Errata Sheet

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

Product Family*:*   DL405

Manual Number   D4-ANLG-M

Revision and Date   5th Ed., Rev. A; July 2004

Date: September 12, 2018

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.
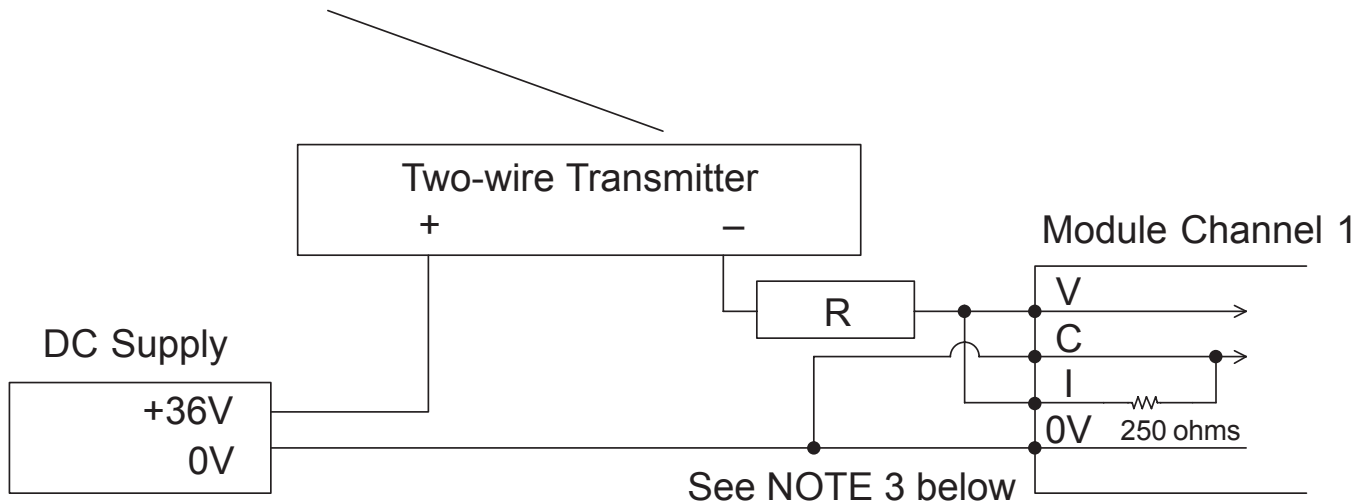
$R = Tr - Mr$ R – resistor to add

$R = 750 - 250$ Tr – Transmitter Requirement

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

Two-wire Transmitter
+   −

Module Channel 1

DC Supply
+36V
0V

R

V
C
I
0V   250 ohms

See NOTE 3 below

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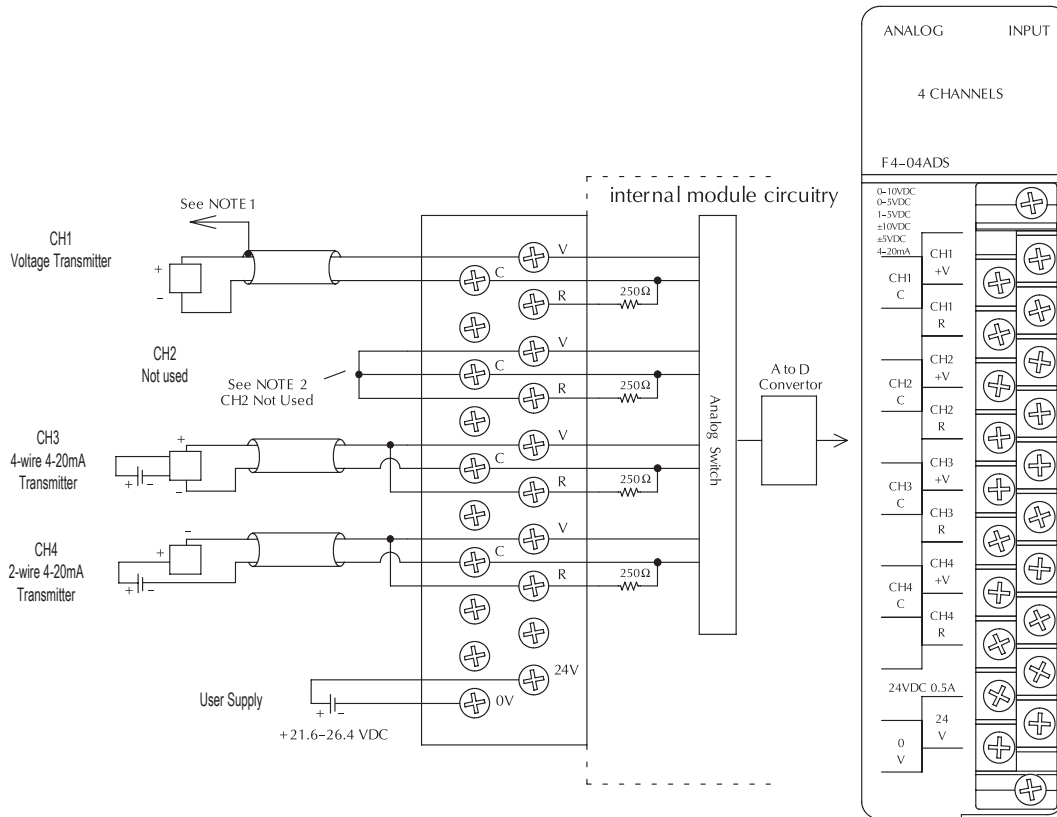
Errata Sheet

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

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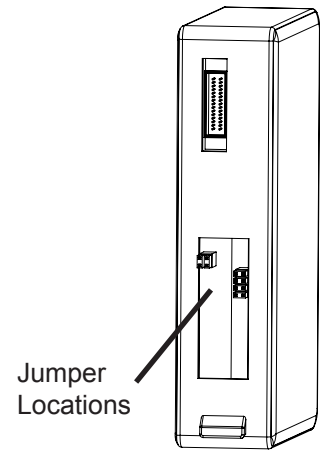
Changes to Chapter 8: F4-08THM-n 8-Channel Thermocouple Input

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.

