

# 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
-

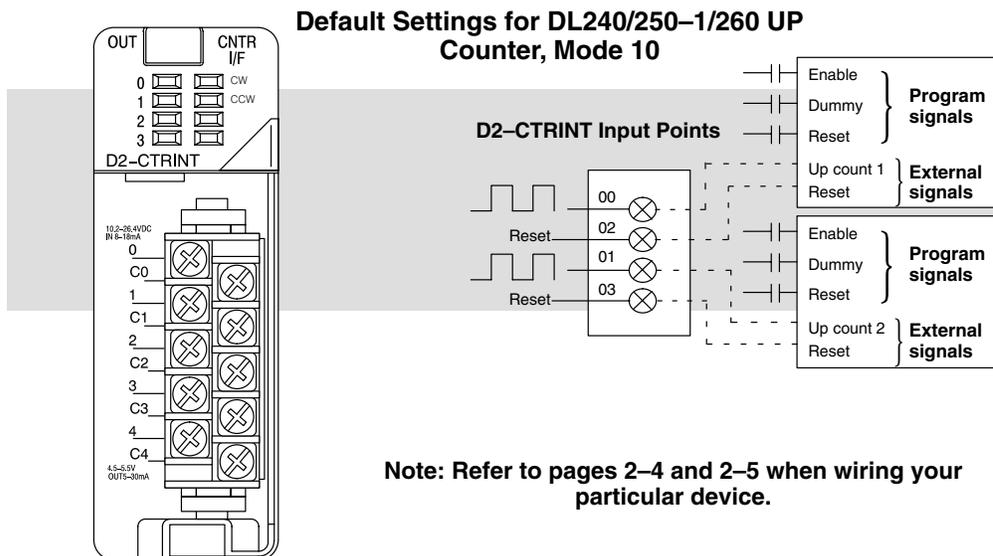
# Using the UP Counters, Mode 10

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

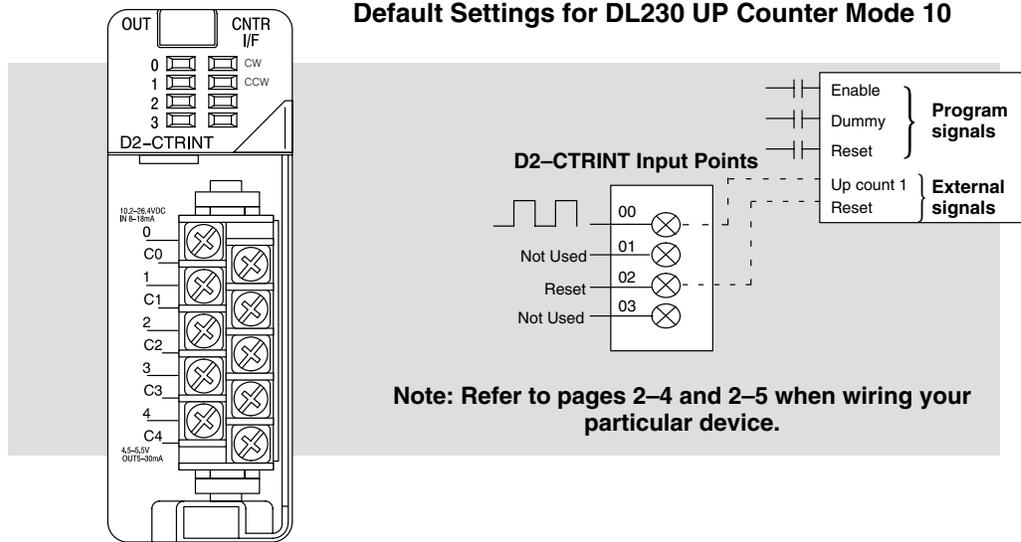
Mode 10  
UP Counter



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.

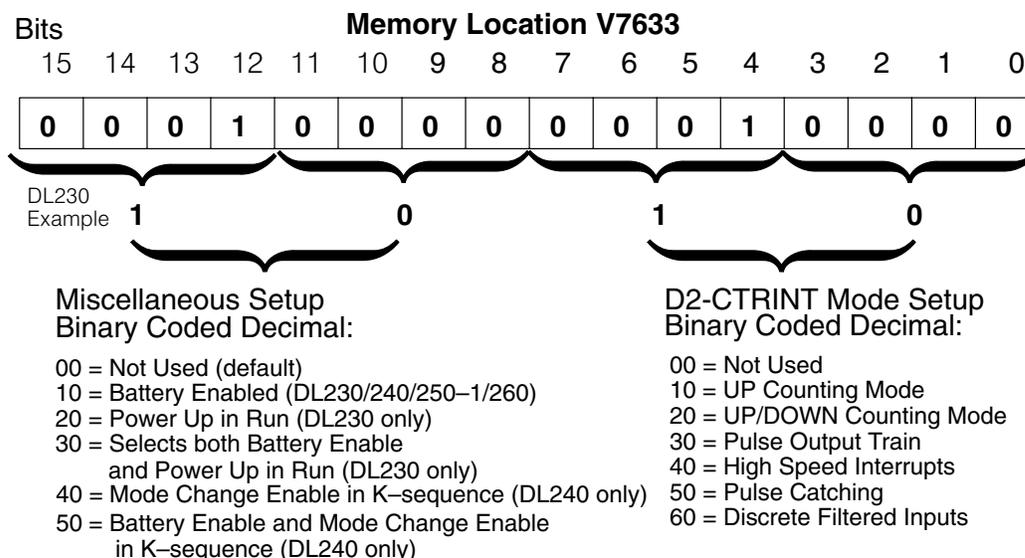


In this example, the first channel (point 00) receives the input from the device or equipment sending the DC level signals for counting. The third channel (point 02) is configured so that it can be wired to some device that will trigger an external reset. The other two points (points 01 and 03) are not 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.

