

F2-04AD-1, F2-04AD-1L 4-Channel Analog Current Input

2

In This Chapter. . . .

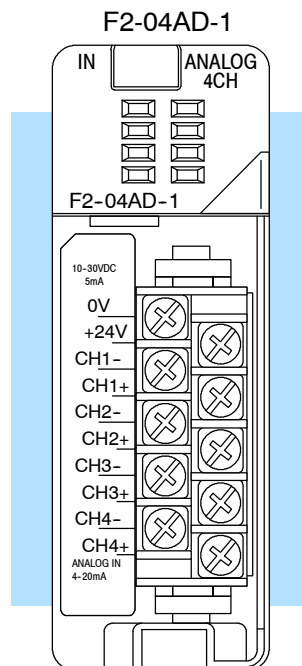
- Module Specifications
- Setting the Module Jumpers
- Connecting the Field Wiring
- Module Operation
- Writing the Control Program

Module Specifications

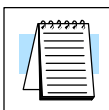
F2-04AD-1

The F2-04AD-1 analog Input module provides several hardware features.

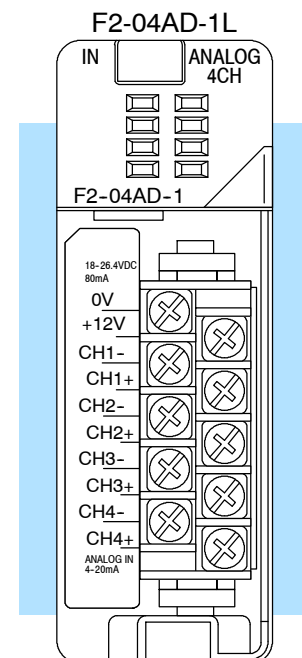
- On-board 250 ohm, 1/2 watt precision resistors provide substantial over-current-protection for 4–20mA current loops.
- Analog inputs are optically isolated from the PLC logic.
- The module has a removable terminal block so the module can be easily removed or changed without disconnecting the wiring.
- With a DL240/250–1/260 CPU, you can read all four channels in one scan.
- On-board active analog filtering and RISC-like microcontroller provide digital signal processing to maintain precision analog measurements in noisy environments.



F2-04AD-1L (Obsolete)



NOTE: In 2009 the F2-04AD-1L was discontinued. A re-designed F2-04AD-1 was released at the same time which can be powered by either 12 VDC or 24VDC input power supplies. This new module is a direct replacement for prior F2-04AD-1 and all F2-04AD-1L modules. The new module is a single circuit board design and the jumper link locations are different. See *Setting the Module Jumpers* on page 2-5. Also, some specifications were changed on page 2-3. Otherwise, the re-designed module functions the same as the prior designs.



Input Specifications

These tables provide specifications for both the F2-04AD-1 and F2-04AD-1L Analog Input Modules (all specifications are the same for both modules except for the input voltage requirements). Review these specifications to make sure the module meets your application requirements.

Number of Channels	4, single ended (one common)
Input Range	4 to 20 mA current
Resolution	12 bit (1 in 4096)
Step Response	4.9 ms (*4.0 ms) to 95% of full step change
Crosstalk	-80 dB, 1/2 count maximum
Active Low-pass Filtering	-3 dB at 120Hz (*80Hz), 2 poles (-12 dB per octave)
Input Impedance	250Ω ± 0.1%, ½W current input
Absolute Maximum Ratings	-40 mA to +40 mA, current input
Converter type	Successive approximation
Linearity Error (End to End)	± 1 count (0.025% of full scale) maximum
Input Stability	± 1 count
Full Scale Calibration Error (Offset error not included)	± 12 counts maximum, @ 20mA current input
Offset Calibration Error	± 7 counts maximum, @ 4mA current input
Maximum Inaccuracy	± .5% @ 25°C (77°F) ± .65% 0 to 60°C (32 to 140°F)
Accuracy vs. Temperature	± 50 ppm/°C maximum full scale calibration (including maximum offset change)
Recommended Fuse (external)	0.032 A, Series 217 fast-acting, current inputs

One count in the specification table is equal to one least significant bit of the analog data value (1 in 4096).

PLC Update Rate	1 channel per scan maximum (DL230 CPU) 4 channels per scan maximum (DL240/250-1/260 CPU)
Digital Inputs Input points required	12 binary data bits, 2 channel ID bits, 2 diagnostic bits 16 point (X) input module
Power Budget Requirement	100 mA (*50 mA) maximum, 5 VDC (supplied by base)
External Power Supply	5mA (*80mA) max., 10 (*18) to 30 VDC (F2-04AD-1) 90mA maximum, 10 to 15 VDC (F2-04AD-1L)
Operating Temperature	0 to 60° C (32 to 140° F)
Storage Temperature	-20 to 70° C (-4 to 158° F)
Relative Humidity	5 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

* Values in parenthesis with an asterisk are for older modules with two circuit board design and date codes 0609F3 or previous. Values not in parenthesis are for single circuit board models with date code 0709G or above.

