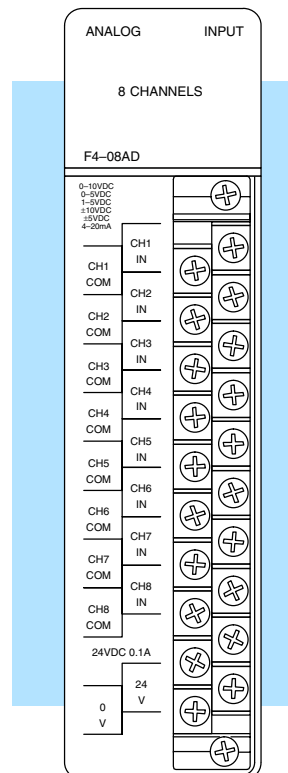# 5

F4-08AD
8-Channel
Analog Input

In This Chapter. . . .
— Module Specifications
— Setting the Module Jumpers
— Connecting the Field Wiring
— Module Operation
— Writing the Control Program

Module Specifications

The F4-08AD Analog Input module provides several features and benefits.

- It accepts eight single-ended voltage or current inputs.

- Analog inputs are optically isolated from PLC logic components.

- The module has a removable terminal block, so the module can be easily removed or changed without disconnecting the wiring.

- All eight analog inputs may be read in one CPU scan (DL440 and DL450 CPUs only).

ANALOG   INPUT

8 CHANNELS

F4–08AD

0–10VDC
0–5VDC
1–5VDC
±10VDC
±5VDC
4–20mA

CH1 IN
CH1 COM
CH2 IN
CH2 COM
CH3 IN
CH3 COM
CH4 IN
CH4 COM
CH5 IN
CH5 COM
CH6 IN
CH6 COM
CH7 IN
CH7 COM
CH8 IN
CH8 COM

24VDC 0.1A

24 V
0 V

Analog Input Configuration Requirements

The F4–08AD Analog Input module requires 16 discrete input points. The module can be installed in any slot of a DL405 system, including remote bases. The limitations on the number of analog modules are:

- For local and expansion systems, the available power budget and discrete I/O points.

- For remote I/O systems, the available power budget and number of remote I/O points.

Check the user manual for your particular model of CPU for more information regarding power budget and number of local or remote I/O points.

The following table provides the specifications for the F4–08AD Analog Input Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

| Number of Channels	8, single ended (one common)
Input Ranges	0–5V, 0–10V, 1–5V, ±5V, ±10V, 0–20 mA, 4–20 mA
Resolution	12 bit (1 in 4096)
Active Low-pass Filtering	–3 dB at 20Hz, –12 dB per octave
Input Impedance	250 ohms ±0.1%, 1/2W current input >20 Megohms voltage input, 1 Megohm minimum
Absolute Maximum Ratings	± 45 mA, current input ±75V, voltage input
Conversion Time	0.4ms per channel (module conversion) 1 ms per selected channel minimum (CPU)
Linearity Error (End to End)	±1 count (0.025% of full scale) maximum
Input Stability	±1/2 count
Full Scale Calibration Error (Offset error not included)	±12 counts maximum , voltage input ±12 counts maximum, @ 20mA current input
Offset Calibration Error	±2 counts maximum, unipolar voltage input
	±4 counts maximum, bipolar voltage input
	±4 counts maximum, 4 mA current input

General Specifications

PLC Update Rate	8 Channel per scan max.
Digital Input Points Required	16 (X) input points total 12 binary data bits, 3 active channel bits,
Power Budget Requirement	75 mA (power from base)
External Power Supply	18–30 VDC, 120 mA, class 2
Recommended Fuse	0.032 A, Series 217 fast-acting, current inputs
Accuracy vs. Temperature	±50 ppm / °C maximum full scale (including maximum offset change of 2 counts)
Operating Temperature	0 to 60°C (32 to 140° F)
Storage Temperature	–20 to 70°C (–4° F 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

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

Setting the Module Jumpers
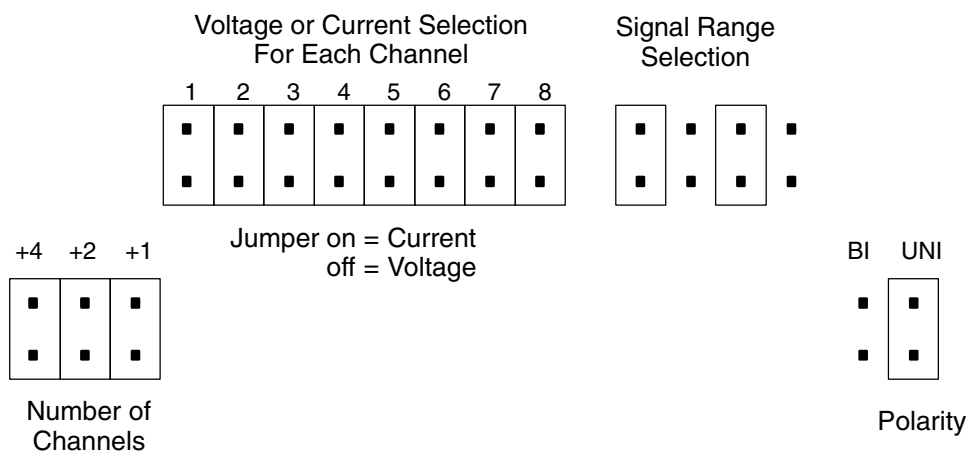
Jumper Locations
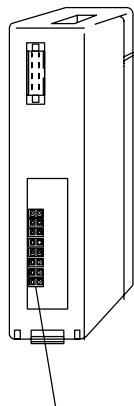
Jumper Locations

If you examine the rear of the module, you will notice four banks of jumpers. The module has several options that you can select by installing or removing these jumpers:

- A bank of eight jumpers to set voltage or current input for each channel.
- A bank of four jumpers to select the signal range for all active channels.
- A bank of three jumpers to select the number of channels used.
- A bank of two jumpers to select unipolar or bipolar signal range for all active channels.

