F2-4AD2DA 4-Ch. In / 2-Ch. Out Analog Combination

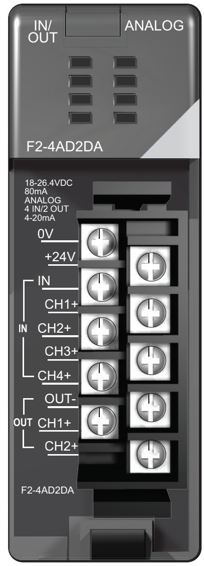
Chapter 14

In This Chapter...

Module Specifications

The F2-4AD2DA analog current input/output module provides several hardware features:

- On-board 250 ohm, 1/2 watt precision resistors provide substantial over-current-protection for 4–20 mA current loops.
- Analog inputs and outputs are optically isolated from the PLC logic.
- The module has a removable terminal block so the module can be easily removed or changed without disconnecting the wiring.
- All input and output channels can be updated in one scan if either a D2-240, a D2-250-1, a D2-260 or a D2-262 CPU is used in the PLC.
- On-board active analog filtering and RISC-like microcontroller provide digital signal processing to maintain precision analog measurements in noisy environments.
- Low-power CMOS design requires less than 80mA from an external 24VDC power supply.



F2-4AD2DA

The following tables provide the specifications for the F2-4AD2DA analog current input/output module. Review these specifications to make sure the module meets your application requirements.

Input Specifications	
Number of Input Channels	4, single ended (one common)
Range	4–20 mA
Resolution	12 bit (1 in 4096)
Input Impedance	250Ω, ±0.1%, 1/2 W, 25ppm / °C current input resistance
Maximum Continuous Overload	±40mA, each current input
Input Stability	±1 count
Crosstalk	-70dB, 1 count maximum
Common Mode Rejection	-50dB @ 800Hz
Active Low-Pass Filter	-3dB @ 50Hz, 2 poles (-12dB per octave)
Step Response	10ms to 95%
Full Scale Calibration Error	±12 counts maximum, @ 20mA current input
Offset Calibration Error	±8 counts maximum, @ 4mA current input
Maximum Inaccuracy	±0.3% @ 25°C (77°F) ±0.45% @ 0–60°C (32–140°F)
Recommended External Fuse	0.032 A, series 217 fast-acting, current inputs

Output Specifications	
Number of Output Channels	2, single ended (one common)
Range	4–20 mA
Resolution	12 bit (1 in 4096)
Peak Withstanding Voltage	75VDC, current outputs
External Load Resistance	0Ω minimum, current outputs
Loop Supply Voltage Range	18–30 VDC, current outputs
Maximum Load / Power Supply	910Ω / 24V, 620Ω / 18V, 1200Ω / 30V, current outputs
Linearity Error (best fit)	±1 count (± 0.025% of full scale) maximum
Settling Time	100µs maximum (full scale change)
Maximum Inaccuracy	±0.1% @ 25°C (77°F) ±0.3% 0–60°C (32–140°F)
Full Scale Calibration Error	±5 counts @ 20mA current output
Output Calibration Error	±3 counts @ 4mA current output

General Module Specifications	
Digital Input and Output Points Required	16 point (X) Inputs 16 point (Y) Outputs
PLC Update Rate	4 input channels per scan maximum (D2-240, D2-250-1, D2-260 and D2-262 CPU) 2 output channels per scan maximum (D2-240, D2-250-1, D2-260 and D2-262 CPU) 1 input and 1 output channels per scan maximum (D2-230 CPU)
Power Budget Requirement	60mA @ 5VDC (supplied by the base)
External Power Supply Requirement	24VDC (±10%), 80mA max. plus 20mA per loop output
Accuracy vs. Temperature	±45 ppm / °C full scale calibration range (including maximum offset change).
Operating Temperature	0–60°C (32–140°F)
Storage Temperature	–20°C to 70°C (-4°F to 158°F)
Relative Humidity	5–95% (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

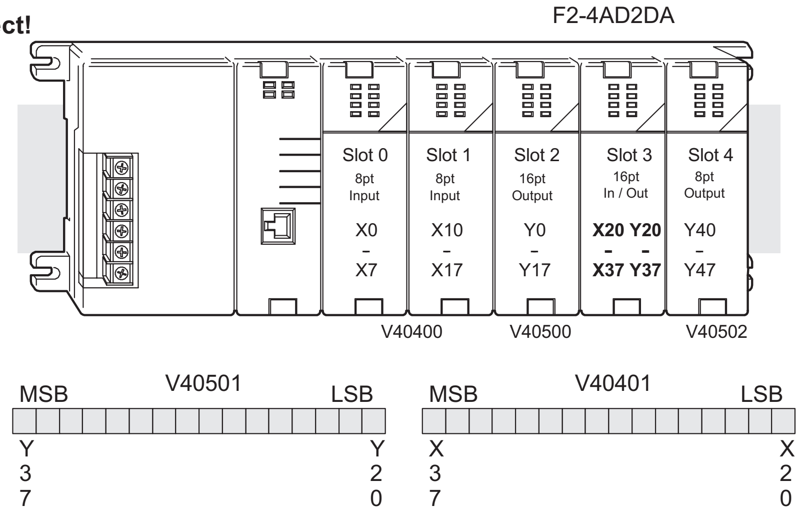
Combination Analog Configuration Requirements

The F2-4AD2DA analog current input/output module requires 16 discrete input points and 16 discrete output points. The module can be installed in any slot of a DL205 system, except when the D2-230 CPU is used. The available power budget may also be a limiting factors. Check the DL205 PLC User Manual for the particular model of CPU and I/O base being used for more information regarding power budget and number of local, local expansion or remote I/O points.