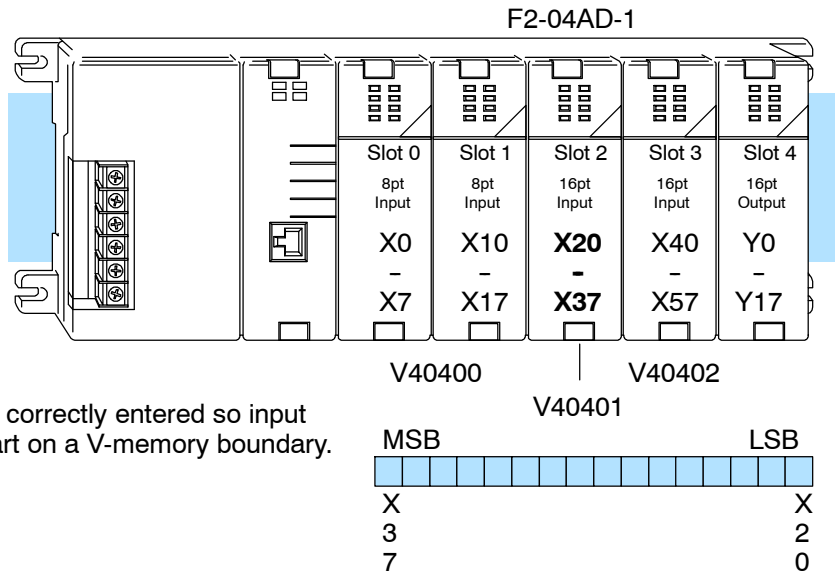
Analog Input Configuration Requirements

Appears as a 16-point discrete input module and can be installed in any slot of a DL205 system. The available power budget and discrete I/O points are the limiting factors. Check the user manual for your particular model of CPU and I/O base for more information regarding power budget and number of local, local expansion or remote I/O points.

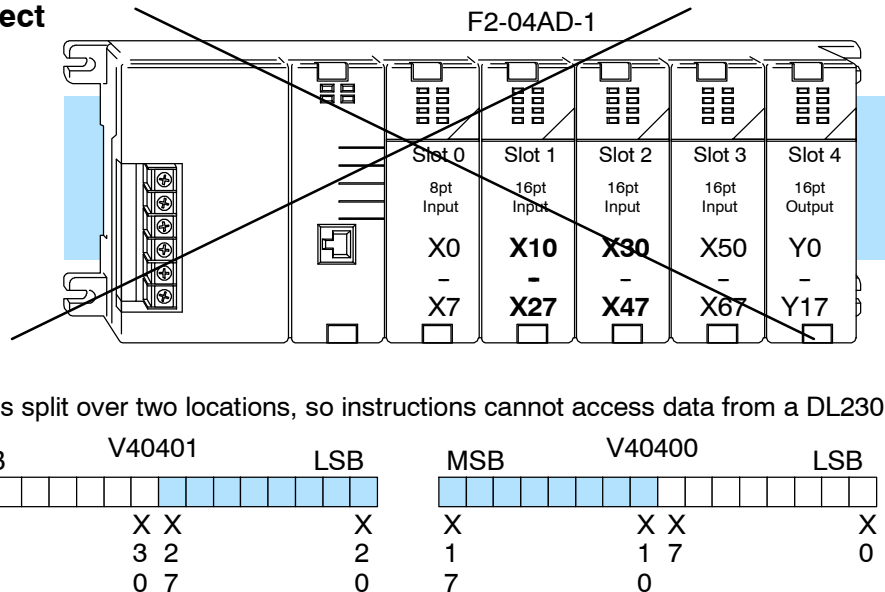
Special Placement Requirements (DL230 and Remote I/O Bases)

Even though the module can be placed in any slot, it is important to examine the configuration if you are using a DL230 CPU. As you can see in the section on writing the program, you use V-memory locations to extract the analog data. If you place the module so that the input points do not start on a V-memory boundary, the instructions cannot access the data. This also applies when placing this module in a remote base using a D2-RSSS in the CPU slot.

Correct!



Incorrect



To use the V-memory references required for a DL230 CPU, the *first* input address assigned to the module must be one of the following X locations. The table also shows the V-memory addresses that correspond to these X locations.

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

Setting the Module Jumpers

Selecting the Number of Channels

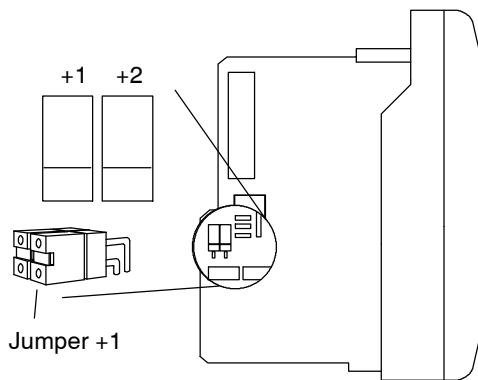
There are two jumpers, labeled +1 and +2, that are used to select the number of channels that will be used. See the figures below to find the jumpers on your module. The module is set from the factory for four channel operation.

