 and Y12 respectively. The goal here is to write the appropriate relay logic to accomplish the above start and stop of the drive to match the curve shown above.



Example Program

DirectSOFT32 Display



Mode 10
UP Counter

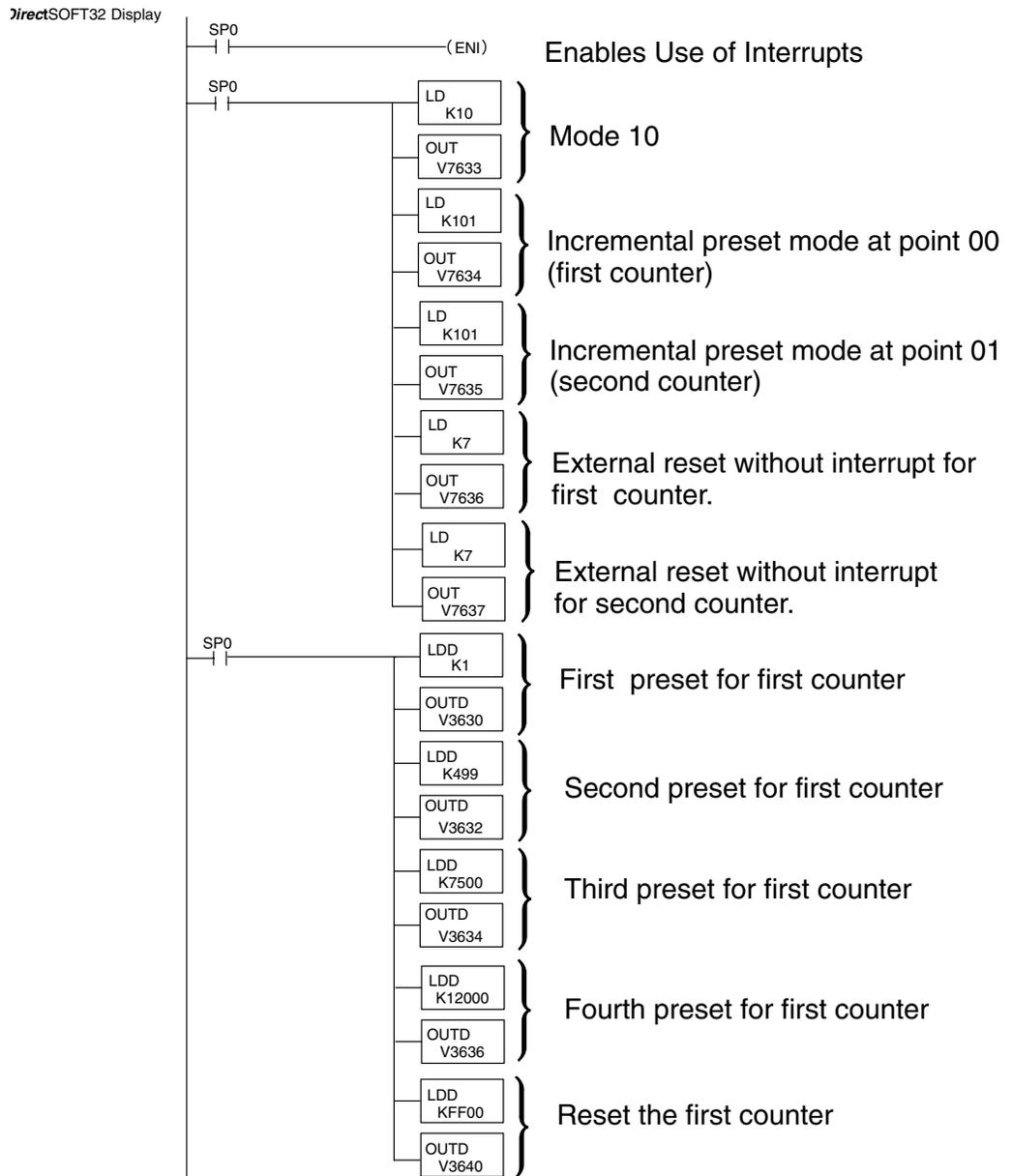
Explanation of RLL

The following describes the events as they are scanned and executed in the example program on the preceding page.

1. The START button, which is connected to X10, is pressed. Y10 latches ON and the inverter starts ramping to low speed.
2. When the D2-CTRINT counts 100 pulses, which is the first preset (100), SP540 goes HIGH and latches Y11 ON. This switches the inverter to medium speed mode.
3. When the D2-CTRINT counts 4000 pulses, the second preset, SP541 goes HIGH and latches Y12 ON. This switches the inverter to the high speed mode.
4. When the D2-CTRINT counts 10000, SP542 goes HIGH and unlatches (resets) Y12 OFF. This switches the inverter back into the medium speed mode.
5. When 17000 pulses are counted, the fourth preset, SP543 goes HIGH and resets outputs (Y10, Y11) OFF. This turns the inverter OFF.

Example 4: Counting Pulses at a High Frequency Where Accuracy is Critical

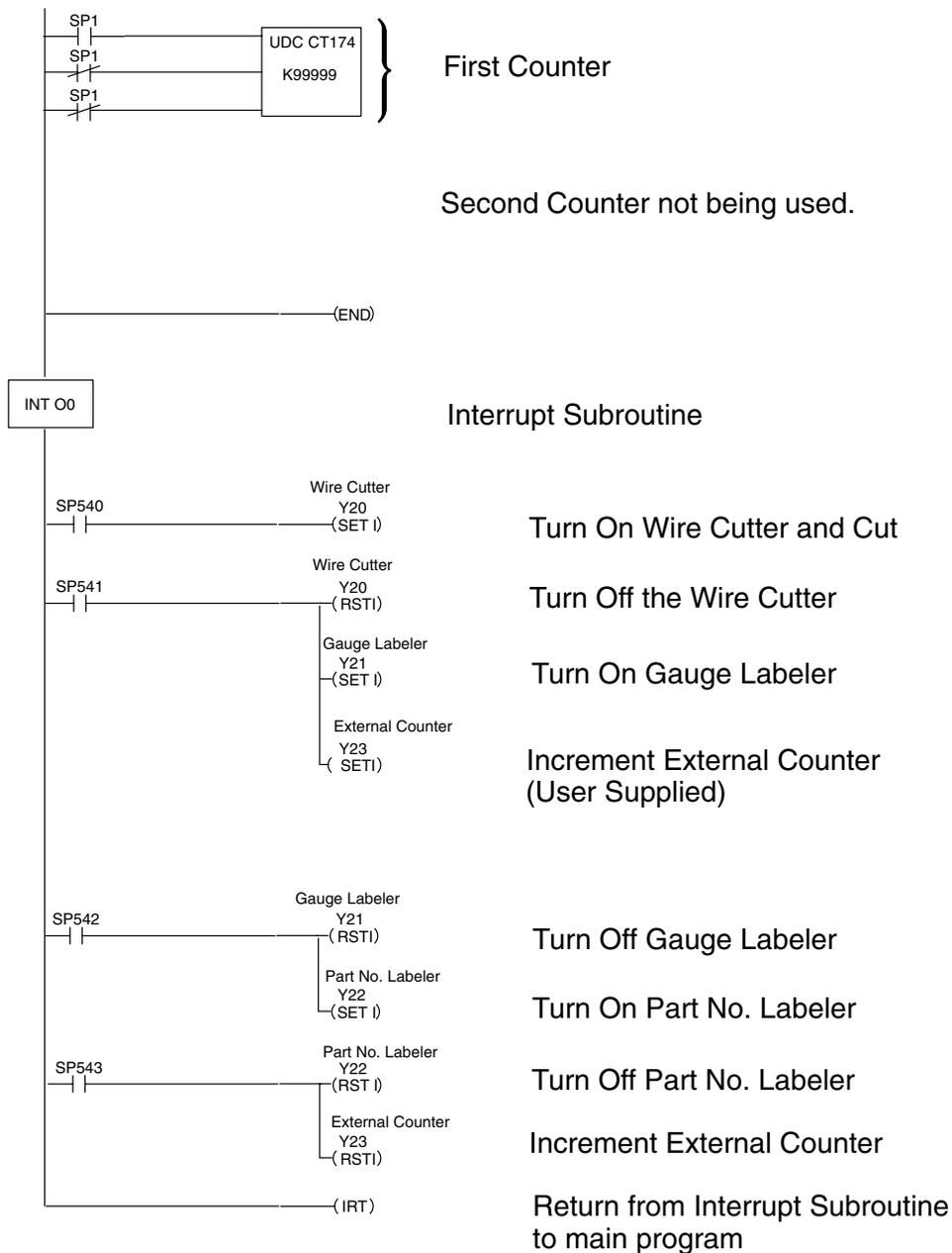
In this example, wire being drawn from a spool is constantly being labeled and cut to length. The process requires continuous unwinding of the wire for cutting. An encoder is connected to a roller shaft and is producing pulses at a high frequency which are used to determine the length of the wire for cutting. A reset of the counter at these high frequencies would cause a loss of pulses (depending on CPU scan time) and result in some wire lengths to be inaccurate. To avoid this problem, a program can be written that resets the counter immediately at the proper preset (to be determined) with no loss of pulses.



Continued on next page

Mode 10
UP Counter

DirectSOFT32 Display



Equal Relays SP540, SP541, SP542 and SP543 turn ON when their respective presets are reached. They are Incremental and not Absolute; each preset is added to the previous preset. For example SP542 goes ON when (1+499+7500) or 8000 total pulses have been reached. Note the use of SET I and RST I instructions inside the interrupt routine.

Troubleshooting

Things to check

The following information may provide some assistance in dealing with problems which may be encountered when setting up the D2-CTRINT module, should they occur. ***Experience has shown that most problems occur because of improper configuration. Always re-check the CPU setup before anything else.***

For verifying types of inputs (or outputs) besides those related to UP counting, see the Chapters in this manual covering the specific function. Listed below are some things that could possibly go wrong with the UP counter inputs:

1. The UP counter is not counting when the encoder or other device is making low to high transitions.
2. The status indicator LED is not lighting for the input point where the UP counter is wired (i.e. points 00 and 01).
3. The counter does not appear to be counting synchronously with the input device's transitional states.
4. The counter is not resetting itself after reaching the preset.
5. The counter is not jumping to the subroutine as expected when the preset is reached.
6. The counter is counting properly and executing the interrupt properly, but it does not continue counting after the interrupt subroutine has been completed.

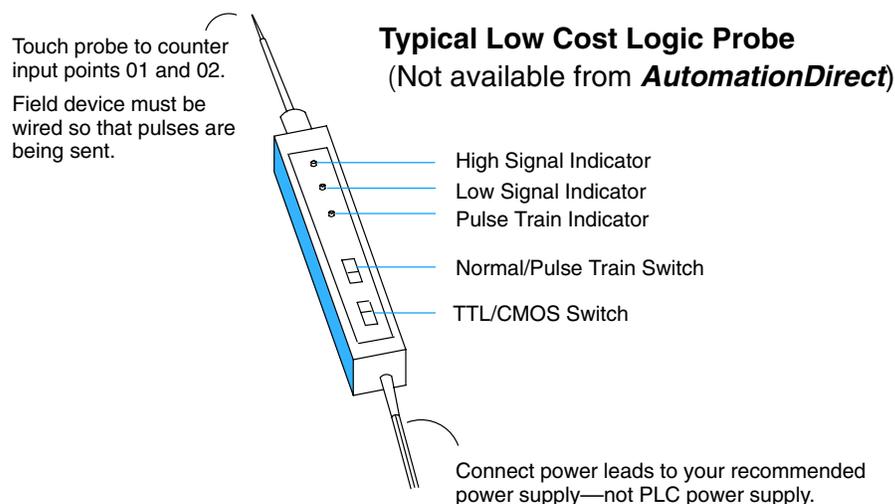
Counter Doesn't Count

The actual count should be in view when monitoring the counter with ***DirectSOFT32***. If this is not happening:

1. The software configuration may be incorrect. Re-check the setup.
2. The field device may be defective.
3. The field device may be O.K., but it is too fast for the counter.
4. The wiring may be defective.
5. The input voltage may not be within specification.
6. The D2-CTRINT module is defective.

No Input Signal

Verify that there is a signal present at point 01 or 02. Check the encoder, or input device, which is connected to the module. Instrumentation can be used to check the 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.



A rotary encoder may also not be operating properly because of a poor coupling between the encoder and the motor shaft. Check to be certain that the coupling is not defective. Check the specifications of the field device. Make sure that the output signal matches up the specifications of the D2-CTRINT module.

Too fast – The pulse rate cannot exceed 5 kHz. If this is suspect, try to slow down the pulse rate to see if the problem is solved. The pulse width may also be too narrow. The pulse must remain high for at least 100ms in order for the module to detect it.

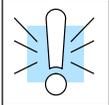
Wiring - Simple as this might seem, quite often poor wiring is the cause of many problems. Make sure that there is a complete electrical loop between the device and the input module. Along with visual inspection, use a voltmeter to check this.

Input Voltage - If the input device is producing a signal which is less than 12 volts, the counter will either not function properly or function improperly. Use a field device with the proper signal level if necessary.

Improper Configuration - The module may be looking for a counter which doesn't exist. Check the RLL program and be sure that the counter is addressed properly. Be certain that the inputs to the counter are properly configured. Is the counter enabled? If using a DL230, only one High Speed Counter can be used.

LED's Do Not Light Make sure the PWR or BAT LED's are not lit on the CPU module. If either the battery power or the external power indicators are not illuminated, there is either a defective power supply, the batteries are worn out or there is no power to the PLC.

Check and be sure that the status indicators are blinking as pulse signals are received at the proper input point on the module (i.e. 0 or 1). Check the field device to make sure it is operating. Use an oscilloscope or digital counter to verify the presence of a pulse train at the input point(s) or use the Change Value feature in **DirectSOFT32** to force the input point ON. Refer to the **DirectSOFT32 Software Programmers Manual** to use this feature. If after forcing the signal ON, and the appropriate LED does not light, the module is defective.



WARNING: Take all necessary precautions to protect personnel and equipment when forcing inputs and outputs, since the equipment on the output side of the system may be energized.

Non-Synchronous Pulsing If the counter is suspect of counting faster than the field device is sending pulses, then the interface may be experiencing noise. Try connecting a shielded wire between the field device and the module input. The wire should be shielded at one end only-- normally the encoder side. If the PLC is interfaced to a rotary shaft encoder, a loose coupling could also be causing the problem. Check the coupling.

No Reset If a reset is not present when expected, check the configuration. Is the proper value for a reset being used? Is the module looking for an external reset? If so, is the field device supplying the reset signal (which should be connected to either points 02 or 03) operating properly. Manually operate the reset device.

Not Jumping to Interrupt Is the interrupt subroutine labeled properly? The interrupt subroutine must have the same number as the I/O point to which the counter is connected. Is there supposed to be an external interrupt, but the external device is not sending the signal? Does the external interrupt device meet the input criteria? Check all of these possibilities.

Not Returning from Interrupt Does the interrupt subroutine end with an Interrupt Return instruction in the RLL program? Is there an endless loop inside the subroutine? Is there a conditional return and the condition has not been met?

Mode 20 – High Speed UP/DOWN Counter

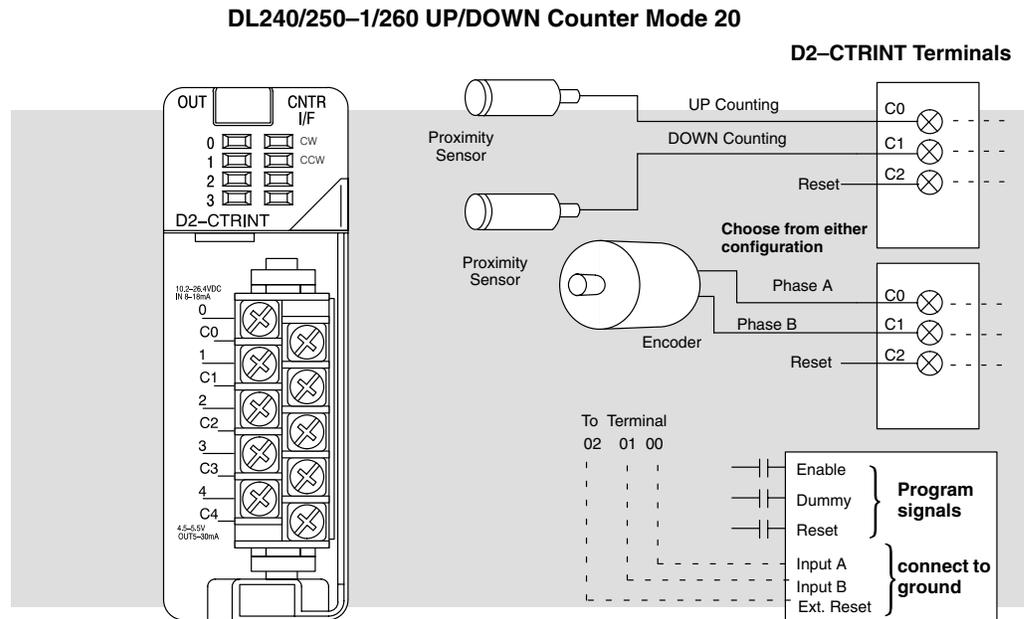
In This Chapter. . . .

- Wiring the UP/DOWN Counter
 - Configuring the UP/DOWN Counter Parameters
 - Writing the Control Program
 - Verification of Proper Operation
 - Troubleshooting
-

Using the UP/DOWN Counter, Mode 20 (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 20, UP/DOWN Counter will be the only mode discussed 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 more about the D2-CTRINT's UP/DOWN counting function.



The above diagram and illustration shows points C0 and C1 which are the two primary UP/DOWN counter connecting points for the pulse input devices. Point C2 is the reset input. The counter can be reset either through the operating program using relay ladder logic or they can be reset by an external field device. Points 00 through 03 are the respective *common* connections for the counter inputs and optional external reset. Point 04 and C4 are for pulse output signals and are not used in this mode.

UP/DOWN Counting Overview

Encoders In General

Encoders are used in all types of applications, including machine tooling, semiconductor positioning and multi-axis positioning. In the most general sense, the encoder puts out a certain number of pulses for each revolution of the shaft. Encoders can be easily interfaced to the D2-CTRINT module for accurate position monitoring and control. There are two types of encoders available, incremental and absolute. These encoders have a rotating shaft which are normally coupled (mechanically) to motors for machine control.

The incremental encoder has a pulse output which corresponds to the rotation of the shaft. This pulse output is the encoder resolution corresponding to the number of pulses per revolution. There are two outputs from an incremental encoder, A and B. There is also an output for home position. The encoder outputs can be used for either counting or for direction control (quadrature).

The absolute encoder is used for more precise positioning. The output from the absolute encoder is a binary value relating to degrees per revolution (360/revolution).

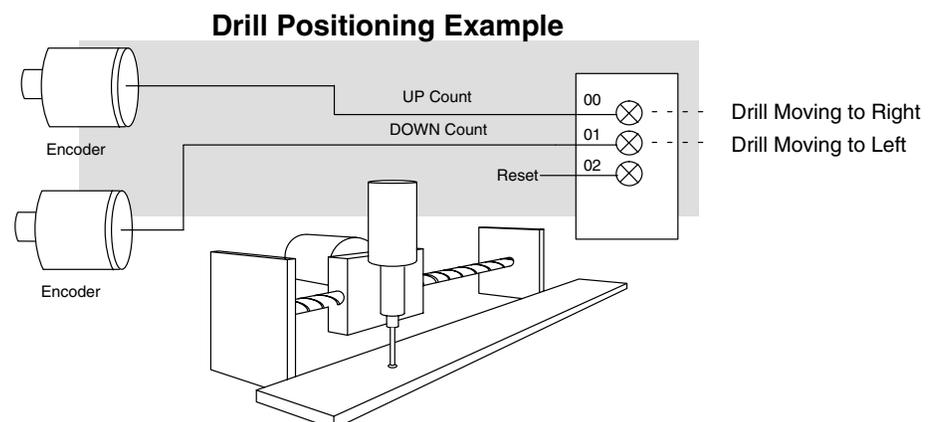
Both types of encoders, incremental and absolute, are available from **AutomationDirect**. More information about the encoders is available from both our website and catalog.

There are two basic ways to use the UP/DOWN counter, either for standard UP/DOWN counting from two separate sources (i.e. single pulse from two encoders) or for quadrature UP/DOWN counting using encoder outputs A and B.

Standard UP/DOWN Counting

The first way, using the D2-CTRINT module as a standard UP/DOWN counter, is commonly used for position control. For example, a drill head may need to be moved in a certain direction horizontally. At the same time, the drill head position may also need to be tracked. This can be done by counting pulses received from two encoders. When the drill moves in one direction, an encoder will increment an accumulated count (UP count). When the drill moves in the opposite direction, a second encoder will decrement the accumulated count (DOWN count). The drill position will be known by looking at the accumulated count at any given time.

Using the high speed positioning example, point 00 would receive signals from an encoder that keeps track of movement in one direction, to the right. Point 01 would keep track of movement to the left via a second encoder. Incrementing and decrementing the UP/DOWN counter can happen simultaneously or during separate intervals of time.

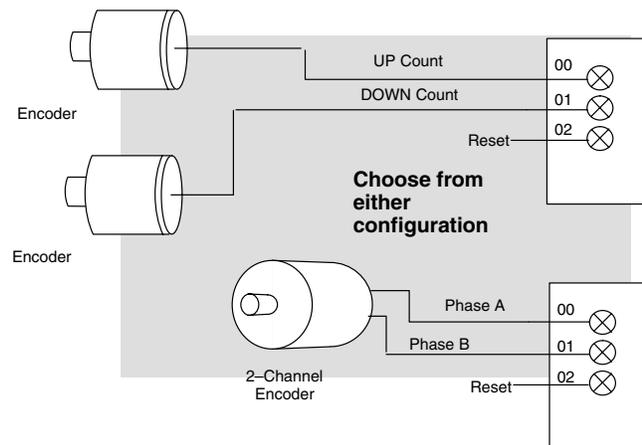


Quadrature UP/DOWN Counting

The second way to use the D2-CTRINT is to keep track of the direction which a motor is turning. This is accomplished by using a quadrature encoder with the A and B outputs connected to the UP/DOWN counter. The A and B pulse trains which are processed have a 90 degree phase difference. This means that one pulse train leads the other pulse train by 90 degrees. From this information, the D2-CTRINT can be configured to know if the shaft is turning clockwise or counterclockwise.

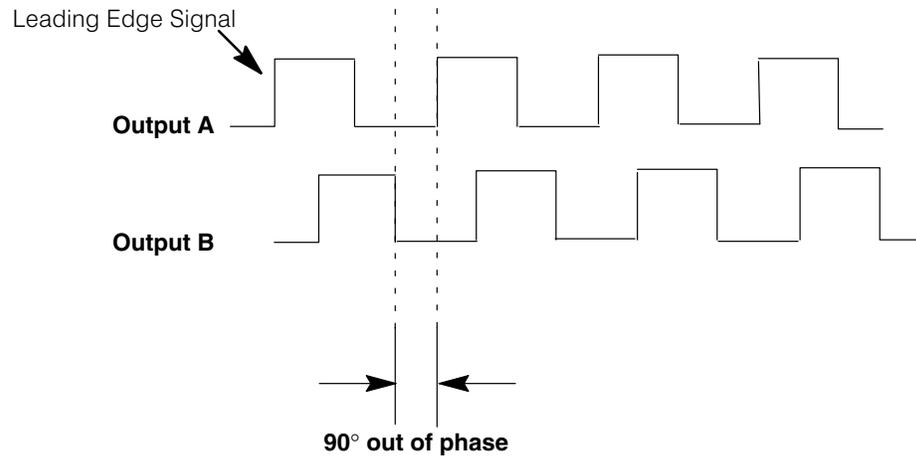
The diagram below, shows both standard UP/DOWN counting and the quadrature UP/DOWN counting being performed with encoders. The encoders shown for the standard UP/DOWN counting could be replaced with photoelectric sensors, or even limit switches.

Quadrature counting, on the other hand, is confined specifically to shaft encoders.