The module is set at the factory for a 4–20 mA signal range on all eight channels with unipolar polarity. The following diagram shows how the jumpers are set at the factory and describes the function of each jumper. When removing a jumper, store it by placing it on a single pin to prevent losing it.
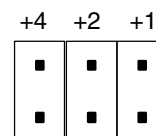

Voltage or Current Selection For Each Channel

Jumper on = Current
off = Voltage

Signal Range Selection

Number of Channels

Polarity

BI   UNI

Selecting the Number of Channels

The jumpers labeled +1, +2 and +4 are used to select the number of channels that will be used.

Any unused channels are not processed. For example, if you only select the first four channels, then the last four channels will not be active. Use this table to determine jumper settings.

Number of Channels

Jumpers installed as shown selects 8-channel operation.

For example: To select 3-channel operation, remove the +4 and +1 jumpers and install the +2 jumper.

Number of Channels Selected	Yes = jumper installed No = jumper removed		
	+4	+2	+1
1	No	No	No
2	No	No	Yes
3	No	Yes	No
4	No	Yes	Yes
5	Yes	No	No
6	Yes	No	Yes
7	Yes	Yes	No
8	Yes	Yes	Yes

Selecting Current or Voltage

Notice the eight jumpers for selecting current or voltage settings for each individual channel. For each channel install the jumper when you are using a current input or remove the jumper if using a voltage input.

Selecting Input Signal and Ranges

The following table shows the jumper selections for the various ranges and are grouped by bipolar and unipolar. The top portion of the table shows signal range settings for when the polarity jumper is installed in the Bi (bipolar) position, and the lower portion of the table shows settings for when the polarity jumper is installed in the Uni (unipolar) position. These settings will apply to all active channels.

Bipolar Signal Range	Jumper Settings
–2 VDC to +2 VDC (–8mA to +8 mA)	
–2.5 VDC to +2.5 VDC (–10mA to +10 mA)	
–5 VDC to +5 VDC (–20mA to +20 mA)	
–10 VDC to +10 VDC	
Unipolar Signal Range	**Jumper Settings**
4 to 20mA (1 VDC to 5 VDC)	
0 to +5 VDC (0 to +20 mA)	
0 to +10 VDC	

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.
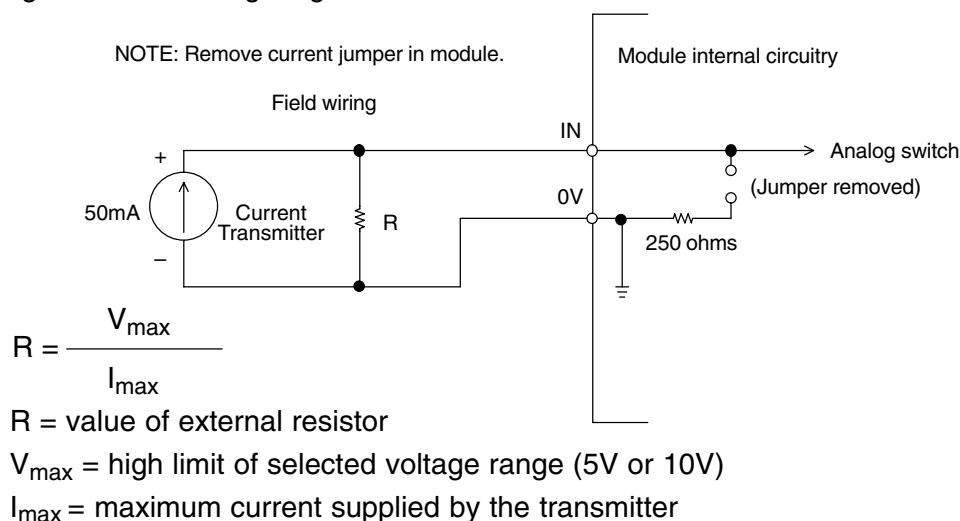
- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- 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 F4-08AD module requires a separate power supply. The Series DL405 CPUs, D4-RS Remote I/O Controller, and D4-EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. If you only have a couple analog modules, you can use this power source instead of a separate supply. If you have more than four analog modules, or you would rather use a separate supply, choose one that meets the following requirements: 24 VDC $\pm$10%, Class 2, 100 mA current (per module).

Custom Input Ranges

Occasionally you may have the need to connect a transmitter with an unusual signal range. By changing the wiring slightly and adding an external resistor to convert the current to voltage, you can easily adapt this module to meet the specifications for a transmitter that does not adhere to one of the standard input ranges. The following diagram shows how this works.

NOTE: Remove current jumper in module.

Field wiring

Module internal circuitry

IN

Analog switch

+

50mA

Current Transmitter

R

0V

(Jumper removed)

250 ohms

−

$$R = \frac{V_{max}}{I_{max}}$$

R = value of external resistor

V_{max} = high limit of selected voltage range (5V or 10V)

I_{max} = maximum current supplied by the transmitter

Example: current transmitter capable of 50mA, 0 − 10V range selected.

$$R = \frac{10V}{50mA} \qquad R = 200 \text{ ohms}$$

NOTE: Your choice of resistor can affect the accuracy of the module. A resistor that has $\pm$0.1% tolerance and a $\pm$50ppm / °C temperature coefficient is recommended.