Special Placement Requirements (D2-230 and Remote I/O Bases)

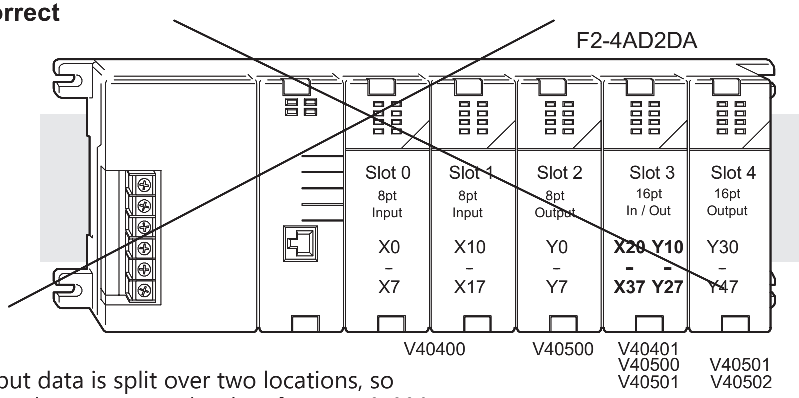
It is important to examine the configuration if a D2-230 CPU is being used with a multiplexing program. As can be seen in the section on Writing the Control Program, V-memory locations are used to manage the analog data. If the module is placed in a slot so that either the input or the output points do not start on a V-memory boundary, the program instructions aren't able to access the data. This also applies when placing this module in a remote base using a D2-RSSS in the CPU slot.

Correct!



Data can be read and written correctly because the input and output points start on a V-memory boundary address as seen in the table on the following page.

Incorrect



Output data is split over two locations, so instructions cannot write data from a D2-230.

To use the V-memory references required for a D2-230 CPU, the first input and output addresses assigned to the module must be one of the following X and Y locations. The table also shows the V-memory addresses that correspond to these locations.

X	X0	X20	X40	X60	X100	X120	X140	X160
V	V40400	V40401	V40402	V40403	V40404	V40405	V40406	V40407

Y	Y0	Y20	Y40	Y60	Y100	Y120	Y140	Y160
V	V40500	V40501	V40502	V40503	V40504	V40505	V40506	V40507

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, check the guidelines before beginning the installation. Here are some general things to consider:

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. Do not ground the shield at both the module and the source.
- Do not run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

Loop Power Supply Requirements

The F2-4AD2DA module requires at least one field-side power supply. The same or separate power sources can be used for the module supply and the current transmitter supply. The F2-4AD2DA module requires 24VDC (at 80mA) and each current loop requires 20mA (a total of 120mA for six current loops), from the external power supply.

The DL205 AC bases have a built-in 24VDC power supply that provide up to 300mA of current. This can be used instead of a separate supply. Check the power budget to be safe.

It is desirable in some situations to power the transmitters separately in a location remote from the PLC. This will work as long as the transmitter supply meets the current requirements, and the transmitter’s minus (-) side and the module supply’s minus (-) side are connected together.

WARNING: If the internal 24VDC base power is used, be sure to calculate the power budget. Exceeding the power budget can cause unpredictable system operation that can lead to a risk of personal injury or equipment damage.

The DL205 base has a switching type power supply. As a result of switching noise, ±3–5 counts of instability may be noticed in the analog input data if the base power supply is used. If this is unacceptable, try one of the following:

- Use a separate linear power supply.
- Connect the 24VDC common to the frame ground, which is the screw terminal marked "G" on the base.

When using these methods, the input stability is rated at ±1 count.

Current Loop Transmitter Impedance

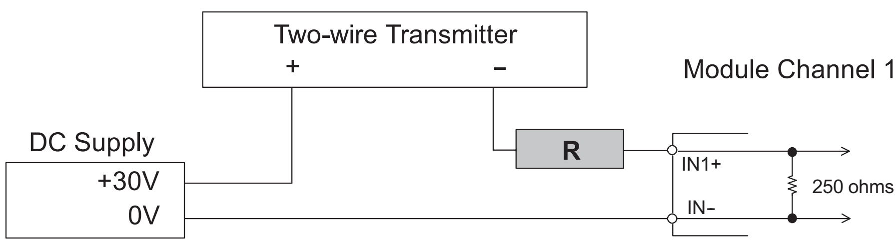
Standard 4–20 mA transmitters and transducers can operate from a wide variety of power supplies. Not all transmitters are alike and the manufacturers often specify a minimum loop or load resistance that must be used with the transmitter.

The F2-4AD2DA provides 250 ohm resistance for each input channel. If the transmitter being used requires a load resistance below 250 ohms, adjustments do not have to be made. However, if the transmitter requires a load resistance higher than 250 ohms, add a resistor in series with the module.

Consider the following example for a transmitter being operated from a 30VDC supply with a recommended load resistance of 750 ohms. Since the module has a 250 ohm resistor, add an additional resistor.

Example:

$R = Tr - Mr$		R – resistor to add
$R = 750 - 250$		Tr – Transmitter total resistance requirement
$R \geq 500$		Mr – Module resistance (internal 250 ohms)



In the example, add a 500 ohm resistor (R) in series with the module.

