Errata Sheet

This Errata Sheet contains corrections or changes made after the publication of this manual.

Product Family:	DL405	**Date**:	September 12, 2018
Manual Number	D4-ANLG-M		
Revision and Date	5th Ed., Rev. A; July 2004		

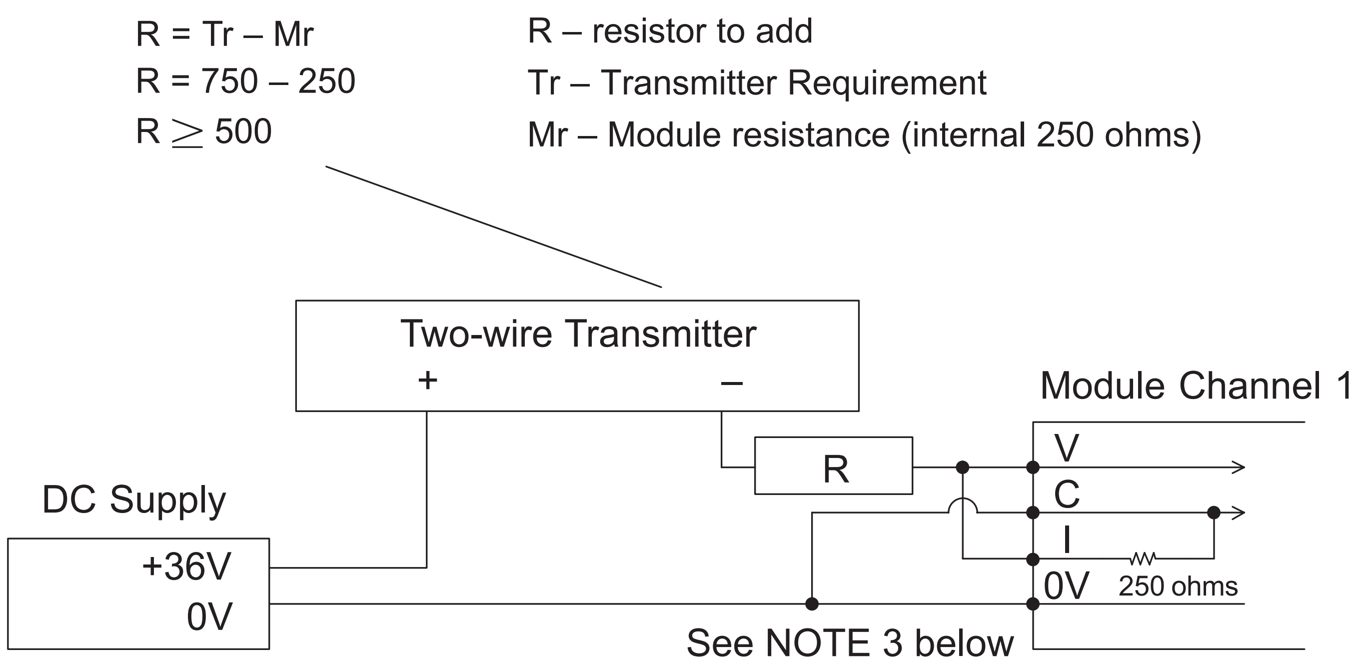
Changes to Chapter 3: F4–04AD 4-Channel Analog Input

Page 3-3. Module Specifications; General Specifications

In the table, change the Power Budget Requirement value from “85 mA (power from base)” to “150 mA (power from base)”.

Page 3-11. Current Loop Transmitter Impedance

Replace the example drawing with this one. Connections were added between the power supply 0V terminal, the 0V CH1 terminal, and the CH1 common terminal. Also, the “See NOTE 3 below” note was added.



NOTE 3: When a differential input is not used, 0V should be connected to C of the channel.

Errata Sheet

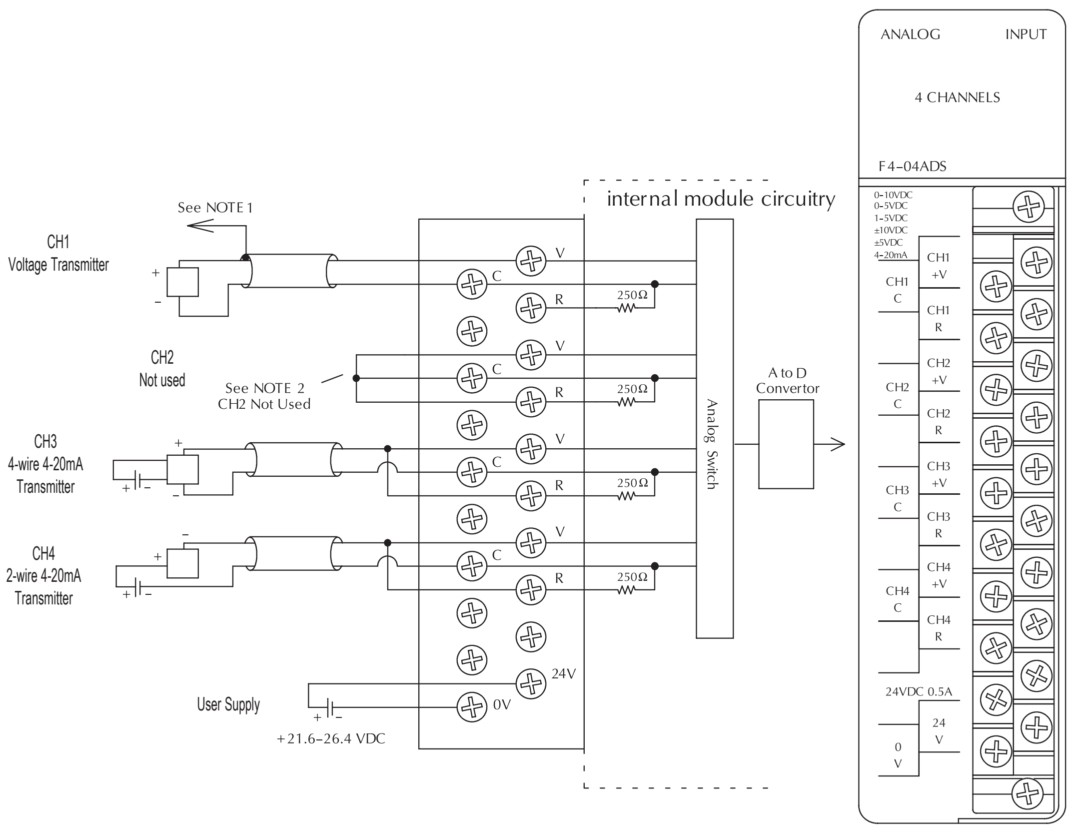
Changes to Chapter 4: F4-04ADS 4-Channel Isolated Analog Input

Page 4-3. Module Specifications; General Specifications

In the table, change the Power Budget Requirement value from “270 mA at 5 VDC (from base)” to “370 mA at 5 VDC (from base).”

Page 4-8. Wiring Diagram

Replace the wiring diagram with this one. The connections for CH3 and CH4 were incorrect. They did not show that external power is required. Examples for wiring 2-wire and 4-wire current transmitters was added.



Errata Sheet

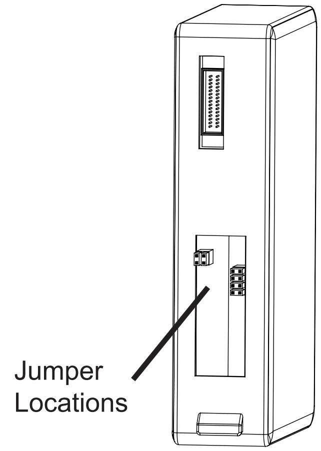
Changes to Chapter 6: F4–16AD-1 16-Channel Analog Input

Page 6-4. Setting the Module Jumpers

Changes to Chapter 7: F4–16AD-2 16-Channel Analog Input

Page 7-4. Setting the Module Jumpers

For both modules, the jumpers are now arranged differently. They are no longer in a straight line like the drawings on pages 6-4 and 7-4 show. They are now next to each other as shown here.