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 F4-08AD provides 250 ohm resistance for each channel. If your transmitter requires a load resistance below 250 ohms, then you do not have to make any adjustments. However, if your transmitter requires a load resistance higher than 250 ohms, then you need to add a resistor in series with the module.

Consider the following example for a transmitter being operated from a 36 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 – resistor to add

$R = 750 - 250$ Tr – Transmitter Requirement

$R \geq 500$ Mr – Module resistance (internal 250 ohms)
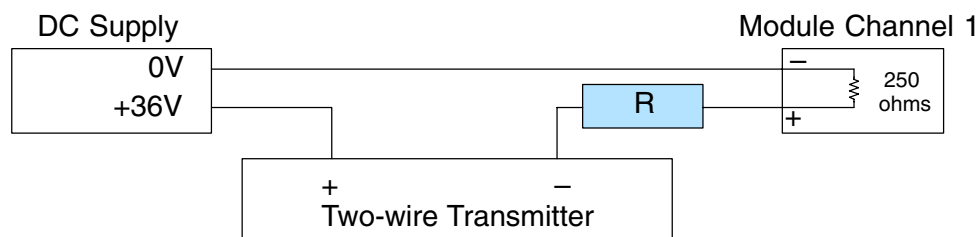
Removable Connector

The F4–08AD module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

Wiring Diagram

NOTE 1: Shields should be grounded at the signal source.
NOTE 2: Unused channels should be connected to 0V
or have current jumpers installed.
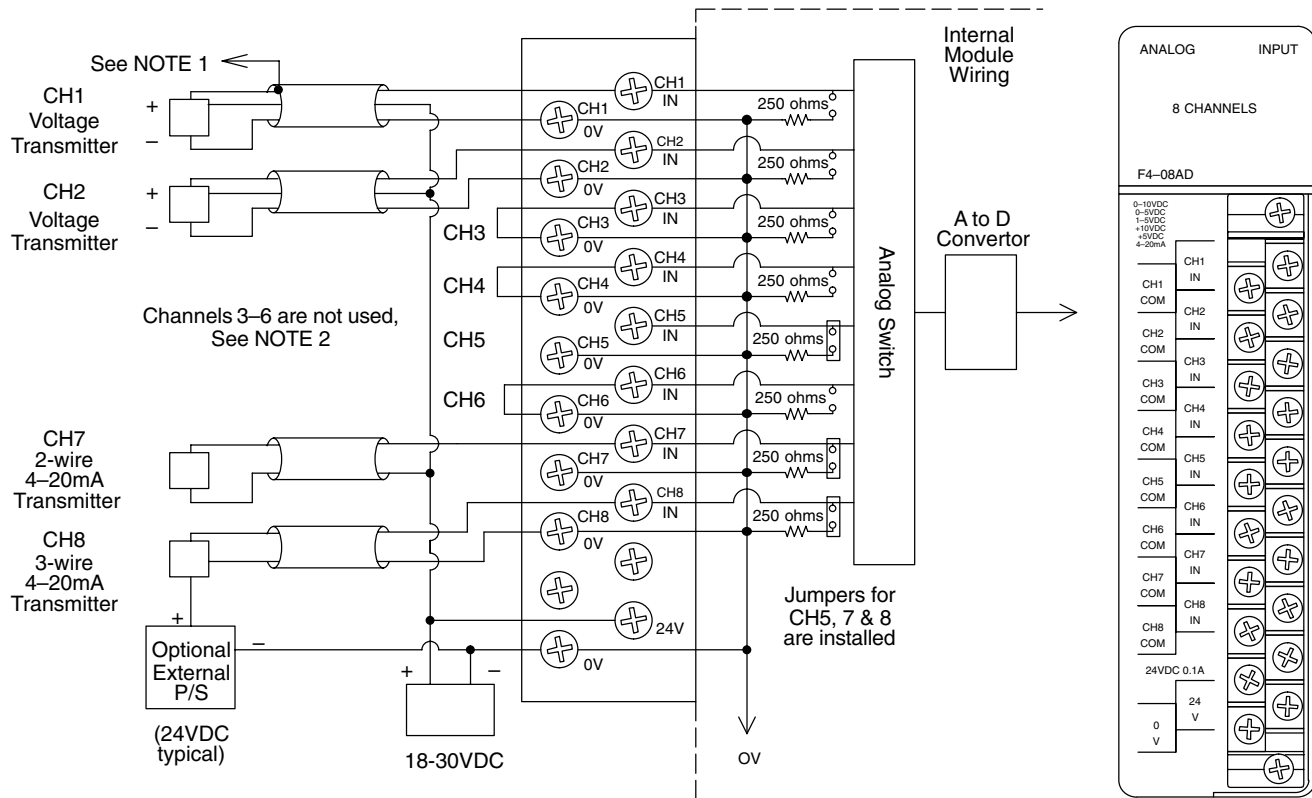
More than one external power supply can be used (see channel 8).

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 4–20mA 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.

Module Operation

DL430 Special Requirements

Even though the module can be placed in any slot, it is important to examine the configuration if you are using a DL430 CPU. As you will see in the section on writing the program, you use V-memory locations to extract the analog data. As shown in the following diagram, if you place the module so the input points do not start on a V-memory boundary, the instructions cannot access the data.

Correct!

F4–08AD

		8pt Input	8pt Input	16pt Input	16pt Input	16pt Output	16pt Output
		X0 – X7	X10 – X17	**X20 – X37**	X40 – X57		

V40400   V40402

V40401

MSB   LSB

X3 7   X3 0   X2 7   X2 0

Wrong!