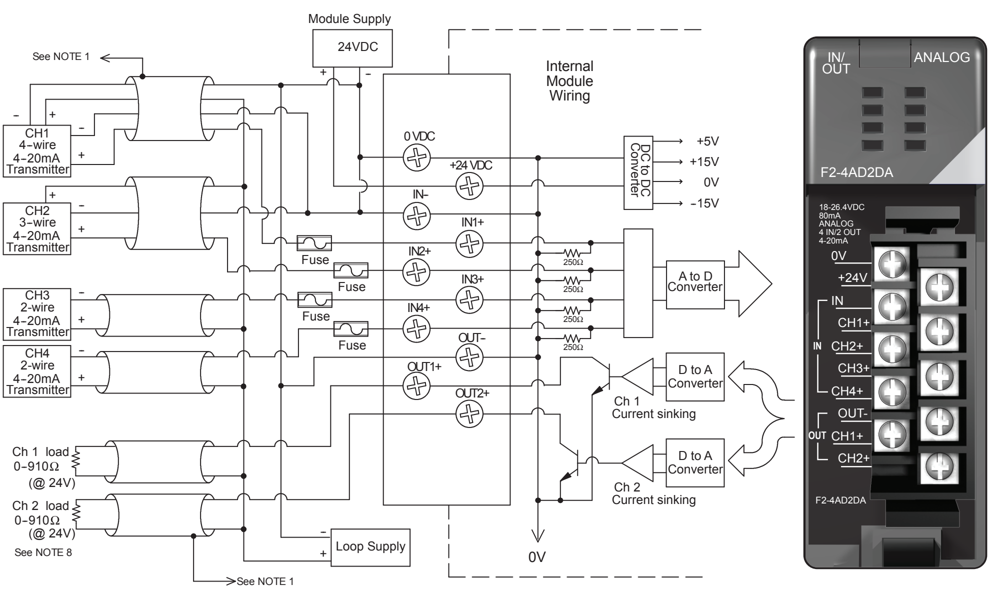
Wiring Diagram

The F2-4AD2DA module has a removable connector to simplify wiring. Simply squeeze the top and bottom retaining clips and gently pull the connector from the module. Use the following diagram to connect the field wiring. The diagram shows separate module and loop power supplies. If it is desired to use only one external supply, just combine the supply's positive (+) terminals into one node, and remove the loop supply.

NOTE 1: *Shields should be connected at their respective signal source.*
NOTE 2: *Unused channels should remain open (no connections) for minimum power consumption.*
NOTE 3: *More than one external power supply can be used provided all the power supply commons are connected together.*
NOTE 4: *A series 217, 0.032 A, fast-acting fuse is recommended for 4–20 mA current input loops.*
NOTE 5: *If the power supply common of an external power supply is not connected to 0V on the module, then the output of the external transmitter must be isolated. To avoid "ground loop" errors, recommended 4–20 mA transmitter types are:*

a. For 2 or 3 wire: Isolation between input signal and power supply.
b. For 4 wire: Isolation between input signal, power supply, and 4–20 mA output.

NOTE 6: *If an analog channel is connected backwards, then incorrect data values will be returned for that channel. Input signals in the ±4mA range return a zero value. Signals in the -4 to -40 mA range return a non-zero value.*
NOTE 7: *To avoid small errors due to terminal block losses, connect 0V, IN– and OUT– on the terminal block as shown. The module's internal connection of these nodes is not sufficient to permit module performance up to the accuracy specifications.*
NOTE 8: *Choose an output transducer resistance according to the maximum load / power supply listed in the Output Specifications table.*

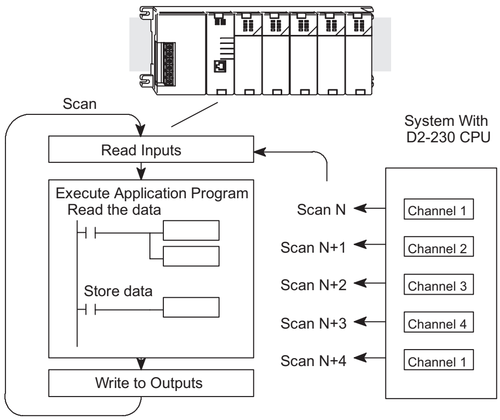


Module Operation

Before beginning to write the control program, it is important to take a few minutes to understand how the module processes the analog signals.

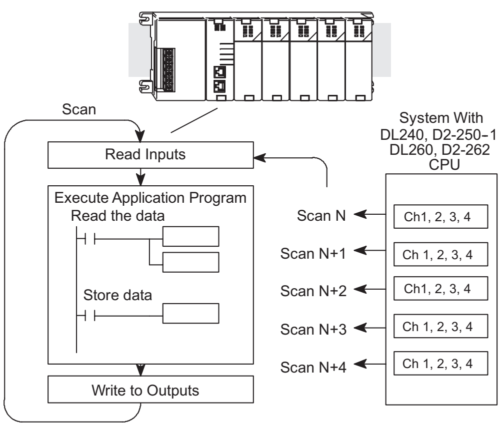
Input Channel Scanning Sequence (Multiplexing) for a D2-230 CPU

The F2-4AD2DA module can supply different amounts of data per scan, depending on the type of CPU being used. The D2-230 can obtain one channel of input data per CPU scan. Since there are four channels, it can take up to four scans to get the data for all channels. Once all channels have been scanned the process starts over with channel 1. Unused channels are not processed, so if only two channels are selected, each channel will be updated every other scan.