Changes to Chapter 7: Title page

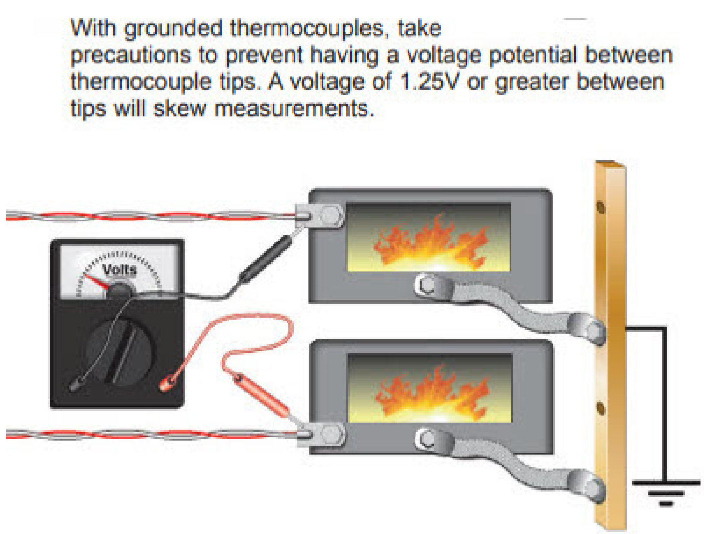
The title page mistakenly calls this an 8-point module; it is actually 16 points

Changes to Chapter 8: F4-08THM-n 8-Channel Thermocouple Input

Changes to Chapter 10: F4-08THM 8-Channel Thermocouple Input

Pages 8-7 and 10-10. Wiring Diagram

Add the following note and drawing to the wiring diagrams for both of these thermocouple modules.

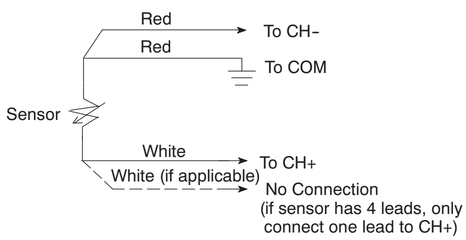


Errata Sheet

Changes to Chapter 9: F4-08RTD 8-Channel RTD Input

Page 9-7. Connecting the Field Wiring; RTD - Resistance Temperature Detector; Lead Detection for RTD Sensors

Replace the wiring diagram with this one. The wire lead colors changed. (The two black leads changed to red and the two red leads changed to white.)

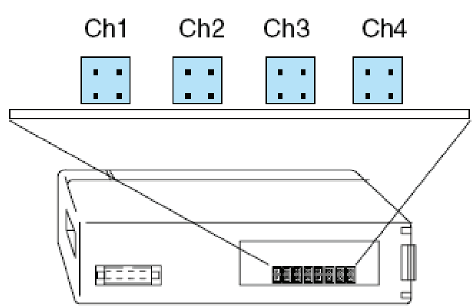


Changes to Chapter 18: F4-04DAS-2 4-Channel Isolated 0–5V, 0–10V Output

Page 18-4. Setting the Module Jumpers

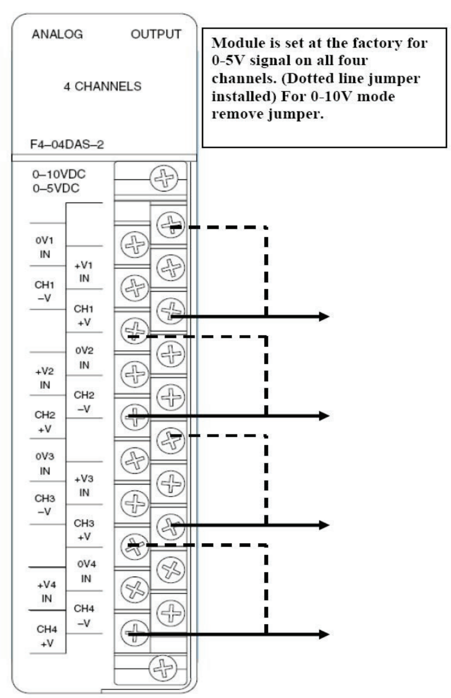
In 2008 the module was redesigned and the range selection jumpers on the back of the module (as described below on the left and on page 18-4) were eliminated. The range selection is now done by a wire jumper on the terminal block as shown here on the right.

"Old Version"



Channel Signal Range	"Old Version Jumper Settings"	"New Version Jumper Settings"
0-5 VDC Place jumper on LEFT		"Install on TB"
0-10 VDC Place jumper on RIGHT		"Remove from TB"

"New Version"