F4–08AD

		8pt Input	16pt Input	8pt Input	16pt Input	16pt Output	16pt Output
		X0 – X7	**X10 – X27**	X30 – X37	X40 – X57		
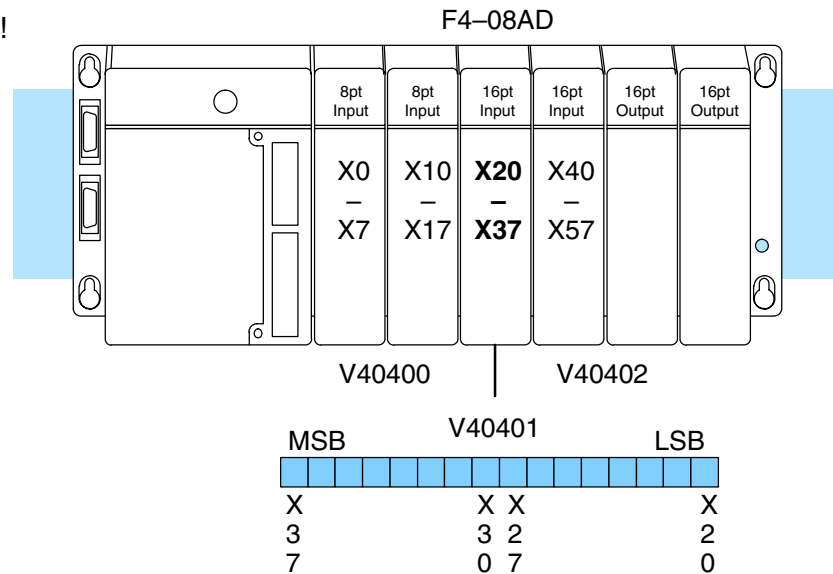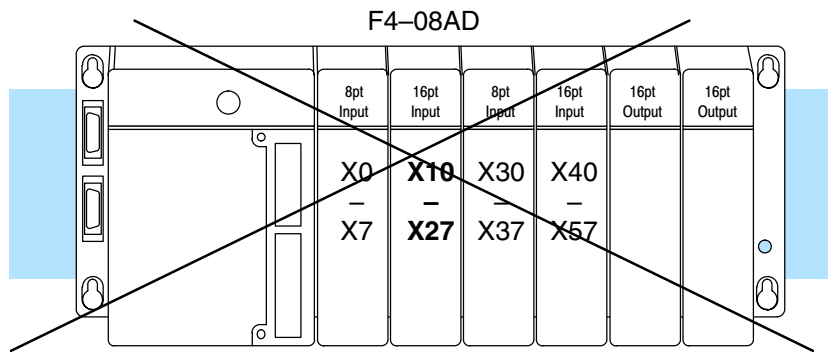
Data is split over two locations, so instructions cannot access data from a DL430.

V40401

MSB   LSB

X3 7   X3 0   X2 7   X2 0

V40400

MSB   LSB

X1 7   X1 0   X7   X0

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.
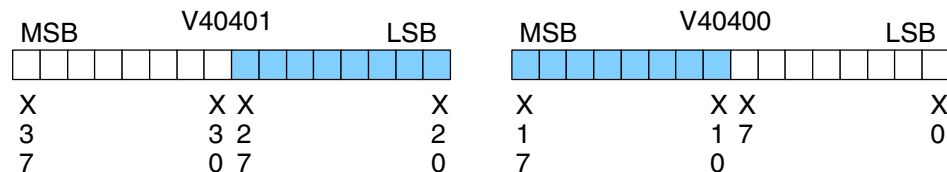
Channel Scanning Sequence

The F4-08AD module supplies one channel of data per each CPU scan. Since there are eight channels, it can take up to eight scans to get data for all channels. Once all channels have been scanned the process starts over with channel 1. There are ways around this. Later we'll show you how to write a program that will get all eight channels in one scan.

Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan.
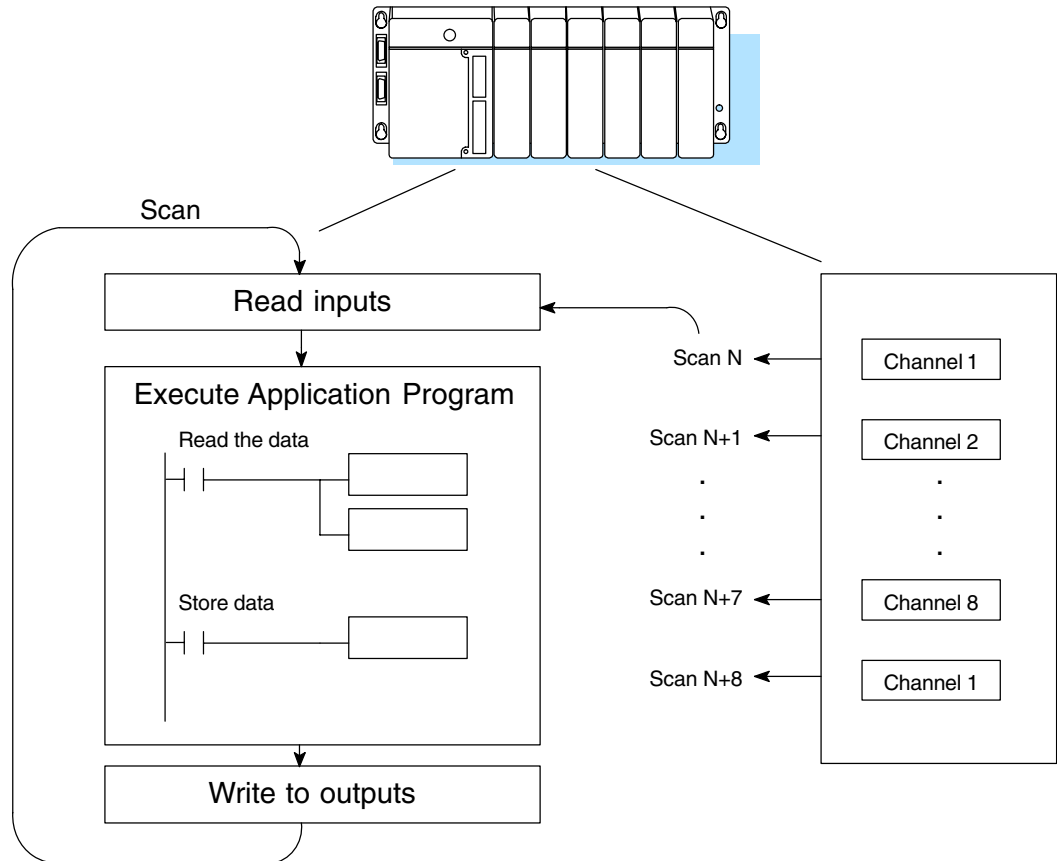
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