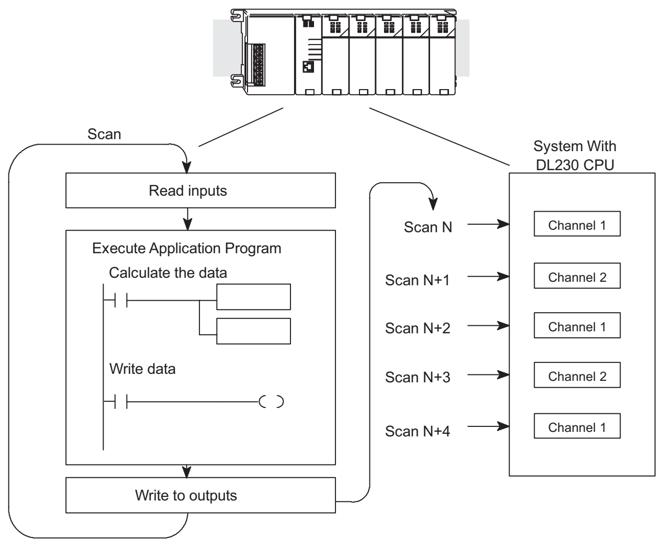
Input Channel Scanning Sequence (Pointer Method) for D2-240, D2-250-1, D2-260 and D2-262 CPUs

If a D2-240, a D2-250-1, a D2-260or a D2-262 CPU is being used, the input data for all four channels can be obtained in one scan. This is because the D2-240, D2-250-1, D2-260 and D2-262 CPUs supports special V-memory locations that are used to manage the data transfer. This is discussed in more detail in the section on Writing the Control Program later in this chapter.



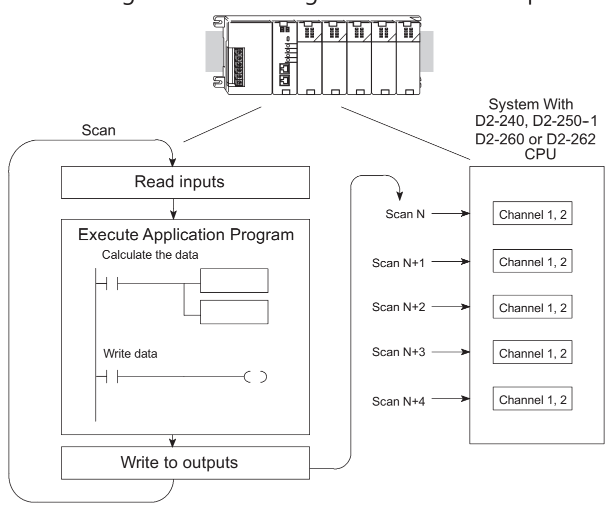
Output Channel Update Sequence (Multiplexing) for a D2-230 CPU

If a D2-230 CPU is used, only one channel of data can be sent to the output module on each scan. Since there are two channels, it can take two scans to update both channels. However, if only one channel is being used, then that channel can be updated on every scan.



Output Channel Update Sequence (Pointer Method) for D2-240, D2-250-1, D2-260 and D2-262 CPUs

If either a D2-240, D2-250-1, D2-260 or D2-262 CPU is used with the pointer method, both channels can be updated on every scan. This is because these CPUs support special V-memory locations that are used to manage the data transfer. This is discussed in more detail in the section on Writing the Control Program later in this chapter.



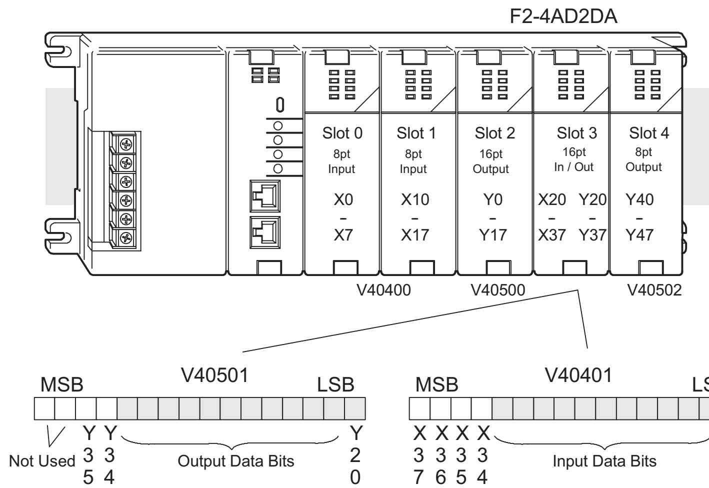
Understanding the I/O Assignments

Remember that the F2-4AD2DA requires 16 discrete input points and 16 discrete output points. These points can be used to obtain:

- An indication of which channel is active,
- The digital representation of the analog signal and,
- Module diagnostic information.

If a D2-240, D2-250-1, D2-260 or D2-262 CPU is being used, these bits may never have to be used, but it may be an aid to help understand the data format.

Since all I/O points are automatically mapped into V-memory, the location of the data words that will be assigned to the module can simply be determined.

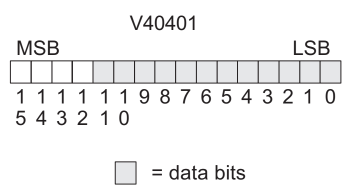


The individual bits in this data word location represent specific information about the analog signal.

Input Data Bits

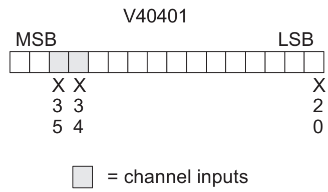
The first twelve bits of the input word represent the analog data in binary format.

Bit	**Value**	**Bit**	**Value**
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



Active Channel Indicator Bits

Two of the inputs are binary encoded to indicate the active input channel. Remember, the V-memory bits are mapped directly to discrete inputs. The module automatically turns these inputs On and Off to indicate the active input channel for each scan.