With grounded thermocouples, take precautions to prevent having a voltage potential between thermocouple tips. A voltage of 1.25V or greater between tips will skew measurements.
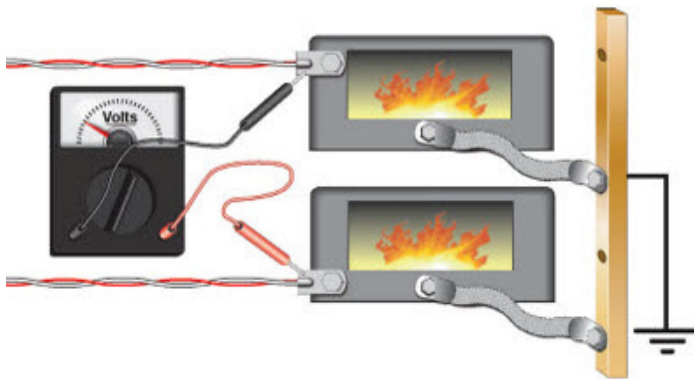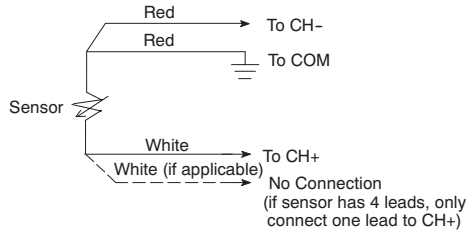
Errata Sheet

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.

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"

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.

F4-08THM-n
8-Channel
Thermocouple Input

8
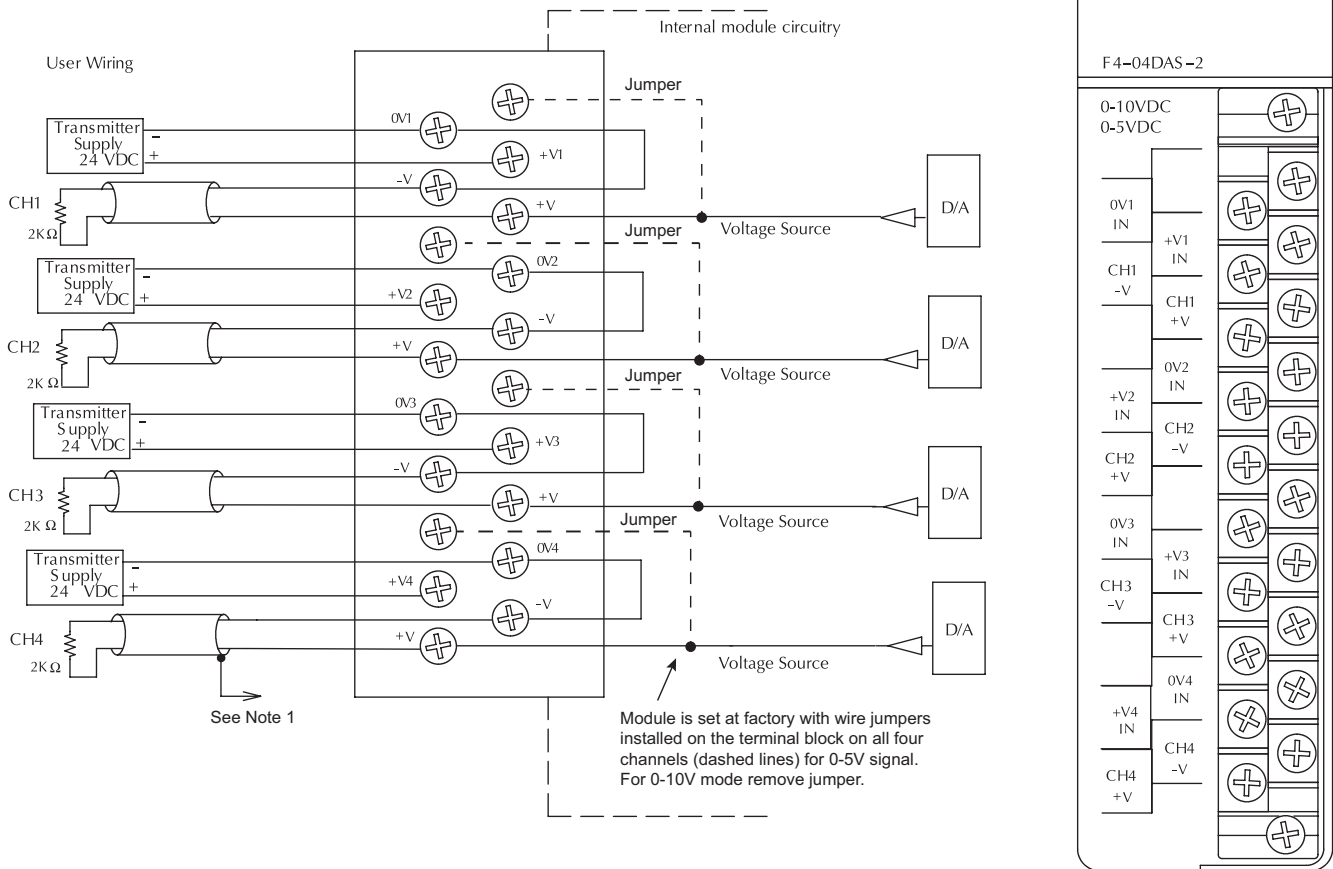
In This Chapter. . . .