**Input Bit
Assignments**

You may recall the F4–08AD module requires 16 discrete input points from the CPU. These 16 points provide:

- An indication of which channel is active.
- The digital representation of the analog signal.

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.

F4-08AD

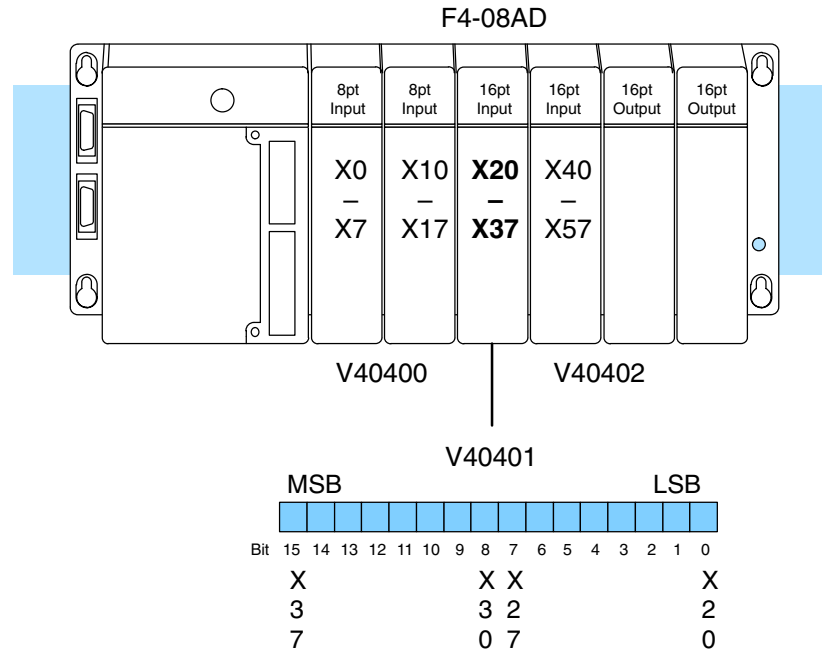|						
	8pt Input	8pt Input	16pt Input	16pt Input	16pt Output	16pt Output
	X0 – X7	X10 – X17	**X20 – X37**	X40 – X57		

V40400   V40402

V40401

MSB   LSB

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	X37							X30	X27							X20

Within this word location, the individual bits represent specific information about the analog signal.

**Active Channel
Indicator Inputs**

The bits (inputs) shown in the diagram indicate the active channel. The next to last three bits of the V-memory location indicate the active channel. The inputs are automatically turned on and off on each CPU scan to indicate the active channel.

V40401

MSB   LSB

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
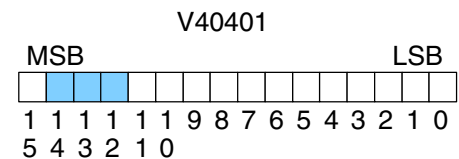
■ – channel inputs

Scan	Channel Inputs	Channel
N	000	1
N+1	001	2
N+2	010	3
N+3	011	4
N+4	100	5
N+5	101	6
N+6	110	7
N+7	111	8

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

V40401

MSB   LSB

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

☐ – data bits
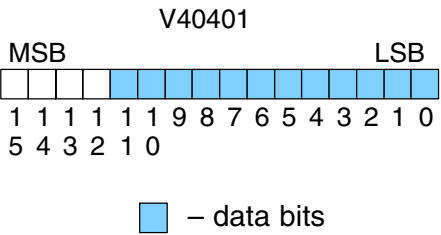
Since the module has 12-bit resolution, the analog signal is converted into 4096 counts ranging from 0-4095 (2^{12}). For example, with a 0 to 10V scale, a 0V signal would be 0, and a 10V signal would be 4095. This is equivalent to a a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The following diagram shows how this relates to each signal range.

−10V to +10V
−5V to +5V

0V − 10V
0V − 5V

1V − 5V

4 − 20mA
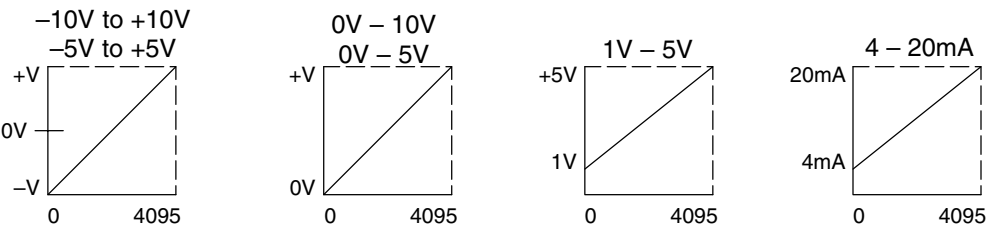
Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal levels that will result in a change in the data value for each signal range.