Any unused channels are not processed, so if you only select channels 1 thru 3, channel 4 will not be active. The following table shows how to use the jumpers to select the number of channels.

No. of Channels	+1	+2
1	No	No
1, 2	Yes	No
1, 2, 3	No	Yes
1, 2, 3, 4	Yes	Yes

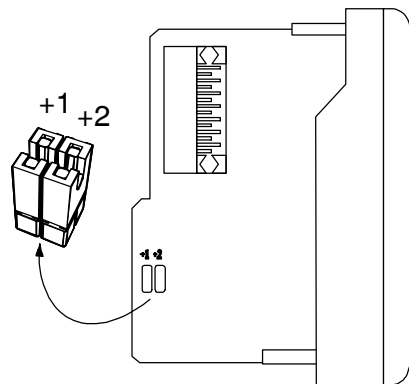
For example, to select all 4 channels (1 – 4), leave both jumpers installed. To select channel 1, remove both jumpers.

Jumper Location on Modules Having Date Code 0609F3 and Previous (Two Circuit Board Design)



These jumpers are located on the motherboard, the one *with* the black D-shell style backplane connector.

Jumper Location on Modules Having Date Code 0709G and Above (Single Circuit Board Design)



Connecting the Field Wiring

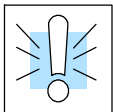
Wiring Guidelines Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider:

- 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.

User Power Supply Requirements

The module requires at least one field-side power supply. You may use the same or separate power sources for the module supply and the current transmitter supply. The F2-04AD-1 module requires 18–30VDC, at 80 mA. The DL205 bases have built-in 24 VDC power supplies that provide up to 300mA of current. You may use this with F2-04AD-1 modules instead of a separate supply if you are using only a couple of analog modules.

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 voltage and current requirements, and the transmitter's minus (–) side and the module supply's minus (–) side are connected together.



WARNING: If you are using the 24 VDC base power supply, make sure you calculate the power budget. Exceeding the power budget can cause unpredictable system operation that can lead to a risk of personal injury or damage to equipment.

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

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

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

The F2-04AD-1L module requires 10–15VDC, at 90 mA and must be powered by a separate power supply.

Current Loop Transmitter Impedance

Standard 4 to 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 F2-04AD-1, (L) provides 250 ohm resistance for each channel. If your transmitter requires a load resistance below 250 ohms, you do not have to make any adjustments. However, if your transmitter requires a load resistance higher than 250 ohms, you need to add a resistor in series with the module.

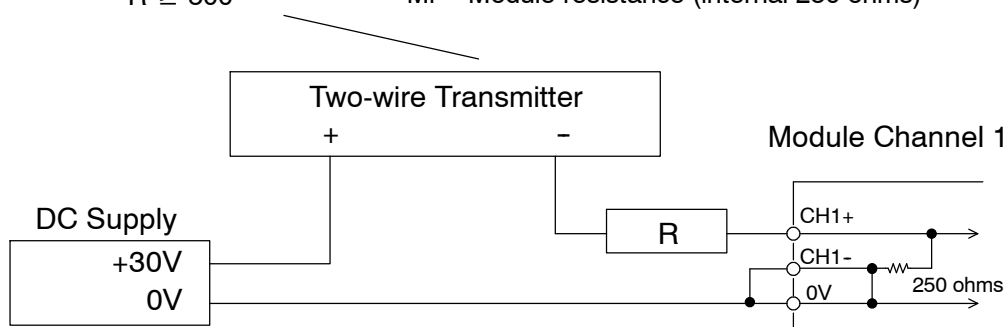
Consider the following example for a transmitter being operated from a 30 VDC supply with a recommended load resistance of 750 ohms. Since the module has a 250 ohm resistor, you need to add an additional resistor.