Module Specifications

The F4–08THM–n 8-Channel Thermocouple Input module provides several features and benefits.

- It provides eight thermocouple input channels with 12-bit resolution.

- It automatically converts type E, K, R, S, J, B, C, P or T thermocouple signals into direct temperature readings. No extra scaling or complex conversion is required.

- Temperature data format is selectable between °F or °C or count operation.

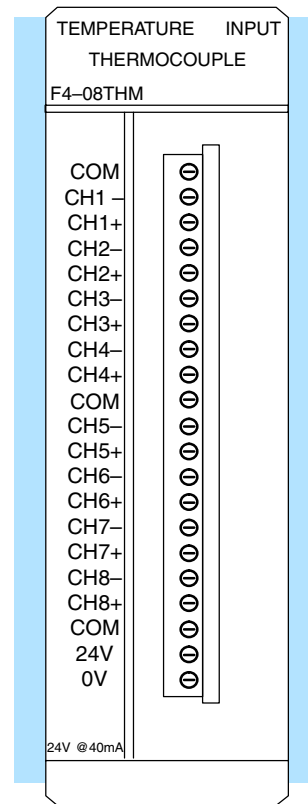- This module is also available in either 0–25mV or 0–100mV versions. They specifically convert millivolt signal levels into digital (0–4095) values.

- Signal processing features include automatic cold junction compensation, thermocouple linearization, and digital filtering.

- The temperature calculation and linearization are based on data provided by the National Institute of Standards and Technology (NIST).

- Diagnostic features include detection of thermocouple burnout or disconnection.

- Thermocouple burnout indication is a value of 4095. This will also indicate if the temperature goes below the minus (–) reading.

```
┌─────────────────────────┐
│ TEMPERATURE    INPUT    │
│      THERMOCOUPLE       │
│ F4–08THM                │
│                         │
│   COM      ⊖            │
│   CH1 –    ⊖            │
│   CH1+     ⊖            │
│   CH2–     ⊖            │
│   CH2+     ⊖            │
│   CH3–     ⊖            │
│   CH3+     ⊖            │
│   CH4–     ⊖            │
│   CH4+     ⊖            │
│   COM      ⊖            │
│   CH5–     ⊖            │
│   CH5+     ⊖            │
│   CH6–     ⊖            │
│   CH6+     ⊖            │
│   CH7–     ⊖            │
│   CH7+     ⊖            │
│   CH8–     ⊖            │
│   CH8+     ⊖            │
│   COM      ⊖            │
│   24V      ⊖            │
│    0V      ⊖            │
│                         │
│ 24V @40mA               │
└─────────────────────────┘
```

NOTE: This F4–08THM–n module differs from the F4–08THM module in that this module requires a specific module for each thermocouple type. For example, an F4–08THM–J only works with "J" type thermocouples. The F4–08THM module can be used with the common thermocouple types (J, K, E, etc.) by setting internal jumpers.

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

Input Specifications

Number of Channels	8, differential inputs
Input Ranges	Type E: −270/1000 °C, −450/1832 °F Type J: −210/760 °C, −350/1390 °F Type K: −270/1370 °C, −450/2500 °F Type R: 0/1768 °C, 32/3214 °F Type S: 0/1768 °C, 32/3214 °F Type T: −270/400 °C, −450/752 °F Type C: 60/2320 °C, 149/4208°F Type B: 529/1820 °C, 984/3594°F Type P: −99/1395 °C, −146/2543 °F −1: 0–50 mV −2: 0–100 mV −3: 0–25 mV
Resolution	12 bit (1 in 4096)
Input Impedance	27KΩ DC
Absolute Maximum Ratings	Fault-protected input, 130 Vrms or 100 VDC
Cold Junction Compensation	Automatic
Conversion Time	15ms per channel, minimum 1 channel per CPU scan
Converter Type	Successive approximation
Linearity Error	±1 count (0.03% of full scale) maximum
Full Scale Calibration Error	0.35% of full scale
Maximum Inaccuracy*	±1° C type J,K,E,T thermocouples
	±3° C type R,S,B,C,P thermocouples

* Maximum Inaccuracy rating is guaranteed for temperatures above −220°C for types E, T, J, and K, and above +100°C for types R and S.

General Specifications

PLC Update Rate	8 channel per scan max.
Digital Input Points Required	16 (X) input points, including 12 binary data bits, 3 channel ID bits, 1 sign bit
Power Budget Requirement	120 mA @ 5 VDC (from base)
External Power Supply	24 VDC ±10%, 50 mA current
Operating Temperature	0° to 60° C (32° to 140° F)
Storage Temperature	−20° to 70° C (−4° to 158° F)
Accuracy vs. Temperature	±57 ppm / °C maximum full scale
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