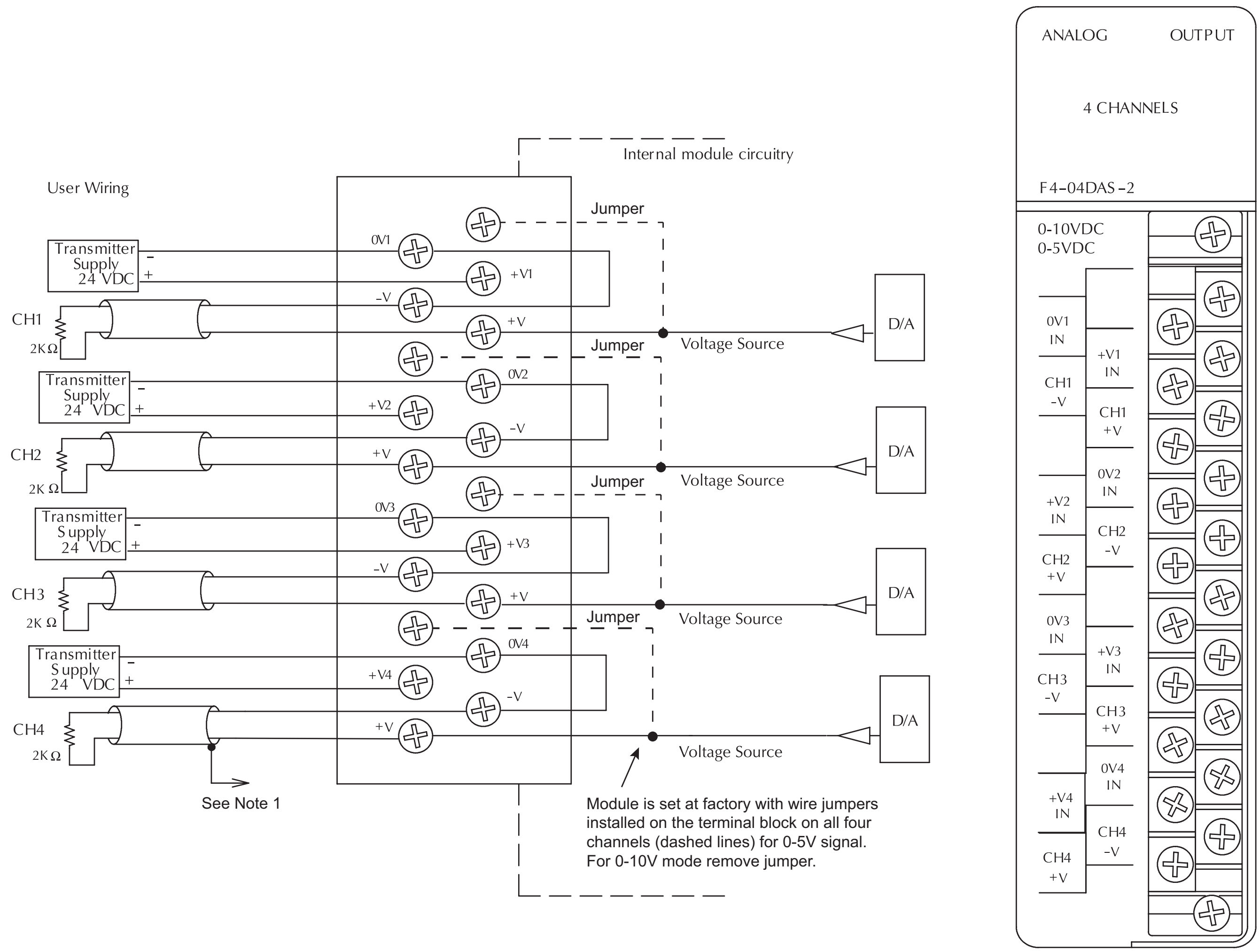


Errata Sheet

Changes to Chapter 18: F4-04DAS-2 4-Channel Isolated 0–5V, 0–10V Output (continued)

Page 18-5. Wiring Diagram

In 2008 the module was redesigned and the range selection jumpers on the back of the module were eliminated. The range selection is now done by a wire jumper for each channel located on the terminal block. This wiring diagram was revised to show these jumpers.

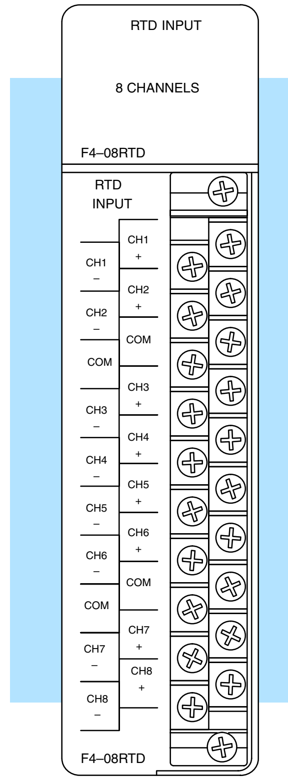


9

F4-08RTD 8-Channel RTD Input

In This Chapter. . . .

Module Specifications



The F4–08RTD 8 Differential Channel RTD Input module provides several features and benefits.

- It provides eight RTD input channels with 16-bit resolution.
- It automatically converts 10 Ω , 25 Ω, 100 Ω, 1000 Ω RTD signals into direct temperature readings. No extra scaling or complex conversion routines are required.
- Temperature data format is selectable between °F or °C , magnitude plus sign or twos complement.
- The module is capable of converting both European and American type 100 Ω RTDs and European type1000 Ω RTDs.
- Precision lead wire resistance compensation by dual matched current sources and ratiometric measurements.
- Temperature calculation and linearization are based on data provided by NIST (National Institute of Standards and Technology).
- Diagnostics features include detection of RTD short or disconnection.

Module Calibration

The F4–08RTD module requires no calibration; however, if your process requires calibration it is possible to correct the RTD tolerance using ladder logic to subtract or add a constant to the actual reading for that particular RTD.

RTD Input Configuration Requirements

The F4–08RTD Input Module requires 32 discrete input points from the CPU. 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 tables provide the specifications for the F4–08RTD Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

Number of Channels	8 differential inputs
Input Ranges	Pt100 –200°C/850°C (–328°F/1562°F) Pt 1000 –200°C/595°C (–328°F/1103°F) jPt100 –38°C/450°C (–36°F/842°F) 10ΩCu. –200°C/260°C (–328°F/500°F) 25ΩCu. –200°C/260°C (–328°F/500°F)
Display Resolution	±0.01 °C, ±0.01 °F (±3276.7)
Resolution	15-bit (1 in 32768)
Absolute Maximum Ratings	Fault-protected input, ±22 VDC
Converter Type	Charge balancing, 24-bit
Sampling Rate	160 msec per channel
Temperature Drift	±5ppm per °C (maximum)
Common Mode Range	0–5 VDC
Linearity Error	±.05° C maximum, ±.01° C typical
Full Scale Calibration	±1° C

General Specifications

PLC Update Rate	8 Channels/Scan max. DL440/DL450 CPUs 1 Channel/Scan max. DL430 CPU
Digital Input Points Required	32 (X) input points, 16 binary data bits, 3 channel ID bits, 8 fault bits
Power Budget Requirement	80 mA @ 5 VDC (from base)
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

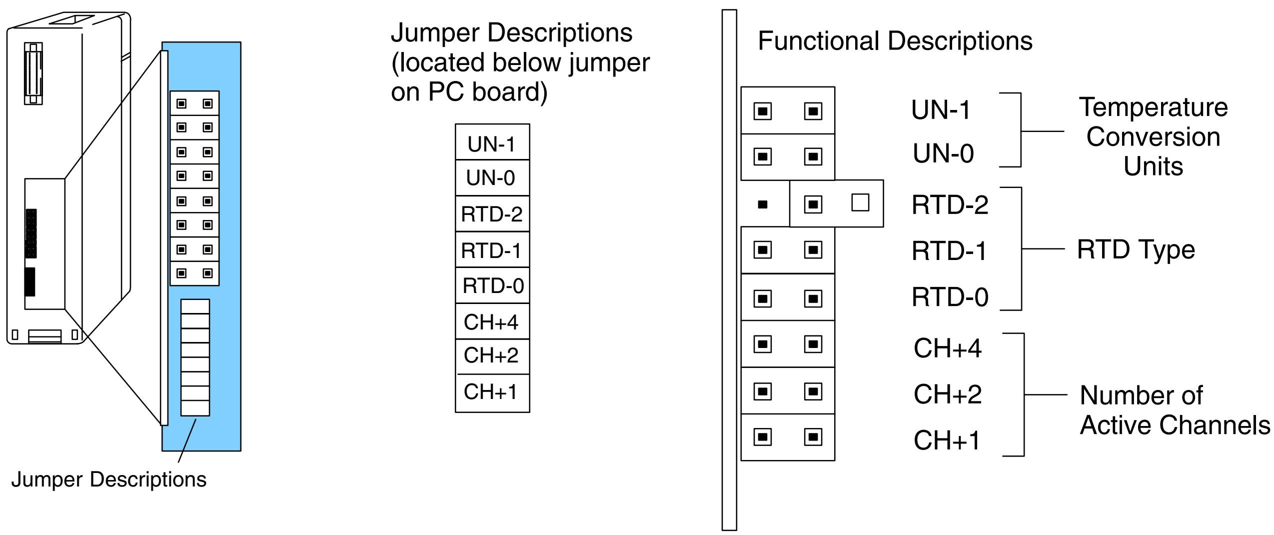
Setting the Module Jumpers

Jumper Locations

The module has several options that you can select by installing or removing jumpers. At the rear of the module is a bank of eight jumpers. You can select the following options by installing or removing the jumpers:

- Number of channels: 1 thru 8.
- The input type: 10 Ω (ohms) or 25 Ω copper RTDs; Pt100 Ω , jPt100 Ω, Pt1000 Ω RTDs.
- Temperature conversion: two's complement or magnitude plus sign format in Fahrenheit or Celsius.

To prevent losing a jumper when it is removed, store it near its original location by sliding one of its sockets over a single pin (like the RTD-2 jumper setting below).