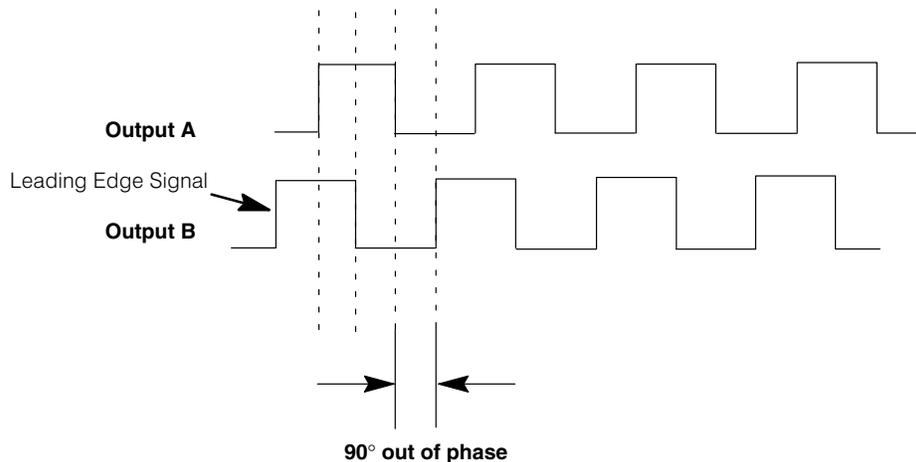


How a Quadrature Encoder Works

As mentioned previously, a quadrature encoder has two outputs, A and B. These outputs can be used to sense position and direction of a motor driven device. This is possible because of the quadrature outputs, A and B, are 90° out of phase to each other as shown in the figure below:



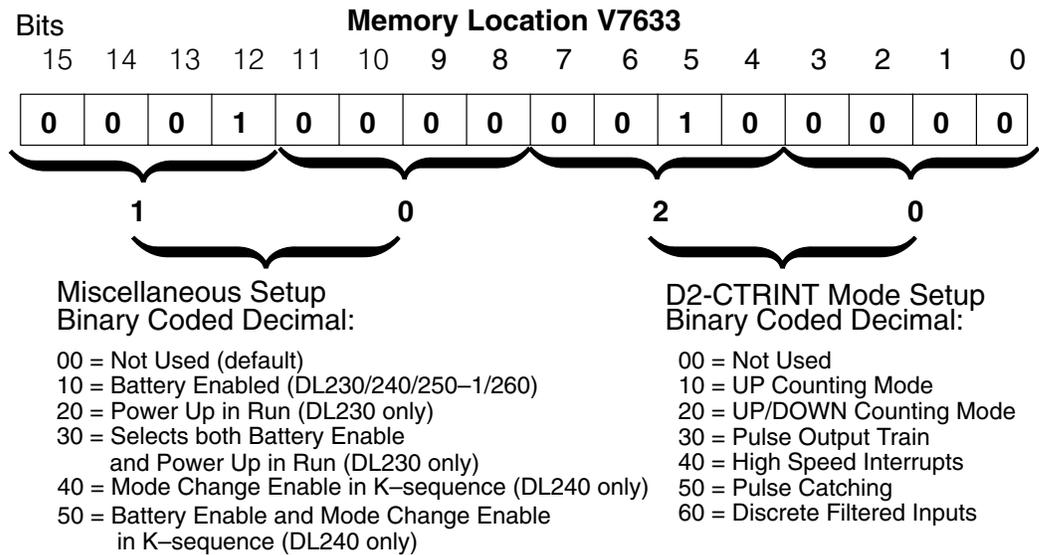
Like any quadrature encoder, four unique logic states are created internal to the encoder. This is based on the rising edge to rising edge (one cycle) on output A or B. The rising edge of Output A, the rising edge of Output B, the falling edge of Output A and the falling edge of Output B form the complete quadrature. In the example above, Output A is the leading signal, and it will cause the counter to count up. This would indicate to the CPU that the motor shaft is turning *clockwise*. However, suppose instead that the signals looked like the diagram below:



Now Output B is the leading signal, and it will cause the counter to count down. This would indicate to the CPU that the motor shaft is turning *counter-clockwise*.

Understanding V-Memory Setup Locations

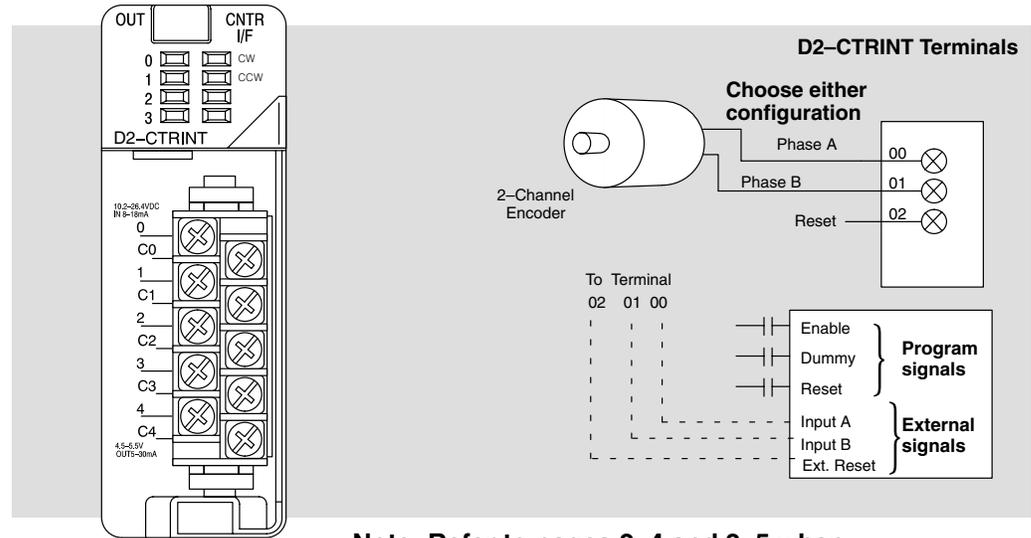
V-memory location V7633 is the most important of all the reserved memory areas because it stores the numeric value which lets the CPU know which mode has been selected. The following diagram shows the 16-bit word and the various information it stores—including the values used for the Counter Interface Module. The example shown here uses the UP/DOWN counting mode. The lower bits are set to 20 and the upper bits are set to 10 so the battery backup is enabled. Together they form the number 1020.



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.

DL240/250-1/260 Quadrature Counter Configuration



Note: Refer to pages 2-4 and 2-5 when wiring your particular device.

The diagram above shows the physical layout of the front of the module. The actual wire connecting points on the module can be visually related to the various functions that are associated through the default counter configuration.

There are programming examples in this chapter which show how to change some of these defaults.

Default settings with V7633 set to XX20:

V-Memory Location and I/O points on the module	Default Value in DL240/250-1/260	Description
V7634 (point 00)	0002	UP/DWN Input
V7635 (point 01)	0000	Not Used
V7636 (point 02)	0007	Reset
V7637 (point 03)	1006	Discrete Input

The values shown above have the following meaning:

0002 =UP/DOWN quadrature, Absolute counter mode

0007 = External reset input without an interrupt

1006 = Discrete input with 10 ms filter

While discussing memory configuration, it is important to mention the reserved memory for the presets. The default memory locations are shown below.

CPU	Channel 1 (pt.00) & (pt.01)
DL240/250-1/260	V3630 thru V3707

Setting Up the CPU

Configuring the V-Memory

The DL240/250-1/260 CPUs check the V-memory to see if there is a High Speed Counter Interface Module present. There will be a hexadecimal number 10, 20, 30, 40, 50 or 60 in V7633 if a module has been properly configured. This is the value which is entered in the RLL program setup. If the CPU finds that a Counter Interface Module is present, other V-memory locations will be checked to see how each point of the module has been configured.

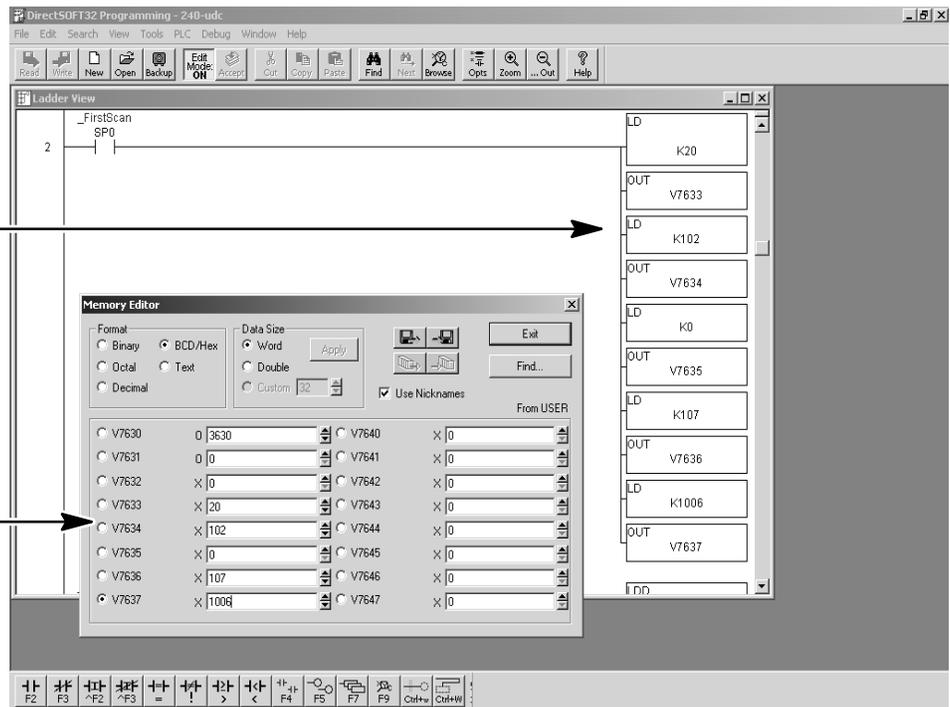
The values can be entered into memory by using either a handheld programmer or by editing them into a control program using **DirectSOFT32**. The following examples will show how to use **DirectSOFT32** to configure the UP/DOWN Counter.

Step 1: Entering the Mode Selected

The UP/DOWN Counter is Mode 20 which is the value to be set into V7633. The following **DirectSOFT32** diagram shows the setup procedures for communicating with your DL240/250-1/260 PLC. Refer to the **DirectSOFT32 Programmers User Manual** for more details.

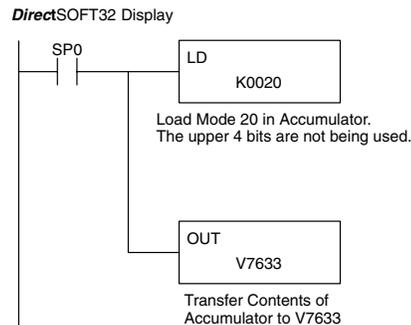
Setting the
V-Memory
using RLL

Setting the
V-Memory
using the
Memory
Editor



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.

The following RLL example shows how to set the UP/DOWN Counter, Mode 20, in V-memory location V7633.



Two 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, 20 is to be placed in V7633. This value is loaded into the accumulator, LD K20. 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. Thus, it will load the values in memory initially, thereby keeping the scan time to a minimum

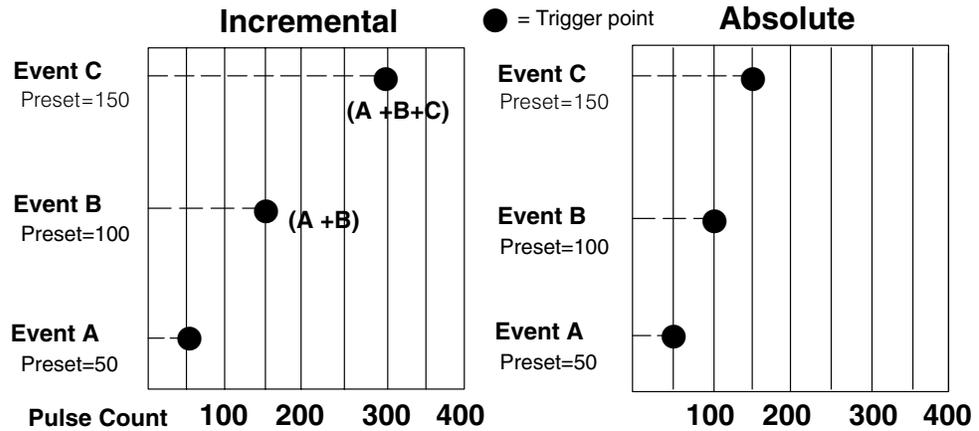
**Step 2:
Select the
Preset Mode**

There are two different preset modes to choose from—either Absolute or Incremental. If the default Absolute mode of presets has been accepted, skip this step and go on to Step 3.

To understand the concepts of using the Absolute and Incremental preset modes, it is essential to know some basics about the counter's presets. Inside the PLC's memory up to twenty-four (24) preset values can be setup for each counter. A preset is the number of pulses which are chosen to be counted before an event is to be initiated.

Presets are entered into successive areas of V-memory. With the Absolute mode, the presets are all independent. That is, the counter compares the actual total count received from the D2-CTRINT module to a preset, when the two are equal, the event is triggered. With an Incremental preset mode, however, the presets are related to each other. In such case, the counter reaches preset A and triggers event A, then preset B is added to preset A and that becomes the number of pulses that must be counted before event B is triggered. Preset C is added to the sum of presets A and B and that is the number of pulses required for event C to be triggered. The process of adding all the presets continues until the CPU is notified that there are no more presets to satisfy.

Below is an example showing the difference between using the presets Incrementally and Absolutely.



Assume that a limit switch (SW1) is being connected to one of the high speed counters. The counter is to initiate three different events at certain points in time determined by the pulse count received. Assume that the presets are stored (associated with each of the three events) in successive memory as 50, 100, and 150. In this example, these values are to be attended to in an Incremental fashion. The counter would, in such case, trigger the 1st event when it counts 50 pulses, the 2nd event when it receives 150 pulses total (50 + 100) and the 3rd event when it receives 300 pulses total (50 + 100 + 150).

The result would have been different if the counter had been configured to count in the Absolute mode. In such case, the counter would trigger the 1st event when it counted 50 pulses, the 2nd event when it received 100 pulses and the 3rd event when it counted 150 pulses.

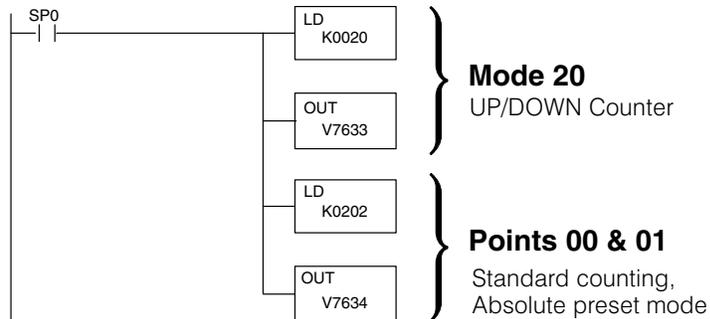
Configuring the Preset Mode

Placing a preset value in memory location V7634 lets the CPU know which preset mode (Absolute or Incremental) to use. The same 4-digit hex value in that same position will also indicate whether the select is either standard or quadrature counting. The following values are available:

- 0002 = Quadrature counting, Absolute preset mode
- 0102 = Quadrature counting, Incremental preset mode
- 0202 = Standard counting, Absolute preset mode
- 0302 = Standard counting, Incremental preset mode

Below is the ladder logic for entering the information for an application where standard UP/DOWN counting is to be used with the Incremental preset mode.

DirectSOFT32 Display

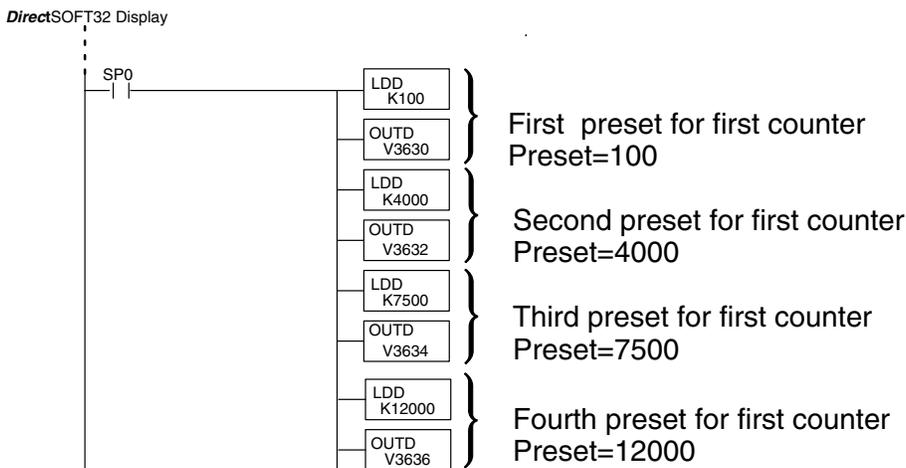


**Step 3:
The Presets**

The final step for setting up the CPU is to configure the presets. There are up to 24 presets available for UP/DOWN counting. These presets can be changed in the relay ladder logic, but are only seen by the counter at a PLC mode change or reset of the counter.

Loading the Presets

The presets are loaded into consecutive V-memory locations, starting with the default memory location of V3630. Use LDD and OUTD instructions as indicated in the example below:



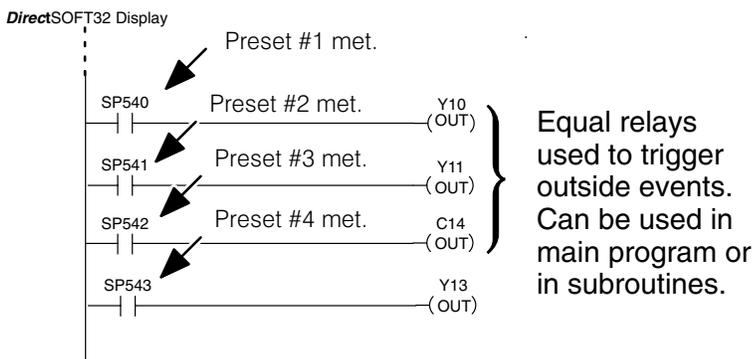
Using Negative Presets

Presets can be changed at any time; but in order for the counter to recognize the changed values, the counter must be reset or the CPU must go through a mode change.

Negative presets can be used, but in order to do so, the MSB in the upper word of the preset must be made HIGH. Example: $-250-1 = 80000250$.

Triggering Presets to Outside Events

Each of the presets are associated with special relays called equal relays. They can be referenced in your relay ladder logic just like any other relay. Outside events are triggered whenever the preset assigned to a particular relay is satisfied by the pulse count, it closes. Unlike the UP counter Mode 10 (where there are two high speed counters and two sets of equal relays), there is only one UP/DOWN counter and one set of equal relays for Mode 20.



Mode 20:
UP/DOWN Counter

This table shows the label for each equal relay and the default V-memory address assigned to each.

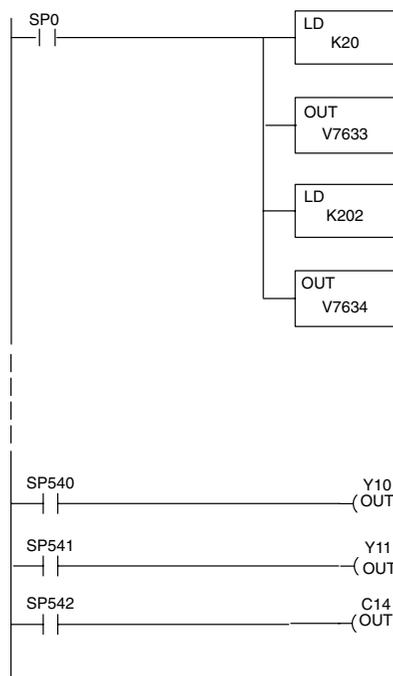
DL240/250-1/260 Equal Relays
Channel 1 (points 00 & 01)
SP540 (V3631/V3630)
SP541 (V3633/V3632)
SP542 (V3635/V3634)
SP543 (V3637/V3636)
SP544 (V3641/V3640)
SP545 (V3643/V3642)
SP546 (V3645/V3644)
SP547 (V3647/V3646)
SP550 (V3651/V3650)
SP551 (V3653/V3652)
SP552 (V3655/V3654)
SP553 (V3657/V3656)
SP554 (V3661/V3660)
SP555 (V3663/V3662)
SP556 (V3665/V3664)
SP557 (V3667/V3666)
SP560 (V3671/V3670)
SP561 (V3673/V3672)
SP562 (V3675/V3674)
SP563 (V3677/V3676)
SP564 (V3701/V3700)
SP565 (V3703/V3702)
SP566 (V3705/V3704)
SP567 (V3707/V3706)



NOTE: Pointers for the start of these addresses are stored by the CPU at V7630. If there is a conflict of addresses because of pre-existing code written to these addresses, change the default block of addresses by placing a different pointer value in V7630. For example, to change the starting address for the relays belonging to Channel 1 to V2500, change the program to write an octal 2500 to V7630. The results are that the CPU will reserve 48 consecutive 16-bit memory addresses (32 bits per preset for the 24 presets available) for SP540 to SP567 equal relays.

The example below shows how the relay labels are used in a relay ladder program. A portion of the program uses three of the equal relays to trigger specific output points.

DirectSOFT32 Display



Equal relays used to trigger outside events. Can be used in the main program and in subroutines.

Also notice that the 0202 is used to tell the CPU that the standard method of counting (as opposed to quadrature counting) is used and each preset is to be related in an Absolute manner (as opposed to an Incremental relationship).

The Other Channels

Filtered Input – Point 03

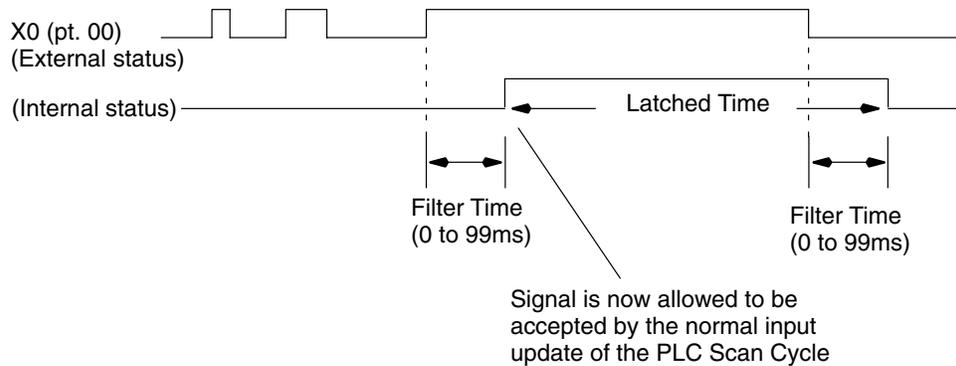
When Mode 20 is selected, the CPU automatically writes the value of 1006 to V7637. The digits 06 means discrete input, and 10 selects a 10 ms filter. Point 03 is setup by default.

Discrete Filtered Inputs

Some applications have inputs from field devices which produce noise. Filtering can help to reduce or eliminate the noise from the inputs caused by switch bounce or other sources.

When an input signal is first detected at point 03, a programmable filter is activated which begins a timed countdown. The ON status of the signal is temporarily prevented from being read by the input update of the CPU. The ON signal must stay present for a certain amount of time for the filter to “time out”.

Referring to the diagram below, once the signal has remained ON for the required time, it is latched and allowed to be accepted by the CPU during the normal input update of the PLC scan cycle. The signal is latched for the remaining duration of the ON signal plus an amount of time equal to the filter time. The filter time can be programmed for 0 to 99ms in 1ms increments. It is set to 10 ms by default. The filtering time can be changed or a different function, Pulse Catching, can be selected for this input point.



Note: If zero is put in V-memory as the filter time. The CPU will treat the configuration as having no filter.

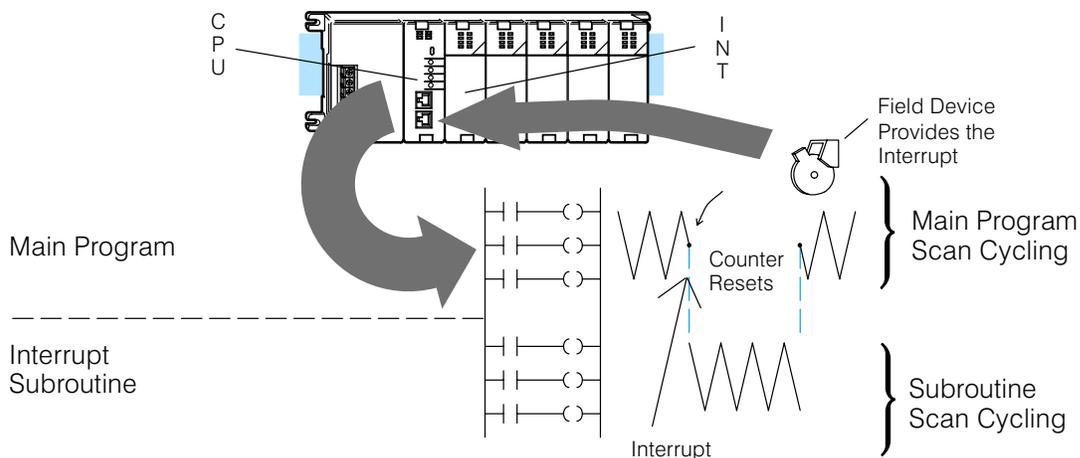


**Point 02 – Reset
w/o an External
Interrupt**

**External Reset
with Interrupt**

When Mode 20 is selected, the CPU automatically writes 0007 to V7636. This value represents a reset at point 02 without an interrupt. If the reset is to trigger an interrupt, replace 0007 with 0107.

An external reset with interrupt sets the counter to zero and allows temporary suspension of the normal scan process in the main program while an interrupt subroutine is executed. When the subroutine is complete, the CPU automatically resumes its routine scan cycle, starting where it was interrupted. The following diagram illustrates the process. For a complete discussion of interrupts, read Chapter 6.



Custom Configurations

Up to this point, only Mode 20 default settings have been discussed. The default settings will be suitable for many applications, they will not require custom configuring. 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.

Mode 20 Options

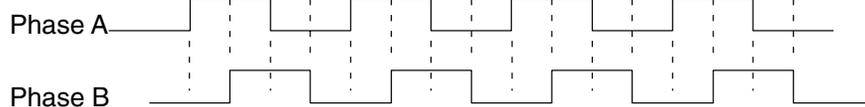
Mode 20
Custom
Configuring

Point Number	V-Memory	Definition (One per point)	Hex Value
point 00	V7634	UP/DOWN Counter having Phase A (quadrature) or 1st Encoder (standard)	0002 (quadrature, Absolute) default
			0102 (quadrature, Incremental)
			0202 (standard, Absolute)
			0302 (standard, Incremental)
			1002 (quadrature, Absolute) 4x counting*
			1102 (quadrature, Incremental) 4x counting*
point 01	V7635	Phase B or 2nd Encoder Input Depending on value in V7634	0000 default
point 02	V7636	Reset UP/DOWN Counter default	0007** (no interrupt) default no Z pulse or index marker recognition
			0107 (interrupt) can recognize Z pulse or index marker
			0207 (interrupt) no Z pulse or index marker recognition
			0307 uses Z pulse recognition in the interrupt
		Pulse Catcher	0005
		Discrete Input	xx06 (xx=filter time=0 to 99ms)
point 03	V7637	Discrete Input	xx06 (xx=filter time=0 to 99ms)
		Pulse Catcher	0005
point 04	Not Used	Pulse Output (CCW)	Not Used

Note: The lower byte of V7633 is set to 20.