Module Calibration

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

Thermocouple Input Configuration Requirements

The F4–08THM–n Thermocouple Input Module requires16 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.
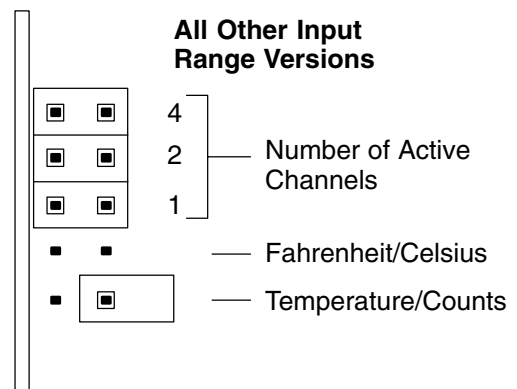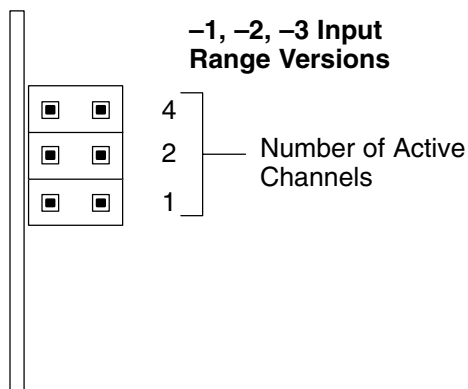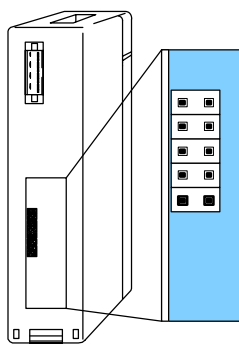- For remote I/O systems, the available power budget and number of remote I/O points.

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.

Setting the Module Jumpers

Jumper Locations

At the rear of the module is a bank of three or five jumpers, depending on the module version. The module has options that you can select by installing or removing these jumpers:

- All modules may be set to select from one to eight active channels.
- All but –1, –2, and –3 version modules may be set to select between Fahrenheit or Celsius temperature conversion, and between temperature or counts data format.

–1, –2, –3 Input Range Versions

4
2 — Number of Active
1 — Channels

All Other Input Range Versions

4
2 — Number of Active
1 — Channels

—— Fahrenheit/Celsius

—— Temperature/Counts

Factory Default Settings

By default, the –1 (50mV), –2 (100mV), and –3 (25mV) input version modules arrive from the factory as shown above with all three jumpers installed. With these jumpers installed, the module has eight active channels.

All other modules arrive from the factory with the top three jumpers installed and the bottom two jumpers not installed as shown above. Notice there is an extra jumper placed over one of the bottom pins as shown (this is a good way to store removed jumpers so they do not get lost). With the top three jumpers installed and the bottom two jumpers removed, the module has eight active channels, and converts temperatures into Celsius readings.

Selecting the Number of Channels

The three jumpers closest to the top of the module are binary encoded to select the number of channels that will be used. Channels must be used contiguously, starting with channel 1. For example, if you are using three channels, you must use channels 1 thru 3, not 2 thru 4 or 5 thru 7, etc.

Any unused channels are not processed, so if you only select the first four channels, then the last four channels will not be active. Use the following table to set jumpers. For example, to select 5 channel operation, install jumper 4 and remove jumpers 1 and 2.

yes = jumper installed

empty space = jumper removed

Number of Channels	Jumper Pins		
	4	2	1
1			
2			yes
3		yes	
4		yes	yes
5	yes		
6	yes		yes
7	yes	yes	
8	yes	yes	yes
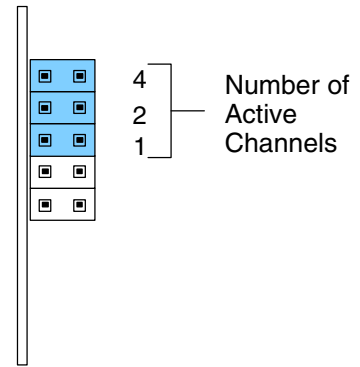
**Selecting Fahrenheit or Celsius**

The fourth jumper down selects between Fahrenheit or Celsius units.

For Celsius, **remove** the jumper.

For Fahrenheit, **install** the jumper.

If the bottom jumper is installed (set for Counts), then this jumper is inactive and can be installed or removed with no effect on the module operation.

Remember, –3, –2, and –1 input range versions do not have this jumper.
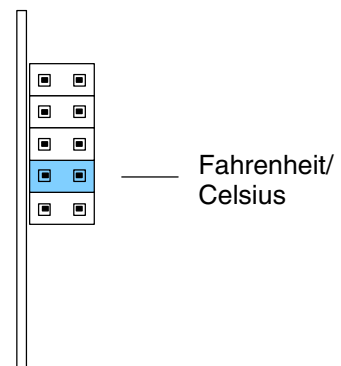
Selecting Temperature or Counts

The jumper closest to the bottom of the module selects between conversion to units of temperature or to a binary count ranging from 0 to 4094.

For Temperature format, **remove** the jumper.

For Counts format, **install** the jumper.

Remember, –3, –2, and –1 input range versions do not have this jumper.

Temperature/ Counts
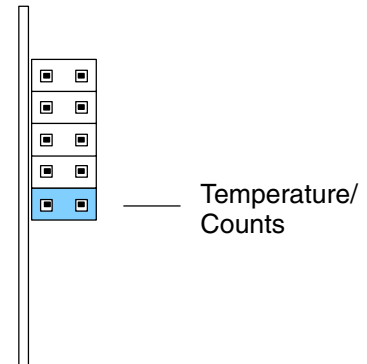
Connecting the Field Wiring

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.
- 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 F4–08THM–n requires a separate power supply. The CPU, 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. You can use this supply to power the Thermocouple Input module. If you already have modules that are using all of the available power from this supply, or if you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ±10%, Class 2, 50mA current.