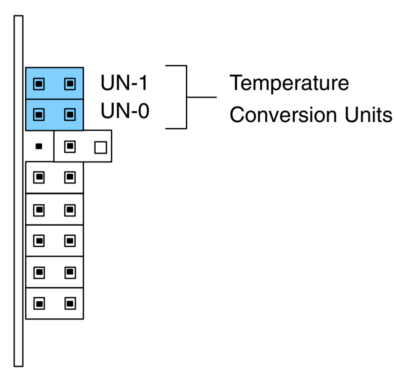


Factory Default Settings

By default, the module arrives from the factory as shown above with all jumpers installed except for the RTD-2 jumper (the third jumper from the top), which has the jumper removed. These settings select eight active channels, convert temperatures into Fahrenheit readings using magnitude plus sign, with Pt100Ω RTD type.

Selecting Fahrenheit or Celsius

The top two jumpers, **UN-0** and **UN-1**, select the conversion unit. The options are magnitude plus sign or two's complement, in Fahrenheit or Celsius. The module comes from the factory with both jumpers installed for magnitude plus sign conversion in Fahrenheit.



All RTD types are converted into a direct temperature reading in either Fahrenheit or Celsius. The data contains one implied decimal place. For example, a value in V-memory of 1002 would be 100.2° C or F.

Negative temperatures can be represented in either two's complement or magnitude plus sign form. If the temperature is negative, the most significant bit in the V-memory location is set (X37).

The two's complement data format may be required to correctly display bipolar data on some operator interfaces and HMI software packages. This data format could also be used to simplify averaging a bipolar signal.

The table shows how to arrange the jumpers.

X = jumper installed, empty space = jumper removed.

Jumper	Temperature Conversion Units			
	Magnitude + Sign °F	°C	2's Complement °F	°C
UN-1	X	X		
UN-0	X		X	

Selecting RTD Type

The jumpers labeled **RTD-2, RTD-1,** and **RTD-0** are used to select the type of RTD. The module can be used with many types of RTDs. All channels of the module must be the same RTD type.

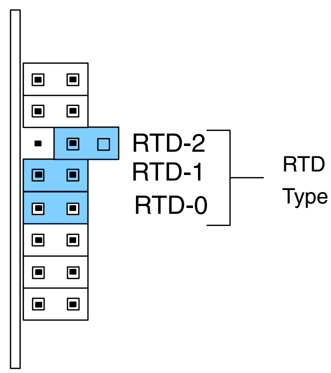
The default setting from the factory is Pt100 Ω (RTD-2 comes with the jumper removed). This selects the DIN 43760 European type RTD. European curve type RTDs are calibrated to DIN 43760, BS1905, or IEC751 specifications which is .00385 Ω /Ω / ° C (100° C = 138.5Ω).

The jPt100 Ω type is used for the American curve (.00392 Ω/Ω/° C), platinum 100 Ω RTDs. The 10 Ω and 25 Ω RTD settings are used with copper RTDs.

The table shows how to arrange the jumpers. For example, to select 10Ω, remove all three jumpers.

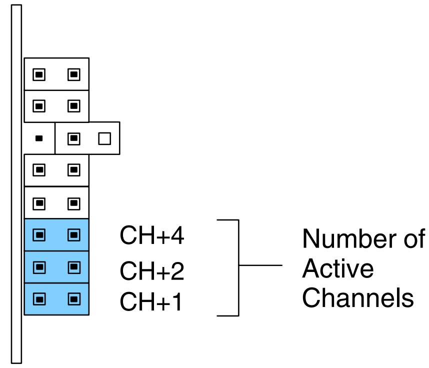
X = jumper installed, empty space = jumper removed.

Jumper	RTD Type 10Ω	25Ω	jPt100Ω	Pt100Ω	Pt1000Ω
RTD-2					X
RTD-1			X	X	
RTD-0		X		X	



Selecting the Number of Channels

The three jumpers labeled **CH+4, CH+2,** and **CH+1** are binary encoded to select the number of channels that will be used. Channels must be used sequentially, starting with channel 1. For example, if you are going to use only two channels, you must use channels 1 and 2 (not 2 and 3, 5 and 7, etc.).



The module comes factory-set with all jumpers installed for eight-channel operation.

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. The following table shows how to arrange the jumpers. For example, to select channels 1 thru 4, remove jumper CH+4 and install jumpers CH+2 and CH+1.

X = jumper installed,

empty space = jumper removed.

Number of Channels	Jumper		
	CH+4	**CH+2**	**CH+1**
1			
2			X
3		X	
4		X	X
5	X		
6	X		X
7	X	X	
8	X	X	X

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 signal 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.

RTD – Resistance Temperature Detector

Use shielded RTDs whenever possible to minimize noise on the input signal. Ground the shield wire at one end only. Connect the shield wire to the COM (common) terminal.

Lead Configuration for RTD Sensors

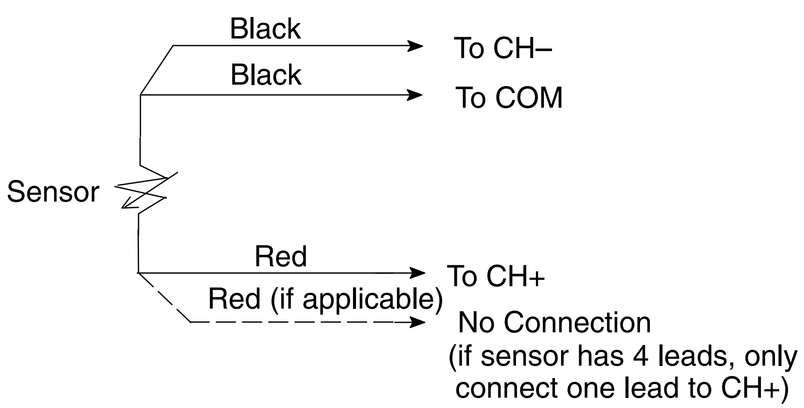
The suggested three-lead configuration shown below provides one lead to the CH+ terminal, one lead to the CH– terminal, and one lead to the COM (common) terminal. Compensation circuitry nulls out the lead length for accurate temperature measurements.

Some sensors have four leads. When making connections, do not connect the second red lead to the CH+ input; leave that lead unconnected.

Do not use configurations having only one lead connected to each input (there is no compensation and temperature readings will be inaccurate).

Wiring Connections For Typical RTD Sensor

See Errata Sheet at the beginning of this file. There is a revised drawing. The wire leads colors were revised.