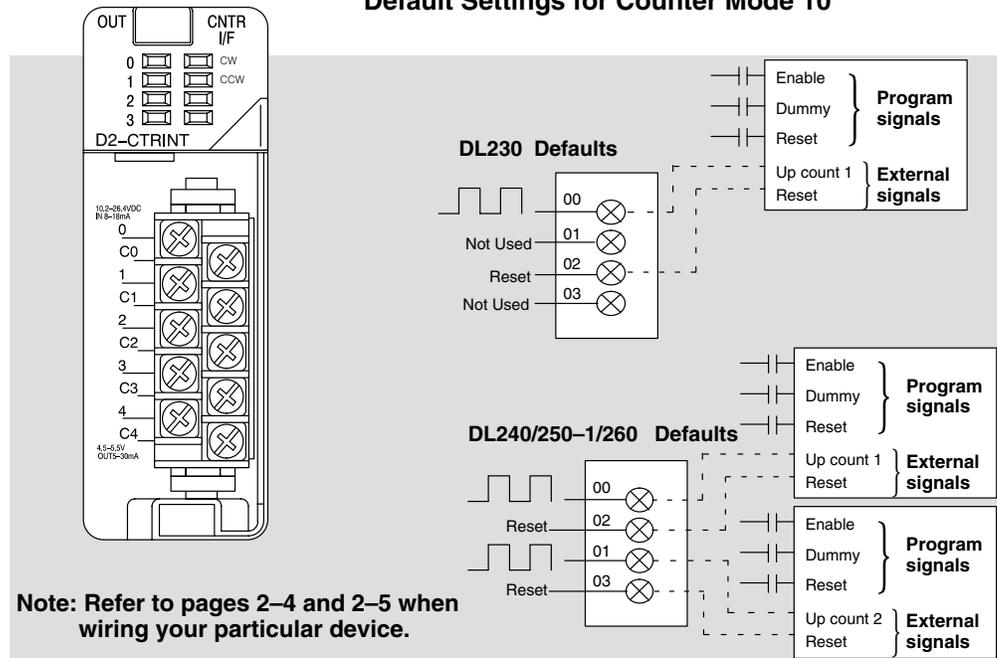
V-memory location V7633 is the most important of all the reserved memory areas because it stores the numeric value which lets the CPU know which mode has been selected. The following diagram shows the 16-bit word and the various information it stores, including the values used for the D2-CTRINT. The 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.

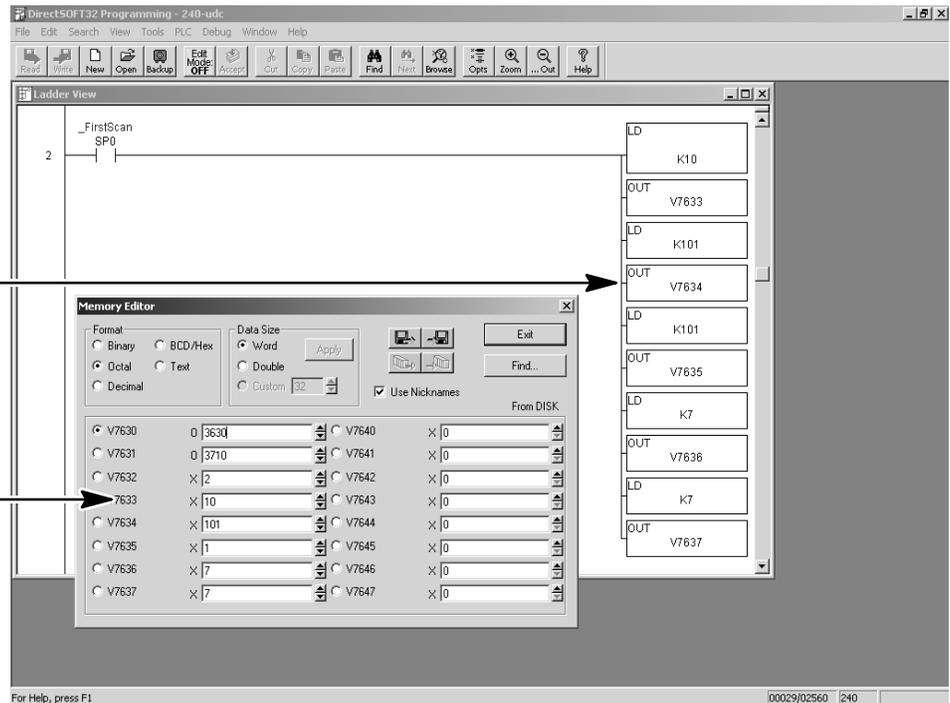
The values can be inserted into memory by using either a handheld programmer or a program edited with **DirectSOFT32**. There are program examples in this chapter for each counter function and mode. The following steps is a guide for configuring the V-memory for the UP counter function.