$$R = Tr - Mr$$

$$R = 750 - 250$$

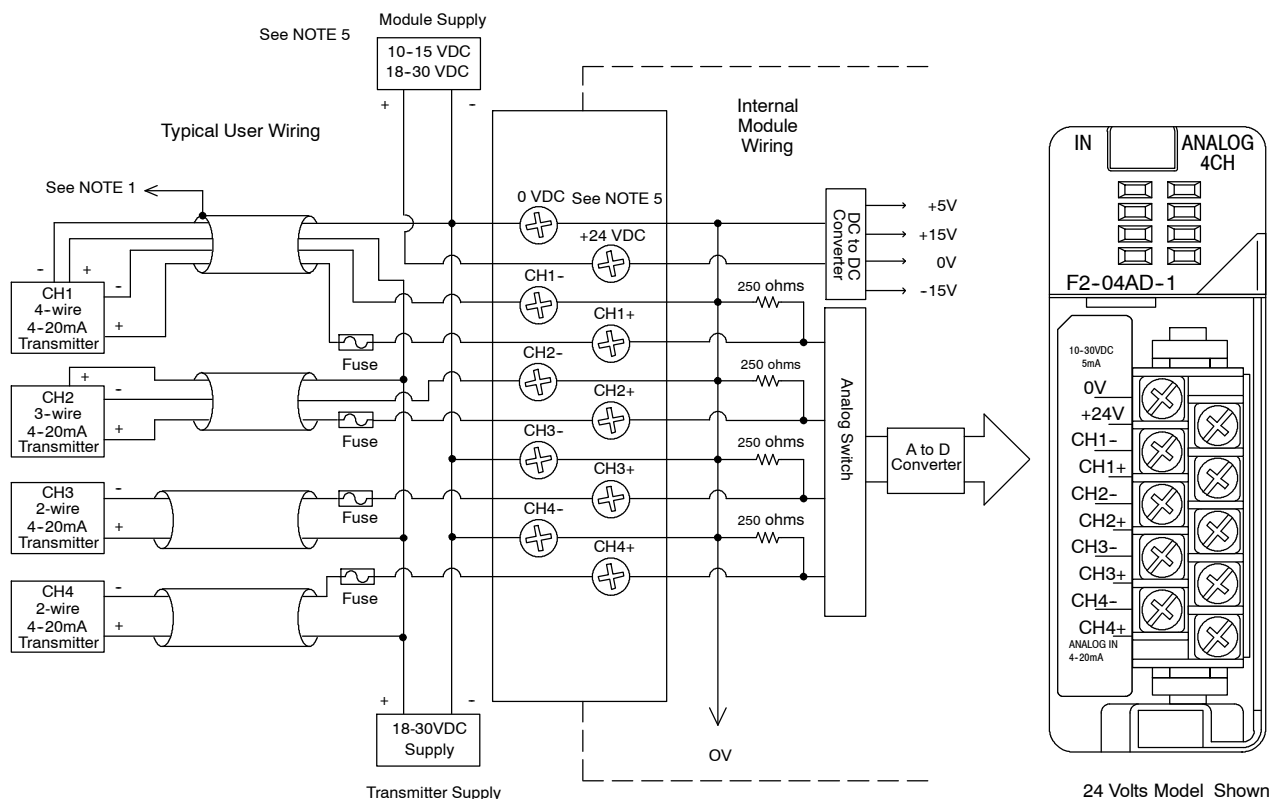
$$R \geq 500$$

R - resistor to add
 Tr - Transmitter Requirement
 Mr - Module resistance (internal 250 ohms)



Wiring Diagram

The F2-04AD-1, (L) module has a removable connector to make wiring easier. 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 transmitter power supplies. If you desire to use only one field-side supply, just combine the supplies' positive (+) terminals into one node, and remove the transmitter supply.



NOTE 1: Shields should be grounded at the signal source.

NOTE 2: More than one external power supply can be used, provided all the power supply commons are connected.

NOTE 3: A Series 217, 0.032A fast-acting fuse is recommended for 4-20 mA current loops.

NOTE 4: If the power supply common of an external power supply is not connected to 0VDC 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:

2 or 3 wire: Isolation between input signal and power supply.

4 wire: Isolation between input signal, power supply, and 4-20mA output.

NOTE 5: Use 10-15VDC for F2-04AD-1L

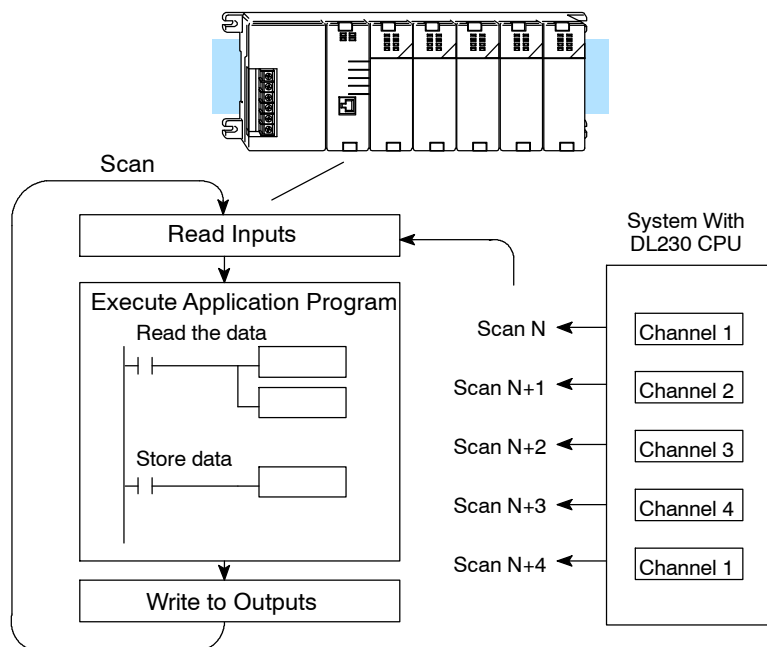
Use 18-30VDC for F2-04AD-1

Module Operation

Channel Scanning Sequence for a DL230 CPU (Multiplexing)