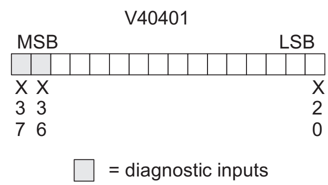


Scan	X36	X37	Channel
N	Off	Off	1
N+1	Off	On	2
N+2	On	Off	3
N+3	On	On	4
N+4	Off	Off	1

Diagnostic Indicator Inputs

The last two inputs are used for module diagnostics.

Module Busy – The first diagnostic input (X36 in this example) indicates a “busy” condition. This input will always be active on the first PLC scan to tell the CPU the analog data is not valid. After the first scan, the input will normally turn on when environmental (electrical) noise problems are present. The programming examples in the next section will show how this input can be used. The wiring guidelines presented earlier in this chapter provide steps that can help reduce noise problems.



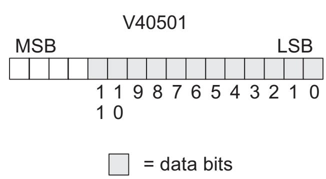
NOTE: *When using the pointer method, the value placed into the V-memory location will be 8000 instead of the bit being set.*

Module Failure – The last diagnostic input (X37 in this example) indicates that the analog module is not operating. For example, if the 24VDC input power is missing, or if the terminal block is loose, then the module will turn on this input point. The module will also return a data value of zero to further indicate there is a problem. This input point cannot detect which individual channel is at fault. If the cause of the failure goes away, the module turns this bit off.

Output Data Bits

The first twelve bits of the output word represent the analog data in binary format.

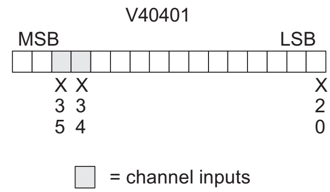
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



Output Channel Selection Bits

Two of the outputs select the active channel. Remember, the V-memory bits are mapped directly to discrete outputs. Turning a bit Off selects its channel. By controlling these outputs, the channel(s) to be updated can be selected.

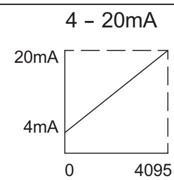
Y35	Y34	Channel
On	Off	1
Off	On	2
Off	Off	1 & 2 (same data to both channels)
On	On	None (both channels hold current values)



Module Resolution

Since the module has 12-bit resolution, the analog signal is converted into 4096 counts ranging from 0–4095 (2^{12}). For example, a 4mA signal would be 0, and a 20mA signal would be 4095. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The diagram shows how this relates to the signal range.

Each count can also be expressed in terms of the signal level by using the equation shown.



$$\text{Resolution} = \frac{H - L}{4095}$$

H = high limit of the signal range

L = low limit of the signal range

16mA / 4095 = 3.907µA per count

Writing the Control Program

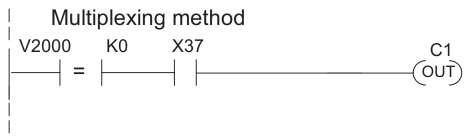
Before starting to write the program, some supplemental examples can be very helpful to the programmer, such as:

- Input power failure detection
- Output data calculation
- Input data scaling

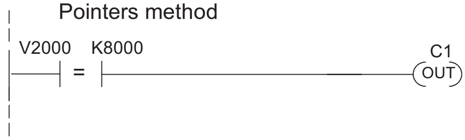
Analog Input Failure Detection

The analog module has a microcontroller that can diagnose analog input circuit problems. Ladder logic can be written to detect these problems. The following rung shows an input point that would be assigned if the module was used as shown in the previous and following examples.

Calculating the Digital Value



V-memory location V2000 holds channel 1 data. When a data value of zero is returned and input X37 is on, then the analog channel is not operating properly.



V-memory location V2000 holds channel 1 data. When a data value of 8000 is returned, then the analog channel is not operating properly.

The control program must calculate the digital value that is sent to the analog output. Several methods can be used to do this, but the best method is to convert the values to engineering units. This is accomplished by using the formula shown.

Adjustments may need to be made to the formula depending on the scale of the engineering units.

Consider the following example which controls pressure from 0.0–99.9 PSI. Using the formula will calculate the digital value to be sent to the analog output. The example shows the conversion required to yield 49.4 PSI. The multiplier of 10 is used because the decimal portion of 49.4 cannot be loaded in the program, so it is shifted right one decimal place to make a usable value of 494.

$$A = U \frac{4095}{H-L}$$

A = Analog Value (0–4095)

U = Engineering Units

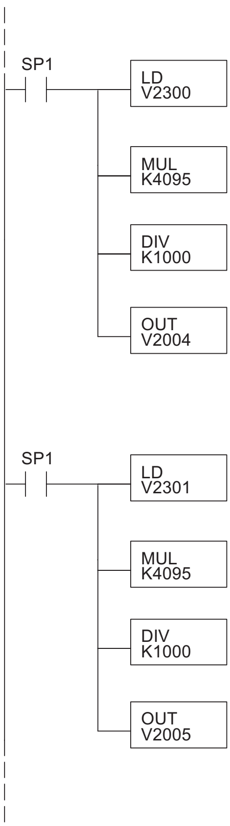
H = High limit of the engineering unit range

L = Low limit of the engineering unit range

$$A = 10U \frac{4095}{10(H-L)} \qquad A = 494 \frac{4095}{(1000-0)} \qquad A = 2023$$

The example program below shows how to write the program to perform engineering unit conversions. This example will work with all CPUs and assumes that the engineering unit values have been calculated or loaded and stored in V2300 and V2301 for channels 1 and 2 respectively. Also, the final values are moved to V2004 and V2005, which are memory locations that are used in the following examples. Any user V-memory locations can be used, but they must match the locations that are specified as the source for the output data (see the next section for an example).