### 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.

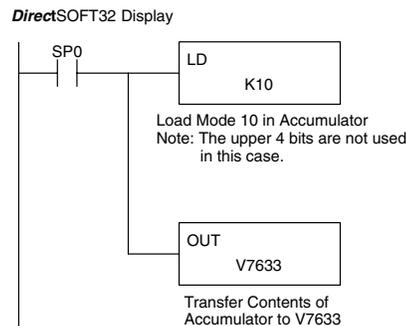
Setting the  
V-Memory  
using RLL

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)**

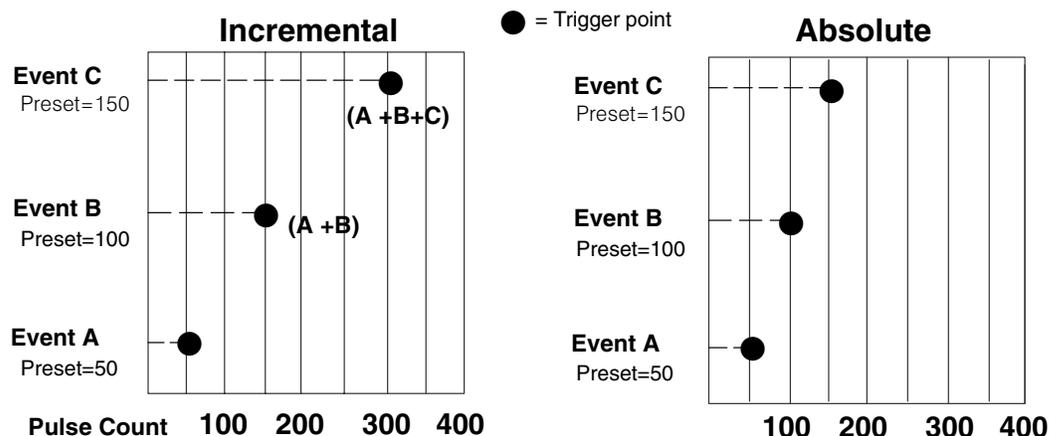
If a DL230 is being used or if the Absolute counting mode is accepted for a DL240, DL250-1 or a DL260, you can skip this step. The CPU automatically assumes that the absolute presets are being used and configures the V-memory accordingly.

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.

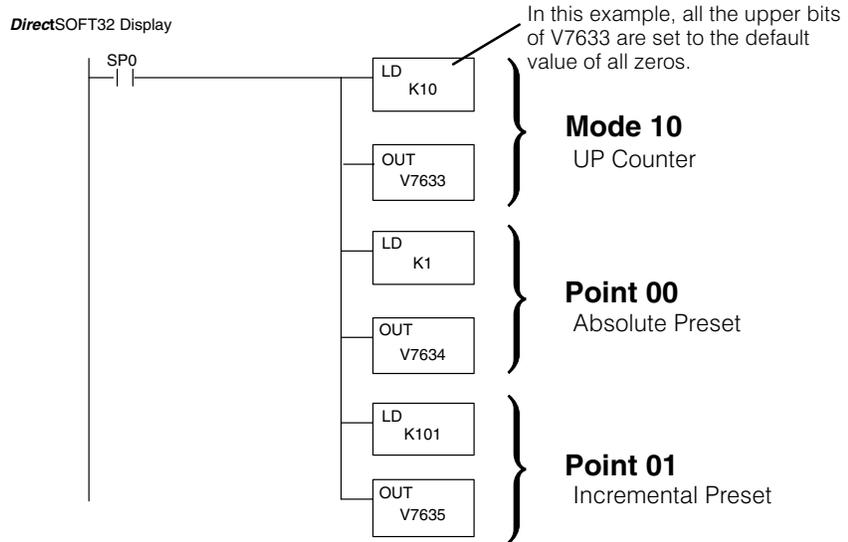


Assume that a limit switch (SW1) is being connected to one of the high speed counters. The counter is to initiate three different events at certain points in time determined by the pulse count received. Also, assume that the presets are stored (associated with each of the three events) in successive 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.

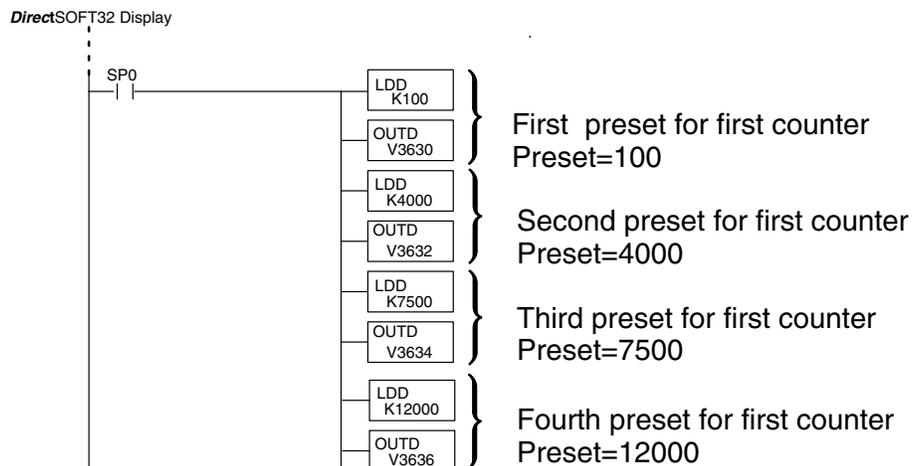


The previous RLL example which set the D2-CTRINT to Mode 10 now has the two preset values added to it. The CPU is set for the counter to use the Absolute preset mode at point 00 and to use the Incremental preset mode at point 01.

### 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.

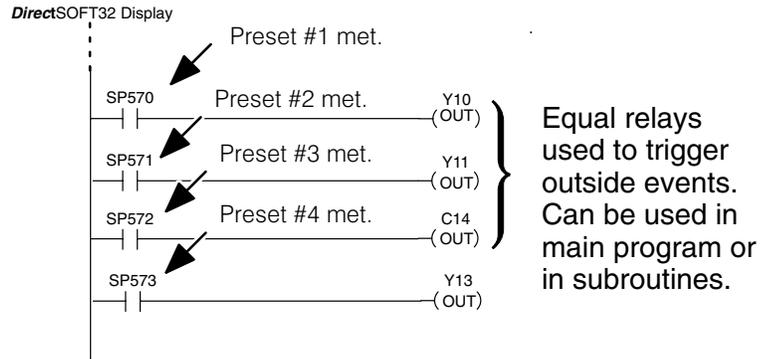
The following RLL diagram loads presets into consecutive V-memory locations, starting with the default memory location V3630. Use LDD and OUTD instructions as indicated in the example below:



**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.

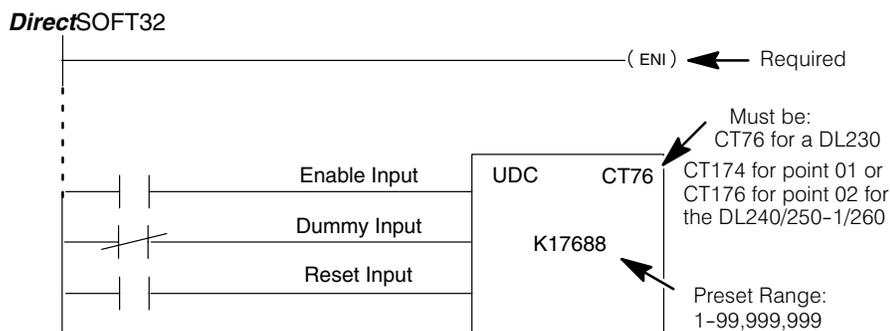
When using **K207** or **K307**, the counter module reads the preset V-memory only at power up, or when a CPU mode change occurs (i.e. PROGRAM to RUN transition).

**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.

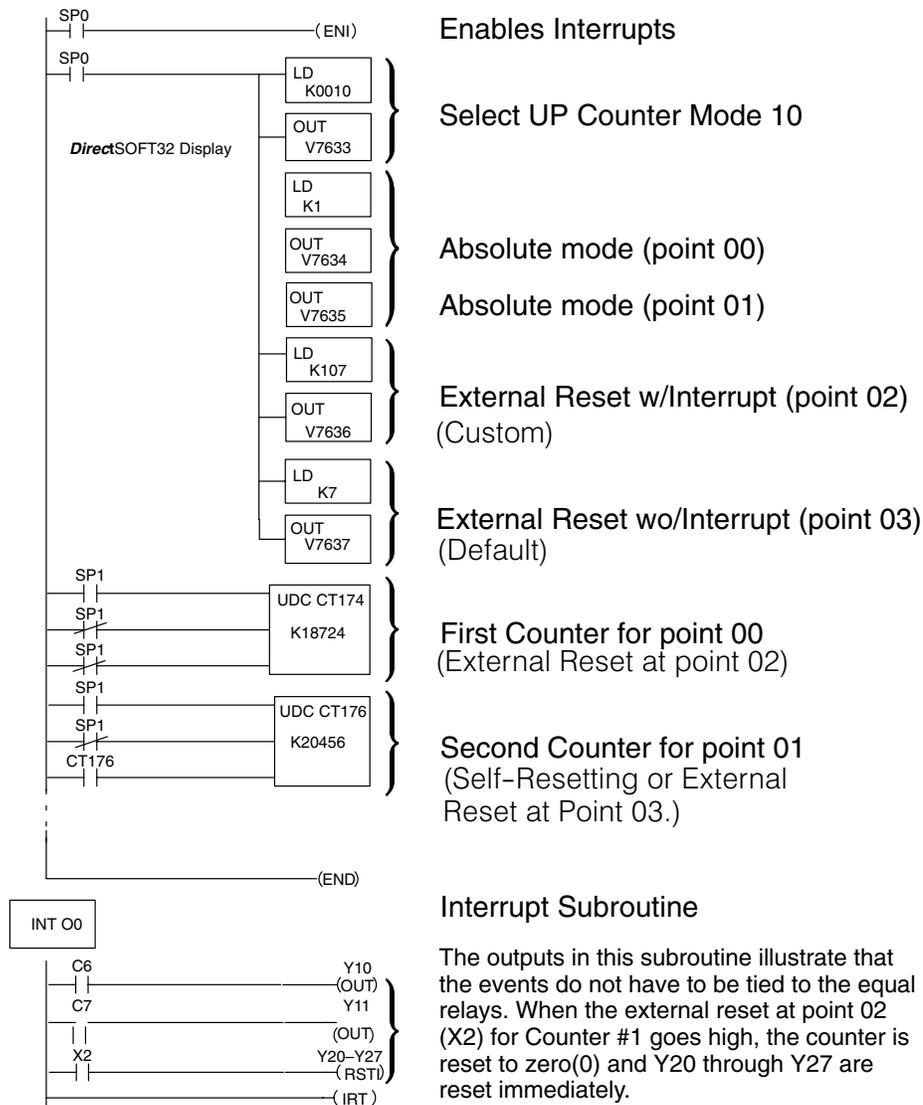
After entering the counter address and preset, enter the contact addresses for the contacts which are automatically drawn in the RLL program.