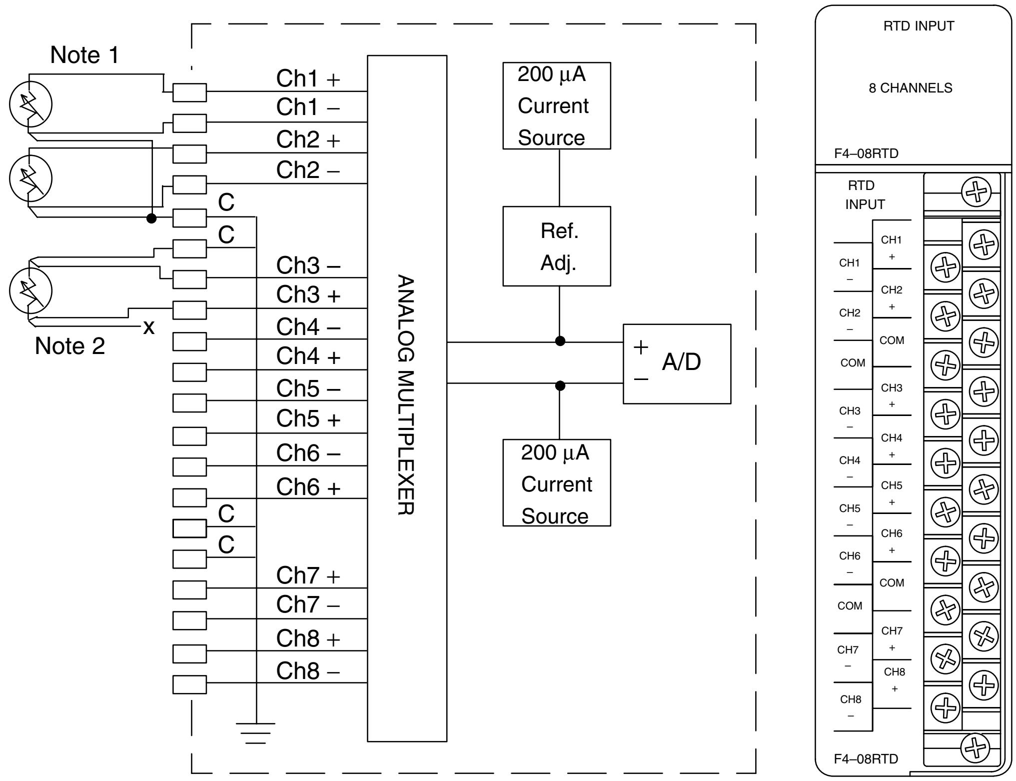


Ambient Variations in Temperature

The F4–08RTD module has been designed to operate within the ambient temperature range of 0°C to 60°C.

Precision analog measurement with no long-term temperature drift is assured by a chopper-stabilized programmable gain amplifier, ratiometric referencing, and automatic offset and gain calibration.

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

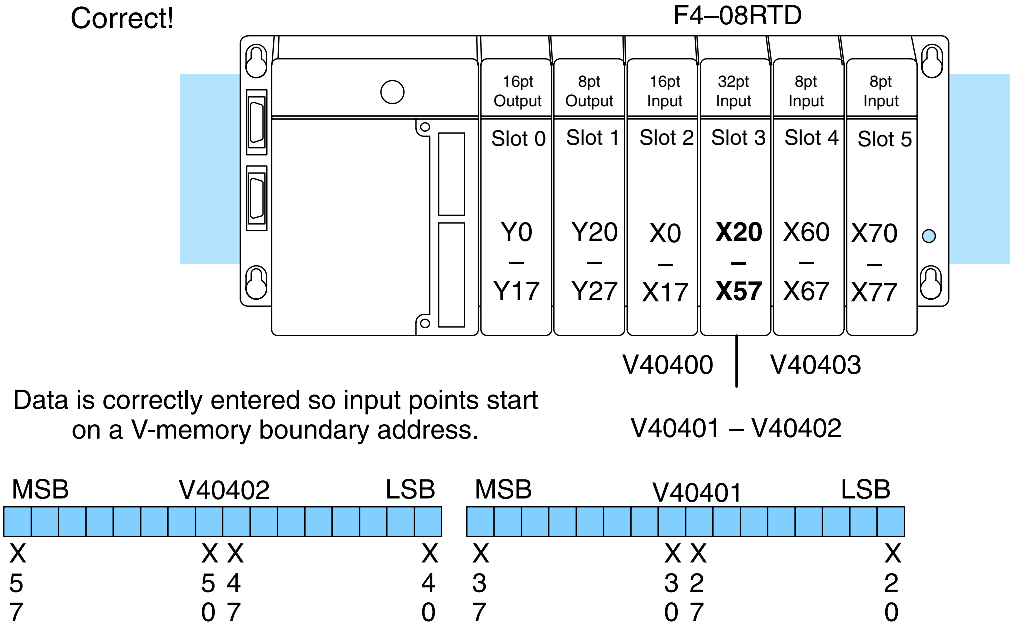


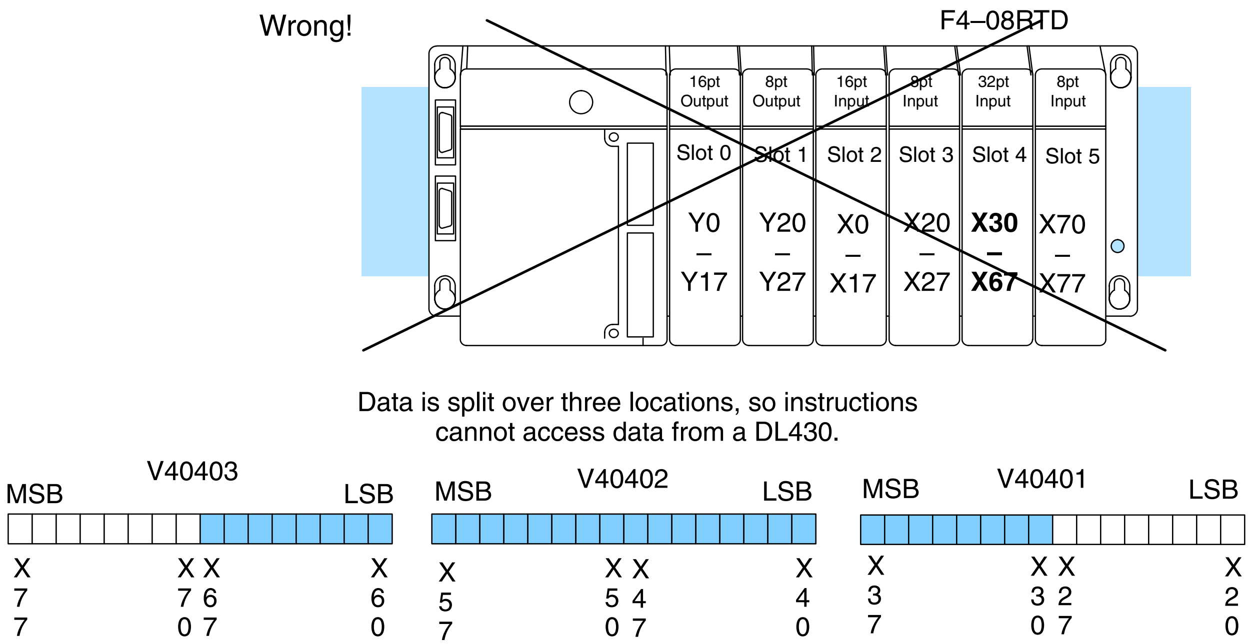
Notes:
1. The three wires connecting the RTD to the module must be the same type and length. Do not use the shield or drain wire for the third connection.
2. If a RTD sensor has four wires, the extra plus (+) sense wire should be left unconnected as shown.

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 that the input points do not start on a V-memory boundary, the instructions cannot access the data.