NOTE: *Since the D2-250 can do math operations in BCD format, it is better to perform the math calculations in BCD.*



The LD instruction loads the engineering units used with channel 1 into the accumulator. This example assumes the numbers are BCD. Since SP1 is used, this rung automatically executes on every scan. You could also use an X, C, etc. permissive contact.

Multiply the accumulator by 4095 (to start the conversion).

Divide the accumulator by 1000 (because we used a multiplier of 10, we have to use 1000 instead of 100).

Store the BCD result in V2004 (the actual steps required to send the data are shown later).

The LD instruction loads the engineering units used with channel 2 into the accumulator. This example assumes the numbers are BCD. Since SP1 is used, this rung automatically executes on every scan. You could also use an X, C, etc. permissive contact.

Multiply the accumulator by 4095 (to start the conversion).

Divide the accumulator by 1000 (because we used a multiplier of 10, we have to use 1000 instead of 100).

Store the BCD result in V2005 (the actual steps required to send the data are shown later).

Scaling the Input Data

Most applications usually require measurements in engineering units, which provide more meaningful data. This is accomplished by using the conversion formula shown.

Adjustments to the formula may be needed depending on the scale chosen for the engineering units.

For example, if pressure (PSI) is to be measured with a scale of 0.0–99.9, a multiplication factor of 10 would be needed in order to imply a decimal place when the value is used in the user program.

$$\text{Units} = A \frac{H-L}{4095}$$

U = Engineering Units

A = Analog Value (0–4095)

H = High limit of the engineering unit range

L = Low limit of the engineering unit range

Example without multiplier

$$\text{Units} = A \frac{H-L}{4095}$$

$$\text{Units} = 2024 \frac{100-0}{4095}$$

$$\text{Units} = 49$$

Example with multiplier

$$\text{Units} = 10A \frac{H-L}{4095}$$

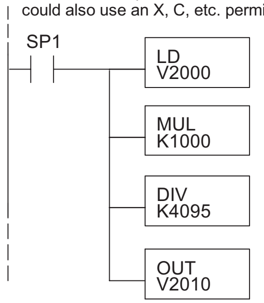
$$\text{Units} = 20240 \frac{100-0}{4095}$$

$$\text{Units} = 494$$

Analog Value of 2024, slightly less than half scale, should yield 49.4 PSI.

The following rung of logic is an example showing how the program can be written to perform the engineering unit conversion. This example assumes the data is in BCD format before being loaded into the appropriate V-memory locations using instructions that apply to the CPU module being used.

Note, this example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.



Load channel 1 data to the accumulator.

Multiply the accumulator by 1000 (to start the conversion).

Divide the accumulator by 4095.

Store the result in V2010.

Read / Write Program (Pointer Method) for D2-240, D2-250-1, D2-260 and D2-262 CPUs

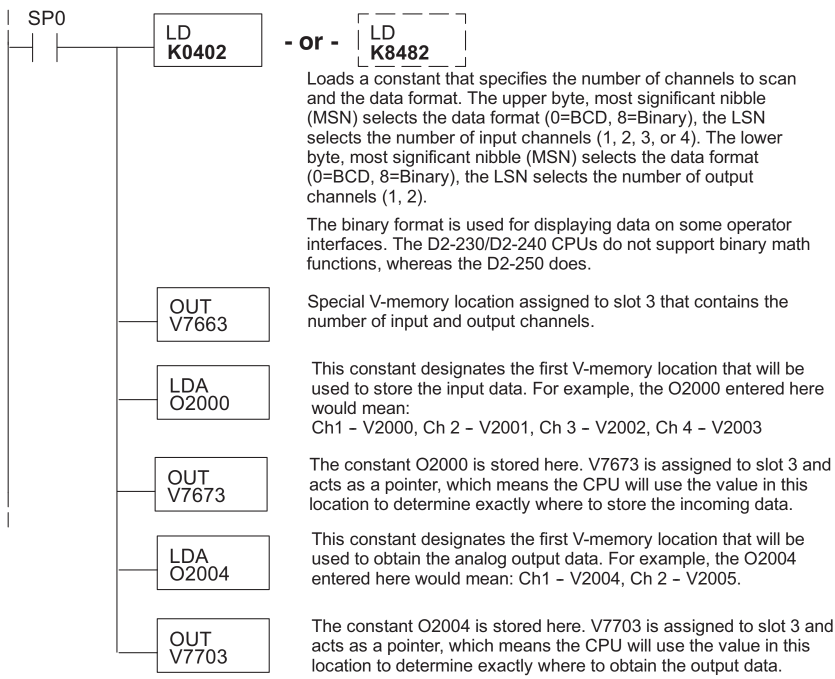
The D2-240, D2-250-1, D2-260 and D2-262 CPUs have special V-memory locations assigned to each base slot that simplifies the programming requirements. The V-memory locations:

- Specify the number of input and output channels to scan.
- Specify the storage location for the input data.
- Specify the source location for the output data.

NOTE: *In order to use the pointer method, D2-250 CPUs must have firmware revision 1.09 or later, and F2-4AD2DA modules must be revision C1 or later*

The following example rung of logic shows how to setup these locations. Place this rung anywhere in the ladder program, or in the initial stage if stage programming instructions are being used.

In this example V2000 and V2004 are used to store the calculated values, but any V-memory location can be used. For this example, the analog module is installed in slot 3. Be sure to use the V-memory locations for which ever slot the module is placed in your system. The pointer method automatically converts values to binary.