Wiring Diagram

Note 1: Terminate shields at the respective signal source.

Note 2: Leave unused channels open (no connection).

Internal Module Wiring

A/D

See Note 1

C

CH1

−1

+1

Examples of differential
Thermocouple wiring

−2

+2

CH3

−3

+3

CH4

−4

+4

C

Analog
Switch

CH5

−5

+5

−6

+6

Examples of grounded
Thermocouple wiring

−7

+7

−8

CH8

+8

COM

User Supply

24V+

+

0V

−

24VDC+ 10% Class 2

TEMPERATURE   INPUT

THERMOCOUPLE

F4–08THM–n

COM
CH1 −
CH1+
CH2−
CH2+
CH3−
CH3+
CH4−
CH4+
COM
CH5−
CH5+
CH6−
CH6+
CH7−
CH7+
CH8−
CH8+
COM
24V
0V

24V @40mA
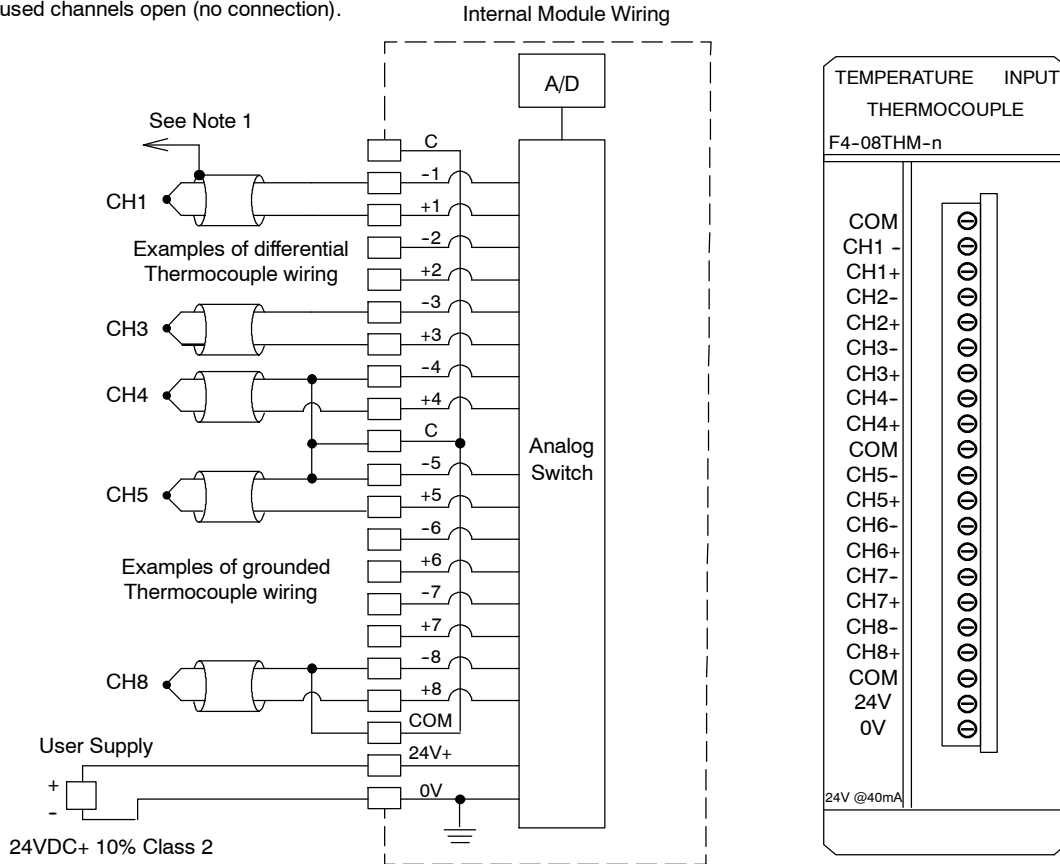
See Errata Sheet at the beginning of
this file. There is a note and drawing
regarding grounded thermocouples.

Module Operation

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.

DL430 Special Requirements

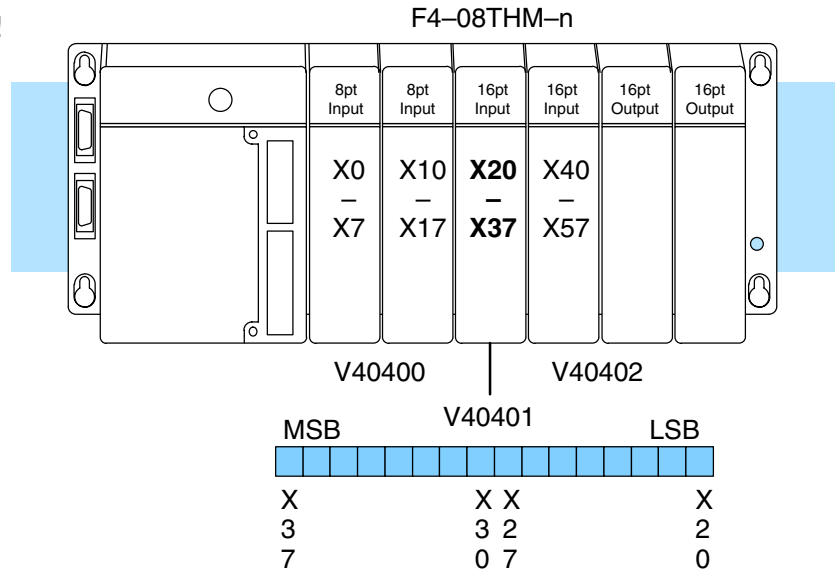Even though the module can be placed in any slot, it is important to examine the configuration if you are using a DL430 CPU. As you will see in the section on writing the program, you use V-memory locations to extract the analog data. As shown in the following diagram, if you place the module so input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

Correct!