* This feature will allow the counter to count 4 times more with the same encoder (250-1/260 only).

4x counting → 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
 Normal counting → 1 2 3 4



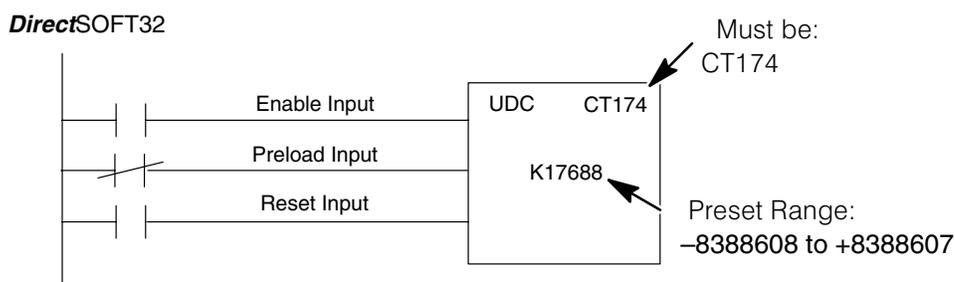
** In a high speed application using a high resolution encoder, with the CPU programmed with presets that span the resolution of the encoder, you **may lose** pulses in the higher range of presets.

When using constants **K7** or **K107** in V7636 or V7637, the counter module will read the preset V-memory each time the counter is reset or at the index marker location.

When using **K207** or **K307**, the counter module reads the preset V-memory only at power up, or when a CPU mode change occurs (i.e. PROGRAM to RUN transition).

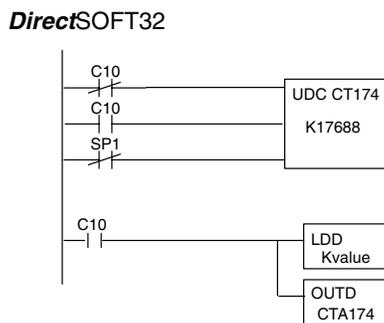
Writing the Control Program for the UP/DOWN Counter

Once the UP/DOWN Counter has been configured, the control program can be written. Writing the RLL program for the high speed counter is much the same as for a regular counter. The embedded high speed counters which are activated through the D2-CTRINT module, have three inputs. The first input (Enable) allows counting when active. The middle input (Preload) allows you to change the current count. The bottom input is to reset the counter. The preload input must be off while the counter is counting.



NOTE: To enter a negative number an 8 is placed in the most significant digit. For example: 80000570 entered in the preset gives a preset value of -570.

The mnemonic for the counter is UDC. It is found in the *DirectSOFT32* instruction browser that pulls down using the hotkey F7 while in the Edit Mode. After selecting the UDC, a box appears asking for the counter address and preset value. The counter address in the above example is CT174. The counter address for Mode 20 is always 174. The preset value of 17688 has been used for this example. This is the number of pulses to be received before the output of the counter goes high. Any value between -8,388,608 and +8,388,607 can be programmed for the preset. After entering the counter address and preset, enter the contact addresses for the contacts which are automatically drawn in the RLL program.

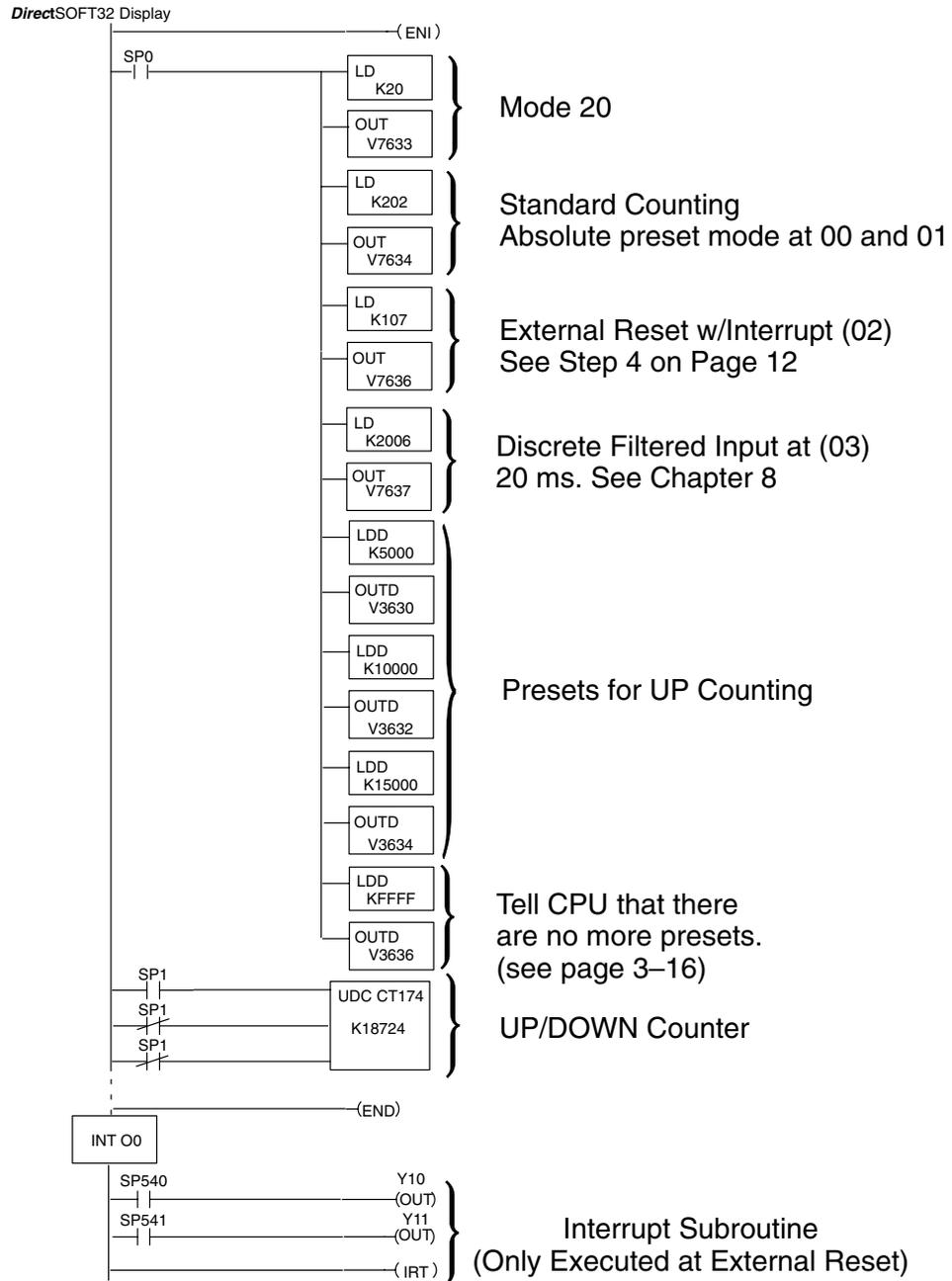


C10 is shown normally closed. When pulses are received at point 00, the current value in CTA174 will increment. When pulses are received at point 01, the current value will decrement. When C10 energizes, CTA174 stops counting and Kvalue is loaded for a new preload value. When an input is received at point 02, the counter will reset.

When the Enable input is energized, the high speed counter will respond to pulses at point 00 and increment the counter. When pulses are received at point 01, the counter will decrement. The reset input behaves in a logical OR fashion with the physical reset input point 02. The high speed counter can receive a reset from either the reset contact in the RLL or the external reset point 02 if it has been configured as an external input.

Example 1: UP/DOWN Counting with an Interrupt

Below is an example of how UP/DOWN counting with an interrupt can be programmed.



Mode 20:
UP/DOWN Counter

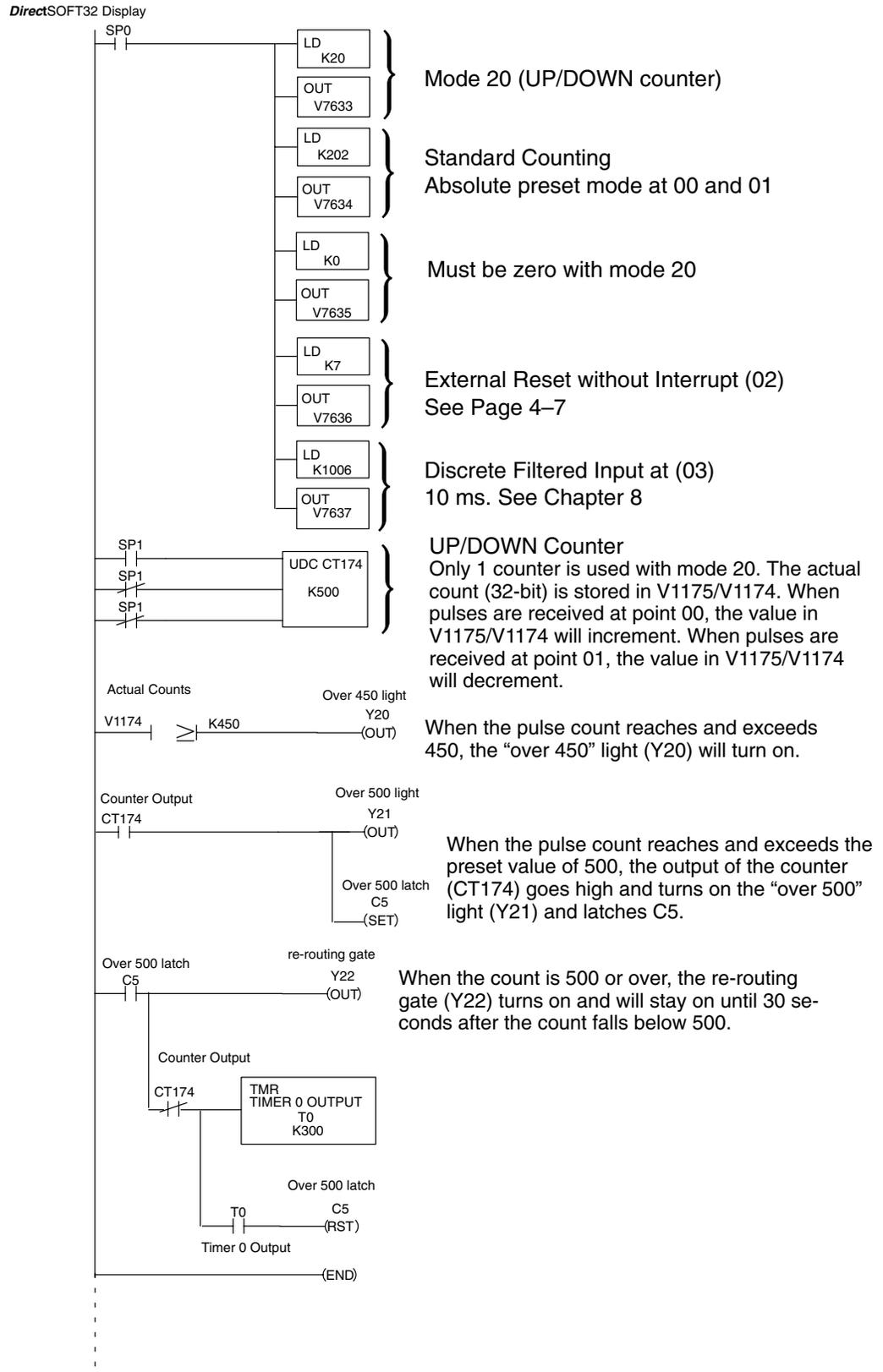
The load accumulator instructions have set up the V-memory as required, i.e. 20 in V7633 for the mode and 0202 in V7634 to designate a standard UP/DOWN counter with the Absolute preset mode. By placing 0107 in V7636, an external reset for counter 174 is selected and it will execute interrupt 0 on the rising edge of the reset. Presets for UP/DOWN Counting have been stored in memory locations V3630 through V3635. The next even-numbered memory location following this has FFFF to indicate there are no more presets.

Example 2: An UP/DOWN Counter with Standard Inputs

In this example, assume there is a conveyor belt A that transports bottles to be inspected. During the course of the process, one sensor is keeping track of the bottles that are going onto belt A for inspection, and another sensor is keeping track of how many bottles are being removed to the finished product line.

When a quantity of 500 bottles has been reached in the process, an OVER 500 light turns on and a re-routing gate is activated to channel the incoming bottles to conveyor belt B. The re-routing gate will stay activated for 30 seconds after the conveyor belt A contains less than 500 bottles.

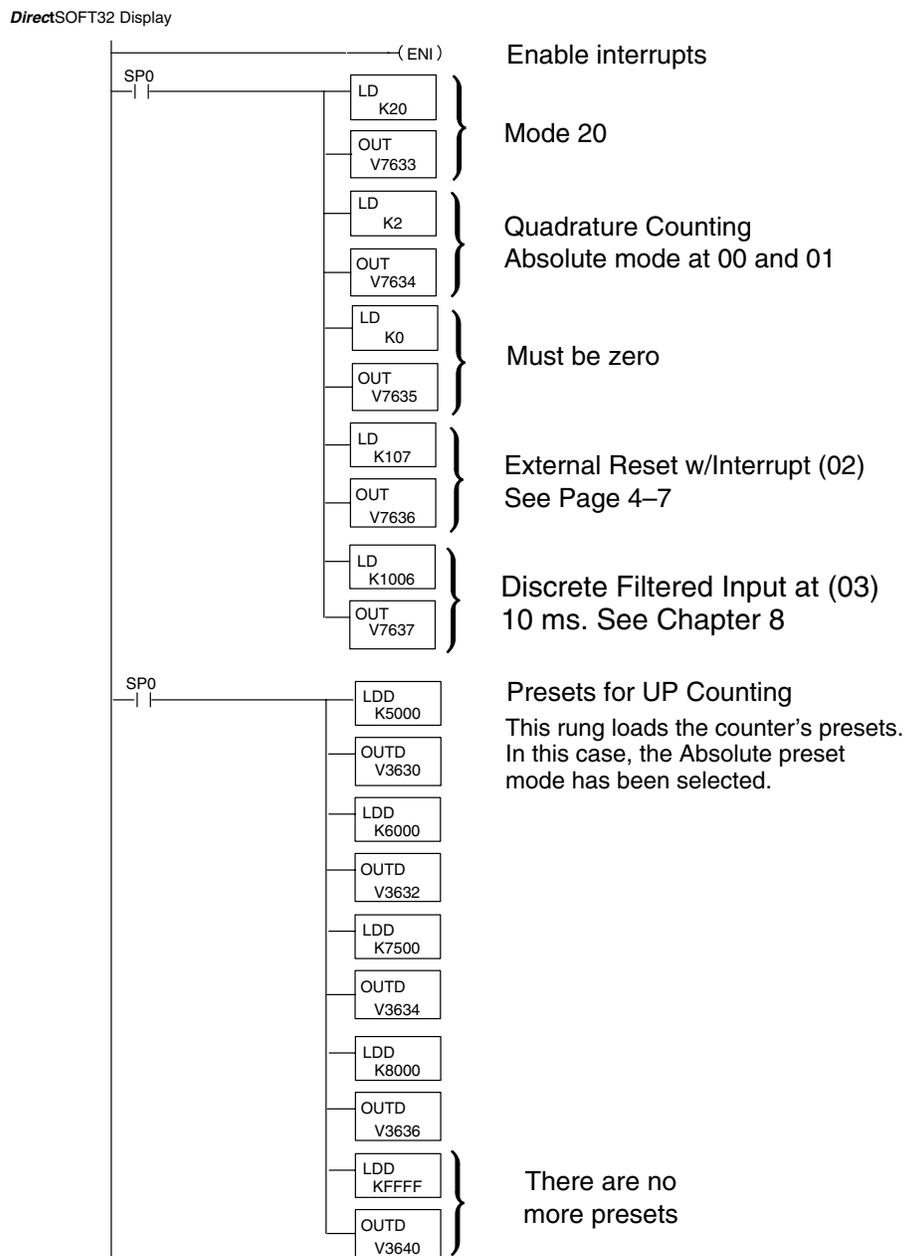
The program on the following page shows how the RLL program might be written for the process. Note the use of V1174. This memory location stores the current count for CT174 which is used with the D2-CTRINT.



Example 3: Quadrature Counting

In this example, a wooden work piece is being drilled with three (3) holes. The holes are injected with glue for dowels to be inserted at another workstation. A quadrature encoder is connected to a positioning table which is moving a drill press horizontally over the work piece. The positioning table will stop and the drill press will lower to drill a hole in the exact location. After the three (3) holes are drilled in the work piece, the positioning table reverses direction and injects glue into the same holes.

The following program shows how this is done:

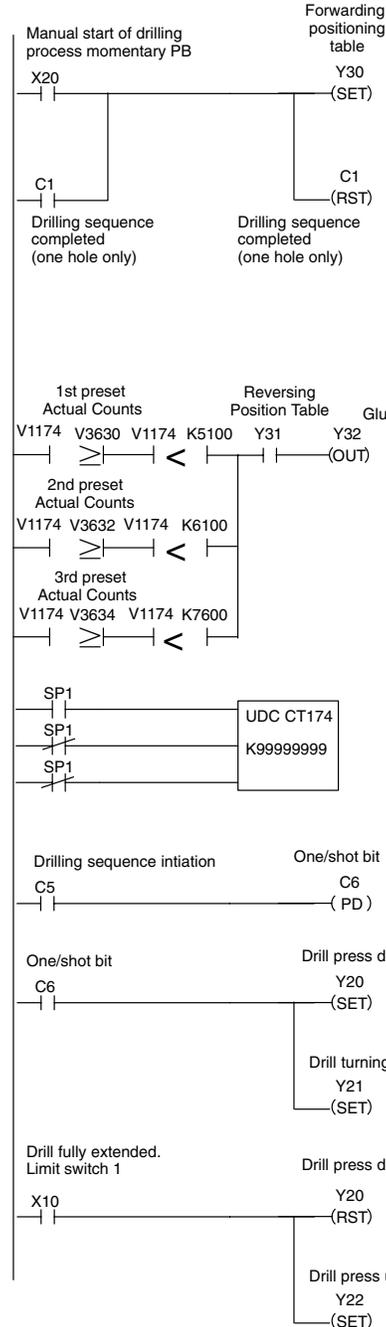


Continued on next page

Mode 20:
UP/DOWN Counter

Continued from previous page.

DirectSOFT32 Display



The operator starts the drilling by moving the positioning table forward. A quadrature encoder is transmitting the pulses to the D2-CTRINT module. After 5000 pulses are counted, the positioning table is stopped and the drilling sequence is started. Two more holes are needed in this work piece so the positioning and drilling are repeated twice automatically. (Stop and jog buttons are left out here for simplicity but should be incorporated in your logic.).

During the reversing of the positioning table, the glue injection is turned on between pulse counts (7600–7499), (6100–5999), (5100–4999) before it reaches its home position and is reset to zero by a limit switch (X2). NOTE: Presets can be added to trigger outputs when counting DOWN as well as counting UP.

When the positioning table moves forward, the current value in UDC174 increments. When the positioning table moves backward, the value decrements.

This rung initiates the drilling portion of our program. When C5 goes high the drill press is lowered and the drill bit starts to turn.

When the drill has reached the proper depth, limit switch 1 (X10) is turned on and stops the downward motion. This, in turn, raises the press up with the drill bit still turning.

Troubleshooting

What Can Go Wrong?

After completing the configuration of the UP/DOWN counter for Mode 20, the counter should work. If there is a problem with the counter operation, the information below and on the following pages may provide some assistance in handling any problems if they should occur. ***Experience has shown that most problems occur because of improper configuration. Always re-check your CPU setup before anything else.***

For verifying types of inputs (or outputs) besides those related to UP/DOWN counting, see the Chapters in this manual covering the specific function. Listed below are some things that could possibly go wrong with the UP/DOWN counter inputs:

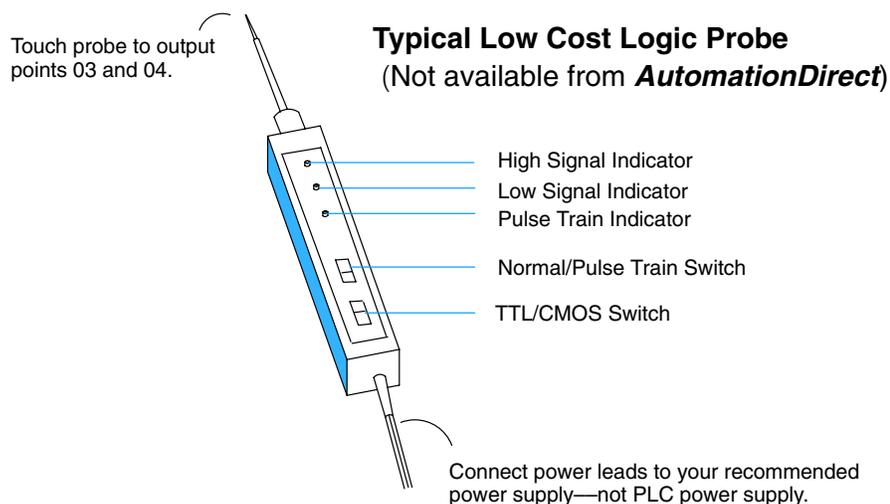
1. The counter is not counting UP or DOWN when configured for standard UP/DOWN counting.
2. When configured for standard UP/DOWN counting, the UP counter seems to be working, but there is no DOWN counting.
3. The status indicator LED is not lighting for the input point where the UP/DOWN counter is wired (i.e. points 00 and 01).
4. The counter does not appear to be counting synchronously with the input device's transitional states.
5. The counter is not resetting itself after it "counts out".
6. The counter is not jumping into its subroutine as expected when it reaches a preset condition.
7. The counter is counting properly and executing the interrupt properly, but does not continue counting after the interrupt subroutine has been completed.

Counter Doesn't Count

You should be able to see counting taking place in the counter block while observing it in the monitor mode with *DirectSOFT32*. If this is not happening:

1. The CPU may be setup improperly. Check the RLL.
2. The field device may be defective.
3. The field device may be O.K., but it is too fast for the counter.
4. The wiring may be defective.
5. The input voltage may not be within specifications.
6. The configuration may be improperly setup.
7. The D2-CTRINT module is defective.

Defective Field Device - If a field device is suspected to be faulty, verify its proper operation first. Examine the characteristics of the pulses being received with an oscilloscope, test equipment type digital counter or an inexpensive logic probe which has a pulse train input capability.



A rotary encoder could also be operating improperly because of a poor coupling between the encoder and the shaft of the motor. Check to make sure the coupling is not defective. Check the specs on the field device. Be certain that its output signal matches up with the specifications of the D2-CTRINT module.

Too fast – The pulse rate cannot exceed 5 kHz. If this is suspect, try to slow down the pulse rate to see if the problem is solved. The pulse width may also be too narrow. The pulse must remain high for at least 100ms in order for the module to detect it.

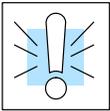
Wiring - Simple as this might seem, quite often poor wiring is the cause of many problems. Make sure there is a complete electrical loop between the device and the input module. Along with visual inspection, use a voltmeter to check this out.

Input Voltage - If the input device is producing a signal which is less than 12 volts, the counter will either not function properly or function improperly. Use a field device with the proper signal level if necessary.

Improper Configuration - The module may be looking for a counter which doesn't exist. Check the RLL program and be sure that the counter is addressed properly. Be certain that the inputs to the counter are properly configured. Is the counter enabled? This function is not available with the DL230.

LED's Do Not Light Make sure the PWR or BAT LED's are not lit on the CPU module. If either the battery power or the external power indicators are not illuminated, there is either a defective power supply, the batteries are worn out or there is no power to the PLC.

Check to be certain that the status indicators are blinking as pulse signals are received at the proper input point on the module (i.e. 0 or 1). Check the field device to make sure it is operating. Use an oscilloscope or digital counter to verify the presence of a pulse train at the input point(s) or use the Change Value feature in **DirectSOFT32** to force the input point ON. Refer to the **DirectSOFT32 Software Programmers Manual** to use this feature. If after forcing the input ON, and the appropriate LED does not light, the module is defective..



WARNING: Take all necessary precautions to protect personnel and equipment when forcing inputs and outputs, since the equipment on the output side of the system may be energized.

Non-Synchronous Pulsing If the counter is subject of counting faster than the field device is sending pulses, then the interface may be experiencing noise. Try connecting a shielded wire between the field device and the module input. The wire should be shielded at one end only-- normally the encoder side. If the PLC is interfaced to a rotary shaft encoder, a loose coupling could also be causing the problem. Check the coupling.

No Reset If a reset is not present when expected, check the configuration. Is the proper value for a reset being used? Is the module looking for an external reset? If so, is the field device supplying the reset signal (which should be connected to either points 02) operating properly. Manually operate the reset device.

Not Jumping to Interrupt Is the interrupt subroutine labeled properly? The interrupt subroutine must be labeled Octal 0. Is there supposed to be an external interrupt, but the external device is not sending the signal? Does the external interrupt device meet the input criteria? Check all of these possibilities.

Not Returning from Interrupt Does the interrupt subroutine end with an Interrupt Return instruction in the RLL program? Is there an endless loop inside the subroutine? Is there a conditional return and the condition has not been met?

Rotary Encoders Rotary encoders can be classified both in terms of how they are coupled to the rotating device for which they are providing information, and how they provide positional information. The rotary encoder can be mechanically coupled to the motor shaft, or it can be optically coupled internally to the encoder.