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

H = high limit of the signal range

L = low limit of the signal range

Range	Signal Span (H - L)	Divide By	Smallest Detectable Change
±10V	20 V	4095	4.88 mV
±5V	10 V	4095	2.44 mV
0 to 5V	5 V	4095	1.22 mV
0 to 10V	10 V	4095	2.44 mV
1 to 5V	4 V	4095	0.98 mV
4 to 20mA	16 mA	4095	3.91 μA

Unusable MSB Bit

When using some instructions, the most significant bit (MSB) is read along with the three active channel bits, and is not available for other uses.

V40401

MSB   LSB

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

☐ – unusable bit   Active Channel Bits
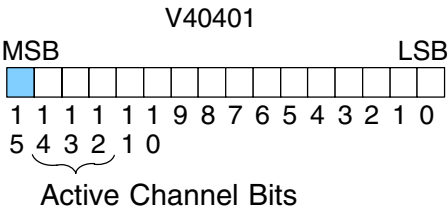
Writing the Control Program

If you have configured the F4–08AD module, use the following examples to get started writing the control program.

Multiple Channels Selected

Since all channels are multiplexed into a single data word, the control program must be set up 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.

F4-08AD

8pt Input	8pt Input	16pt Input	16pt Input	16pt Output	16pt Output
X0 – X7	X10 – X17	**X20 – X37**	X40 – X57		

V40400   V40402

V40401

MSB   LSB

Unusable Bit   Active Channel Bits   Data Bits
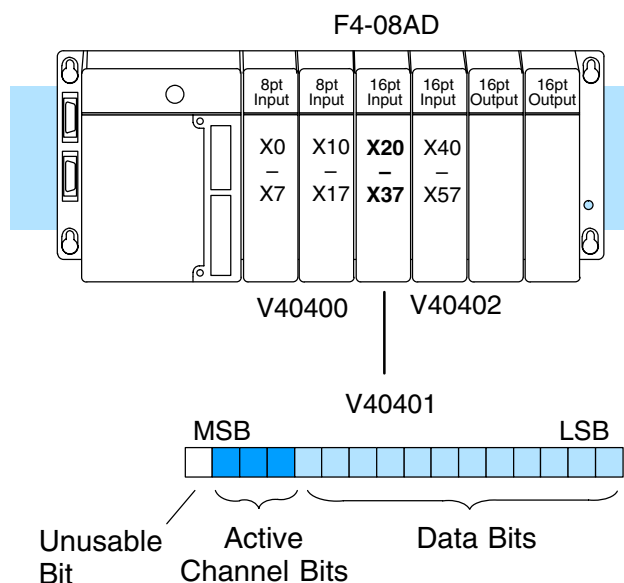
Reading Values, DL430 CPU

✓ ✓ ✓
430 440 450

The following program example shows how to read the analog data into V-memory locations with the DL430 CPU. Since the DL430 does not support the LDF instruction, you can use the LD instruction instead as shown. The example also works for DL440 and DL450 CPUs. This example will read one channel per scan, so it will take eight scans to read all eight channels. Contact SP1 is used in the example because the inputs are continually being updated.

SP1
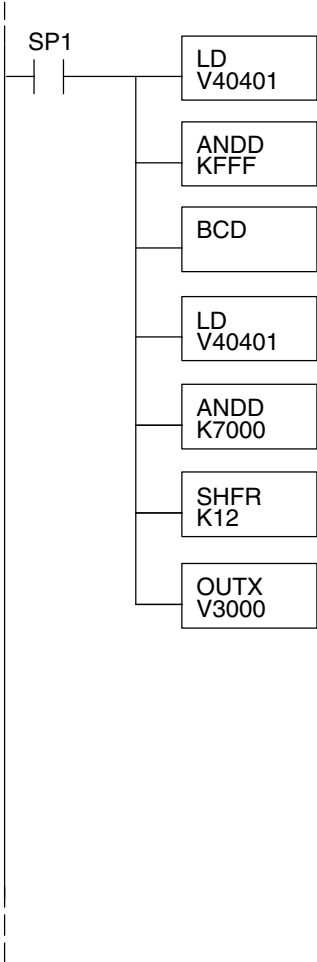
| LD V40401 | Loads the complete channel data word from the module into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map. |

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

| BCD | Since the DL405 CPUs perform math operations in BCD, it is usually best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it (such as for PID loops, which require the process variable to be in binary format). |

| LD V40401 | This load instruction reads the data into the accumulator again. The channel data will be pushed into the first level of the stack. |

| ANDD K7000 | This instruction masks the analog data values and leaves the channel ID bits in the accumulator. |

| SHFR K12 | Now you have to shift the accumulator bits so the channel ID bits will result in a value between 0 and 7 (binary format). This value is the offset and indicates which channel is being processed in that scan. |

| OUTX V3000 | OUTX copies the value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–7) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0) and if the binary value is 6, then the channel 7 data is stored in location V3006 (V3000 + 6). See the following table. |

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

Module Reading Stored in ...	Acc. Bits	Offset	Data
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

Single Channel Selected

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

SP1

LD or LDF

When X34 is on, channel 1 data is being sent to the CPU. Use the LD instruction when using a DL430 CPU.*

BCD

The BCD instruction converts the data from binary to BCD. You can leave out this instruction if your application does not require it.