F4–08THM–n

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

X 3 7   X 3 0   X 2 7   X 2 0

Wrong!

F4–08THM–n

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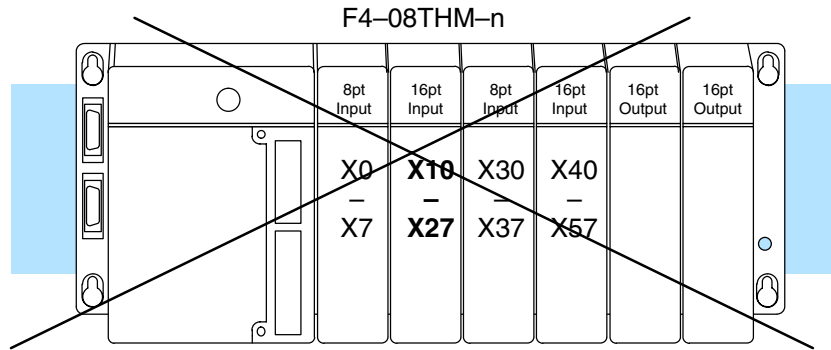
Data is split over two locations, so instructions cannot access data from a DL430.

MSB   V40401   LSB   MSB   V40400   LSB

X 3 7   X 3 0   X 2 7   X 2 0   X 1 7   X 1 0   X 7   X 0

Channel Scanning Sequence

The F4–08THM–n 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 and later we will 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.
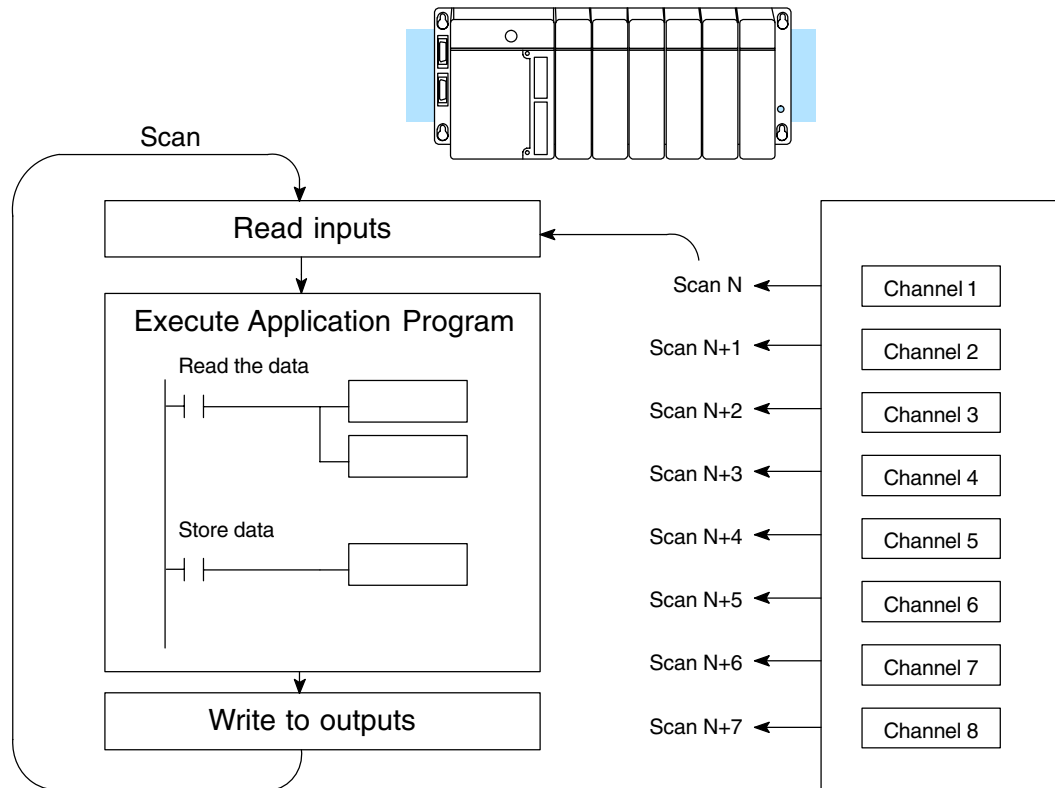
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the thermocouple transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

**Identifying the
Data Locations**

You may recall the Thermocouple Input 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 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.

F4–08THM–n
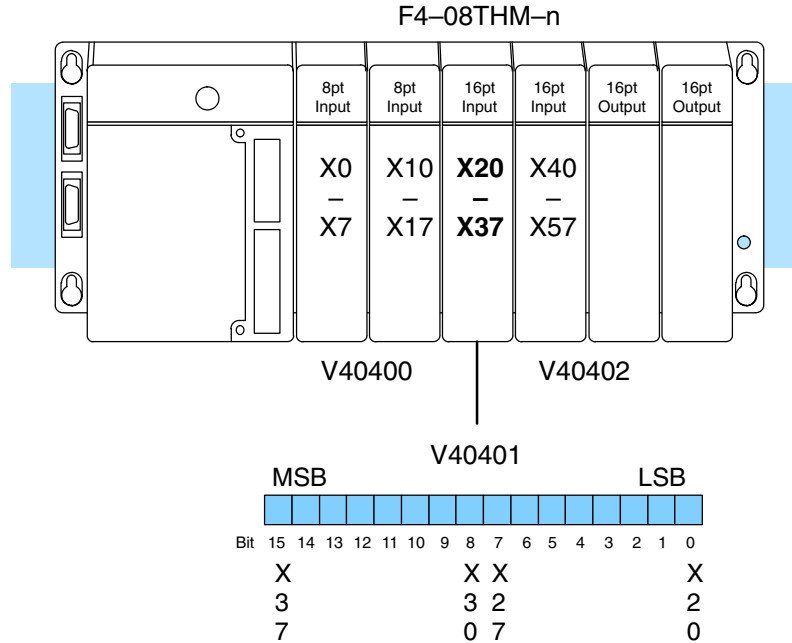
Within this word location, the individual bits represent specific information about the analog signal.