The following tables show the special V-memory locations used by the D2-240, D2-250-1, D2-260 and D2-262 for the CPU base and local expansion base I/O slots. Slot 0 (zero) is the module next to the CPU or D2-CM module. Slot 1 is the module two places from the CPU or D2-CM, and so on. Remember, the CPU only examines the pointer values at these locations after a mode transition. Also, if the D2-230 (multiplexing) method is used, verify that these addresses in the CPU are 0 (zero).

The table below applies to the D2-240, D2-250-1, D2-260 and D2-262 CPU base.

CPU Base: Analog In/Out Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V7660	V7661	V7662	V7663	V7664	V7665	V7666	V7667
Input Pointer	V7670	V7671	V7672	V7673	V7674	V7675	V7676	V7677
Output Pointer	V7700	V7701	V7702	V7703	V7704	V7705	V7706	V7707

The table below applies to the D2-250-1, D2-260 or D2-262 CPU base 1.

Expansion Base D2-CM #1: Analog In/Out Module Slot-Dependent V-memory Locations								
Slot.	0	1	2	3	4	5	6	7
No. of Channels	V36000	V36001	V36002	V36003	V36004	V36005	V36006	V36007
Input Pointer	V36010	V36011	V36012	V36013	V36014	V36015	V36016	V36017
Output Pointer	V36020	V36021	V36022	V36023	V36024	V36025	V36026	V36027

The table below applies to the D2-250-1, D2-260 or D2-262 CPU base 2.

Expansion Base D2-CM #2: Analog In/Out Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36100	V36101	V36102	V36103	V36104	V36105	V36106	V36107
Input Pointer	V36110	V36111	V36112	V36113	V36114	V36115	V36116	V36117
Output Pointer	V36120	V36121	V36122	V36123	V36124	V36125	V36126	V36127

The table below applies to the D2-260 and D2-262 CPU base 3.

Expansion Base D2-CM #3: Analog In/Out Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36200	V36201	V36202	V36203	V36204	V36205	V36206	V36207
Input Pointer	V36210	V36211	V36212	V36213	V36214	V36215	V36216	V36217
Output Pointer	V36220	V36221	V36222	V36223	V36224	V36225	V36226	V36227

The table below applies to the D2-260 and D2-262 CPU base 4.

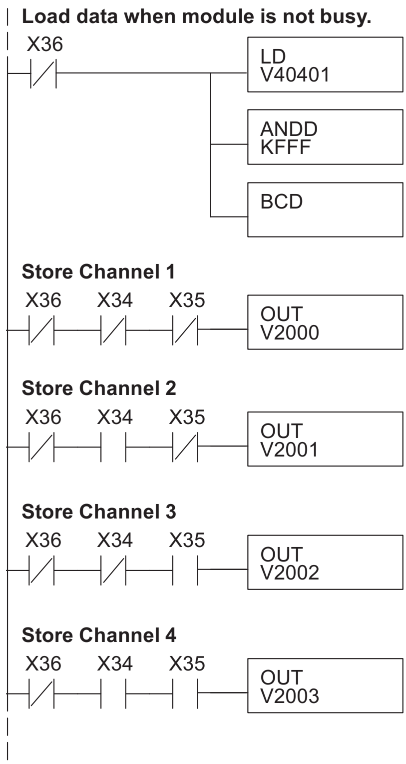
Expansion Base D2-CM #4: Analog In/Out Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36300	V36301	V36302	V36303	V36304	V36305	V36306	V36307
Input Pointer	V36310	V36311	V36312	V36313	V36314	V36315	V36316	V36317
Output Pointer	V36320	V36321	V36322	V36323	V36324	V36325	V36326	V36327

Read Input Values (Multiplexing)

The D2-230 CPU does not use special V-memory locations for transferring data. Since all channels are multiplexed into a single data word, the control program must be setup to determine which channel is being read. Since the module appears as X input points to the CPU, simply use the active channel status bits to determine which channel is being read.

Note, this example is for a module installed in slot 3, as shown in the previous examples. The addresses used would be different if the module was used in a different slot. These rungs can be placed anywhere in the program or if stage programming is being used, place them in a stage that is always active.

The following multiplexing example can be used with all of the DL205 CPUs.



Loads the complete data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map.

This instruction masks the channel identification bits. Without this, the values used will not be correct so do not forget to include it.

It is usually easier to perform math operations in BCD, You can leave out this instruction if your application does not require it.

When the module is not busy and X36, X34 and X35 are off, channel 1 data is stored in V2000.

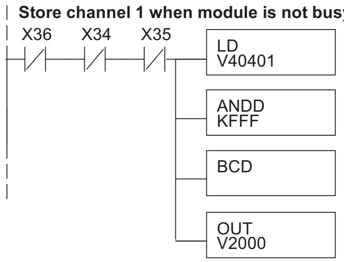
When the module is not busy and X34 is on and X35 and X36 are off, channel 2 data is stored in V2001.

When the module is not busy and X34 and X36 are off and X35 is on, channel 3 data is stored in V2002.

When the module is not busy and both X34 and X35 are on and X36 is off, channel 4 data is stored in V2003.

Single Input Channel Selected (Multiplexing)

Since it isn't necessary to determine which channel is selected, the single channel example shown below can be implemented in the user program.



Loads the complete data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map.

This instruction masks the channel identification bits. Without this, the values used will not be correct so do not forget to include it.

It is usually easier to perform math operations in BCD. You can leave out this instruction if your application does not require it.

When the module is not busy and X34 and X35 are off, channel 1 data is stored in V2000.