OUT
V3000

The OUT instruction stores the data in V3000.

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

* Remember, before the BCD instruction is executed, the DL430 requires an additional instruction to mask out the first four bits that are brought in with the LD instruction. An example of how to do this using an ANDD instruction is shown in the previous section.

Reading Values, DL440/450

The following program example shows how to read the analog data into V-memory locations with DL440 and DL450 CPUs. Once the data is in V-memory, you can perform math on the data, compare the data against preset values, and so forth. This example will read one channel per scan, so it will take eight scans to read all eight channels.

SP1

LDF   X20
K12

Loads the first 12 bits of channel data (starting with location X20) from the module into the accumulator.

BCD

Converts the binary value in the accumulator to BCD and stores the result in the accumulator. Use this BCD conversion if you want the channel data to be stored as BCD. Do not use this instruction if you are going to send the data to an internal PID loop because the PID loop requires the PV (process variable) to be in binary format.

LDF   X34
K3

Loads the binary value of the three channel indicator bits, plus the MSB, into the accumulator and pushes the channel data loaded into the accumulator from the first LDF instruction into the first level of the stack. X34 = X20 + 14.

OUTX
V3000

OUTX copies the 16 bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (which is the offset) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0) and if the binary value is 6, then the channel 7 data is stored in location V3006 (V3000 + 6). See the following table.
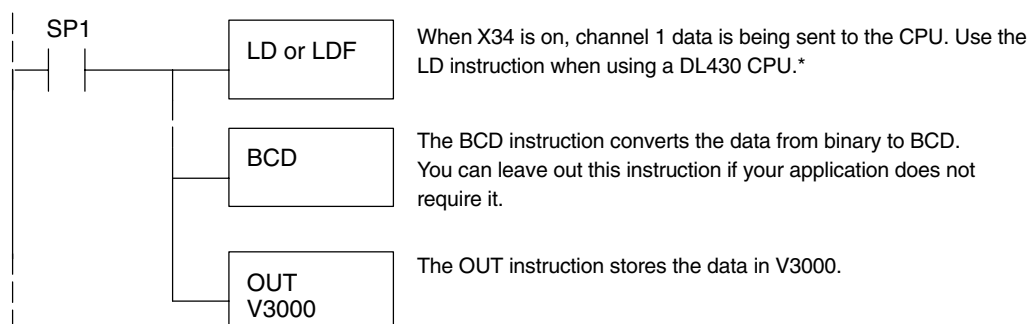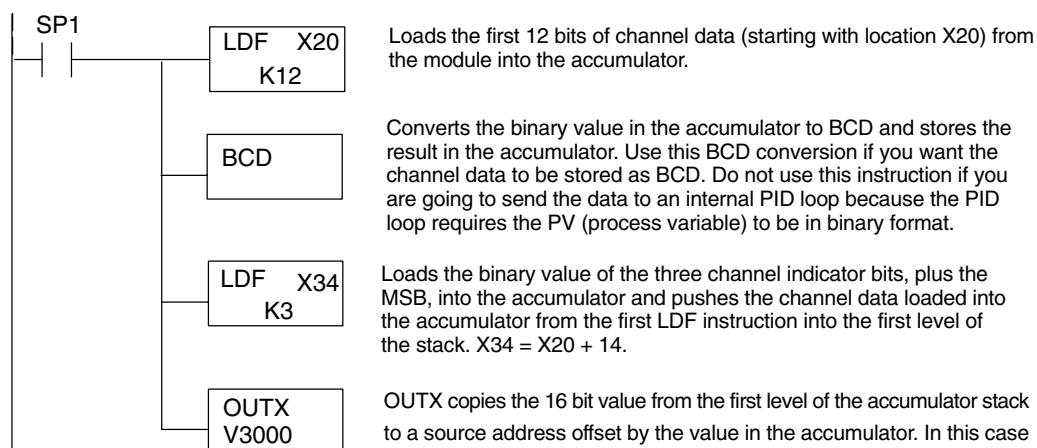
| Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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

Reading Eight Channels in One Scan, DL440/450

✗	✓	✓
430	**440**	**450**

The following program example shows how to read all eight channels in one scan by using a FOR/NEXT loop. Before you choose this method, do consider its impact on CPU scan time. The FOR/NEXT routine shown here will add about 16ms (2ms/loop) to the overall scan time. If you do not need to read the analog data on every scan, change SP1 to a permissive contact (such as an X input, CR, or stage bit) to only enable the FOR/NEXT loop when it is required.

NOTE: Do not use this FOR/NEXT loop program to read the module in a remote/slave arrangement; it will not work. Use one of the programs shown that reads one channel per scan.
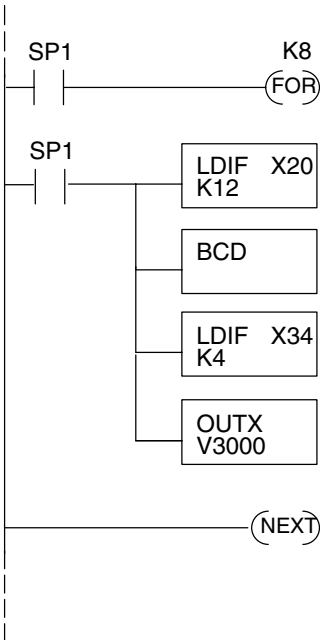
SP1 ── K8 (FOR)

Starts the FOR/NEXT loop. The constant (K8) specifies how many times the loop will execute. Enter a constant equal to the number of channels you are using. For example, enter K4 if you're using 4 channels.