Mode 30 – Pulse Train Output

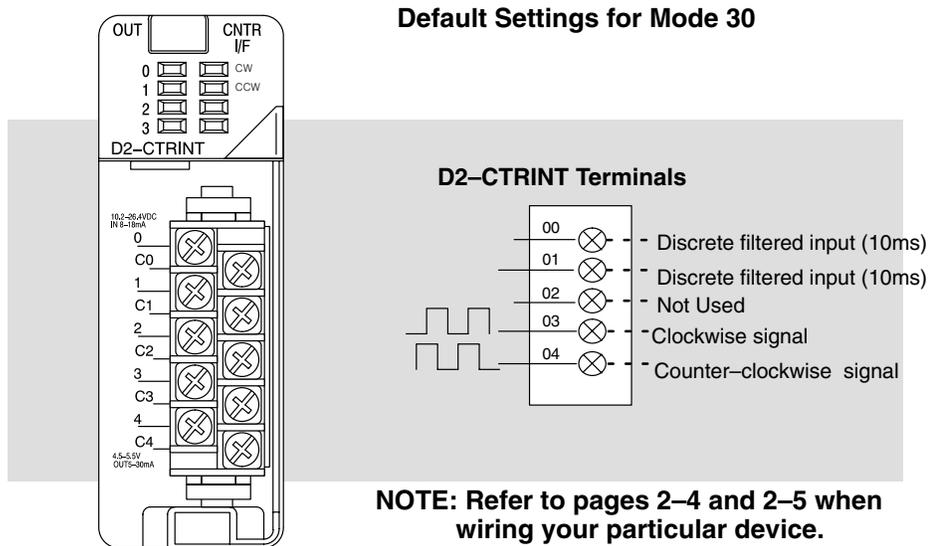
In This Chapter. . . .

- Wiring the Pulse Train Mode
 - System Options and Limitations
 - Configuring the System Parameters
 - Writing the Control Program
 - Verification of Proper Operation
 - Troubleshooting
-

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.



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

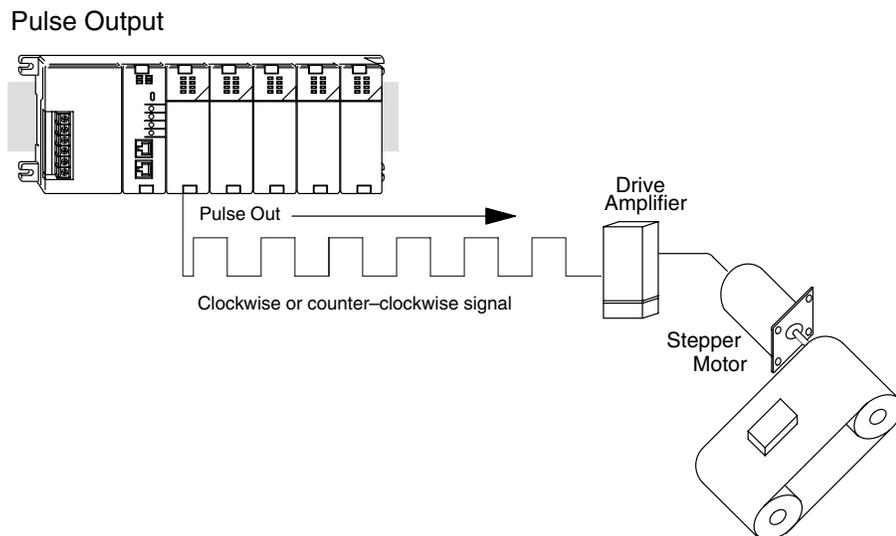
The D2-CTRINT, Counter Interface Module, is a powerful tool for providing solutions for stepper motor applications, however, there is a limitation. **The D2-CTRINT can only be used in open loop stepper motor applications.** The Counter Interface Module can not simultaneous read encoded information while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

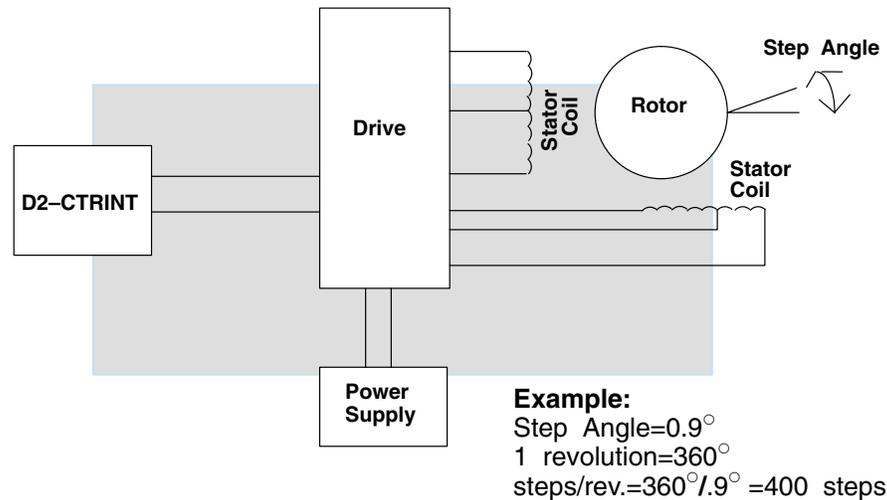
Less than 25% of all stepper motor applications in use today require closed loop control or multiple indexing. The D2-CTRINT is a cost-effective solution for most open loop stepper motor applications.

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.



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 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.



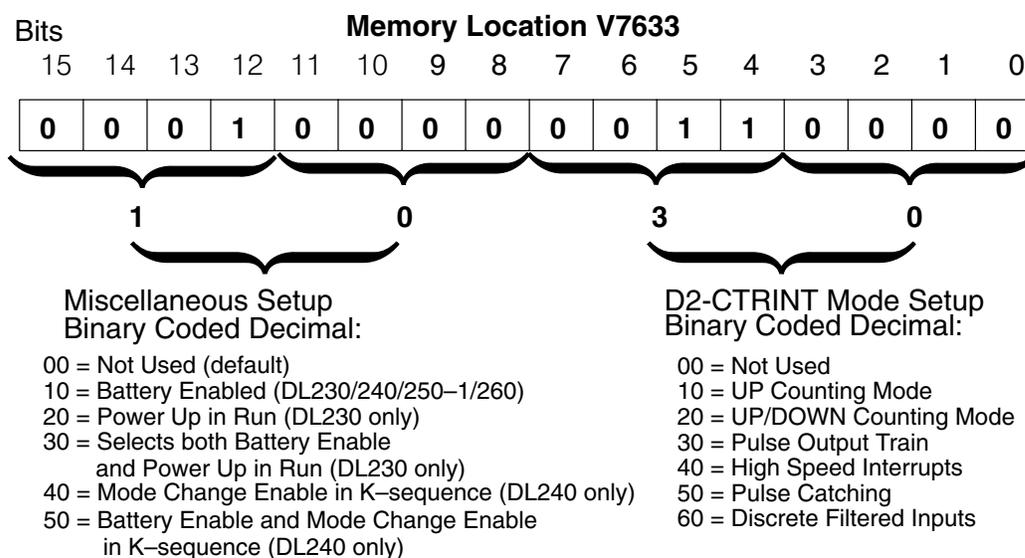
Since the rotation of the stepper motor's shaft is predictable for a given number of pulses, open loop systems are quite reliable. A typical system will use a "home position" which can be determined with a limit switch to orient the load to a start position. After the home position is determined, the 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.



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 **DirectSOFT32**. The following examples will show how to use **DirectSOFT32** to configure the D2-CTRINT module for Pulse Output operation.

Setting the V-Memory using RLL

Setting the V-Memory using the Memory Editor

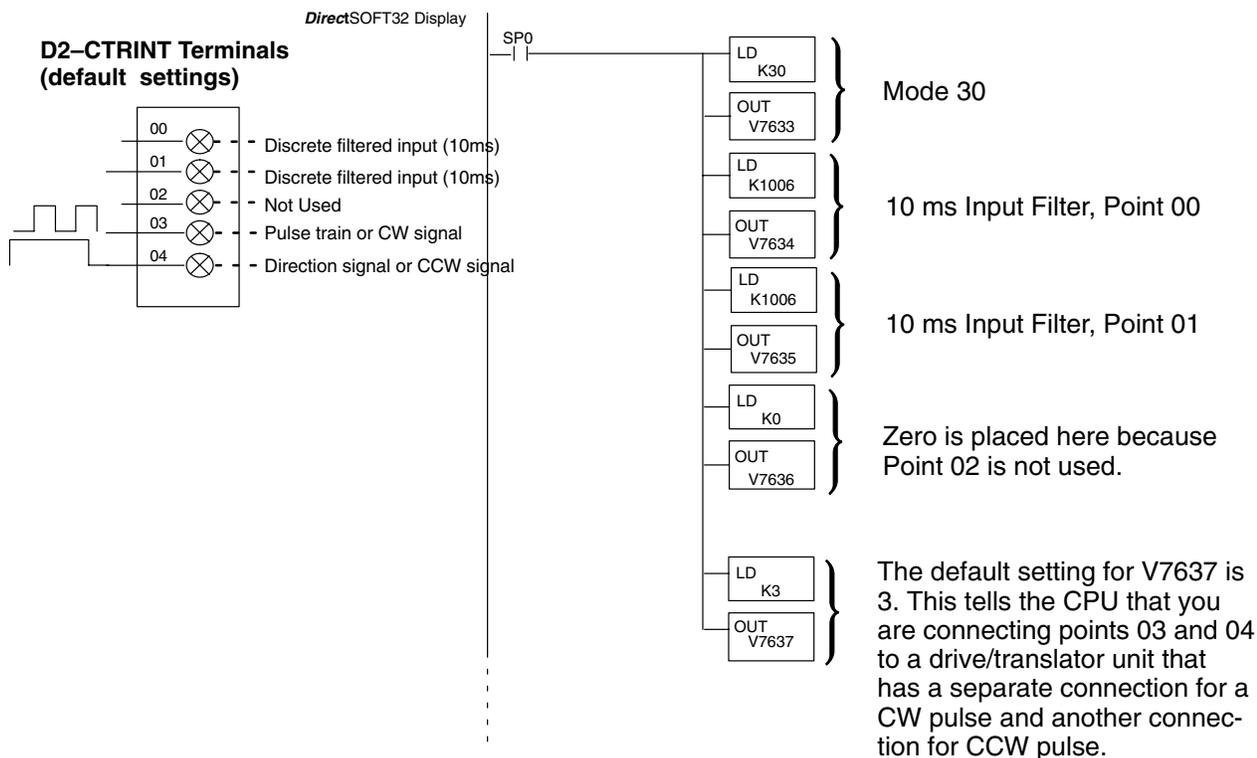
Address	Data Size	Value	Address	Data Size	Value
V7630	Word	3630	V7640	Word	0
V7631	Word	0	V7641	Word	0
V7632	Word	0	V7642	Word	0
V7633	Word	30	V7643	Word	0
V7634	Word	1006	V7644	Word	0
V7635	Word	1006	V7645	Word	0
V7636	Word	0	V7646	Word	0
V7637	Word	3	V7647	Word	0

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.



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.

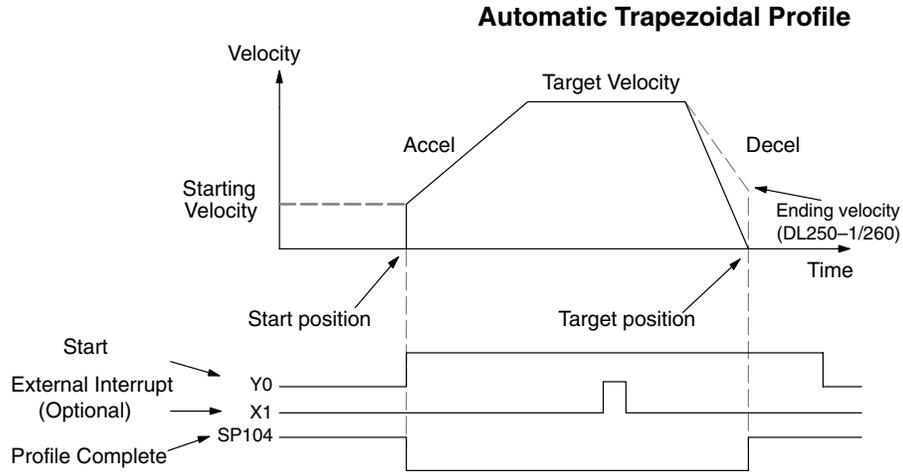
Determine the Positioning Profile

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.

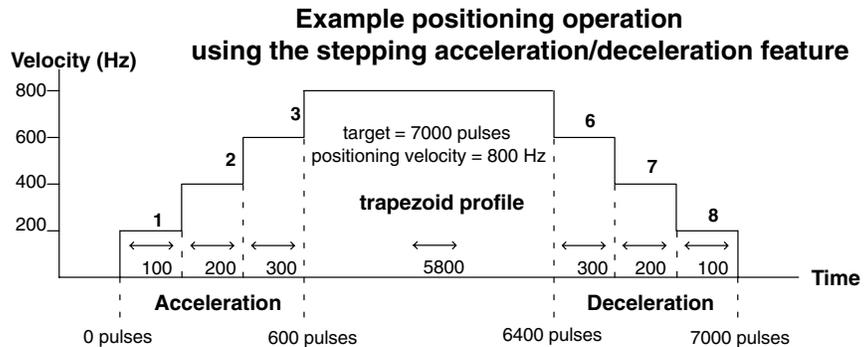
Automatic Accel/Decel Trapezoid Profile



(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



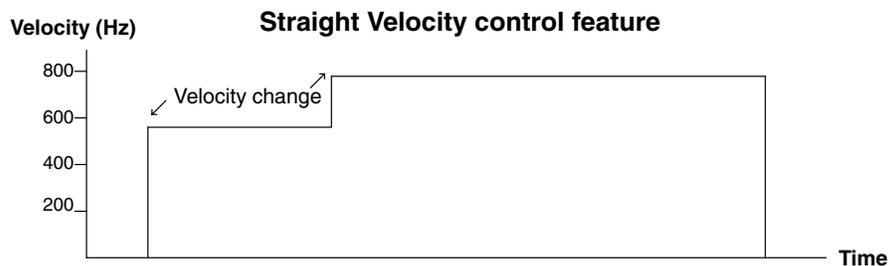
NOTE: Only 3 steps are shown here for both acceleration and deceleration. Up to 4 steps for each and 1 target can be used.



(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



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.

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

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.

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.

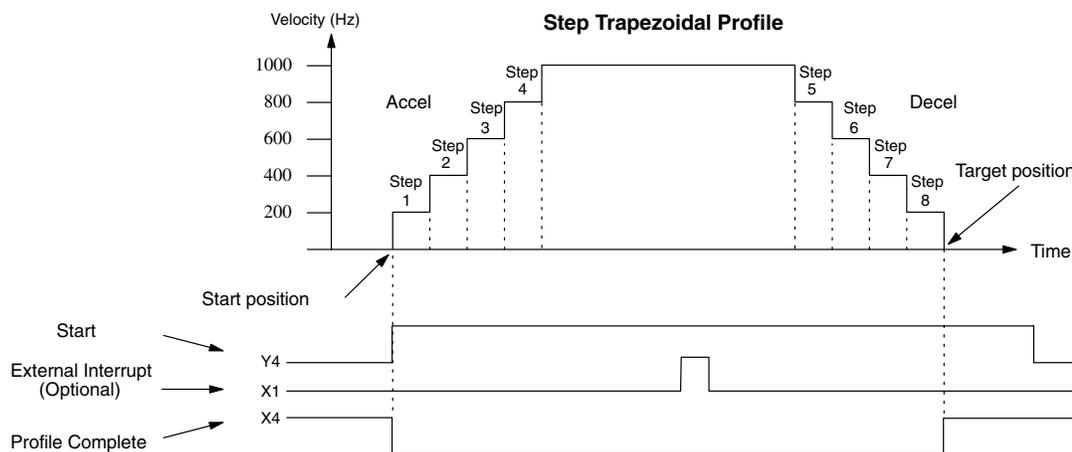
For example, suppose the target is 2000 and Y4 goes HIGH. The actual pulse count goes to 2000 and reaches the target. Then, if Y4 goes HIGH again and the pulse train produces 2000 pulses, the actual pulse count will be 4000 (2000+2000=4000).

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.

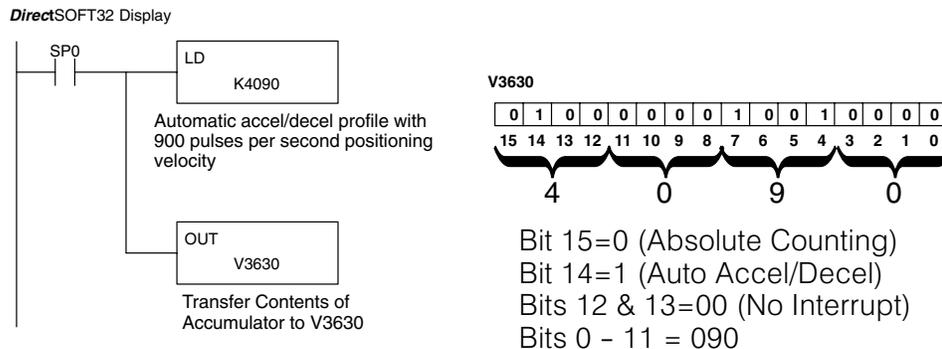
**To set a negative number, put an 8 in the most significant digit. For example: –8388608 would be written as 883388608 in V3631 and V3632.

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.

**Step 1:
Setup V3630**

Implementing **DirectSOFT32** to setup the Automatic acceleration/deceleration profile.

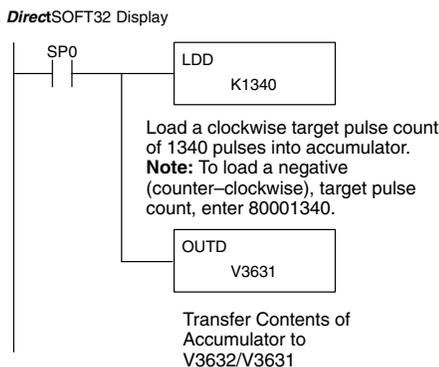
The CPU needs to know that the module is to be used with the Automatic acceleration/deceleration profile. The RLL diagram below shows what needs to be done to setup V3630.



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.

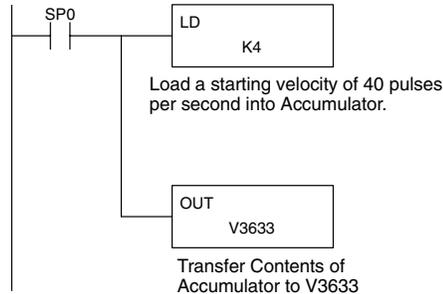


A positive target pulse count was used in the above example. The positive value means the shaft will rotate clockwise. A negative target pulse count could have been used to rotate the shaft counter-clockwise. Whenever a negative target number is used, it will be in the form 8xxxxxxx, where the 8 signifies 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.

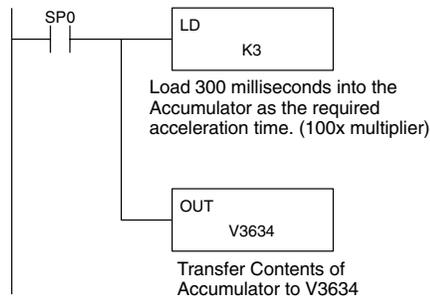
DirectSOFT32 Display



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.

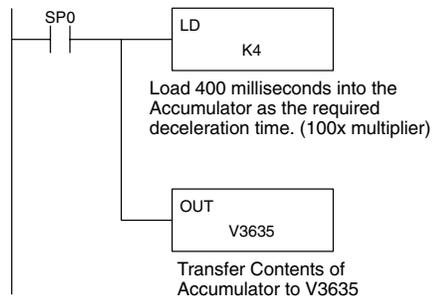
DirectSOFT32 Display



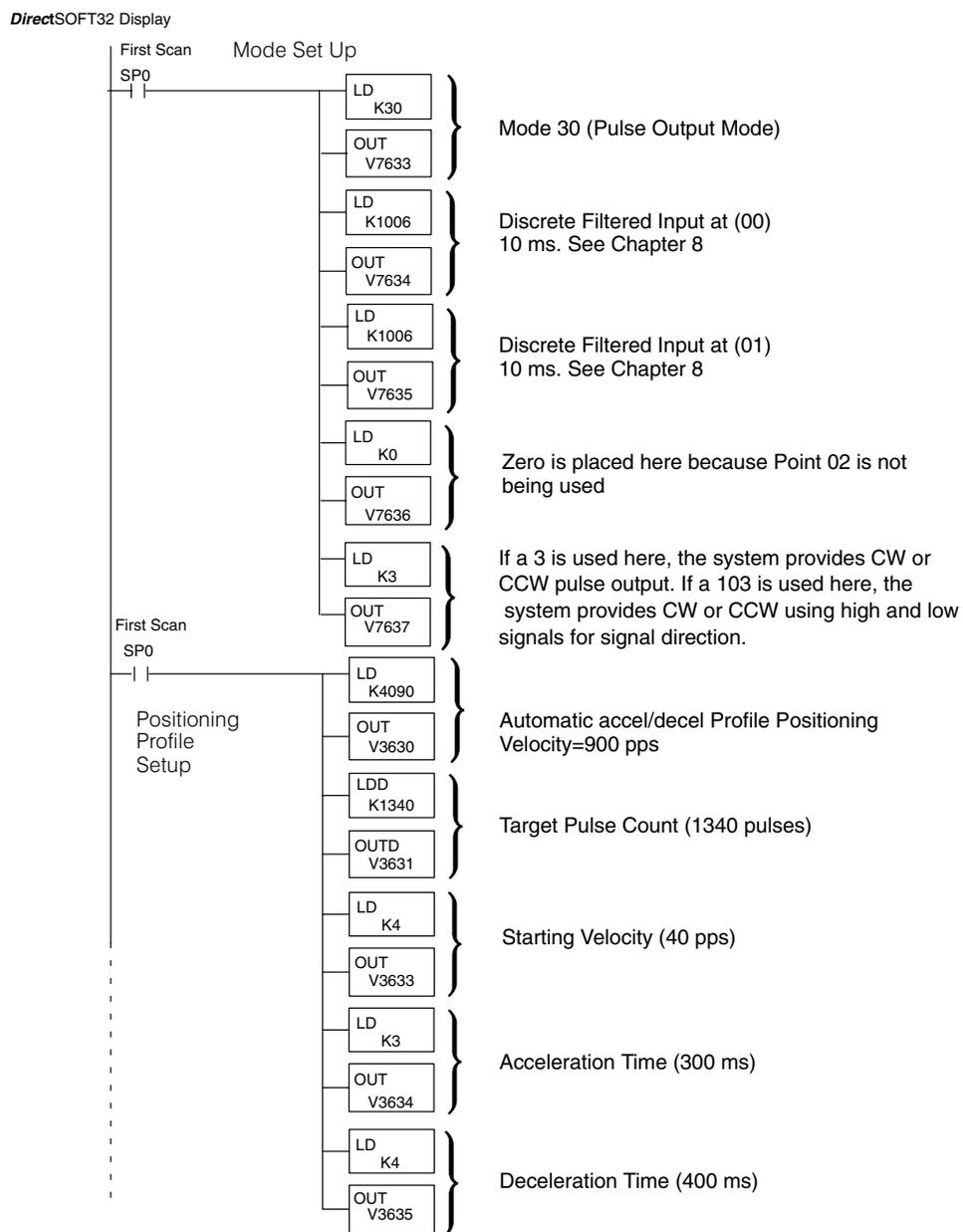
Step 5: Entering the Deceleration Time

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



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



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:

$$\text{Pulses per revolution} = 360 \div 7.5 = 48 \text{ pulses}$$

$$\text{Movement per pulse} = 7 \text{ mm per rev.} \div 48 \text{ pulses per rev.} = 0.146 \text{ mm per pulse}$$

$$\text{Target pulse value} = 1000 \text{ mm} \div 0.146 \text{ mm per pulse} = 6849 \text{ pulses}$$

$$\text{Positioning velocity} = 117 \text{ mm per sec} \div 0.146 \text{ mm per pulse} = 800 \text{ pps}$$

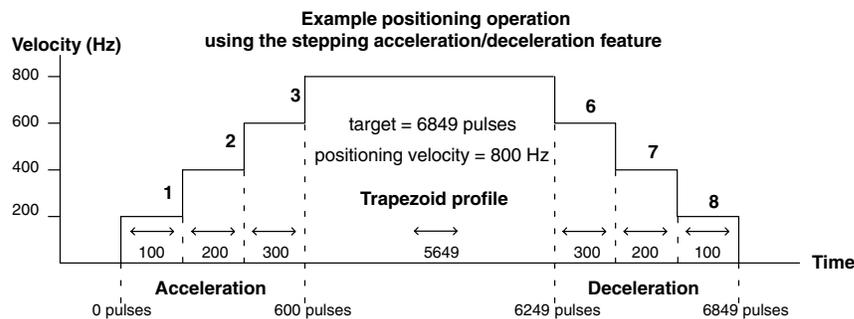
$$\text{Steps used to reach positioning velocity} = 4 \text{ (200 pps increments)}$$

$$\text{Slope of acceleration ramp} = 2:1$$



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 ($800 \div 4 = 200$).

The diagram below shows the characteristics of the example profile.