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

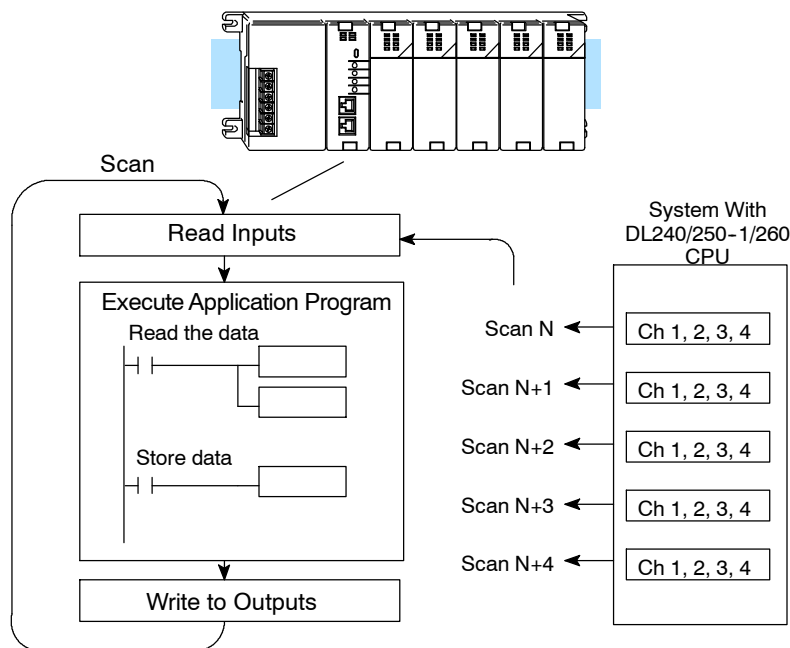
The module can supply different amounts of data per scan, depending on the type of CPU you are using. The DL230 can obtain one channel of data per CPU scan. Since there are four channels, it can take up to four scans to get data for all channels. Once all channels have been scanned the process starts over with channel 1. Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan. The multiplexing method can also be used for the DL240/250-1 and DL260 CPUs.



F2-04AD-1, (L)
4-Ch. Current Input

Channel Scanning Sequence for a DL240, DL250-1 or or DL260 CPU (Pointer Method)

If you are using a DL240/250-1/260 CPU, you can obtain all four channels of input data in one scan. This is because the DL240/250-1/260 CPU 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.



Analog Module Updates

Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

For the vast majority of applications, the values are updated much faster than the signal changes. However, in some applications, the update time can be important. The module takes approximately 4 milliseconds to sense 95% of the change in the analog signal.

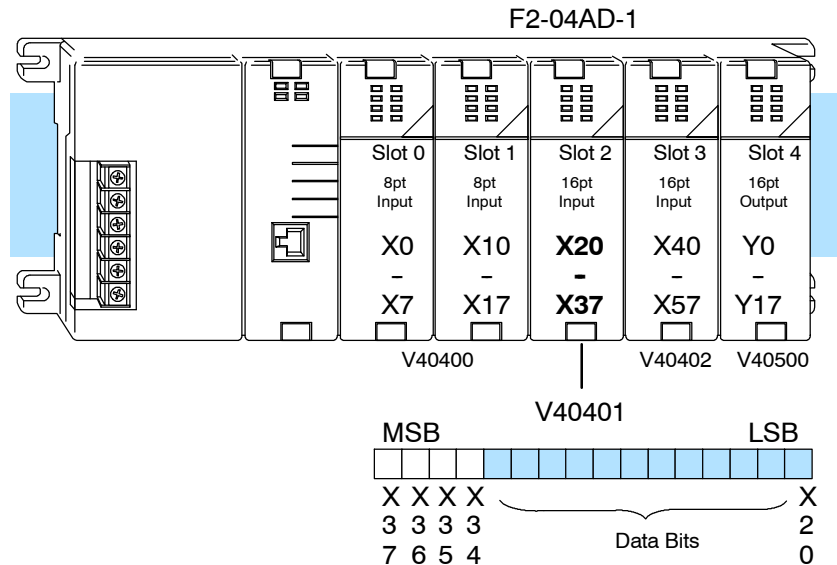
Note, this is *not* the amount of time required to convert the signal to a digital representation. The conversion to the digital representation takes only a few microseconds. Many manufacturers list the conversion time, but it is the settling time of the filter that really determines the update time.