### 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

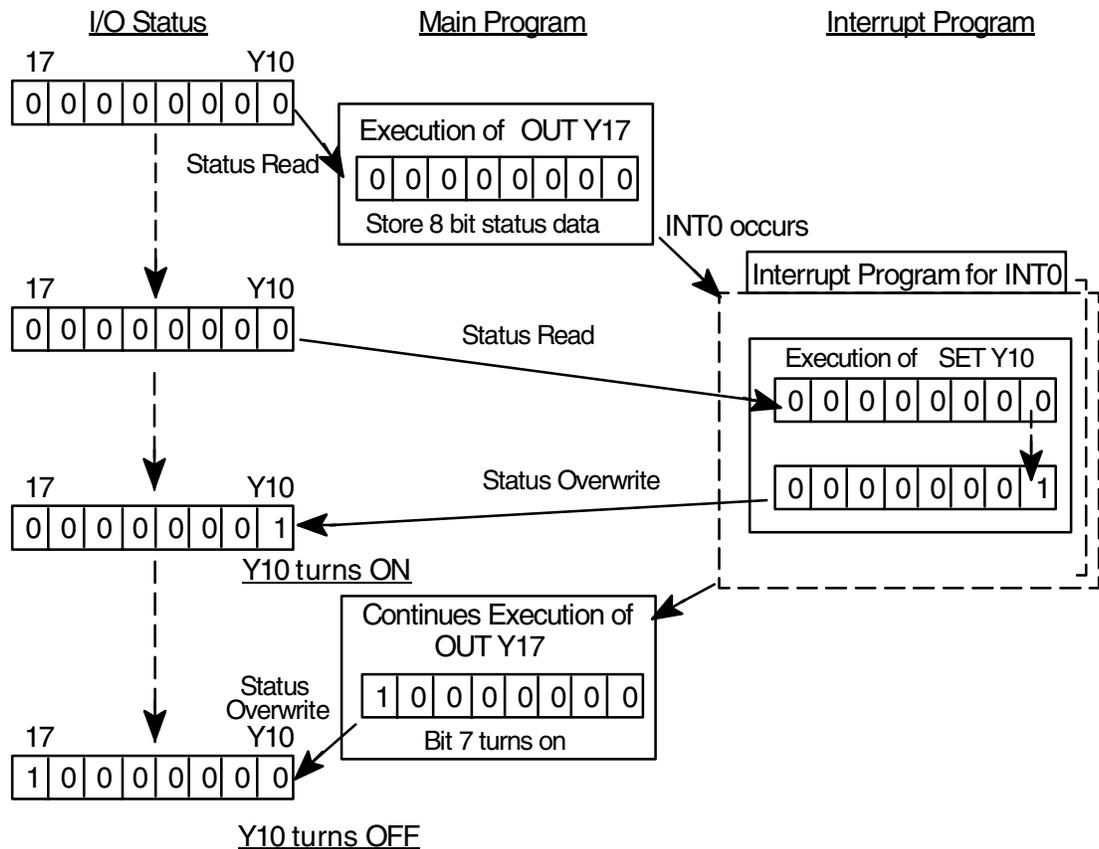
In the previous example, whenever point 02 is enabled, CT174 is reset to zero (0). At the same time, the CPU suspends its normal main program scan and execution, going to the interrupt subroutine. When it finishes executing the subroutine and encounters the return statement, the execution returns to 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.

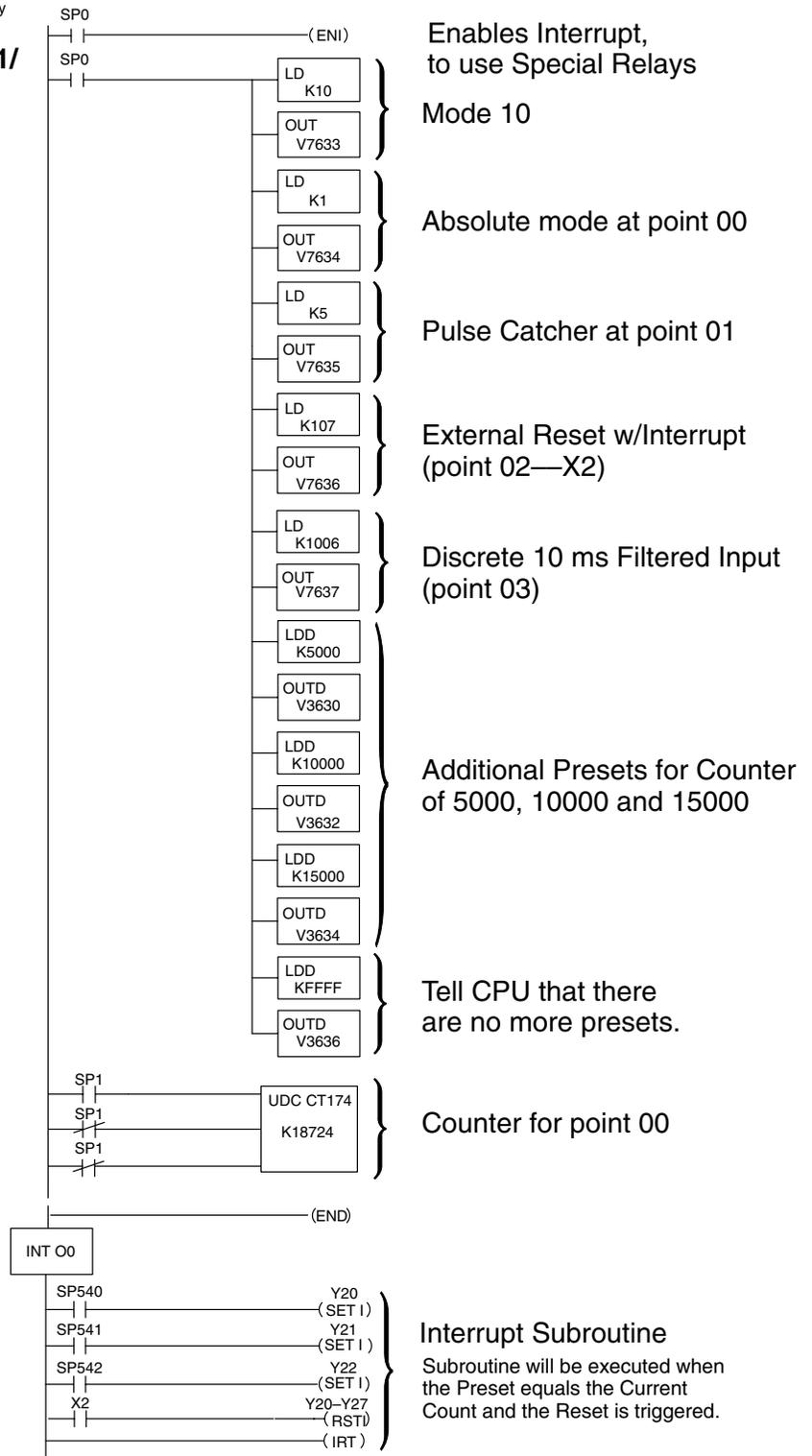
Each preset requires two V-memory locations (32-bits), i.e. V2320 and V2321 are the first preset, V2322 and V2323 are the second, etc. Each of these presets has a special equal relay assigned to it. The numbering assignment for these are one digit apart, i.e. SP540 assigned to the first preset, SP541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