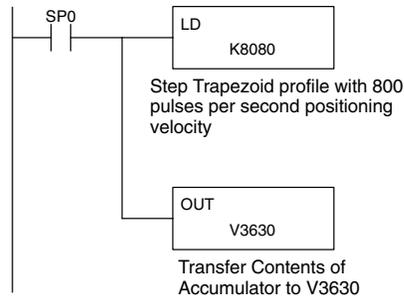


Step 1: Setup V3630

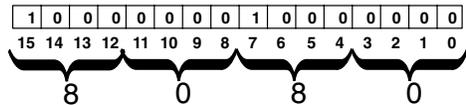
The first memory location that should be configured is V3630. This will setup the Step Trapezoidal profile data for the CPU to operate the module. It will also set the type of numbers stored for each step change (i.e. incremental or absolute), and set the positioning velocity.

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):

DirectSOFT32 Display



V3630



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 (0).

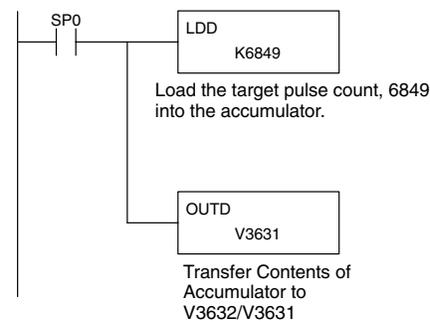


NOTE: There is a 10 pps multiplier for the number placed in the lower 12 bits of this memory location. The code is therefore 080 for 800 pps positioning velocity.

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.

DirectSOFT32 Display



Step 3: Select the Height and Width

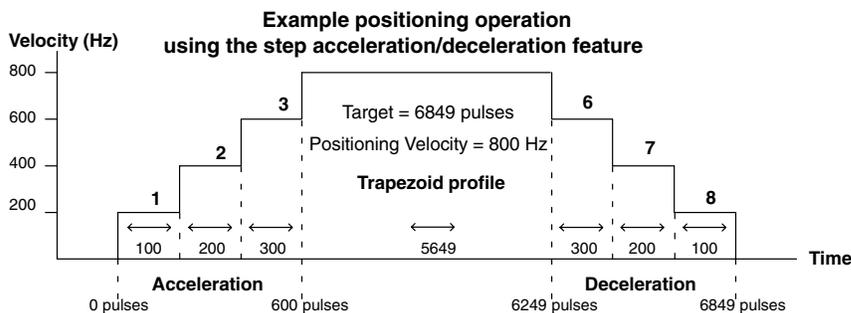
Three steps are used for the acceleration/deceleration and a 2:1 slope for this example. 200 pulses per second increments will be used for the height; this means the width will be half that amount.

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

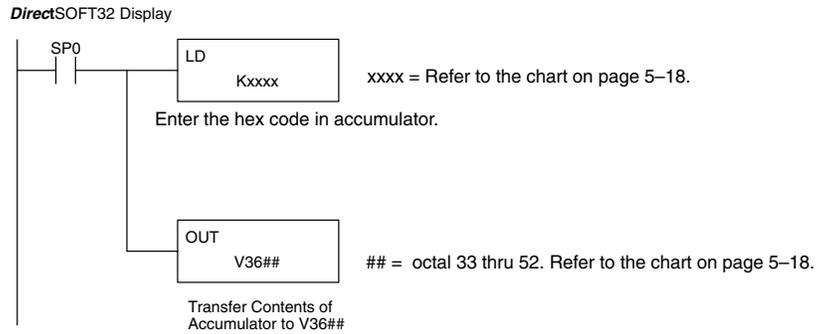
The CPU also calculates the time it takes to move at the target Positioning Velocity:

$(6849 - 1200 = 5649)$ 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.



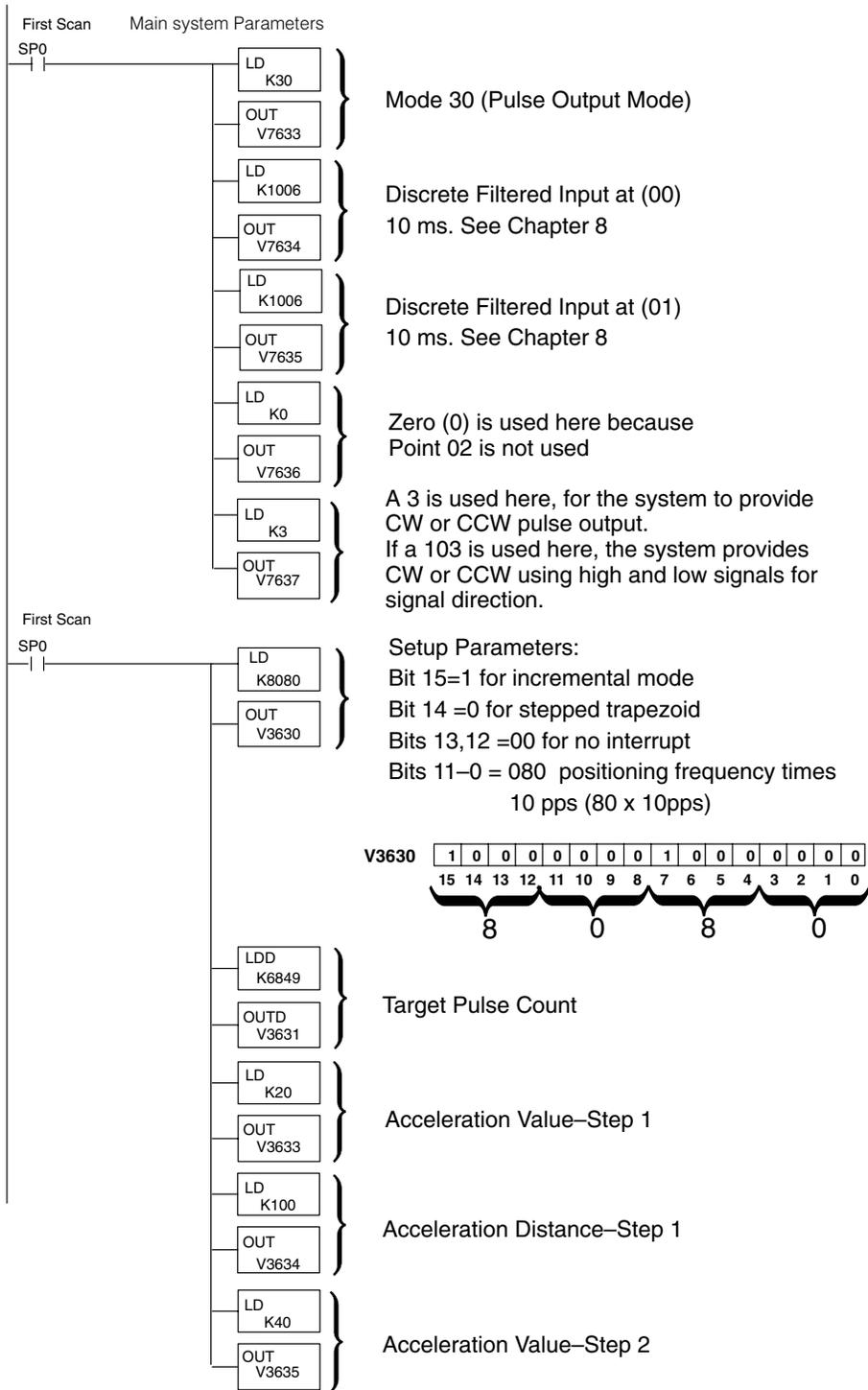
Shortcuts: The data for each step does not have to be entered. For example, if only two steps are used, then enter the data for Step 1 and Step 2 and a zero (0) for each of the V-memory locations from V3643 to V3652. Also, if the acceleration and deceleration are identical, place a zero (0) in each V-memory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

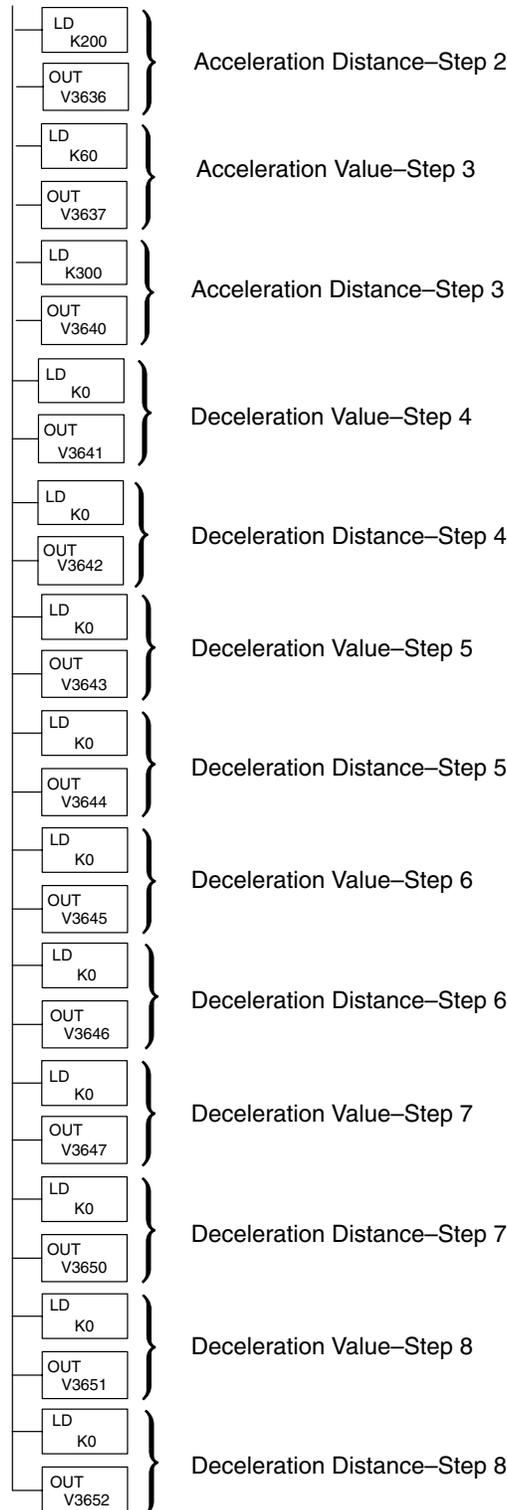
DirectSOFT32 Display



Continued on next page....

Mode 30
Pulse Train Output

Continued from previous page....



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 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.

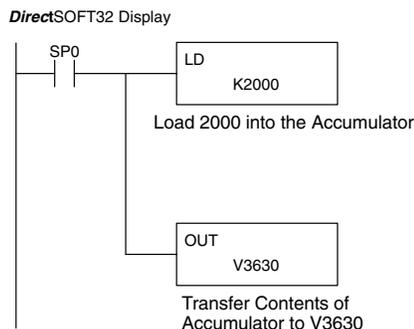
Straight Velocity

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.

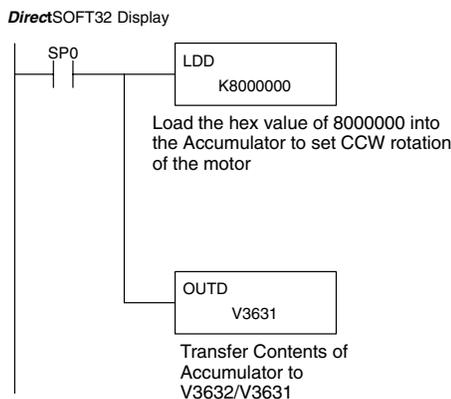
Step 1: Setup V3630

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



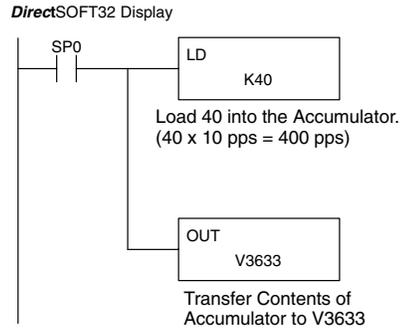
Step 2: Set the Direction for Shaft Rotation

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.



Step 3: Set Shaft Velocity

Set the velocity the motor shaft is to turn.



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.

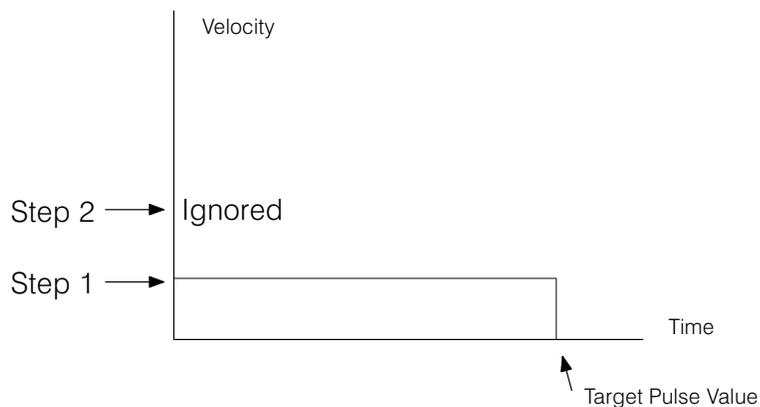
What Happens If the Ramping and Target Are Mismatched?

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:

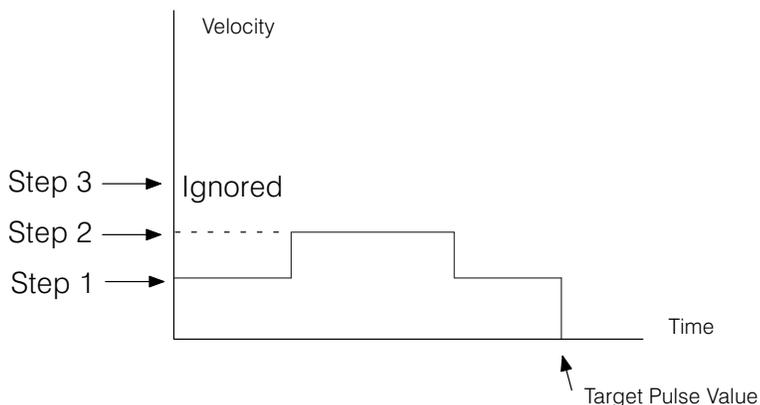
Case 1: The target value is less than the total distance required for acceleration/deceleration Steps 1 and 2.

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



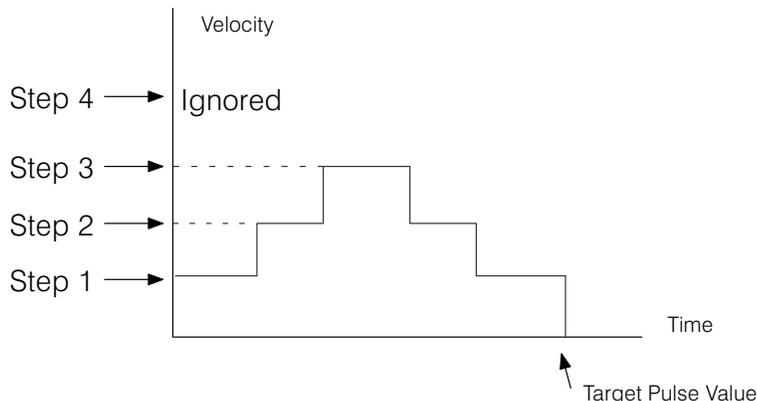
Case 2: The target value is less than the total distance required for acceleration/deceleration Steps 1, 2 and 3.

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.



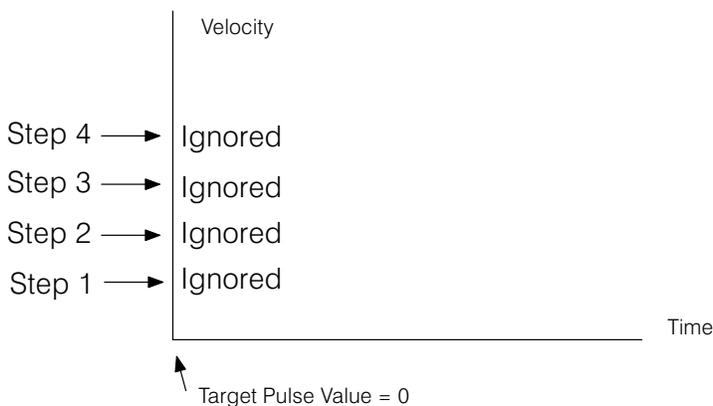
Case 3: The target value is less than the total distance required for acceleration/deceleration Steps 1, 2, 3 and 4.

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.

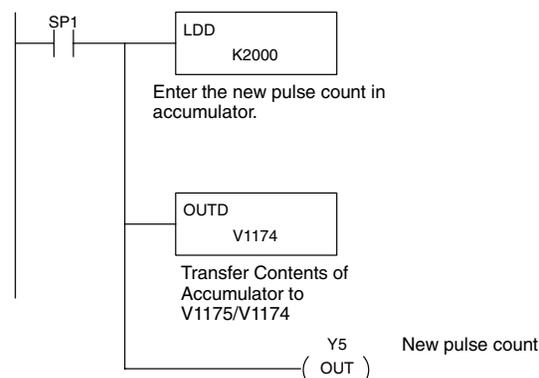
Start Relay – Y4

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



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.

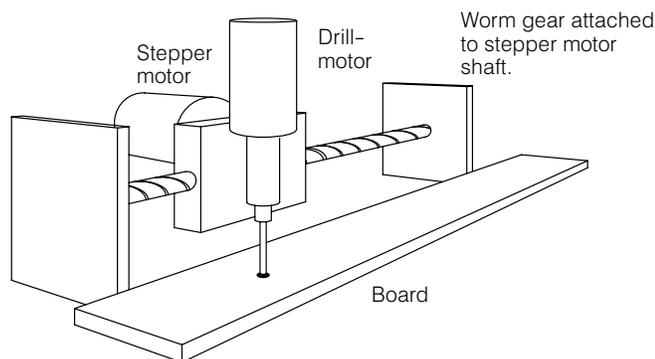
Positioning Complete Relay (X4)

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.

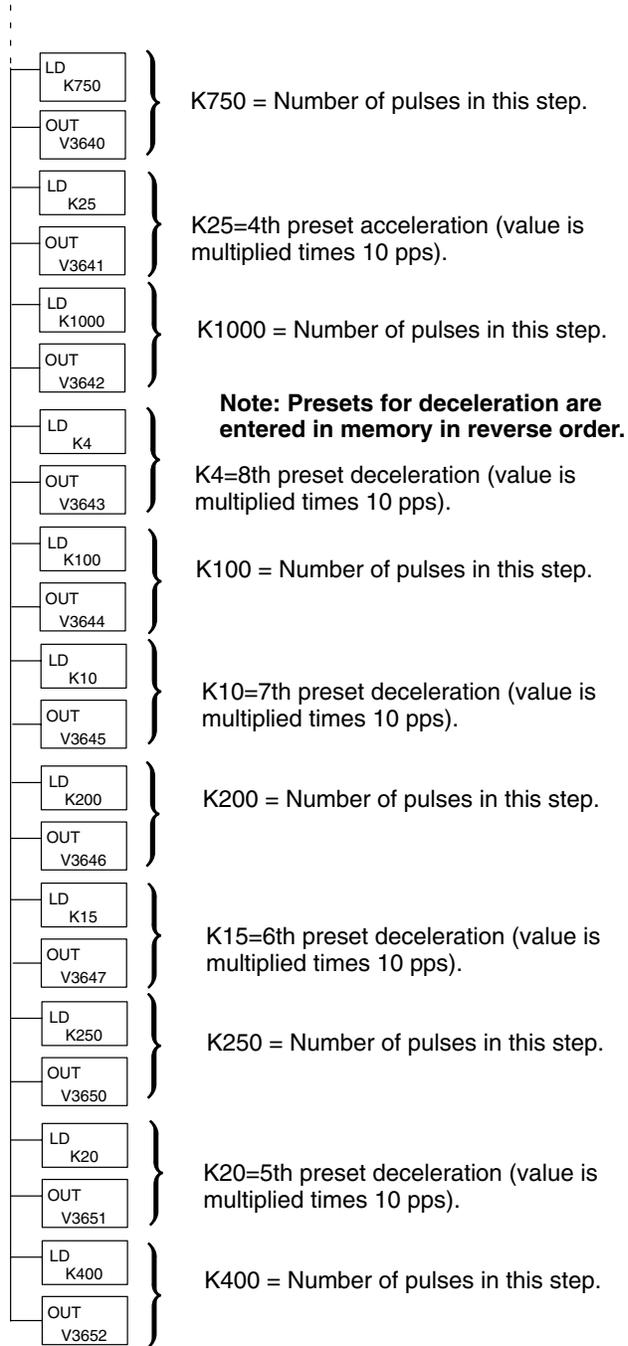
A start button, X10, will be used to initiate the process. Once this pushbutton is pressed, the drill head will accelerate to a traveling speed, then decelerate to the target position. It will do this for each of the three target positions. Anytime it reaches one of these three targets, the table will 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.

Continued from previous page.

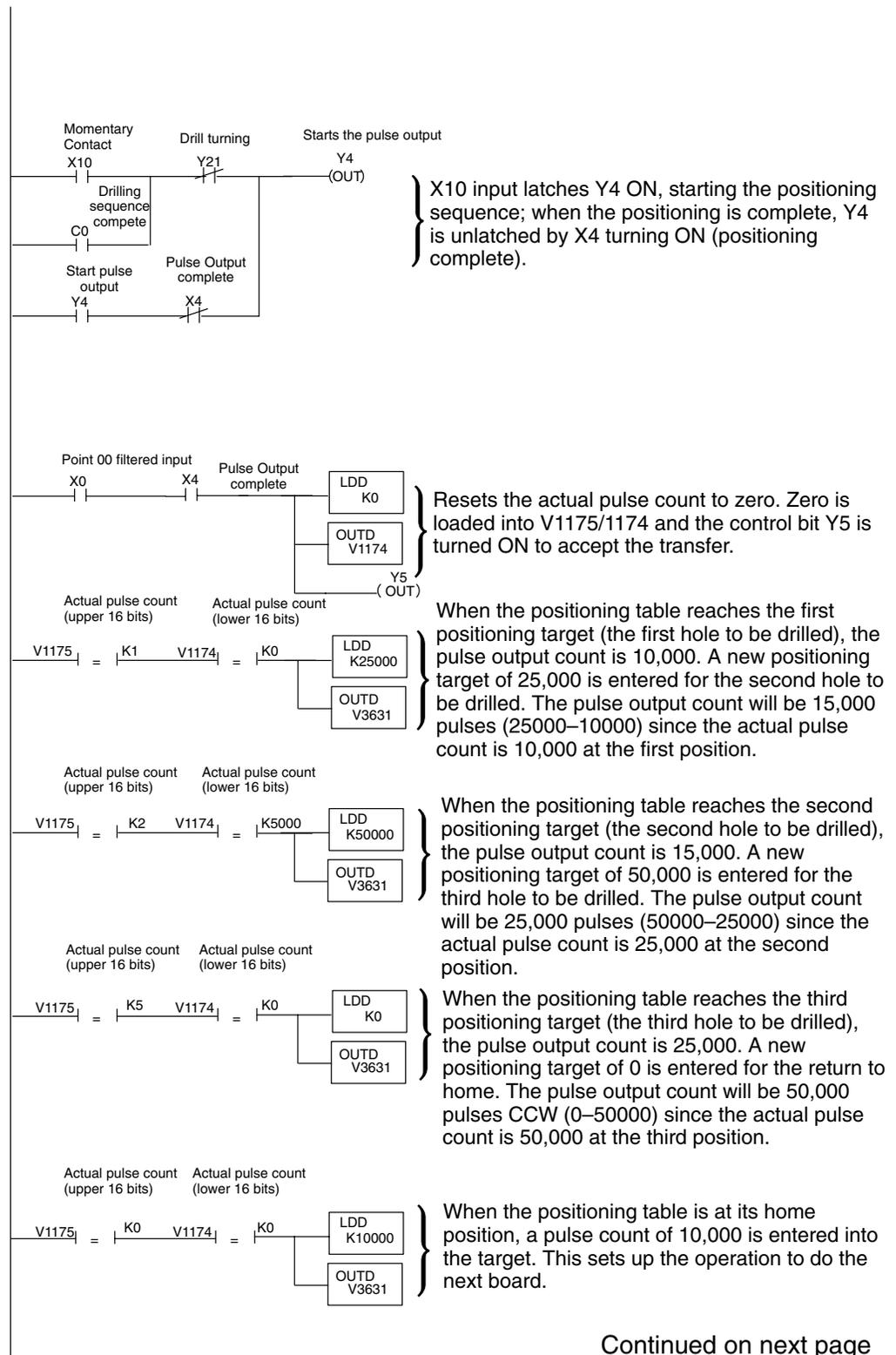
DirectSOFT32 Display



Continued on next page

Continued from previous page.