**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.
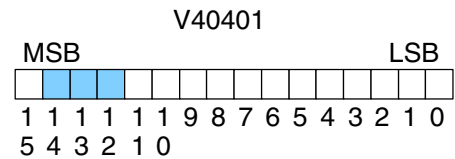
☐ – channel inputs

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
N+8	000	1

Temperature Sign Bit

The most significant bit is used to note the sign of the temperature. If this bit is on, then the temperature is negative. If the bit is off, then the temperature is positive.

V40401

MSB   LSB

1 1 1 1 1 1 9 8 7 6 5 4 3 2 1 0
5 4 3 2 1 0

☐ – temperature sign

Analog Data Bits

The first twelve bits represent the temperature. If you have selected the 0–4095 scale (counts), the following format is used.

V40401

MSB   LSB

1 1 1 1 1 1 9 8 7 6 5 4 3 2 1 0
5 4 3 2 1 0

☐ – data bits
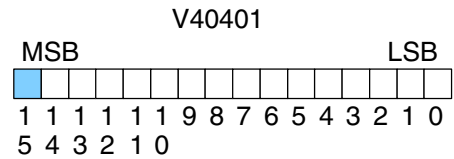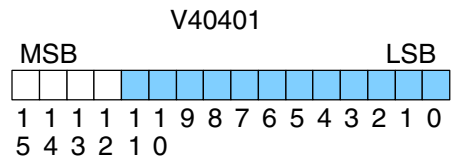
Bit	Value	Bit	Value
0 (LSB)	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048

Temperature Input Resolution

Typically, the F4–08THM–n resolution enables you to detect a 1°F change in temperature. The National Institute of Standards and Technology (NIST) publishes conversion tables that show how each temperature corresponds to an equivalent signal level.

Millivolt Input Resolution

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 –2 (100mV) module a signal of 0 mV would be 0, and a signal of 100 mV 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 diagram shows how this relates to the example signal range.

0–100 mV Scale

100mV

0 mV

0   4095
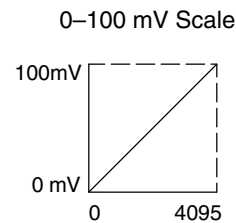
Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal levels that will result in a change in the data value for each signal range.

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

H = high limit of the signal range
L = low limit of the signal range

Range	Signal Span (H – L)	Divide By	Smallest Detectable Change
0 to 25 mV	25 mV	4095	6.1 μV
0 to 50 mV	50 mV	4095	12.2 μV
0 to 100 mV	100 mV	4095	24.4 μV

Writing the Control Program

Multiple Channels Selected

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 demultiplex the individual channel information

F4–08THM–n

		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

Temperature Sign Bit

Channel Indicator Bits

Data Bits
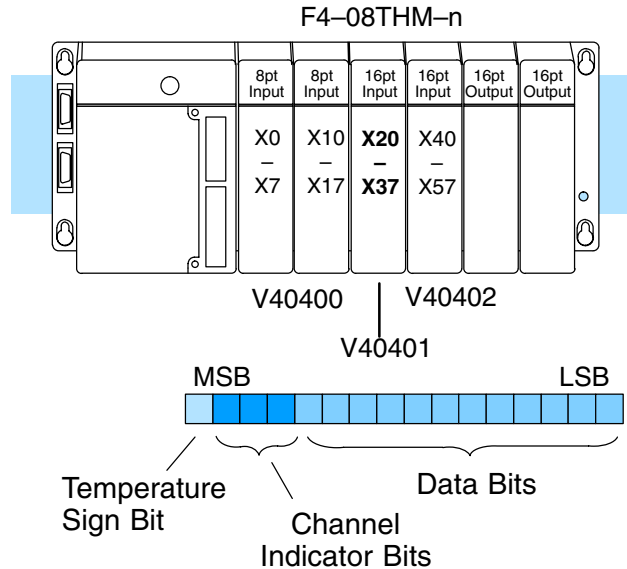
Automatic Temperature Conversion

If you are using the temperature scale (°F or °C) then you do not have to perform any scaling. Once you convert the binary temperature reading to a four-digit BCD number, you have the temperature.

Reading Values, DL430

✓ ✓ ✓
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.