Understanding the Input Assignments

You may recall the F2-04AD-1, (L) module requires 16 discrete input points in the CPU. You can use these points to obtain:

- an indication of which channel is active
- the digital representation of the analog signal
- module diagnostic information

Since all input points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.



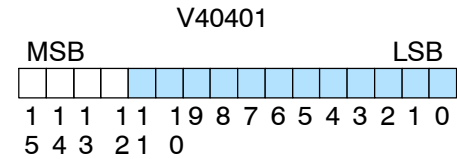
F2-04AD-1, (L)
4-Ch. Current Input

Within these word locations, the individual bits represent specific information about the analog signal.

Analog Data Bits

The first twelve bits 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

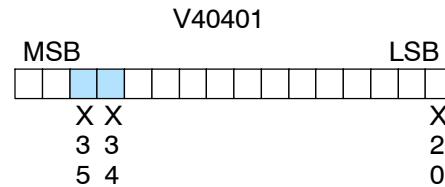


 = data bits

Active Channel Indicator Inputs

Two of the inputs are binary-encoded to indicate the active channel (remember, the V-memory bits are mapped directly to discrete inputs). The inputs are automatically turned on and off to indicate the active channel for each scan.

Scan	X35	X34	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

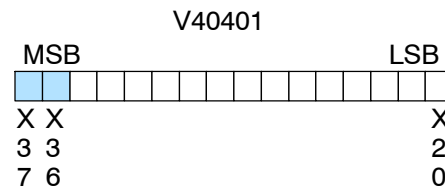


□ = channel inputs

Module Diagnostic 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 usually only comes on when extreme environmental (electrical) noise problems are present. The programming examples in the next section shows how you can use this input. The wiring guidelines shown earlier in this chapter provide steps that can help reduce noise problems.



□ = diagnostic inputs

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

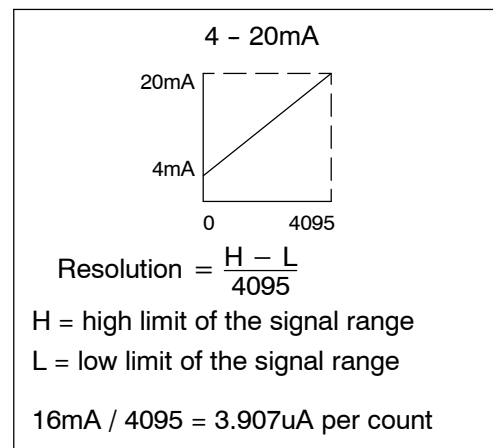
Channel Failure — The last diagnostic input (X37 in this example) indicates the analog channel is not operating. For example, if the 24 VDC input power is missing or if the terminal block is loose, the module will turn on this input point. The module also returns a data value of zero to further indicate there is a problem.

The next section, Writing the Control Program, shows how you can use these inputs in your control program.

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.



Writing the Control Program

Reading Values: Pointer Method and Multiplexing

There are two methods of reading values:

- The pointer method
- Multiplexing

You *must* use the multiplexing method when using a DL230 CPU. You must also use the multiplexing method with remote I/O modules (the pointer method will not work). You can use either method when using DL240, DL250-1 and DL260 CPUs, but for ease of programming it is strongly recommended that you use the pointer method.

The DL205 series has special V-memory locations assigned to each base slot that greatly simplify the programming requirements. These V-memory locations allow you to:

- specify the data format
- specify the number of channels to scan
- specify the storage locations

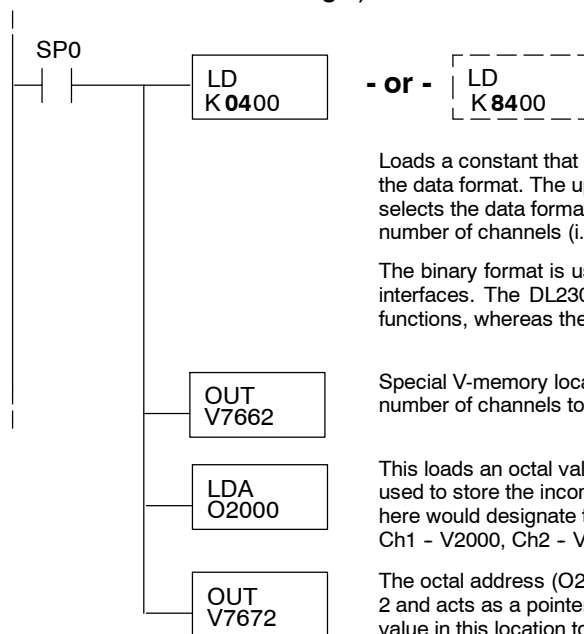
Pointer Method

✗	✓	✓	✓
230	240	250-1	260



NOTE: DL250 CPUs with firmware release version 1.06 or later support this method. If you must use the DL230 example, module placement in the base is very important. Review the section earlier in this chapter for guidelines.