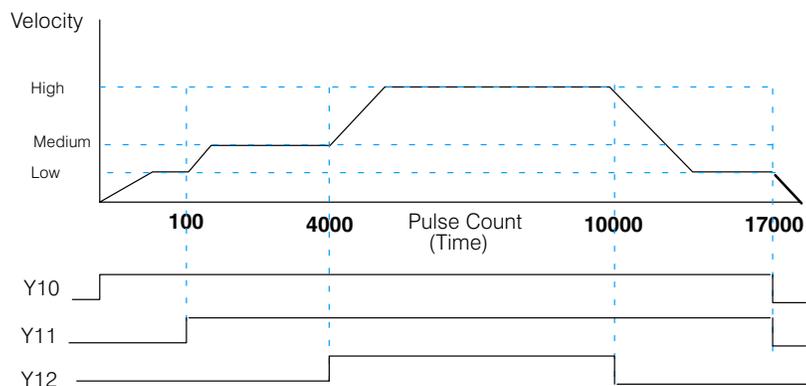
*DirectSOFT32 Display*  
**DL240/250-1/  
260  
Example**



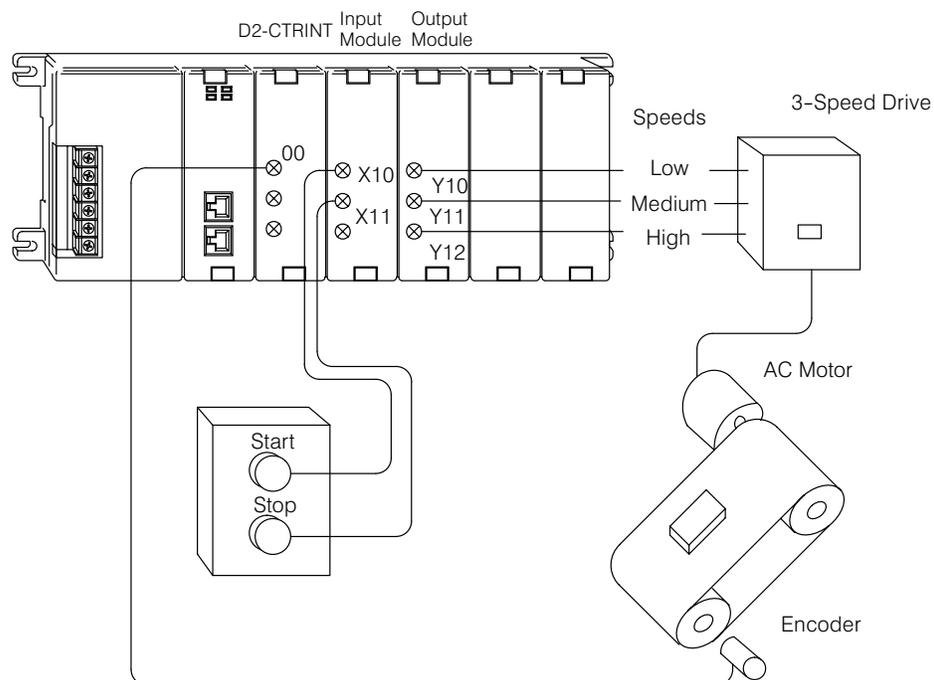
**NOTE:** To trigger outside events with the three presets, assign SP540,541 and 542 respectively to V3630, V3632, and V3634. (See Table on Page 3-11.)