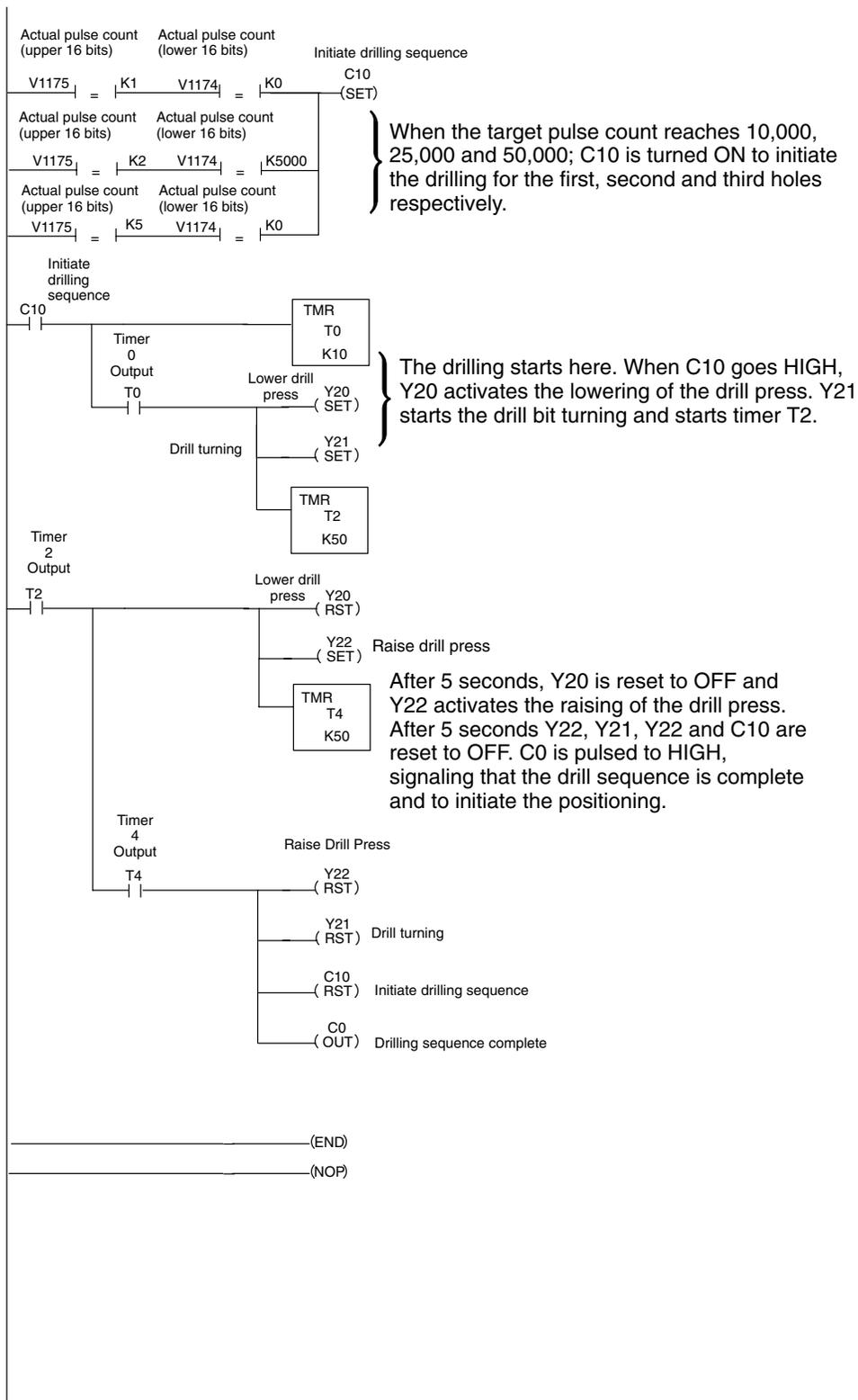
DirectSOFT32 Display



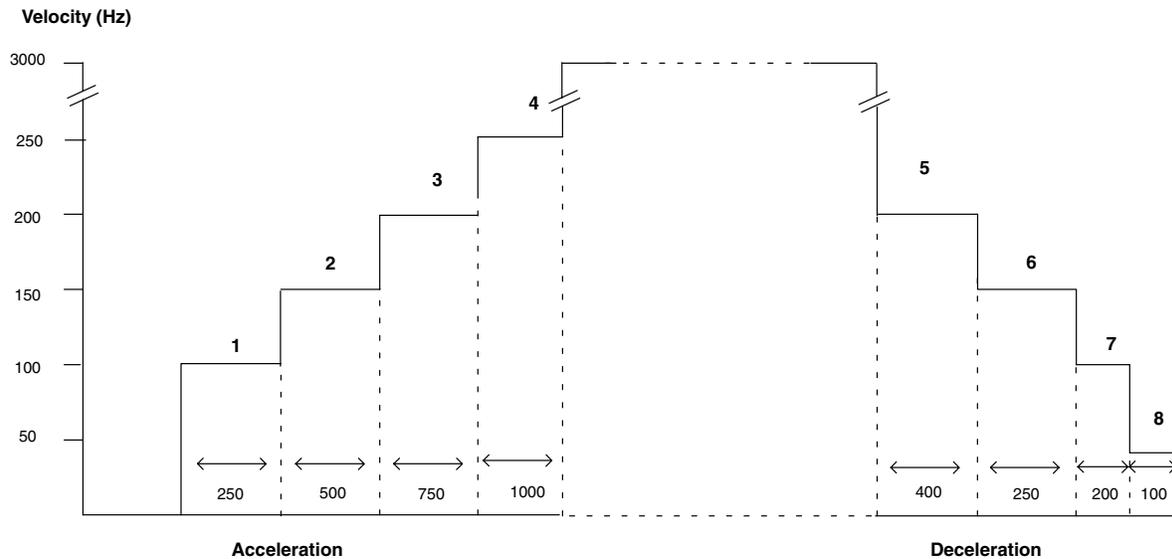
Continued on next page

Continued from previous page.

DirectSOFT32 Display



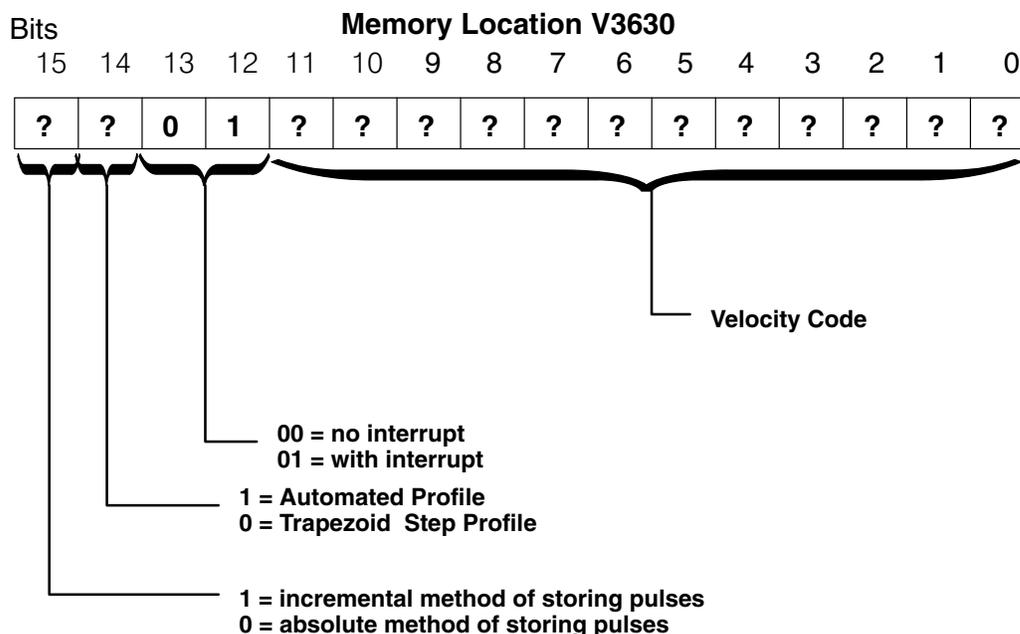
Graphical Diagram of Positioning Profile The diagram for the drilling operation just described is shown below:



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.

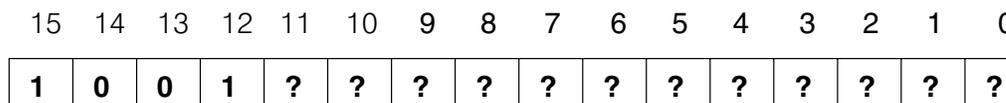
The External Interrupt at X1

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.

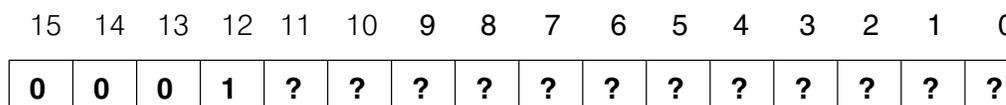


Here are the upper 4-bit hexadecimal values for the possible interrupt configurations used with the positioning profile. The XXX are the digit positions for the velocity:

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



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



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.

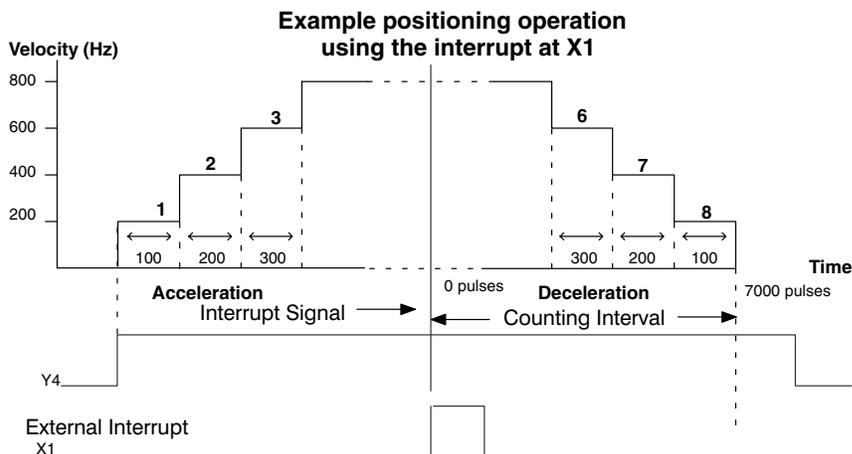
1060 = Step profile with absolute current count mode and interrupt at X1. Positioning velocity is 600 pps.

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

- When Y4 turns ON, the object moves 100 pulses at 200 pps.
- It moves for 200 pulses at 400 pps.
- It moves for 300 pulses at 600 pps.
- It moves at 800 pps.
- 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.
- X4 turns on to indicate that positioning is complete.

Troubleshooting

Is The Module Working Properly?

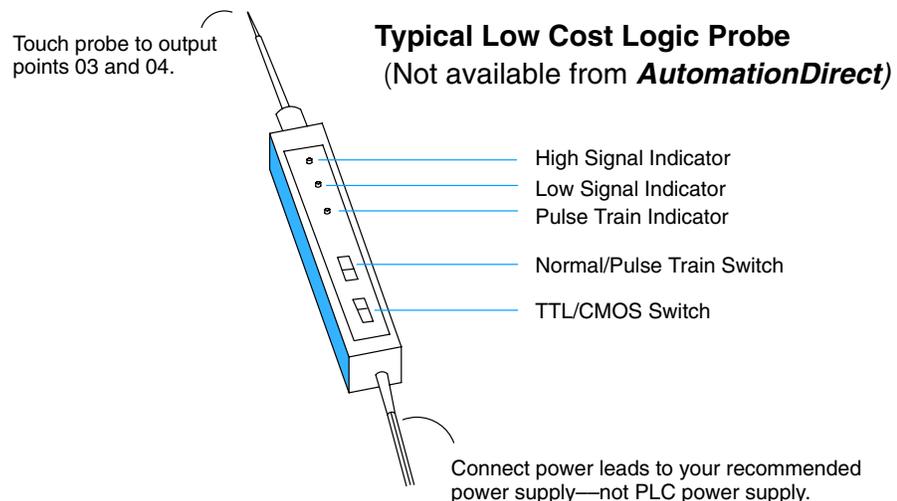
When Mode 30 is used with the D2-CTRINT module, there are a number of areas which can be checked for proper operation. **Experience has shown that most problems occur because of improper configuration. Always re-check the CPU setup before anything else.** The questions below may help to identify any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



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 *DirectSOFT32* to force the input ON. Refer to the *DirectSOFT32 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.

Mode 40 – High Speed Interrupts

In This Chapter. . . .

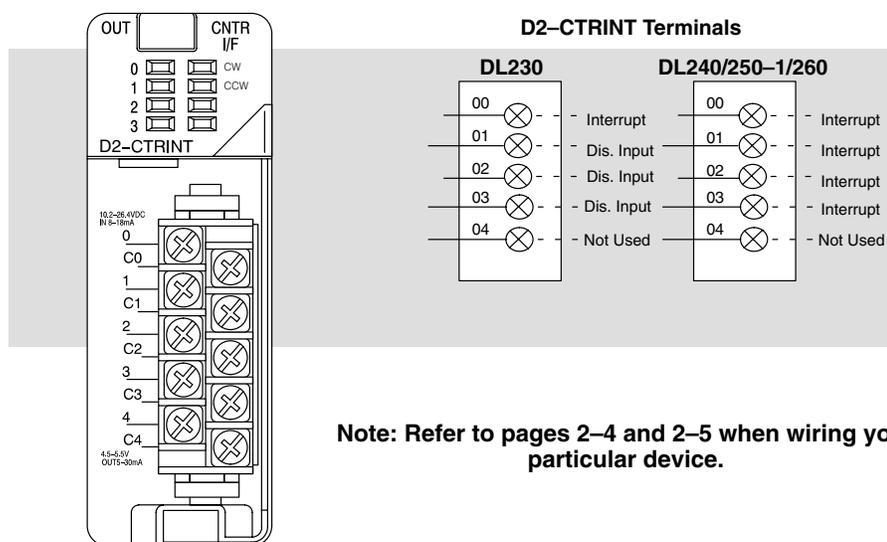
- Wiring the HS Interrupts
 - Configuring the HS Interrupt Parameters
 - Writing the Control Program
 - Verification of Proper Operation
 - Troubleshooting
-

Using the Interrupt Inputs, Mode 40

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 40 —Interrupt Input Mode 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 to make use of the D2-CTRINT's powerful pulse train outputs. A good place to begin is to learn what each channel of the module represents when it is being used in the pulse train mode.

Default Settings for Interrupt Input Mode 40



Note: Refer to pages 2-4 and 2-5 when wiring your particular device.

Shown in the above diagram and illustration are points 00 through 03 which default to high speed interrupt inputs when the module is used with the DL240/250-1/260 and set to operate in Mode 40. When the module is used with the DL230 only point 00 can be used for the interrupt, and the remaining channels can be used as discrete filtered inputs. Chapter 8 contains information about the filter time constant.

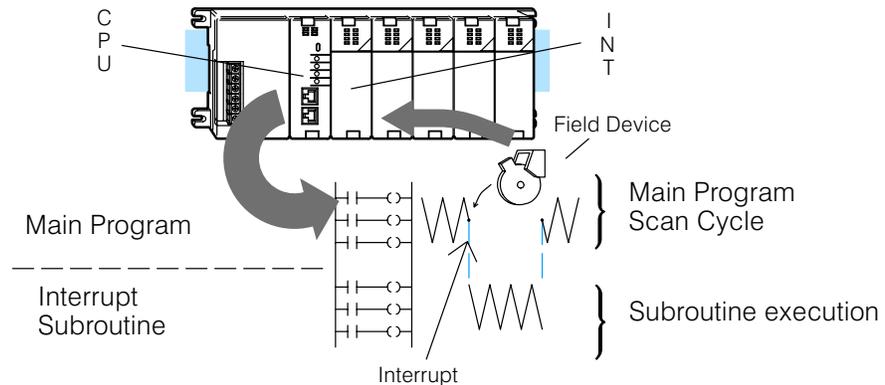
When used with the DL240/250-1/260, the channels which are not configured as interrupt inputs can be configured as “pulse catch” inputs, or as discrete filtered inputs. More about pulse catching in Chapter 7.



Note: All interrupts have a pulse width of 0.1 ms and a pulse period of 0.5ms.

High Speed Interrupts

The high speed interrupts of the D2-CTRINT interface module are intended for applications that have one or more high-priority events which require special operations to be acted upon immediately. When an interrupt occurs, the module tells the CPU to go to a subroutine. The CPU finishes executing the instruction at that point in the program scan, immediately suspends the program scan and jumps to a subroutine which is identified by the interrupt input. Once the subroutine has been executed, the normal program scan will continue at the point where the interrupt occurred. (See diagram below.)



Subroutines can include math instructions, data instructions, or any necessary routine which may be needed. One popular practice is to include *immediate* I/O instructions in the subroutine. These instructions immediately update the I/O points individually instead of waiting for a normal I/O update.

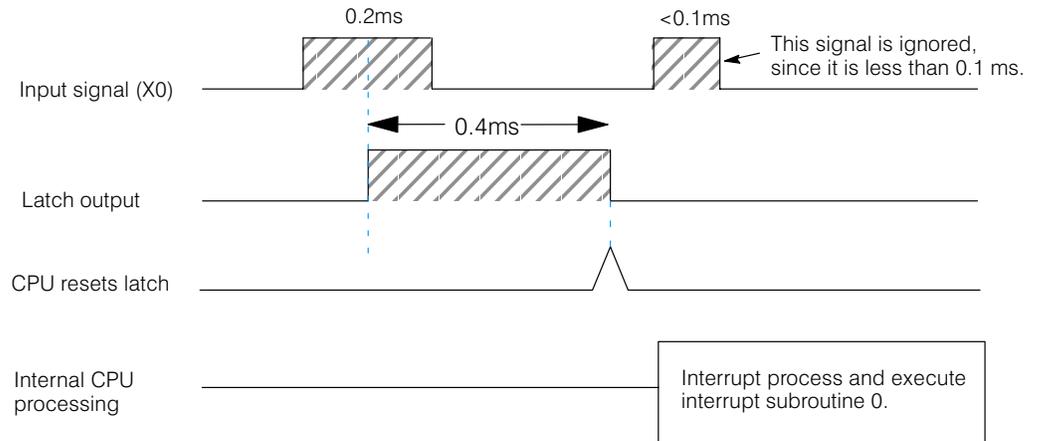
The order in which each input is to be acted upon is prioritized and determined by which input point the user chooses to use as the interrupt. The inputs are given labels X0 through X3 for the DL240/250-1/260, and X0 for the DL230. The highest priority is input X0, with the next highest being X1, then X2, then the lowest priority X3.

Each interrupt input signal is latched when it occurs. This means that as the module senses a DC signal appearing at the input point for at least 0.4 milliseconds, it is held ON until acted upon by the CPU even if the input changes.

High Speed Interrupts Process

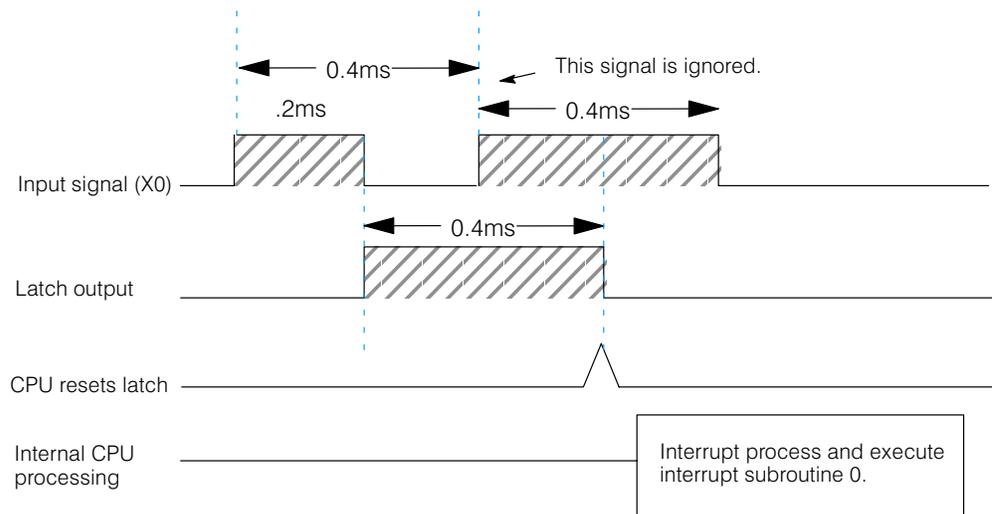
Single Input Signal Processing

When the interrupt input point is turned on for a period of 0.1 milliseconds or longer, the module will latch the signal and send the interrupt request to the CPU. Once the CPU receives the request, the CPU stops the normal scan to execute the interrupt routine and send a reset signal to the D2-CTRINT module.



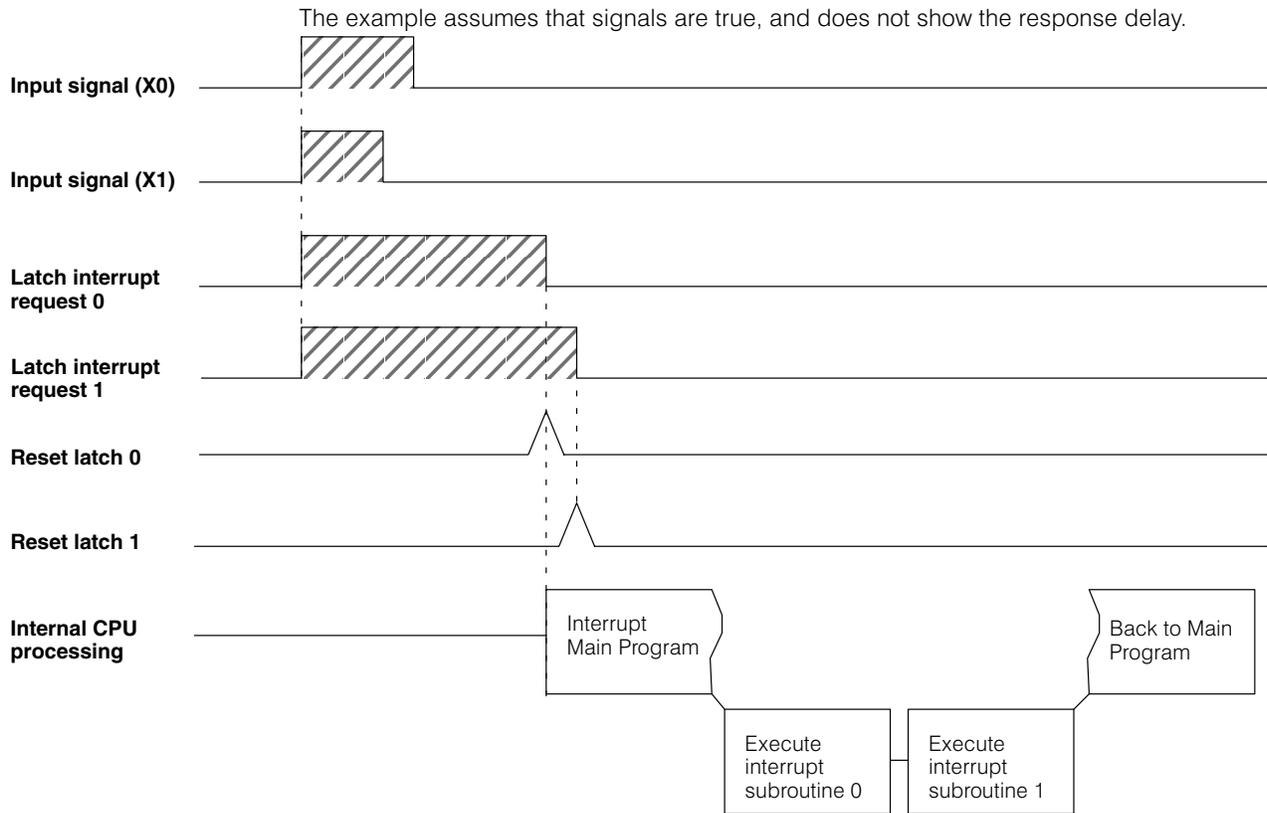
Multiple Signals from Same Input

There may be occasions where several signals are received close together, in a short period of time, from the same input. In the diagram below, the time line reads from left to right. The first signal is latched and takes 0.4 milliseconds to complete. If another signal is received within that time span, the module will only recognize the first signal and ignore the second one as shown below.



Simultaneous Signals from Different Inputs

When the module senses signals simultaneously from several of the X0–X3 input points, the priority system begins to work. The highest priority is given to X0, then X1, X2 and X3. The diagram below shows two signals being received simultaneously.



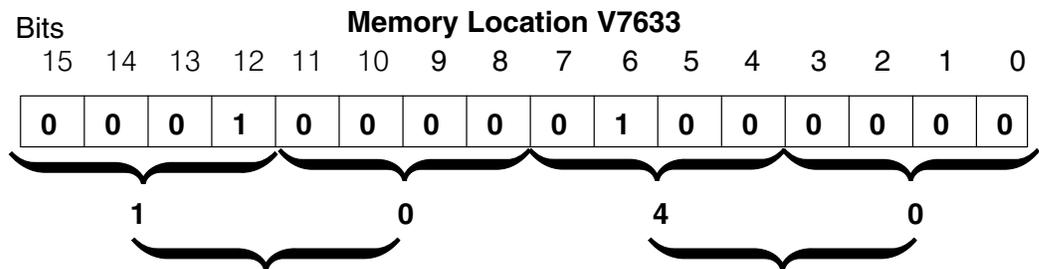
Leading Edge Triggering

The interrupt signals processed by the D2–CTRINT module are leading edge triggered. The response time for the interrupt request acknowledgement is 0.1 milliseconds.

Understanding V-Memory Setup Locations

The High Speed Interrupt, Mode 40, associated with the D2-CTRINT requires V-memory configuration in order to be used.

V-memory location V7633 is the most important of all the reserved memory areas because it stores the 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 is for High Speed Interrupt, Mode 40. The lower bits are set to 40 and the upper bits are set to 10 so the backup battery is enabled. Together they form the hexadecimal number 1040.



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 in 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 in the 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 setup information for each I/O point. The CPU will automatically configure them with default values for the selected mode.