The example program shows how to setup these locations. Place this rung anywhere in the ladder program, or in the initial stage if you are using stage programming instructions. This is all that is required to read the data into V-memory locations. Once the data is in V-memory you can perform math on the data, compare the data against preset values, and so forth. V2000 is used in the example but you can use any user V-memory location. In this example the module is installed in slot 2. You should use the V-memory locations for your module placement. The pointer method automatically converts values to BCD (depending on the LD statement in the ladder logic).



- or -

Loads a constant that specifies the number of channels to scan and the data format. The upper byte, most significant nibble (MSN) selects the data format (i.e. 0=BCD, 8=Binary), the LSN selects the number of channels (i.e. 1, 2, 3, or 4).

The binary format is used for displaying data on some operator interfaces. The DL230/240 CPUs do not support binary math functions, whereas the DL250 does.

Special V-memory location assigned to slot 2 that contains the number of channels to scan.

This loads an octal value for the first V-memory location that will be used to store the incoming data. For example, the O2000 entered here would designate the following addresses.
Ch1 - V2000, Ch2 - V2001, Ch3 - V2002, Ch 4 - V2003

The octal address (O2000) is stored here. V7672 is assigned to slot 2 and acts as a pointer, which means the CPU will use the octal value in this location to determine exactly where to store the incoming data.

The tables below show the special V-memory locations used by the DL240, DL250-1 and DL260 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 you use the DL230 (multiplexing) method, verify that these addresses in the CPU are zero.

The Table below applies to the DL240, DL250-1 and DL260 CPU base.

CPU Base: Analog Input 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
Storage Pointer	V7670	V7671	V7672	V7673	V7674	V7675	V7676	V7677

The Table below applies to the DL250-1 or DL260 expansion base 1.

Expansion Base D2-CM #1: Analog Input 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
Storage Pointer	V36010	V36011	V36012	V36013	V36014	V36015	V36016	V36017

The Table below applies to the DL250-1 or DL260 expansion base 2.

Expansion Base D2-CM #2: Analog Input 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
Storage Pointer	V36110	V36111	V36112	V36113	V36114	V36115	V36116	V36117

The Table below applies to the DL260 CPU expansion base 3.

Expansion Base D2-CM #3: Analog Input 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
Storage Pointer	V36210	V36211	V36212	V36213	V36214	V36215	V36216	V36217

The Table below applies to the DL260 CPU expansion base 4.

Expansion Base D2-CM #4: Analog Input 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
Storage Pointer	V36310	V36311	V36312	V36313	V36314	V36315	V36316	V36317

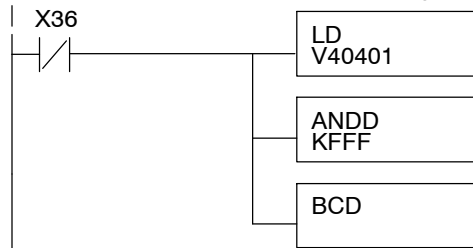
Reading Values (Multiplexing)

✓	✓	✓	✓
230	240	250-1	260

The DL230 CPU *does not* have the special V-memory locations that allow you to automatically enable the data transfer. 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, it is very easy to use the active channel status bits to determine which channel is being monitored.

Note, this example is for a module installed as shown in the previous examples. The addresses used would be different if the module was installed in a different I/O arrangement. You can place these rungs anywhere in the program, or if you are using stage programming place them in a stage that is always active.

Load Data when Module is not busy

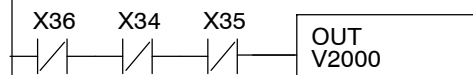


Loads the complete data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map.

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

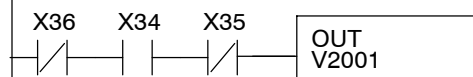
It is usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it.

Store Channel 1



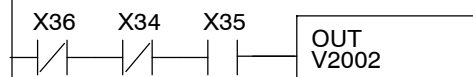
When the module is not busy and X34 and X35 are off, channel 1 data is stored in V2000.

Store Channel 2



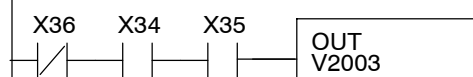
When X34 is on and X35 is off, channel 2 data is stored in V2001.

Store Channel 3



When X34 is off and X35 is on, channel 3 data is stored in V2002.

Store Channel 4

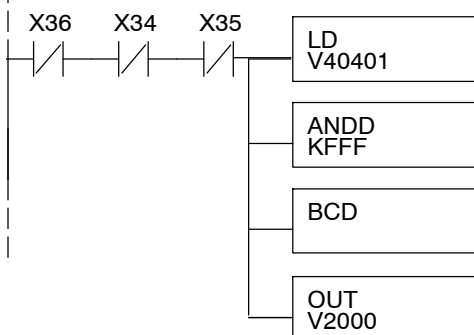


When both X34 and X35 are on, channel 4 data is stored in V2003.