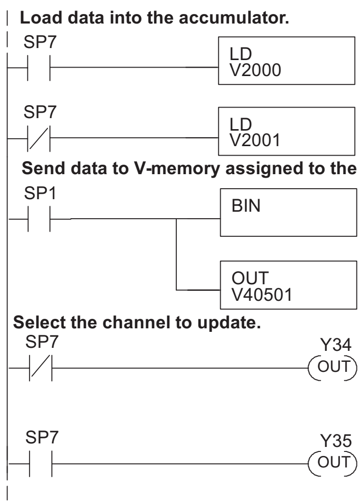
Write Output Values (Multiplexing)

Since all channels are multiplexed into a single data word, the control program can be setup to determine which channel to write the data to. Since the module appears as Y output points to the CPU, it is simple to use the channel selection outputs to determine which channel to update.

Note, this example is for a module installed in slot 3, as shown in the previous examples. The addresses used would be different if the module was used in a different slot. These rungs can be placed anywhere in the program or if stage programming is being used, place them in a stage that is always active.

This example is a two-channel multiplexer that updates each channel on alternate scans. Relay SP7 is a special relay that is On for one scan, then Off for one scan. This multiplexing example can be used with all of the DL205 CPUs.

NOTE: *Binary data must be sent to the output module. If the data is already in binary format, do not use the BIN instruction shown in this example.*



Loads the data for channel 1 into the accumulator.

Loads the data for channel 2 into the accumulator.

Convert the data to binary (you must omit this step if you have converted the data elsewhere). SP1 is always on.

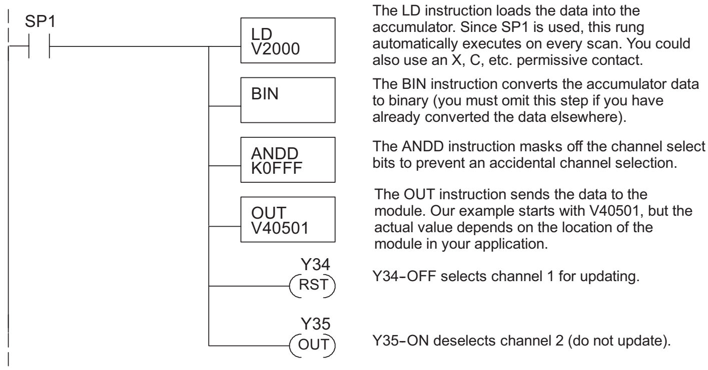
The OUT instruction sends the data to the module. Our example starts with V40501, but the actual value depends on the location of the module in your application.

Selects channel 1 for update when Y34 is OFF (Y35–ON deselects channel 2). Note, Y34 and Y35 are used due to the previous examples. If the module was installed in a different I/O arrangement, the addresses would be different.

Selects channel 2 for update when Y35 is OFF (Y34–ON deselects channel 1). Note, Y34 and Y35 are used due to the previous examples. If the module was installed in a different I/O arrangement, addresses would be different.

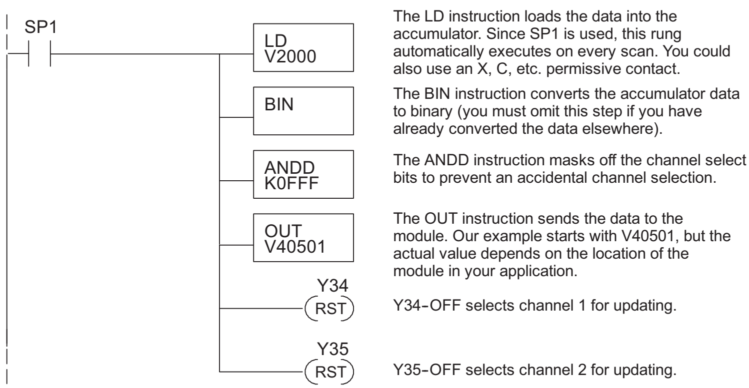
Write Data to One Channel

If only one channel is being used, or if the updates are to be controlled separately, the following logic can be used.



Write the same Data to Both Channels

If both channel select outputs are Off, then both channels will be updated with the same data.



Analog and Digital Value Conversions

It is sometimes useful to do quick conversions between the signal levels and the digital values. This can be helpful during startup and/or troubleshooting. The following table shows some formulas to help with the conversions.

Range	If the digital value is known	If the analog signal level is known.
4–20 mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

For example, if a 10mA signal level is needed, use the formula to the right to determine the digital value "D" to be stored in the V-memory location which is designated to store the data.

$$D = \frac{4095}{16}(A - 4)$$

$$D = \frac{4095}{16}(10\text{mA} - 4)$$

$$D = (255.93)(6) \qquad D = 1536$$

Filtering Input Noise (D2-250-1, D2-260 and D2-262 CPUs Only)

Add the following logic to filter and smooth analog input noise in D2-250-1, D2-260 or D2-262 CPUs. This is especially useful when using PID loops. Noise can be generated by the field device and/or induced by field wiring.

In the following example, the analog value in BCD is first converted to a binary number. Memory location V1400 is the designated workspace in this example. The MULR instruction is the filter factor, which can be from 0.1–0.9. The example uses 0.2. A smaller filter factor increases filtering. A higher precision value can be used, but it is not generally needed. The filtered value is then converted back to binary and then to BCD. The filtered value is stored in location V1402 for use in your application or PID loop.

NOTE: *Be careful not to do a multiple number conversion on a value. For example, if the pointer method is used to get the analog value, it is in BCD and must be converted to binary. However, if the conventional method is used to read a value and the first twelve bits are masked, then it is already in binary and no conversion using the BIN instruction is needed.*

NOTE*: Please review intelligent instructions (IBox) in Chapter 5, which simplify this and other functions. The IBox instructions are supported by the D2-250-1, D2-260 and D2-262.*