## 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.

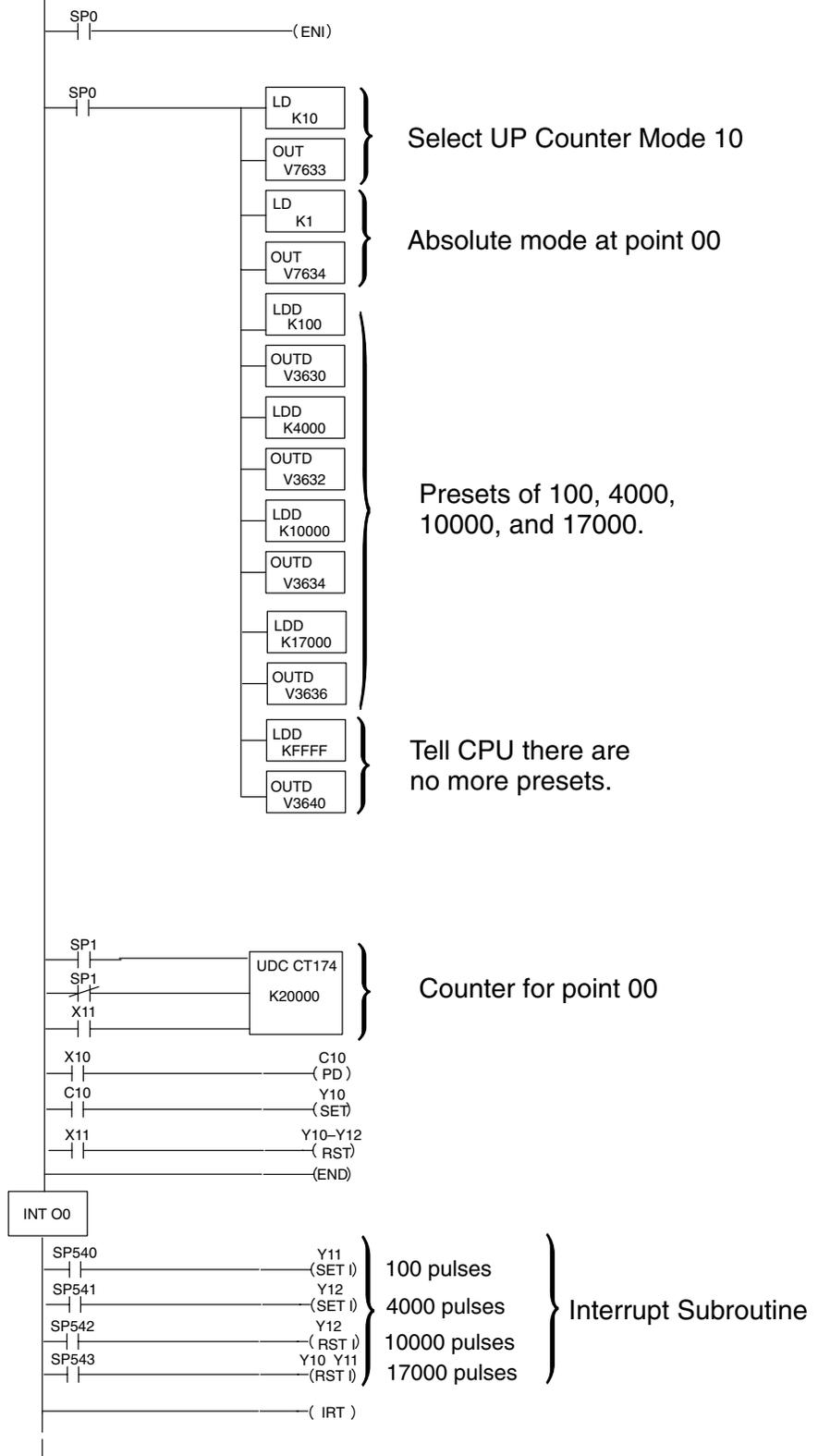


Assume that there are input and output modules in a DL205 base and that X10 and X11 of the input module are wired to a START/STOP switch. The inverter being used in this example has programmable input relays that accept low, medium and high speed ranges that have been programmed into the drive. The drive 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**

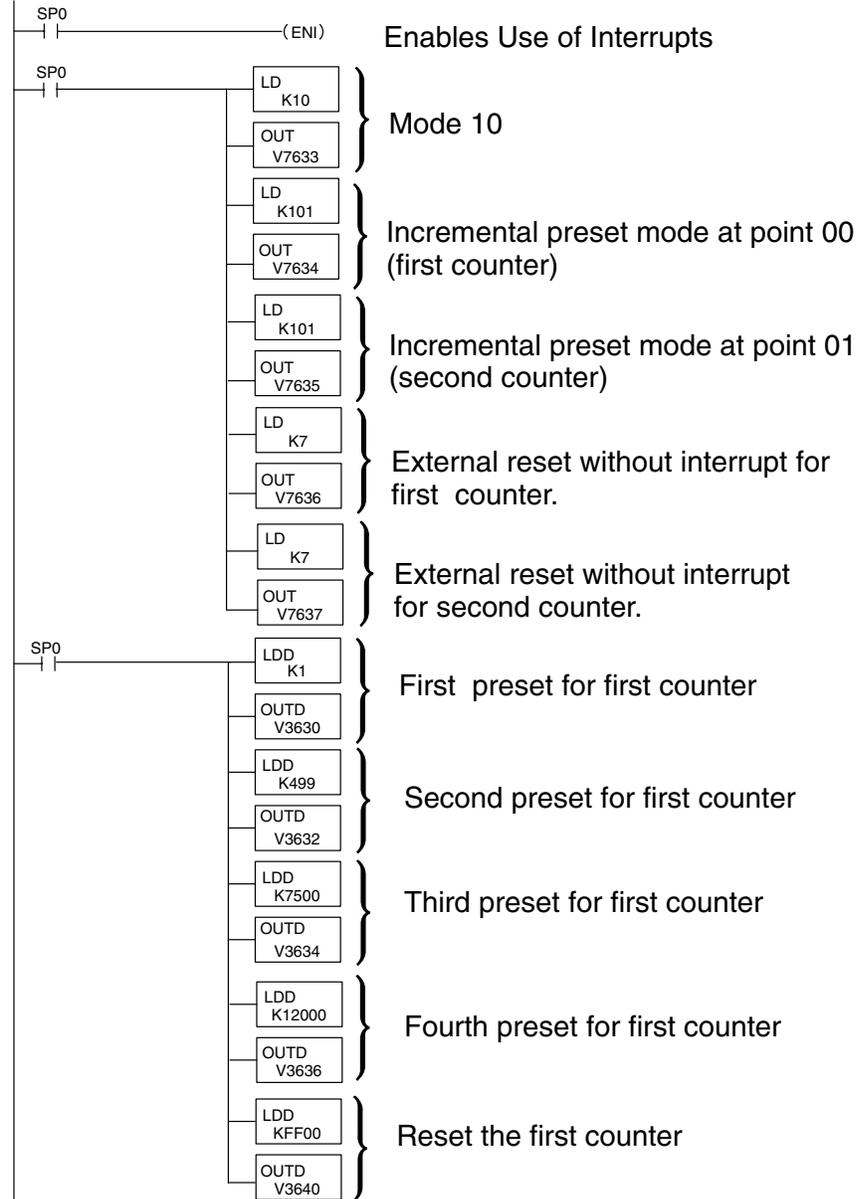
The following describes the events as they are scanned and executed in the example program on the preceding page.