Single Channel Selected

Since you do not have to determine which channel is selected, the single channel program is even more simple.

Store Channel 1 when Module is not busy



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.

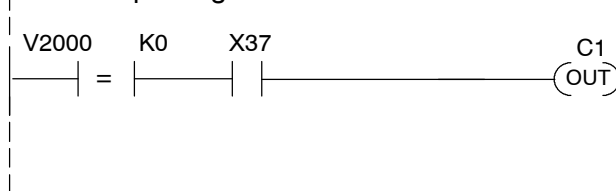
It is usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it.

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

Analog Power Failure Detection

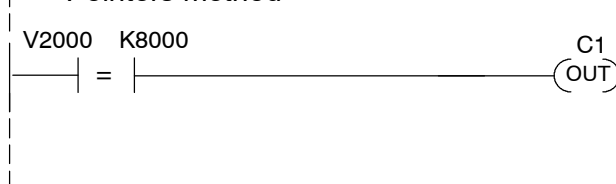
The Analog module has an on-board processor that can diagnose analog input circuit problems. You can easily create a simple ladder rung to detect these problems. This rung shows an input point that would be assigned if the module was installed as shown in the previous examples. A different point would be used if the module was installed in a different I/O arrangement.

Multiplexing method



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

Pointers method



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

Scaling the Input Data

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

You may have to make adjustments to the formula depending on the scale you choose for the engineering units.

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

H = High limit of the engineering unit range

L = Low limit of the engineering unit range

A = Analog value (0 - 4095)

For example, if you wanted to measure pressure (PSI) from 0.0 to 99.9 then you would have to multiply the analog value by 10 in order to imply a decimal place when you view the value with the programming software or a handheld programmer. Notice how the calculations differ when you use the multiplier.

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

Example without multiplier

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

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

$$\text{Units} = 49$$

Handheld Display

V 2001	V 2000
0000	0049

Example with multiplier

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

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

$$\text{Units} = 494$$

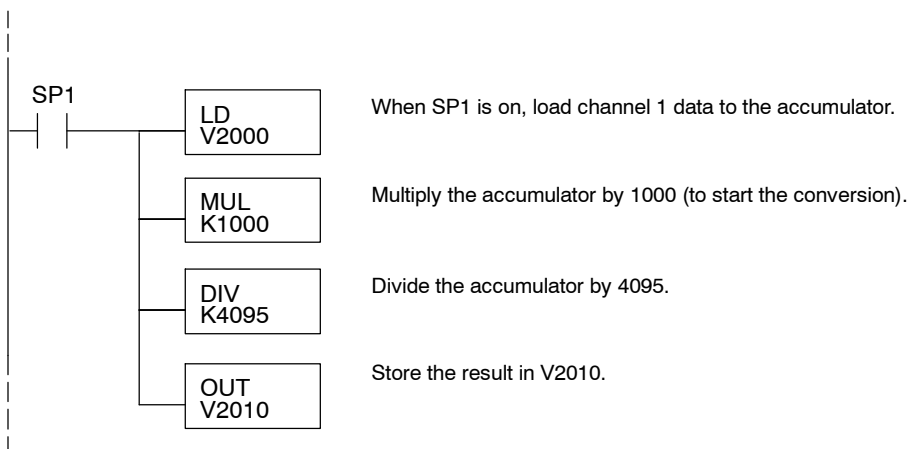
Handheld Display

V 2001	V 2000
0000	0494

This value is more accurate

The following example shows how you would write the program to perform the engineering unit conversion. This example assumes you have BCD data loaded into the appropriate V-memory locations using instructions that apply for the model of CPU you are using.

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



Analog and Digital Value Conversions

Sometimes it is useful to be able to quickly convert between the signal levels and the digital values. This is especially helpful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

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

For example, if you have measured the signal as 10mA, you can use the formula to easily determine the digital value that will be stored in the V-memory location that contains the data.

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

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

$$D = (255.93) (6)$$

$$D = 1536$$

Filtering Input Noise (DL250-1, DL260 CPU Only)

×	×	✓	✓
230	240	250-1	260



Add the following logic to filter and smooth analog input noise in DL250-1 and DL260 CPUs. This is especially useful when using PID loops. Noise can be generated by the field device and/or induced by field wiring.

The analog value in BCD is first converted to a binary number because there is not a BCD-to-real conversion instruction. Memory location V1400 is the designated work space in this example. The MULR instruction is the filter factor, which can be from 0.1 to 0.9. The example uses 0.2. A smaller filter factor increases filtering. You can use a higher precision value, 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 you are using the pointer method to get the analog value, it is in BCD and must be converted to binary. However, if you are using the conventional method of reading analog and are masking the first twelve bits, then it is already in binary and no conversion using the BIN instruction is needed.