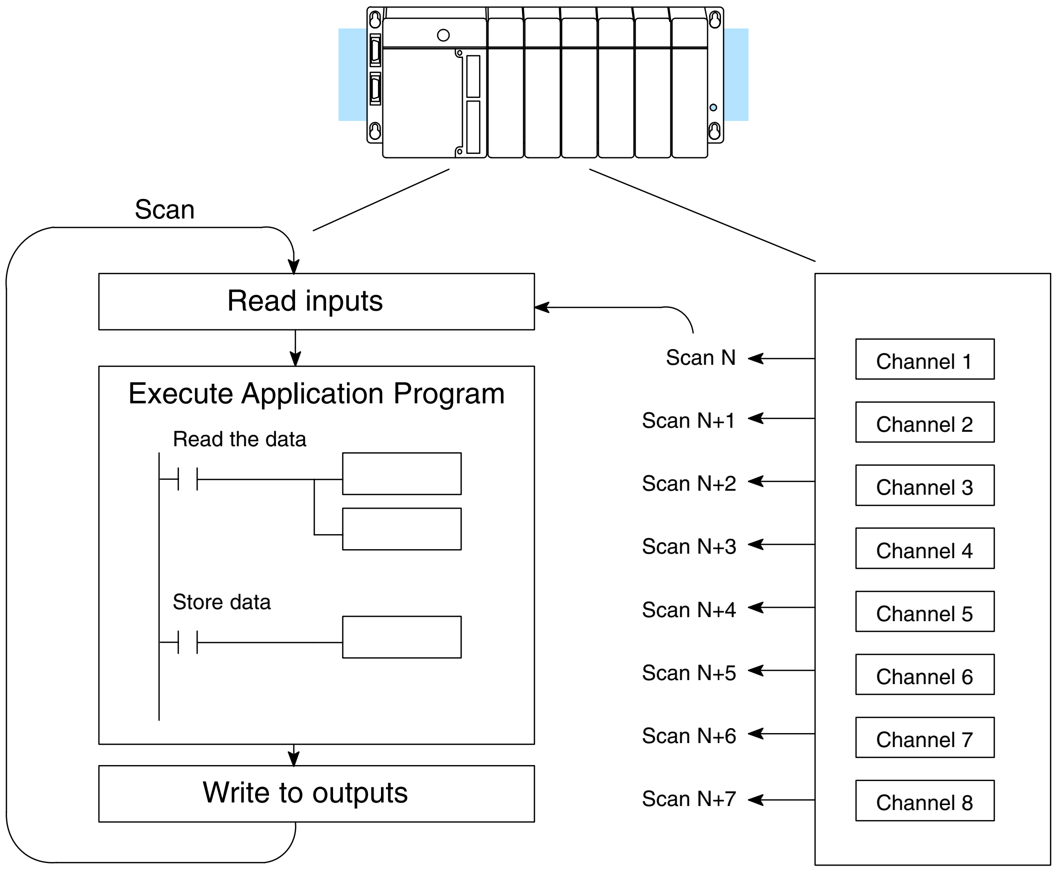


Channel Scanning Sequence

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 F4–08RTD 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.



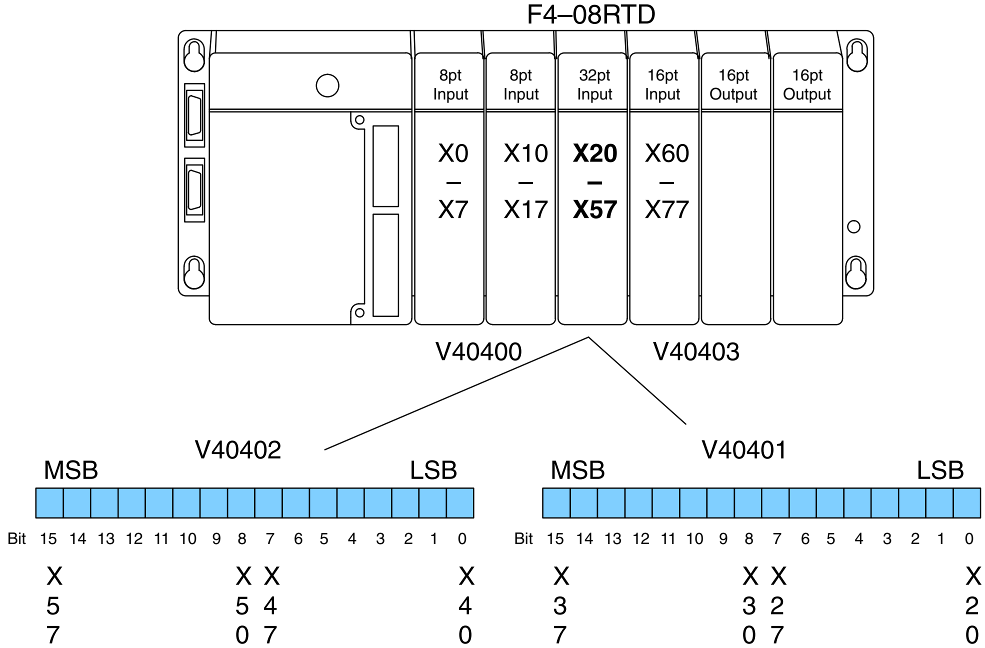
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the RTD transmitter signal and converts the signal to a 16-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

Identifying the Data Locations

The F4–08RTD module requires 32-point discrete input points. These inputs provide:

- Individual active channel bits for each channel.
- A digital representation of the analog signal in various data formats.
- Individual broken transmitter detection bits for each channel.

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



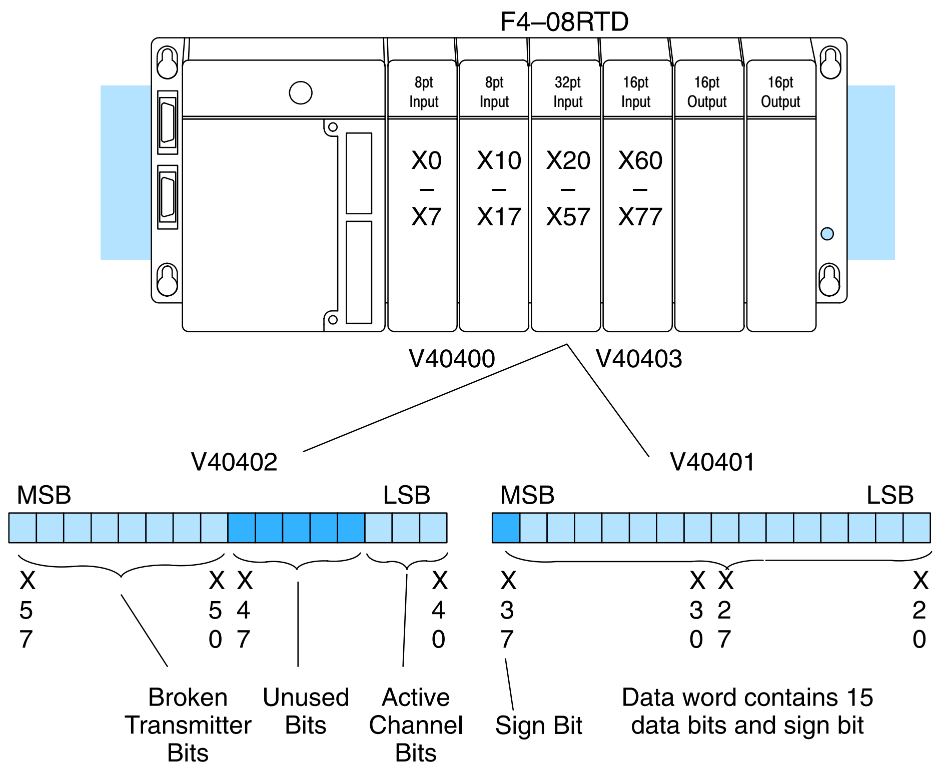
Writing the Control Program

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

Multiple Active Channels

The analog data is multiplexed into the lower word and is presented in 16 bits. The upper word contains three groups of bits that contain active channel status, unused bits, and broken transmitter status.

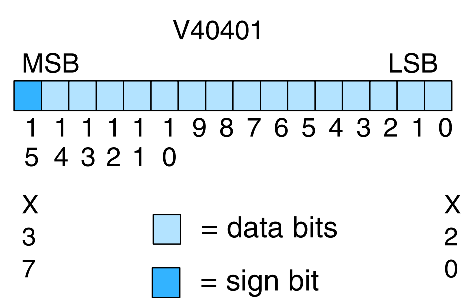
The control program must determine which channel's data is being sent from the module. If you have enabled only one channel, its data will be available on every scan. Two or more channels require demultiplexing the lower data word. Since the module communicates as X input points to the CPU, it is very easy to use the active channel status bits in the upper word to determine which channel is being monitored.