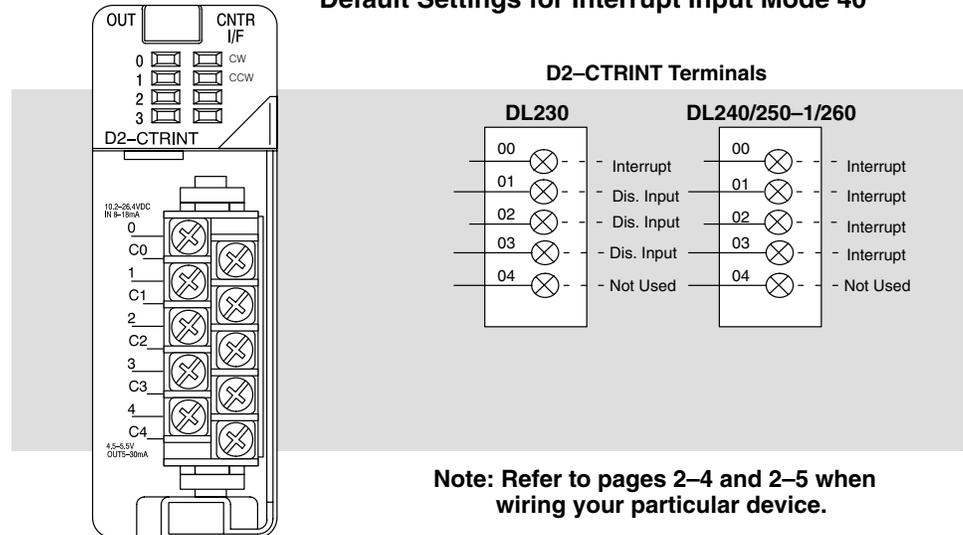
Default Settings When xx40 is written to V7633, the CPU places the following default values in V-memory.

Configuration	Point 00/V7634	Point 01/V7635	Point 02/V7636	Point 03/V7637	Point 04
DL230 Hexadecimal Value	Interrupt 0004	Discrete Input 1006	Discrete Input 1006	Discrete Input 1006	Not Used
DL240/250-1/260 Hexadecimal Value	Interrupt 0004	Interrupt 0004	Interrupt 0004	Interrupt 0004	

Explanation the Values

Chapter 8 explains the discrete filtered inputs in detail. From the above table, 1006 for Point 01, DL230, is in the form xx06 where the 06 indicates the function (discrete input) and the xx indicates the filter time in milliseconds (in this case 10 ms). The value of 0004 is for high speed interrupts.

Default Settings for Interrupt Input Mode 40



Custom Configuration

Up to this point, only Mode 40 default settings have been discussed. The default settings will be suitable for many applications and will not require a custom 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.

Mode 40 Options

Point Number	V-Memory Location	Possibility (One per point)	Hex Value
point 00	V7634	High Speed Interrupt (external)	0004 (default)
		High Speed Interrupt (timed)	ttt4 (ttt=1 to 999ms timer setting)
		Discrete Filtered Input	xx06 (xx=filter time)
		Pulse Catch	0005
point 01	V7635	High Speed Interrupt (DL240/250-1/260)	0004 (default for DL240/250-1/260)
		Discrete Filtered Input	xx06 (xx=filter time) (default for DL230)
		Pulse Catcher (DL240/250-1/260)	0005 (DL240/250-1/260)
point 02	V7636	High Speed Interrupt (DL240/250-1/260)	0004 (default for DL240/250-1/260)
		Discrete Filtered Input	xx06 (xx=filter time) (default DL230)
		Pulse Catcher (DL240/250-1/260)	0005 (DL240/250-1/260)
point 03	V7637	High Speed Interrupt (DL240/250-1/260)	0004 (default for DL240/250-1/260)
		Discrete Filtered Input	xx06 (xx=filter time) (default DL230)
		Pulse Catcher (DL240/250-1/260)	0005 (DL240/250-1/260)
point 04	-----	Not available in Mode 40	-----

When using Mode 40 with a high speed interrupt, it is better to select one or more of the available I/O points. If only one of the points is being used as an interrupt with a DL240/250-1/260 CPU, it may be more advantageous to select a different operating mode (i.e. 10, 20, 30, 50, or 60). If this is done, configure one of the unused terminals as an interrupt. For example, configure a CPU for an UP Counter Mode 10, and use point 03 for a high speed interrupt. Read the individual chapters for an alternate mode to choose; these chapters will be helpful in selecting a high speed interrupt as a secondary feature.

Setting Up the CPU for the Interrupts

Configuring the V-Memory

The DL240 or DL250-1/260 CPUs checks the V-memory to see if there is a D2-CTRINT Module present. The number 40 will reside in V7633 if the module has been configured properly. If the CPU finds that a Counter Interface Module is present, other V-memory locations will be checked to see how each channel 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 **DirectSOFT32**. The following examples will show how to use **DirectSOFT32** to configure the High Speed Interrupts.

Step 1: Enter the Mode

If Mode 40, Interrupt Input, has been chosen as the primary function, 40 must be placed in V7633. The following **DirectSOFT32** diagram shows the setup procedures for communicating with the DL240/250-1/260 CPU. Refer to the **DirectSOFT32 Programmers User Manual** for more details.

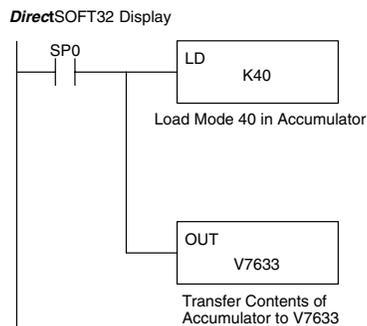
Setting the V-Memory using RLL

Setting the V-Memory using the Memory Editor

Address	Value	Address	Value
V7633	40	V7643	0
V7634	4	V7644	0
V7635	4	V7645	0
V7636	4	V7646	0
V7637	4	V7647	0
V7640	0	V7650	0
V7641	0	V7651	0
V7642	0	V7652	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.

The following RLL example shows how to set V-memory location V-7633 to Mode 40, Interrupt Input. This is the location for all mode values for this module. Only one mode (6 possible) can be entered, i.e. 10,20,30,40,50 or 60.



Two 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, 40 is to be placed in V7633. This value is loaded into the accumulator, LD K40. The CPU then writes this data to the memory location, V7633, once it reads the OUT instruction, 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.

**Step 2:
How Many
Interrupts**

Up to four (4) high speed interrupt inputs can be used with the DL240/250-1/260 CPUs and one (1) high speed interrupt for the DL230 CPU. The following steps will discuss the programming for each channel which has an interrupt device wired to it.

**Step 3:
Configure the
V-Memory**

The table below gives a description for each of the V-memory locations that must be configured for each I/O point which are selected to have high speed interrupt capability.

V-Memory	Description
V7633	Primary Mode (Interrupt=40)
V7634	Point 00 and Timer Value if Used as Timed Interrupt
V7635	Point 01
V7636	Point 02
V7637	Point 03

Two Types of Interrupts

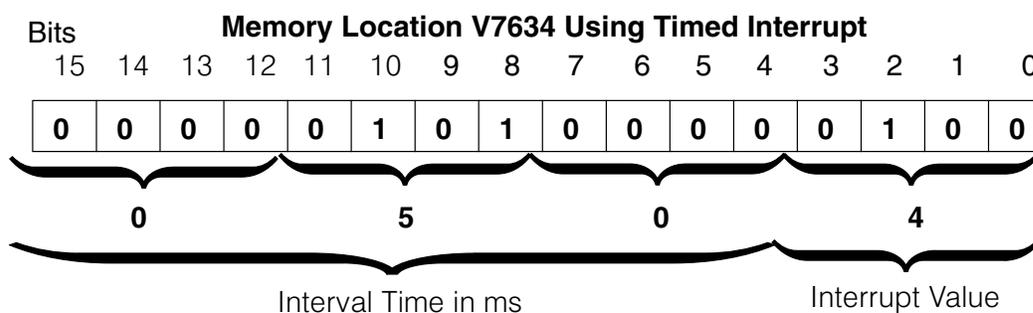
There are two types of interrupts to choose: external interrupt and timed interrupt.

A four digit hex value is written to each of the memory locations shown in the table for each I/O point which will receive interrupt signals. Every input point wired to receive interrupts should have the value of 0004 for the lowest digit.

All four channels can be programmed to respond to external interrupt signals. Channel 1 is the only one which can only be programmed as a timed interrupt. If the timed interrupt is used, the time interval value is written in the first three digits of V7634.

The interval time range is between 3 and 999 milliseconds.

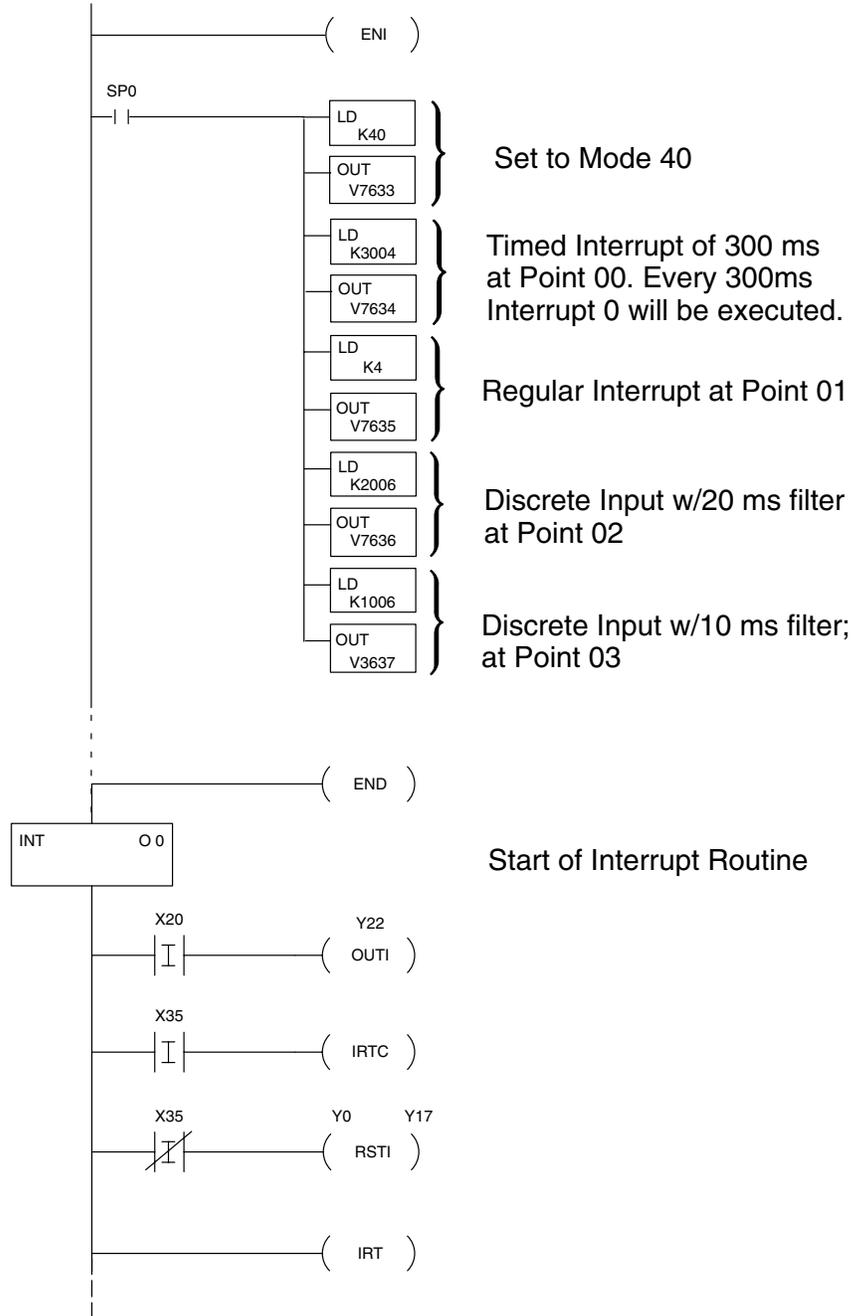
For example the following means that Channel 1 is to be a 50 ms interrupt timer:



Interrupt 0 will be executed every 50ms.

In the RLL example below, Channels 1 and 2 are configured to be high speed interrupt points, and Channels 3 and 4 to be discrete filtered inputs. Channel 1 will be a timed interrupt.

DirectSOFT32 Display



Troubleshooting

The following information may provide some assistance in handling any problems which may be encountered when setting up the D2-CTRINT module, should they occur. ***Experience has shown that most problems occur because of improper configuration. Always re-check configuration before anything else.***

For verifying types of inputs (or outputs) which do not relate to external High Speed Interrupts, see the Chapters in this manual covering the specific function. Listed below are some things that could possibly go wrong with the high speed interrupt inputs:

1. An external field device wired to the interrupt channel seems to be working, but there is no interrupt when the device is operated.
2. The status indicator LED is not lighting for the input point where the interrupt is wired.
3. The CPU is executing the interrupt properly, but it does not continue its normal scan after the interrupt subroutine has been completed.
4. The interrupt is not fast enough.

Interrupt Device Working But No Interrupt

Status Indicators -Is the LED for the channel in question lit when the field device attached to it is closed? If it isn't, then the logic side of the module is either not being energized from that device or the LED is bad. A voltmeter can tell you if there is a signal at the I/O point. If there is, and it does not light the corresponding LED on the top of the module, then the LED must be bad.

Pulse Width – If the signal does not maintain a HIGH level for more than 0.1 ms, then the CPU may never acknowledge that it is there. Use a field device which produces a pulse longer than 0.1ms.

Wiring - Simple as this might seem, quite often poor wiring is the cause of many problems. Make sure there is a complete electrical loop between the device and the input module. Along with visual inspection, use a voltmeter to check the wiring.

Input Voltage - If the input device is sending a signal which is less than 12 volts, most likely the counter will not function or it will function improperly. Replace the field device with one which has a stronger output if necessary.

Improper Configuration - Make sure that the proper configuration has been setup for the mode being used. Also make sure that the function block has been properly labeled for the interrupt subroutine. The function block and the I/O point must have the same octal number assigned.

Interrupt Occurs, But No Normal Scan Afterwards

When this happens there is usually an error in the subroutine itself. Look for endless loops or conditional returns where the condition has not occurred. Make sure that a return has been placed at the end of the subroutine. **DirectSOFT32** will usually indicate this type of error.

Mode 50 – Pulse Catch Inputs

In This Chapter. . . .

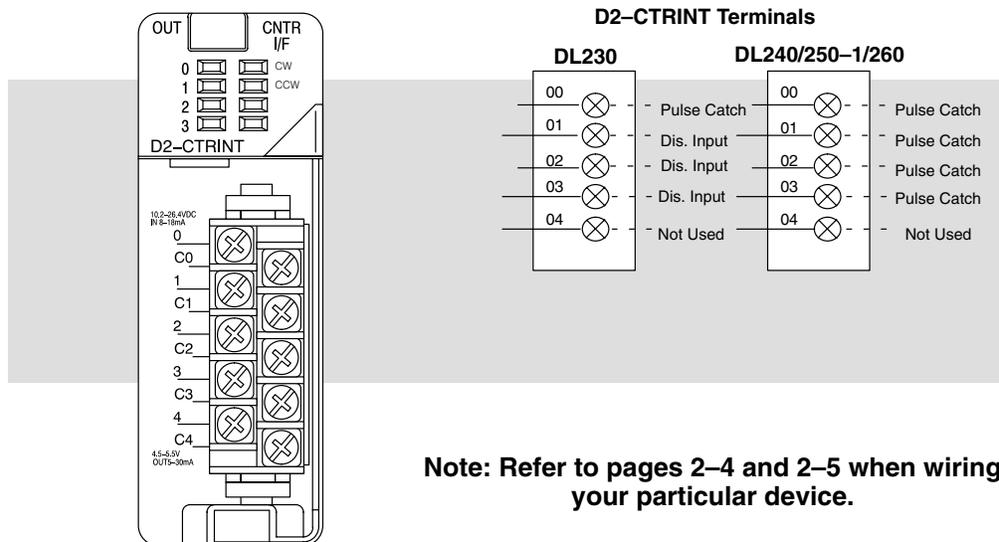
- Wiring the Pulse Catch Inputs
 - Configuring the Pulse Catch Parameters
 - Verification of Proper Operation
 - Troubleshooting
-

Using the Pulse Catch Inputs, Mode 50

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*. Pulse Catch Input, Mode 50 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 to make use of the D2-CTRINT's Pulse Catch Inputs. A good place to begin is to learn what each channel of the module represents when it is being used in the Pulse Catch Input mode.

Default Settings for Pulse Catch Input Mode 50



Note: Refer to pages 2-4 and 2-5 when wiring your particular device.

Shown in the above diagram and illustration are points 00 through 03 which default to pulse catch inputs when the module is used with the DL240/250-1/260 and set to operate in Mode 50. When the module is used with the DL230 only point 00 can be used for the pulse catch input, and the remaining channels can be used as discrete filtered inputs. Chapter 8 contains information about the filter time constant.

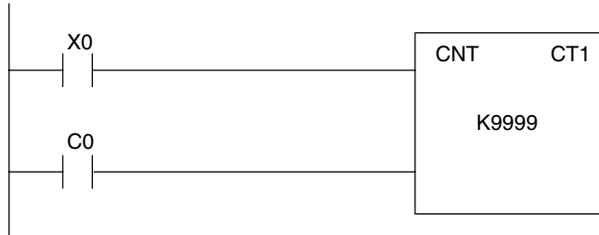
When used with the DL240/250-1/260, the channels which are not configured as pulse catch inputs can be configured for many different functions, or as discrete filtered inputs.

Pulse Catching Explained

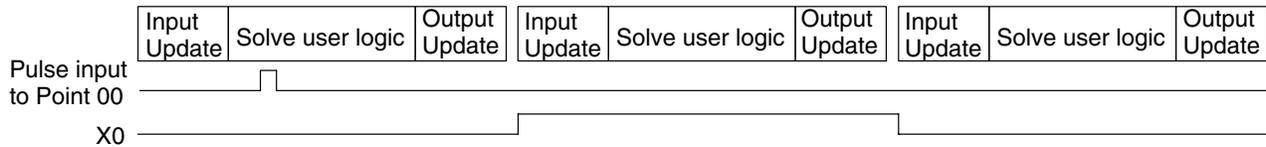
The following example will help to explain the Pulse Catch Mode.

Counting pulses in Mode 50 (Pulse Catch Input)

Counting input pulses in the user program has some problems to be considered.

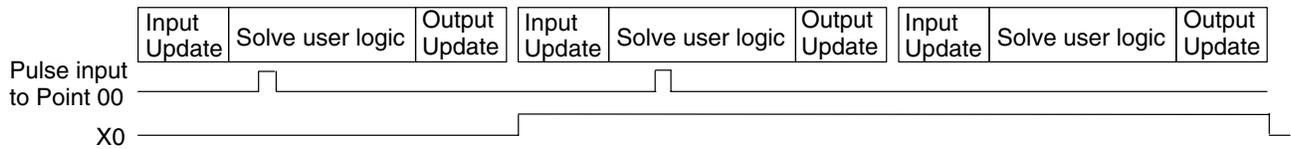


[Case 1] This is okay.



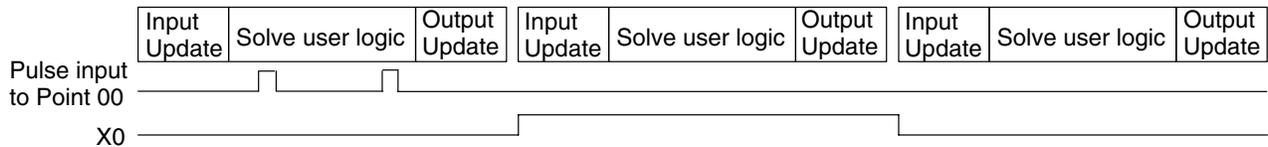
In this case, the pulse can be counted correctly.

[Case 2] This is not okay.



Both of these pulses are counted as one pulse.

[Case 3] This also is not okay.



Both of these pulses are also counted as one pulse.

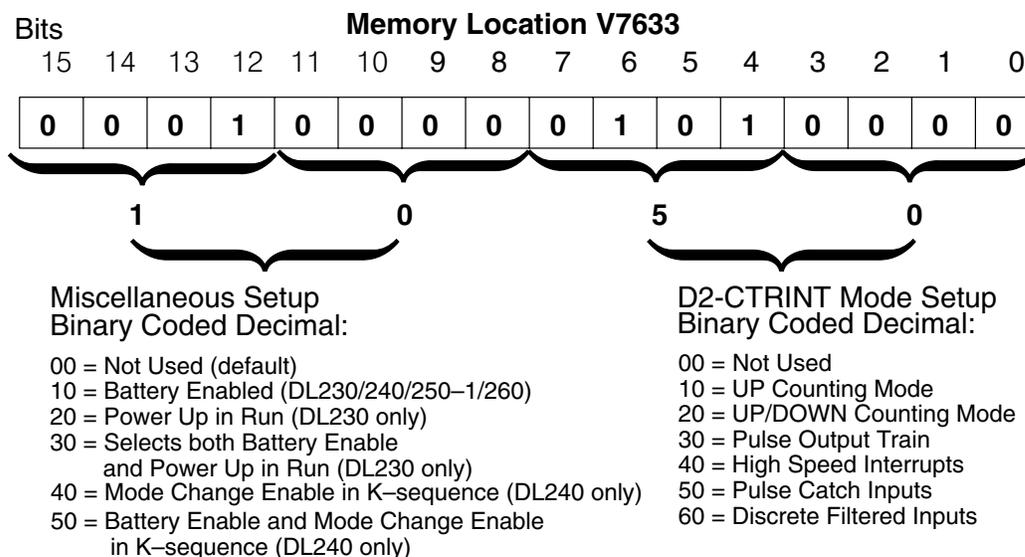
To count the pulses correctly, Mode 10 or Mode 20 should be used.

Understanding V-Memory Setup Locations

The Pulse Catch Input, Mode 50, associated with the D2-CTRINT requires V-memory configuration in order to be used.

V-memory location V7633 is the most important of all the reserved memory areas because it stores the 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 the Pulse Catch Input, Mode 50, the lower and upper bits are set to 10 so the backup battery is enabled. Together they form the hexadecimal number 1050.



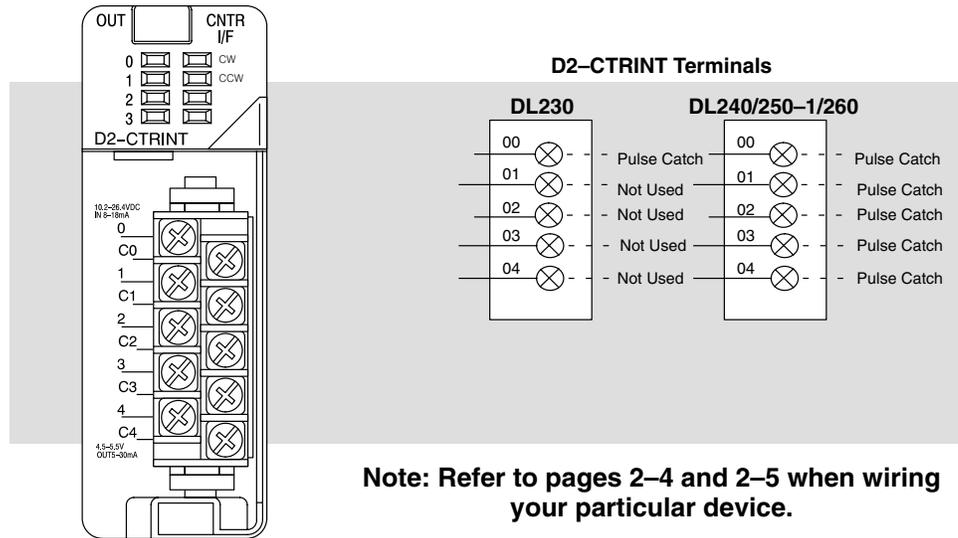
NOTE: It is important to look at the entire 16 bits at V7633. If the RLL program only sets the bits in the lower byte when entering the Counter Interface mode value, the upper bits will be overwritten with zeros (0's). Always enter a 4-digit BCD value in the V-memory. This way, the proper value will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

There are also other V-memory locations which contain Counter Interface setup information for each I/O point. They will be automatically configured with default values for each Counter Interface mode selected.

Default Settings When xx50 is written to V7633, the CPU places the following default codes in V-memory.

Configuration	Point 00/V7634	Point 01/V7635	Point 02/V7636	Point 03/V7637	Point 04
DL230 Hexadecimal Code	Pulse Catch Input 0005	Not Used 0000	Not Used 0000	Not Used 0000	Not Used
DL240/250-1/260 Hexadecimal Code	Pulse Catch Input 0005	Pulse Catch Input 0005	Pulse Catch Input 0005	Pulse Catch Input 0005	Not Used

Default Settings for Interrupt Input Mode 60



Mode 50
Pulse Catch Input

Custom Configuration

Up to this point, only Mode 50 default settings have been discussed. The default settings will be suitable for many applications and will not require a custom 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.

Mode 50 Options

Point Number	V-Memory Location	Possibility (One per point)	Hex Value
point 00	V7634	Pulse Catch Input	0005 (DL240/250-1/260) default
		High Speed Interrupt (timed)	ttt4 (ttt=1 to 999ms timer setting)
		Discrete Filtered Input	xx06 (xx=filter time)
		High Speed Interrupt (DL240/250-1/260)	0004
point 01	V7635	Pulse Catch Input (DL240/250-1/260)	0005 (DL240/250-1/260) default
		Discrete Filtered Input	xx06 (xx=filter time)
		High Speed Interrupt (DL240/250-1/260)	0004
point 02	V7636	Pulse Catch Input (DL240/250-1/260)	0005 (DL240/250-1/260) default
		Discrete Filtered Input	xx06 (xx=filter time)
		High Speed Interrupt (DL240/250-1/260)	0004
point 03	V7637	Pulse Catch Input (DL240/250-1/260)	0005 (DL240/250-1/260) default
		High Speed Interrupt (DL240/250-1/260)	0004
		Discrete Filtered Input	xx06 (xx=filter time)
point 04	V7637	Not available in Mode 50	_____

Setting UP the CPU for the Pulse Catch Inputs

Configuring the V-Memory

The DL240, DL250-1 or DL260 CPUs checks the V-memory to see if there is a D2-CTRINT Module present. There will be the value 10, 20, 30, 40, 50 or 60 in V7633 if the module has been properly configured. If the CPU finds that a Counter Interface module is present, other V-memory locations will be checked to see how 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 **DirectSOFT32**. The following examples will show how to use **DirectSOFT32** to configure the Pulse Catch Inputs.