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.

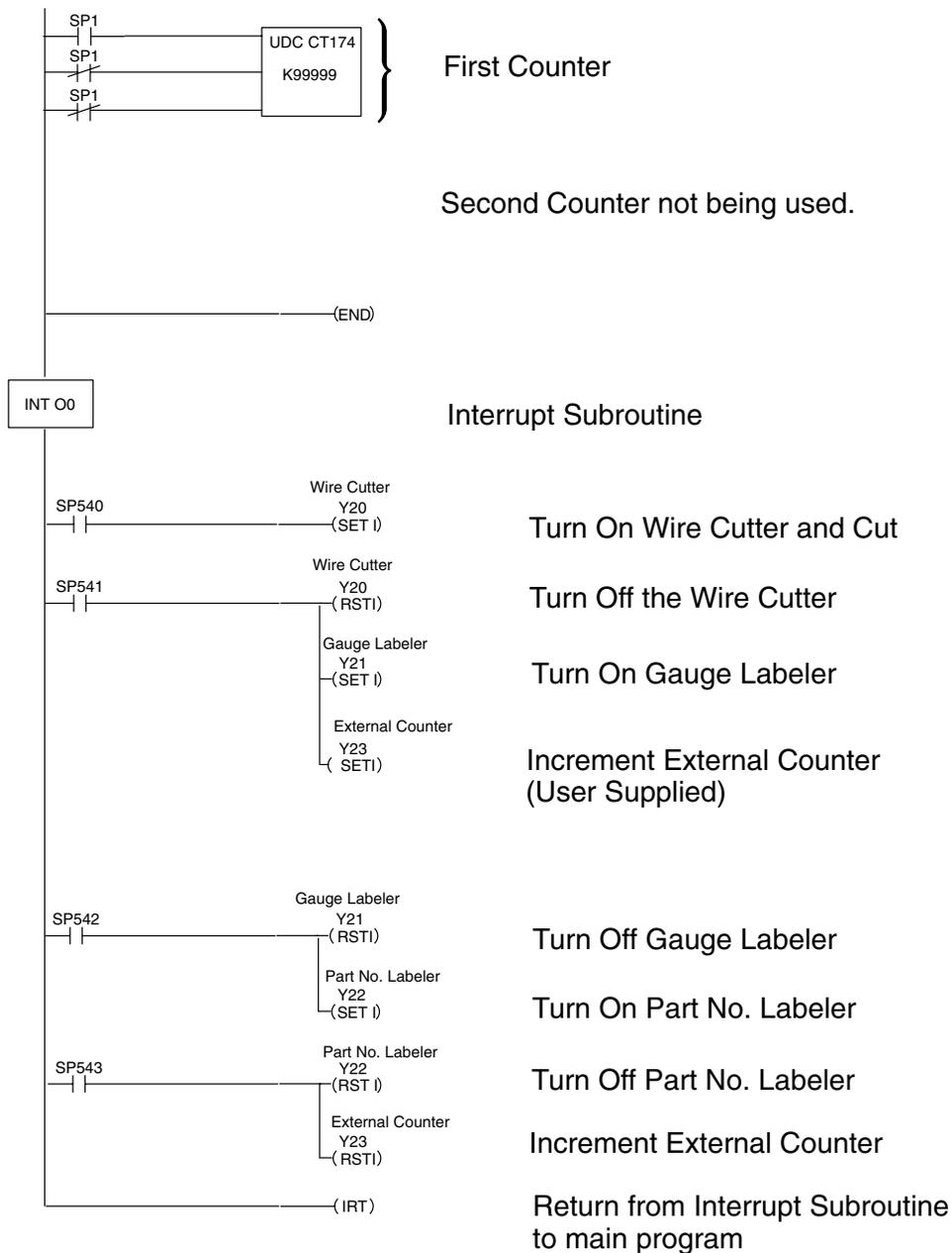
DirectSOFT32 Display



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.

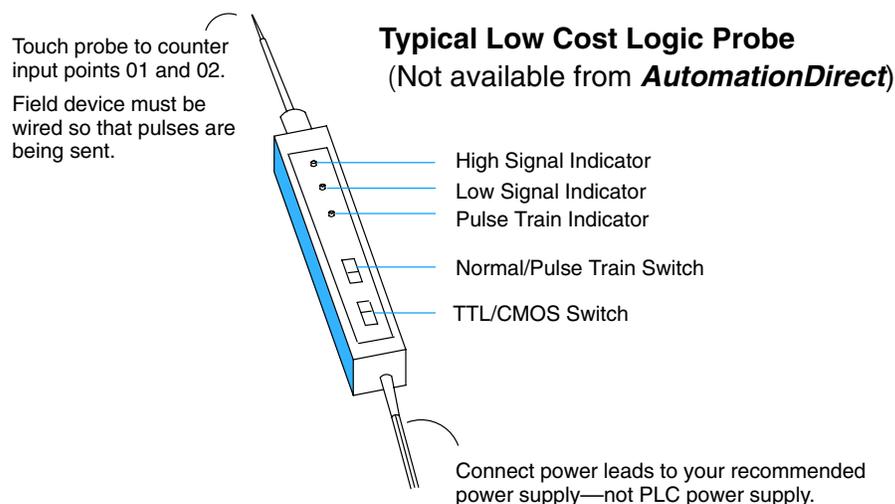
### Counter Doesn't Count

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.

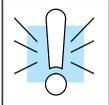
**Wiring** - Simple as this might seem, quite often poor wiring is the cause of many problems. Make sure that there is a complete electrical loop between the device and the input module. Along with visual inspection, use a voltmeter to check this.

**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?