Analog Data and Sign Bits

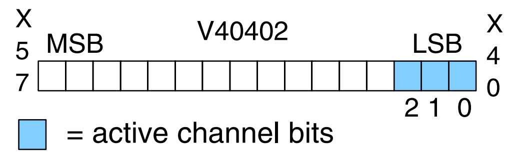
The first 16 bits represent the analog data in binary format. The MSB is the sign bit.

Bit	Value	Bit	Value
0	1	8	256
1	2	9	512
2	4	10	1024
3	8	11	2048
4	16	12	4096
5	32	13	8192
6	64	14	16384
7	128	15	32768



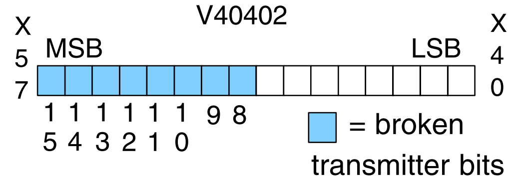
Active Channel Bits

The active channel bits represent the channel selections in binary format (000 = channel 1 is active, 001 = channel 2 is active, 111 = channel 8 is active, etc.).



Broken Transmitter Bits

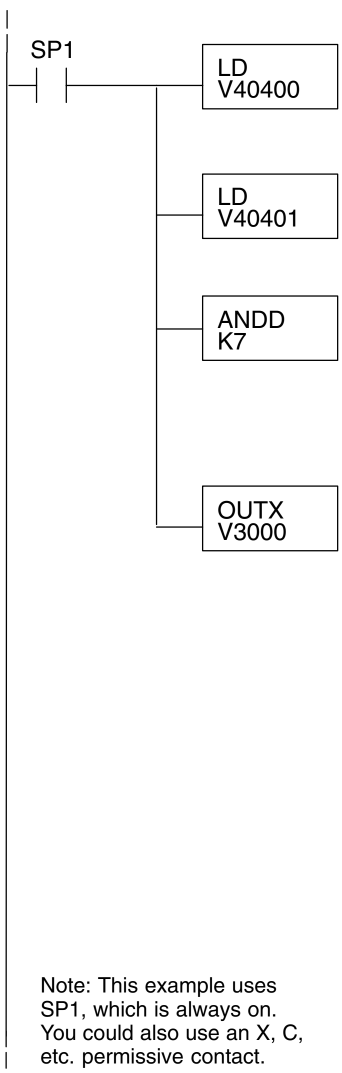
The broken transmitter bits are on when the corresponding RTD is open (00000001 = channel 1 is open, 00000010 = channel 2 is open, 11111111 = all eight channels are open, etc.).



Reading Values, DL430

430 440 450

This program example shows how to read the analog data into V-memory locations with the DL430 CPU (which does not support the LDF instruction) using the LD instruction. The example also works for DL440 and DL450 CPUs. The example reads one channel per scan, so it takes eight scans to read all the channels. Contact SP1 is used in the example because the inputs are continually being updated.



Loads all 16 bits of the channel data (first word) from the module into the lower 16 bits of the accumulator. This example assumes that the module location starts in the X0 position of the base.

Loads all 16 bits of the second data word from the module into the accumulator, and pushes the channel data (V40401) onto the first level of the stack.

ANDDs the value in the accumulator with the constant K7, which masks off everything except the three least significant bits (LSB) of V40401. The result is stored in the accumulator. The binary value of these bits (0–7, which is the offset) indicates which channel is being processed in that particular scan.

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 (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 channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

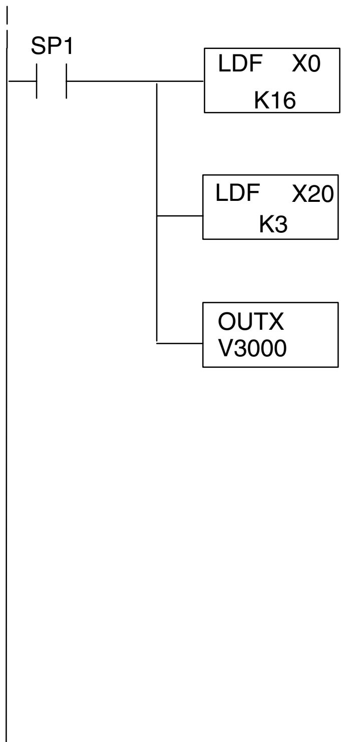
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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

Reading Values, DL440/450

430 440 450

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. Contact SP1 is used in the example because the inputs are continually being updated. This example will not work with DL430 CPUs.



Loads the 16 bits of channel data (starting with location X0) from the module into the accumulator.

Loads the binary value of the active channel bits (0–7) into the accumulator, and pushes the channel data loaded into the accumulator from the first LDF instruction onto the first level of the stack.

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 (0–7, 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 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

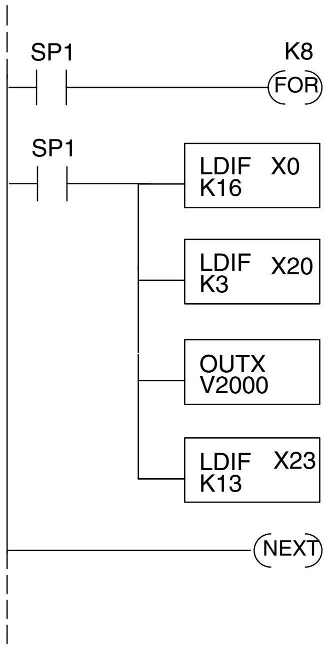
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Reading Eight Channels in One Scan, DL440/DL450

430 440 450

The following program example shows how to read all eight channels in one scan by using a FOR/NEXT loop. This program only works with DL440 and DL450 CPUs. Before you try this method, remember that the FOR/NEXT routine shown here will add about 10–12 ms to the overall scan time. If you don't 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.



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 four channels.

Immediately loads the first 16 bits of the data word (starting with X0) into the accumulator. The LDIF instruction will retreive the I/O points immediately without waiting on the CPU to finish the scan.

Immediately loads the three active channel bits of the status word (starting with X20) into the accumulator, and pushes the data word loaded into the accumulator from the first LDIF instruction into the first level of the stack. The value in the accumulator is the offset (0–7).

The OUTX instruction stores the channel data to an address that starts at V2000 plus the channel offset. For example, if channel 2 was being read, the data would be stored in V2001 (V2000 + 1). See the following table.

Increments the temperature reading to the next channel.