Step 1: Enter the Mode Selected

If Mode 50, Pulse Catch Input, has been chosen as the primary function, the value 50 must be placed in V7633. The following **DirectSOFT32** diagram shows the setup procedures for communicating with the DL240/250-1/260 CPU. Refer to the **DirectSOFT32 Programmers User Manual** for more details.

The screenshot shows the DirectSOFT32 Programming interface in Mode 50. The main window displays a ladder logic rung (Rung 1) with the following text:

```

1
----- ( ENI )
This RUNG loads the parameters to use the Pulse Catch Input of the D2-CTRINT.
V7633 = 50  Sets Pulse Catch Mode
V7634 = 5   Pulse Catch
V7635 = 5   Pulse Catch
V7636 = 5   Pulse Catch
V7637 = 5   Pulse Catch
_VfirstScan
SP0
-----
2

```

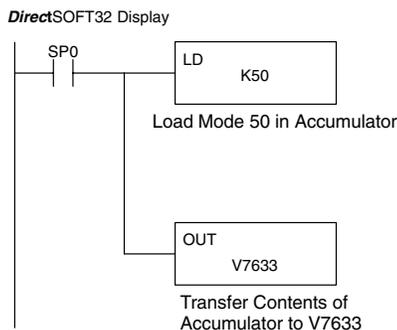
An annotation "Setting the V-Memory using RLL" points to the rung. Below the rung, the Memory Editor dialog box is open, showing a table of V-memory locations:

Address	Value	Address	Value
V7633	50	V7643	0
V7634	5	V7644	0
V7635	5	V7645	0
V7636	5	V7646	0
V7637	5	V7647	0
V7640	0	V7650	0
V7641	0	V7651	0
V7642	0	V7652	0

An annotation "Setting the V-Memory using the Memory Editor" points to the Memory Editor dialog box. The Memory Editor also shows format options (Binary, BCD/Hex, Octal, Text, Decimal) and data size options (Word, Double, Custom).

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.

The following RLL example shows how to set the Pulse Catch Input, Mode 50, in V-memory location V7633.



Two 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, 50 is to be placed in V7633. This value is loaded into the accumulator, LD K50. 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.

Step 2: How Many Pulse Catch Inputs

Up to four(4) pulse catch inputs can be used with the DL240/250-1/260 CPUs and one (1) pulse catch input for the DL230 CPU. The following steps will discuss the programming for each channel which has an interrupt device wired to it.

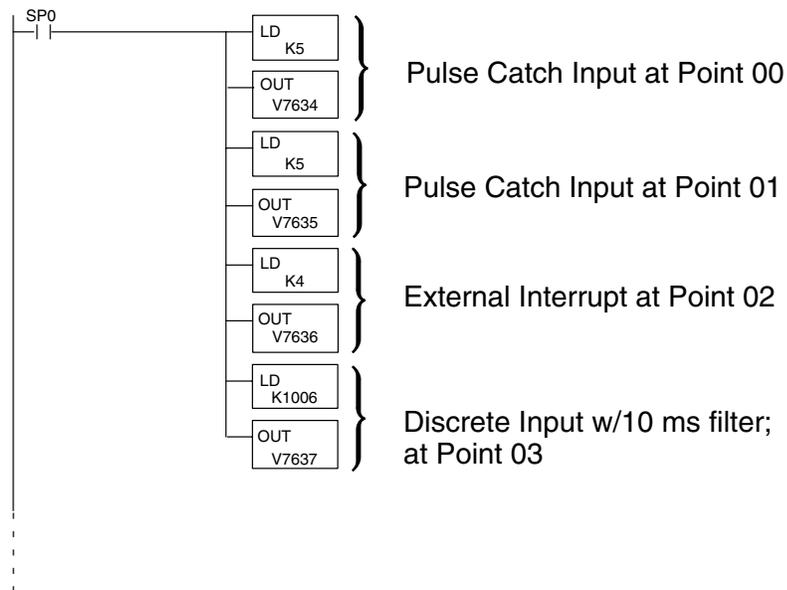
Step 3: Configure the V-Memory

The table below gives a description for each of the V-memory locations that must be configured for each I/O point which are selected to have high speed interrupt capability.

V-Memory	Description
V7633	Primary Mode (Pulse Catch=50)
V7634	Point 00
V7635	Point 01
V7636	Point 02
V7637	Point 03

In the below example, Channels 1 and 2 are configured as Pulse Catch Inputs, Channel 3 is to be a Discrete Filtered Input, and Channel 4 to be an Interrupt Input.

DirectSOFT32 Display



Notice that the hex number 5 is stored in the V-memory locations for each I/O point which are to be Pulse Catch Inputs. The number 4 is used for an external interrupt at an I/O point, and Kxx06 is used for a programmable discrete filtered input (where xx represents the filtering time constant in milliseconds.)

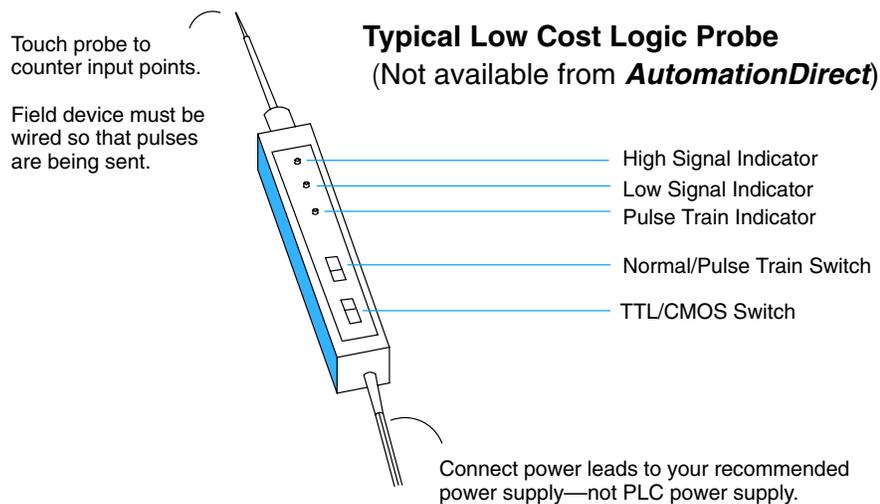
Troubleshooting

The following information may provide some assistance in handling any problems which may be encountered when setting up the D2-CTRINT module, should they occur. **Experience has shown that most problems occur because of improper configuration. Always re-check configuration before anything else.**

For verifying types of inputs (or outputs) which do not relate to the Pulse Catch Input, see the Chapters in this manual covering the specific function. Listed below are some things that could possibly go wrong with the high speed interrupt inputs:

1. No pulse catching appears to be taking place.
2. The status indicator LED is not lit for the input point where the pulse catching is wired (i.e. points 00 and 01).

Defective Field Device - If a field device is suspected to be faulty, verify its proper operation first. Examine the characteristics of the pulses being received with an oscilloscope, test equipment type digital counter, or an inexpensive logic probe.



Check the specifications for the field device. Make certain that the output signal matches the specifications of the D2-CTRINT module.

Pulse Width – The pulse width may be too narrow. The positive transition must remain HIGH for at least 0.1 ms in order for the module to detect its presence.

Wiring - Simple as this might seem, quite often poor wiring is the cause of many problems. Be sure there is a complete electrical loop between the device and the input module. Along with visual inspection, use a voltmeter to check the wiring.

Input Voltage - If the input device is sending a signal that is less than 12 volts, most likely the counter will not function or function improperly. Replace the field device with one which has the proper output level if necessary.

Improper Configuration - Verify that proper values have been used in the configuration. If interfacing a DL230, point 00 is the only point available for pulse catching.

Status Indicators – Make sure the PWR or BAT LED's are not lit on the CPU. Be sure that the status indicators are lit as the pulse signals are received at the proper input point on the D2-CTRINT module. If an LED is not functioning, check the point with a voltmeter to be sure that the I/O point is being energized. .

Mode 60 – Discrete Filtered Inputs

In This Chapter. . . .

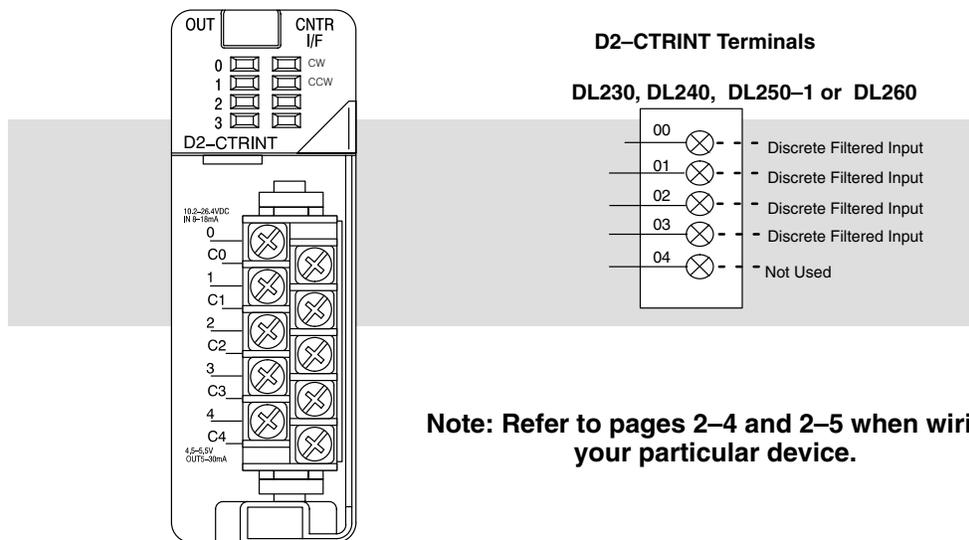
- Wiring the Discrete Filtered Inputs
 - Configuring the Discrete Filtered Parameters
 - Writing the Control Program
 - Verification of Proper Operation
 - Troubleshooting
-

Using the Discrete Filtered Inputs, Mode 60

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*. Discrete Filtered Inputs, Mode 60 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 to make use of the D2-CTRINT's pulse catch inputs. A good place to begin is to learn what each channel of the module represents when it is being used for Discrete Filtered Inputs.

Default Settings for Discrete Filtered Inputs, -Mode 60



Note: Refer to pages 2-4 and 2-5 when wiring your particular device.

Shown in the above diagram and illustration are points 00 through 03 which default to discrete filtered inputs when the module is set for Mode 60.

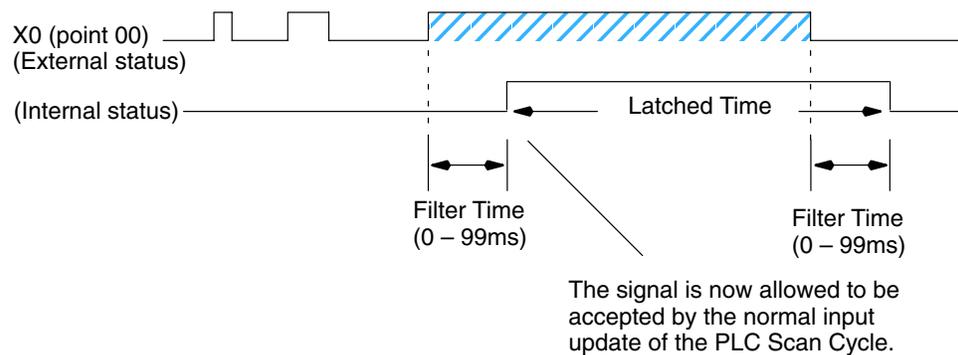
All of the I/O points do not have to be used as Discrete Filtered Inputs; there are other options which can be used.

Discrete Filtered Inputs

Some applications have inputs from field devices which have high noise levels. Adjustable input filtering will help eliminate these nuisance signals caused by switch bounce or other sources.

When an input signal is first detected at any one of the four channels, a programmable filter is activated which begins a timed countdown. The ON status of the signal is temporarily prevented from being read by the input update of the CPU. The ON signal must stay present long enough for the filter to “time out”.

Once the signal has remained ON for the programmed time, it is latched and allowed to be accepted by the CPU during the normal input update of the PLC scan cycle. The signal is latched for the remaining duration of the ON signal plus an amount of time equal to the filter time. The filter time can be programmed from 0 to 99ms in 1ms increments.



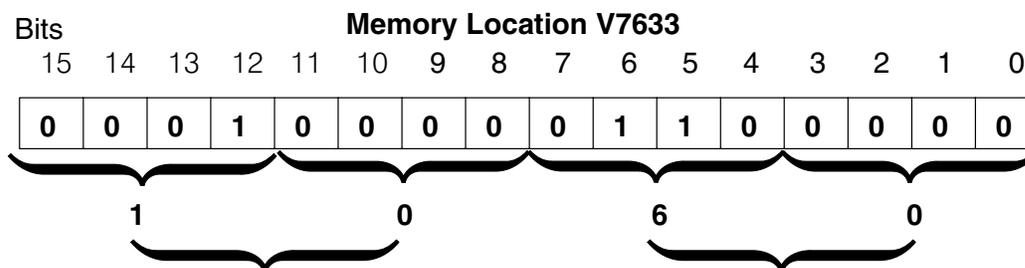
Note: If a zero (0) is stored to V-memory as the filter time, the CPU will treat the configuration as having no filter. The filter time range is 0 – 99ms.



Understanding V-Memory Setup Locations

The Discrete Filtered Input, Mode 60, associated with the D2-CTRINT requires V-memory configuration in order to be used.

V-memory location V7633 is the most important of all the reserved memory areas because it stores the 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 is for the Discrete Filtered Input, Mode 60. The lower bits are set to 60 and the upper bits are set to 10 so the backup battery is enabled. Together they form the hexadecimal number 1060.



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



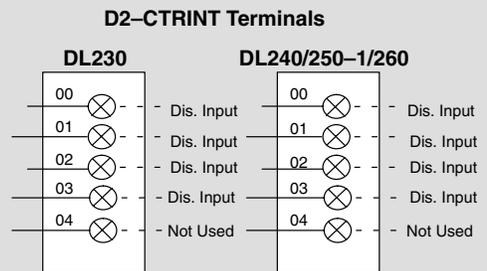
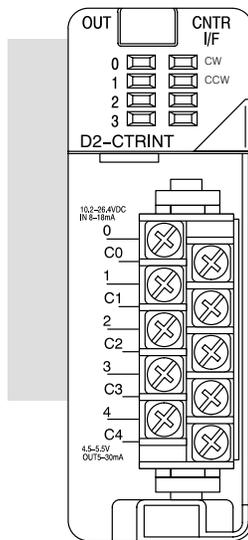
NOTE: It is important to look at the entire 16 bits in V7633. If the RLL program only sets the bits in the lower byte when entering the mode value, the upper bits will be overwritten with zeros (0's). Always enter a 4-digit BCD value in the V-memory. This way, the proper value will be written into the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

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

Default Settings When xx60 is written to V7633, the CPU places the following default codes in V-memory:

Configuration	Point 00/V7634	Point 01/V7635	Point 02/V7636	Point 03/V7637	Point 04
DL230 Hexadecimal Value	Discrete Input 1006	Discrete Input 1006	Discrete Input 1006	Discrete Input 1006	Not Used
DL240/250-1/26 0 Hexadecimal Value	Discrete Input 1006	Discrete Input 1006	Discrete Input 1006	Discrete Input 1006	Not Used

Default Settings for Interrupt Input Mode 60



Note: Refer to pages 2-4 and 2-5 when wiring your particular device.

Mode 60
Discrete/Filtered Inputs

Custom Configuration

Up to this point, only Mode 60 default settings have been discussed. The default settings will be suitable for many applications and will not require a custom 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.

Mode 60 Options

Point Number	V-Memory Location	Possibility (One per point)	Hex Value
point 00	V7634	Discrete Filtered Input	xx06 (xx=filter time) (default)
		High Speed Interrupt (Internal)	0004
		High Speed Interrupt (timed)	ttt4 (ttt=1 to 999ms timer setting)
		Pulse Catcher	0005
point 01	V7635	Discrete Filtered Input	xx06 (xx=filter time) (default)
		High Speed Interrupt (DL240/250-1/260)	0004
		Pulse Catcher (DL240/250-1/260)	0005 (DL240/250-1/260)
point 02	V7636	Discrete Filtered Input	xx06 (xx=filter time) (default)
		High Speed Interrupt (DL240/250-1/260)	0004
		Pulse Catcher (DL240/250-1/260)	0005 (DL240/250-1/260)
point 03	V7637	Discrete Filtered Input	xx06 (xx=filter time) (default)
		High Speed Interrupt (DL240/250-1/260 only)	0004
		Pulse Catcher (DL240/250-1/260)	0005 (DL240/250-1/260)
point 04	-----	Not available in Mode 60	-----

Setting Up the CPU for Discrete Filtered Inputs

Configuring the V-Memory

The CPU checks the V-memory to see if there is a High Speed Counter Interface Module present. There will be the number 10, 20, 30, 40, 50, or 60 in V7633 if a module has been properly configured. If the CPU finds that a Counter Interface module is present, it checks other V-memory locations to see how they have been configured for each channel of the module.

The values can be entered into memory by using either a handheld programmer or by editing them into a control program using **DirectSOFT32**. The following examples will show how to use **DirectSOFT32** to configure the Filtered/Discrete Inputs.

Step 1: Enter the Mode

If Mode 60, Discrete Filtered Inputs, has been chosen as the primary function, then the value 60 must be placed in V7633. The following **DirectSOFT32** diagram shows the setup procedures for communicating with a DL230, DL240, DL250-1 or DL260 CPU. Refer to the **DirectSOFT32 Programmers User Manual** for more details.

The screenshot shows the DirectSOFT32 Programming interface in Mode 60. The main window displays a ladder logic rung with the following text:

```

This RUNG loads the parameters to use the Discrete Filter Inputs
of the D2-CTRINT module.

V7633 = 60   Sets the Discrete Mode
V7634 = 1006 Discrete
V7635 = 1006 Discrete
V7636 = 1006 Discrete
V7637 = 1006 Discrete

_FirstScan
SP0
  
```

Below the rung, a Memory Editor dialog box is open, showing the following table of memory addresses and values:

Address	Value	Address	Value
V7633	60	V7643	0
V7634	1006	V7644	0
V7635	1006	V7645	0
V7636	1006	V7646	0
V7637	1006	V7647	0
V7640	0	V7650	0
V7641	0	V7651	0
V7642	0	V7652	0

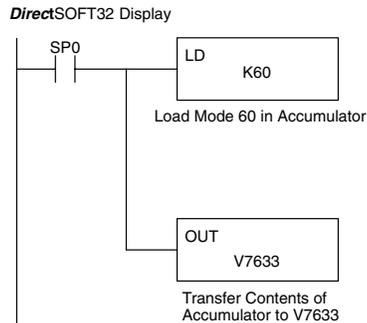
Annotations on the left side of the image point to the rung and the Memory Editor dialog box:

- Setting the V-Memory using RLL (points to the rung)
- Setting the V-Memory using the Memory Editor (points to the Memory Editor dialog box)

On the right side of the image, a vertical label reads: Discrete/Filtered Inputs Mode 60.

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.

The following RLL example shows how to set V-memory location V7633 to Mode 60, Discrete Filtered Input. This is the location for all mode values for this module. Only one mode (6 possible) can be entered, i.e. 10,20,30,40,50 or 60.



Two 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, 60 is to be placed in V7633. This value is loaded into the accumulator, LD K60. The CPU then writes this data to the memory location, V7633, once it reads the OUT instruction, 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.

Up to four (4) Discrete Filtered Inputs can be used with the DL230/240/250-1/260 CPUs. The following steps will discuss the programming for each channel.

**Step 2:
How many
Channels**

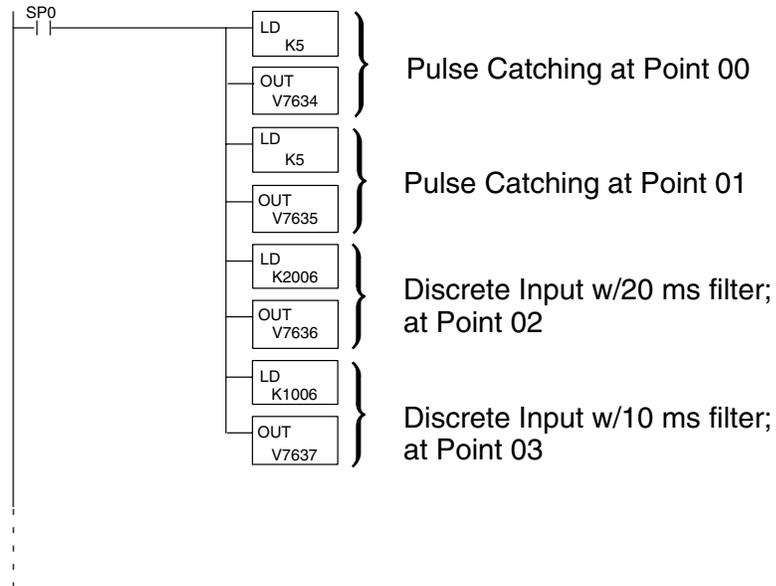
**Step 3:
Configure the
V-Memory.**

The table below gives a description for each of the V-memory locations which must be configured for each I/O point which are selected to be filtered inputs.

V-Memory	Description
V7633	Primary Mode (Discrete Filter=60)
V7634	Point 00
V7635	Point 01
V7636	Point 02
V7637	Point 03

In the below example, Channels 1 and 2 are configured as Pulse Catch Inputs, Channel 3 is to be a Discrete Filtered Input, and Channel 4 to be an Interrupt Input. Channel 3 is set to be a 20ms filter and Channel 4 is set to be a 10ms filter.

DirectSOFT32 Display



The hex number 5 stored in points 00 and 01 is the value for the Pulse Catching feature. Notice that the upper two digits of each hex value stored in V-memory for points 02 and 03 are the timing constants for filtering. The last two digits, 06, is the value for Discrete Filtered Inputs.

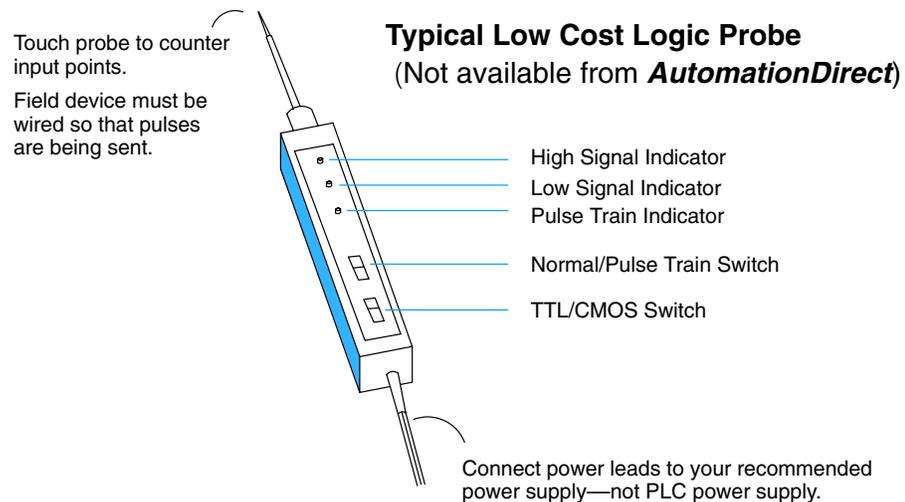
Troubleshooting

The following information may provide some assistance in dealing with problems which may be encountered when setting up the D2-CTRINT module, should they occur. **Experience has shown that most problems occur because of improper configuration. Always re-check configuration before anything else.**

For verifying types of inputs (or outputs) which do not relate to external High Speed Interrupts, see the Chapters in this manual covering the specific function. Listed below are some things that could possibly go wrong with the high speed interrupt inputs:

1. No filtering appears to be taking place.
2. The status indicator LED is not lighting for the input point where the Discrete Filtered Input are connected.

Defective Field Device - If a field device is suspected to be faulty, verify its proper operation first. Examine the characteristics of the pulses being received with an oscilloscope, test equipment type digital counter or an inexpensive logic probe.



Check the specifications for the field device. Make certain that the output signal matches the specifications of the D2-CTRINT module.

Pulse Width – The pulse width may be too narrow. The positive transition must remain HIGH for at least 0.1 ms in order for the module to detect its presence. Select another field device which has a pulse longer than 0.1ms.

Wiring - Simple as this might seem, quite often poor wiring is the cause of many problems. Be sure there is a complete electrical loop between the device and the input module. Along with visual inspection, use a voltmeter to check the wiring.

Input Voltage - If the input device is sending a signal which is less than 12 volts, most likely the counter will not function or function improperly. Replace the field device with one which has the proper output level if necessary.

Improper Configuration - Verify that proper values have been used in the configuration. The value for a filtered input follows the format xx06, where xx is the filtering time in milliseconds.

Status Indicators – Make sure the PWR or BAT LEDs are not lit on the CPU. Be sure that the status indicators are lit as the pulse signals are received at the proper input point on the D2-CTRINT module. If an LED is not functioning, check the point with a voltmeter to be sure that the I/O point is being energized.