SP1 ── LDIF X20 K12

Immediately loads the first 12 bits of the data word (starting with X20) into the accumulator. The LDIF instruction will retreive the I/O points without waiting on the CPU to finish the scan.

BCD

Since the DL405 CPUs perform math operations in BCD, it is usually best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it (such as PID loops).

LDIF X34 K4

This LDIF instruction immediately loads the three channel indicator bits into the accumulator. (For this module, the last bit in the word must be read also, that's why the K4 is used. Otherwise, only one channel will be read).

OUTX V3000

The OUTX instruction stores the channel data to an address that starts at V3000 plus the channel offset. For example, if channel 3 was being read, the data would be stored in V3002 (V3000 + 2).

(NEXT)

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

Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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.

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

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

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 3101 V 3100 |
| V MON   0000   0049 |

Example with multiplier

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

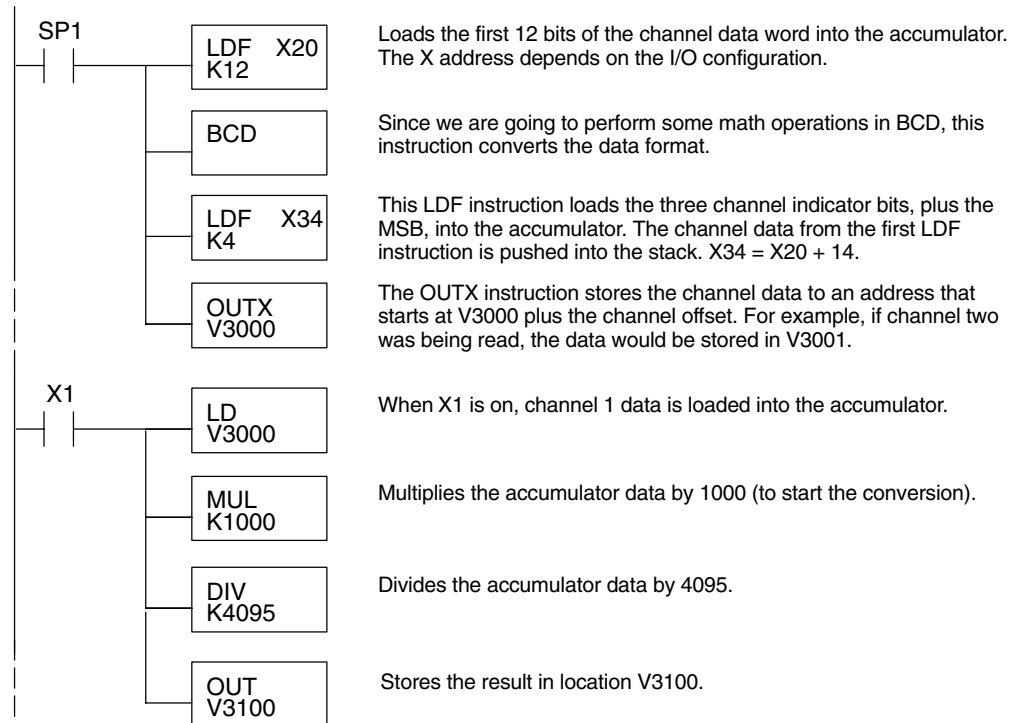$$\text{Units} = 20240 \frac{100 - 0}{4095}$$

$$\text{Units} = 494$$

Handheld Display

| V 3101 V 3100 |
| V MON   0000   0494* |

*Value is more accurate

Here's how you would write the program to perform the engineering unit conversion. This example uses SP1 which is always on. You could also use an X, C, etc. permissive contact.

SP1

| LDF   X20 |
| K12 |

Loads the first 12 bits of the channel data word into the accumulator. The X address depends on the I/O configuration.

| BCD |

Since we are going to perform some math operations in BCD, this instruction converts the data format.

| LDF   X34 |
| K4 |

This LDF instruction loads the three channel indicator bits, plus the MSB, into the accumulator. The channel data from the first LDF instruction is pushed into the stack. X34 = X20 + 14.

| OUTX |
| V3000 |

The OUTX instruction stores the channel data to an address that starts at V3000 plus the channel offset. For example, if channel two was being read, the data would be stored in V3001.

X1

| LD |
| V3000 |

When X1 is on, channel 1 data is loaded into the accumulator.

| MUL |
| K1000 |

Multiplies the accumulator data by 1000 (to start the conversion).

| DIV |
| K4095 |

Divides the accumulator data by 4095.

| OUT |
| V3100 |

Stores the result in location V3100.

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the signal levels and the digital values. This is especially useful 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 signal level ...
−10V to + 10V	$A = \dfrac{20D}{4095} - 10$	$D = \dfrac{4095}{20}(A + 10)$
−5V to + 5V	$A = \dfrac{10D}{4095} - 5$	$D = \dfrac{4095}{10}(A + 5)$
0 to 5V	$A = \dfrac{5D}{4095}$	$D = \dfrac{4095}{5}(A)$
0 to 10V	$A = \dfrac{10D}{4095}$	$D = \dfrac{4095}{10}(A)$
1 to 5V	$A = \dfrac{4D}{4095} + 1$	$D = \dfrac{4095}{4}(A - 1)$
4 to 20mA	$A = \dfrac{16D}{4095} + 4$	$D = \dfrac{4095}{16}(A - 4)$

For example, if you are using the –10V to +10V range and you have measured the signal at 6V, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

$$D = \frac{4095}{20}(A + 10)$$

$$D = \frac{4095}{20}(6V + 10)$$

$$D = (204.75)(16)$$

$$D = 3276$$