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

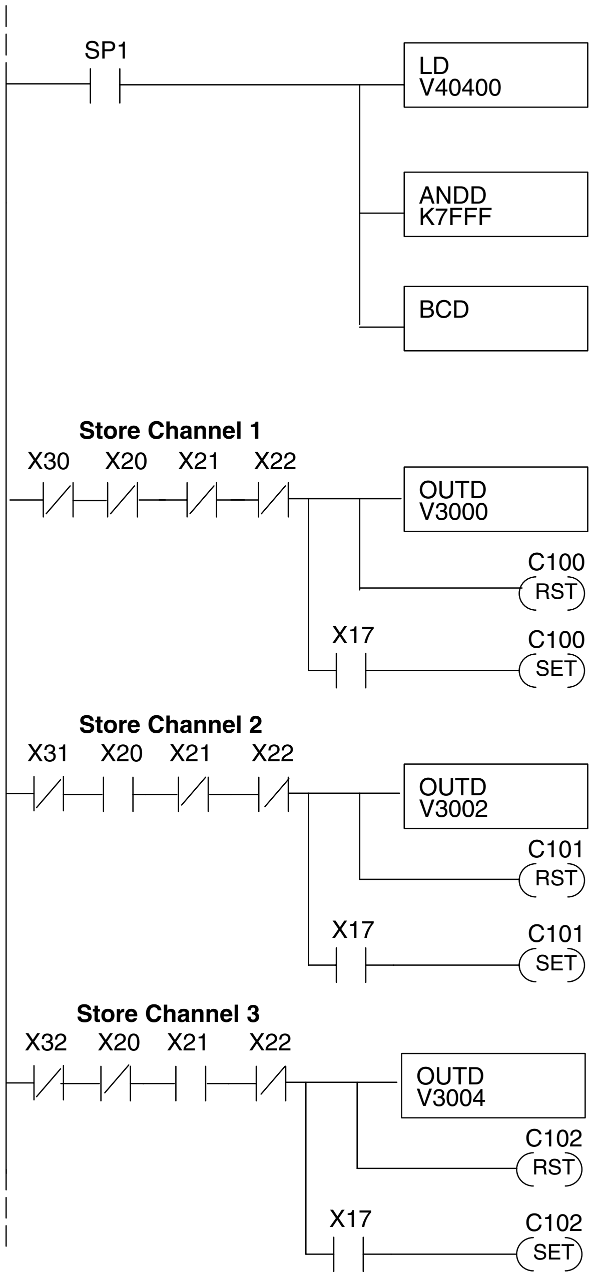
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V2000
Channel 2	001	1	V2001
Channel 3	010	2	V2002
Channel 4	011	3	V2003
Channel 5	100	4	V2004
Channel 6	101	5	V2005
Channel 7	110	6	V2006
Channel 8	111	7	V2007

Using Bipolar Ranges (Magnitude Plus Sign)

✓ ✓ ✓
430 440 450

With bipolar ranges, you need some additional logic because you need to know if the value being returned represents a positive voltage or a negative voltage. For example, you may need to know if the temperature is positive or negative.

The following program shows how you can accomplish this. Since you always want to know when a value is negative, these rungs should be placed *before* any operations that use the data, such as math instructions, scaling operations, and so forth. Also, if you are using stage programming instructions, these rungs should be in a stage that is always active. Although this example shows all eight channels, you only need the additional logic for those channels that are using bipolar input signals.



Loads the complete data word into the accumulator. The V-memory location depends on the I/O configuration. This example assumes the module is in the X0–X37 slot. See the CPU memory map.

This instruction masks off the channel data and excludes the sign bit. 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. Do not use with internal PID loops because the PV requires binary data.

This rung looks at fault bit X30 (the broken transmitter bit for channel 1) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 1 data is stored in V3000.

If the sign bit X17 is on, then control relay C100 is set. C100 can be used to indicate a negative channel 1 value or to call for a different message on an operator interface.

This rung looks at fault bit X31 (the broken transmitter bit for channel 2) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 2 data is stored in V3002.

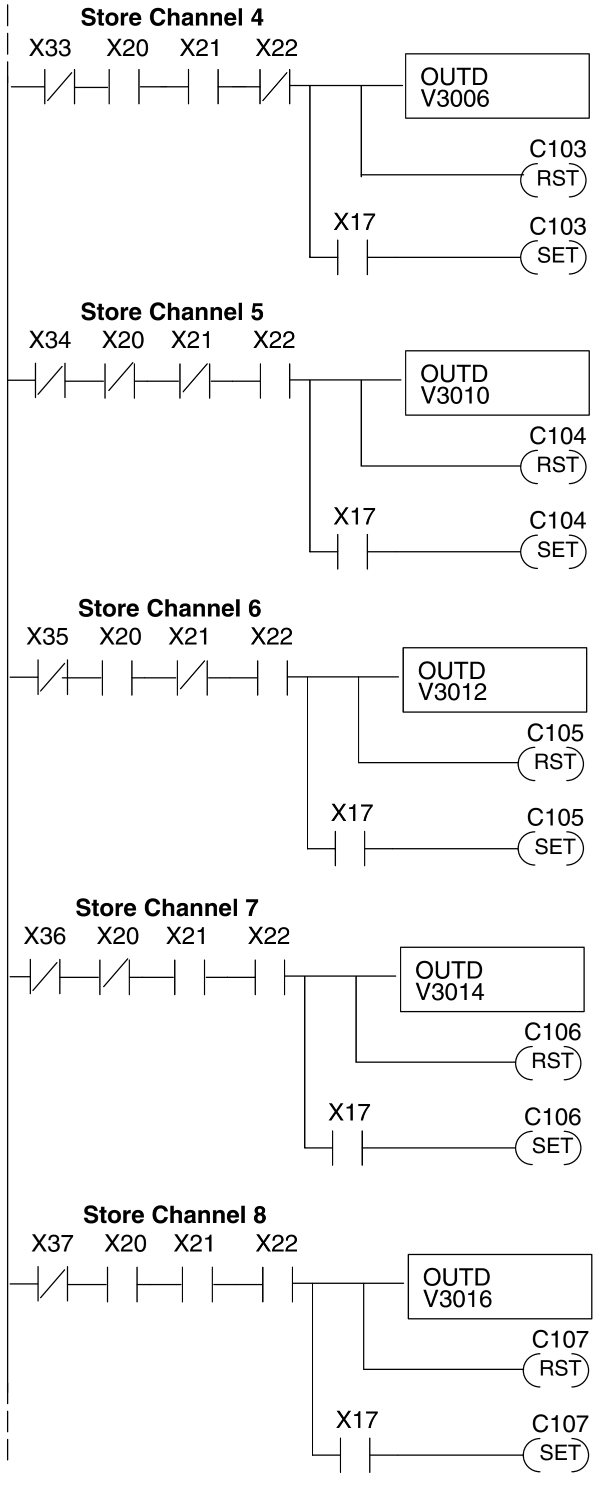
If the sign bit X17 is on, then control relay C101 is set. C101 can be used to indicate a negative channel 2 value or to call for a different message on an operator interface.

This rung looks at fault bit X32 (the broken transmitter bit for channel 3) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 3 data is stored in V3004.

If the sign bit X17 is on, then control relay C102 is set. C102 can be used to indicate a negative channel 3 value or to call for a different message on an operator interface.

Program is continued on the next page.

Using Bipolar Ranges Example Continued



This rung looks at fault bit X33 (the broken transmitter bit for channel 4) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 4 data is stored in V3006.

If the sign bit X17 is on, then control relay C103 is set. C103 can be used to indicate a negative channel 4 value or to call for a different message on an operator interface.

This rung looks at fault bit X34 (the broken transmitter bit for channel 5) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 5 data is stored in V3010.

If the sign bit X17 is on, then control relay C104 is set. C104 can be used to indicate a negative channel 5 value or to call for a different message on an operator interface.

This rung looks at fault bit X35 (the broken transmitter bit for channel 6) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 6 data is stored in V3012.

If the sign bit X17 is on, then control relay C105 is set. C105 can be used to indicate a negative channel 6 value or to call for a different message on an operator interface.

This rung looks at fault bit X36 (the broken transmitter bit for channel 7) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 7 data is stored in V3014.

If the sign bit X17 is on, then control relay C106 is set. C106 can be used to indicate a negative channel 7 value or to call for a different message on an operator interface.

This rung looks at fault bit X37 (the broken transmitter bit for channel 8) ANDed with active channel bits X20–X22. When the active channel bits are true and there is no transmitter fault, channel 8 data is stored in V3016.

If the sign bit X17 is on, then control relay C107 is set. C107 can be used to indicate a negative channel 8 value or to call for a different message on an operator interface.

Reading the Input Data

The RTD module is capable of converting both European and American type 100Ω RTDs and European type 1000Ω RTDs into direct temperature readings in (Fahrenheit or Celsius) for processing by the programmable controller. The temperature readings have one implied decimal point. For example, a reading of 10273 is actually 1027.3 degrees.