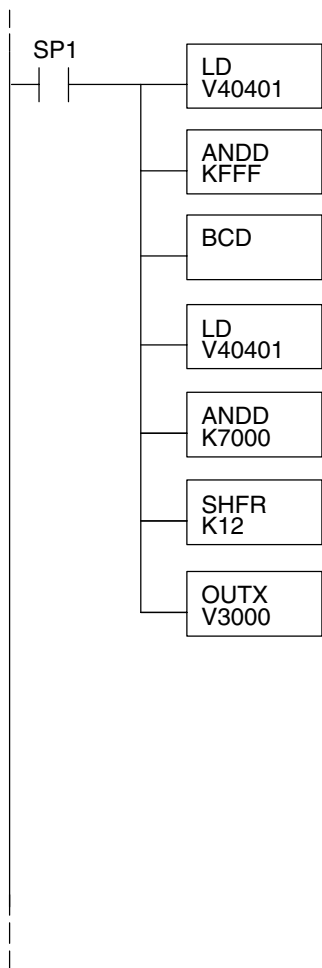| SP1		
	LD V40401	Loads the complete channel data word from the module into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map.
	ANDD KFFF	This instruction masks the channel identification bits. Without this, the values used will not be correct, 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	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

Single Channel Selected

✔ ✔ ✔
430 440 450

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

SP1 — LD or LDF
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

✕ ✔ ✔
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.
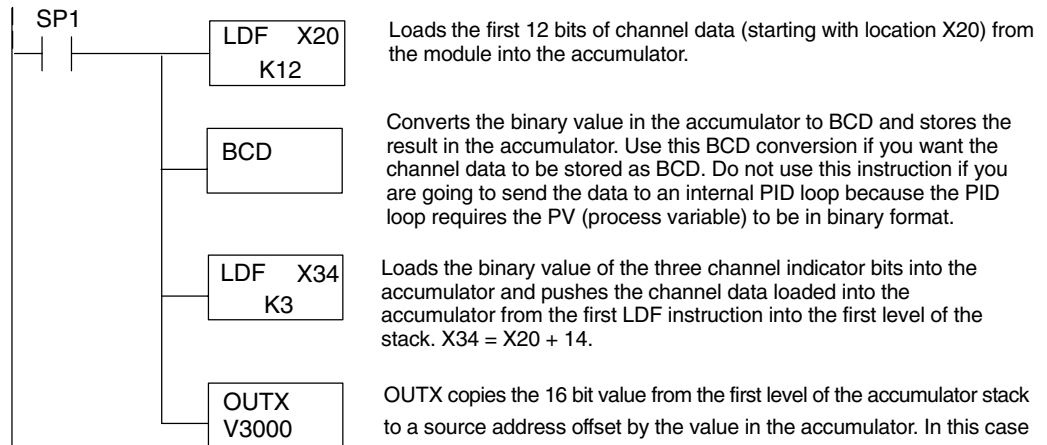
SP1 — LDF   X20
            K12
Loads the first 12 bits of channel data (starting with location X20) from the module into the accumulator.

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 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.
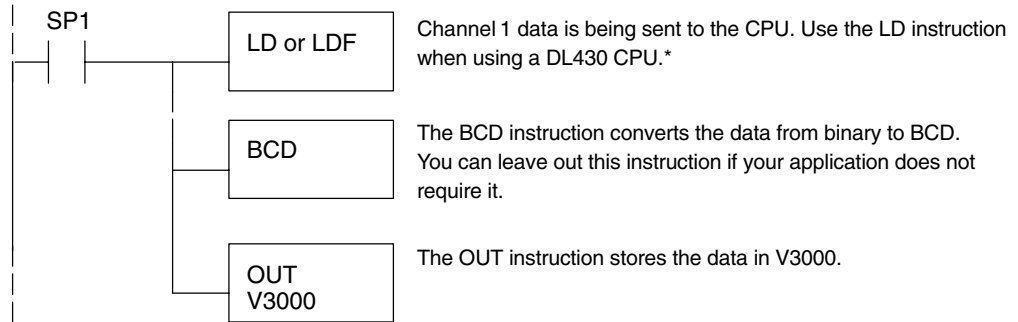
OUTX
V3000
OUTX copies the 16 bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (which is the offset) to V3000. The particular channel data is then stored in its respective location: For example, 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.

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

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 10–12ms 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.
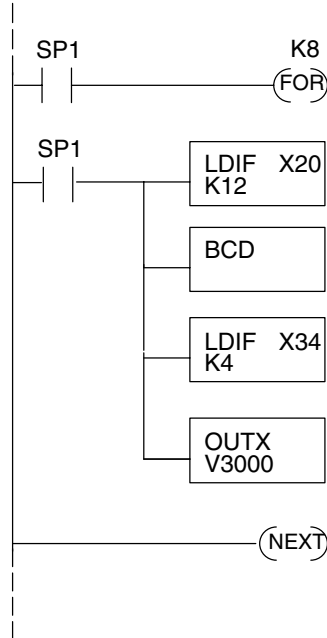
Starts the FOR/NEXT loop. The constant (K8) specifies how many times the loop will execute. Enter a constant equal to the number of channels you are using. For example, enter K4 if you are using 4 channels.

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.

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).

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, which is why K4 is used. Otherwise, only one channel will be read.

The OUTX instruction stores the channel data to an address that starts at V3000 plus the channel offset (0–7). For example, if channel 3 was being read, the data would be stored in V3002 (V3000 + 2). 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.

Ladder diagram:

```
  SP1                    K8
  | |                   (FOR)

  SP1          ┌──────────┐
  | |          │ LDIF  X20│
               │ K12      │
               └──────────┘
               ┌──────────┐
               │ BCD      │
               └──────────┘
               ┌──────────┐
               │ LDIF  X34│
               │ K4       │
               └──────────┘
               ┌──────────┐
               │ OUTX     │
               │ V3000    │
               └──────────┘
                         (NEXT)
```

Using the Sign Bit, DL440/450

✕	✓	✓
430	440	450

By adding a couple of simple rungs you can easily monitor the temperature for positive vs. negative readings. For example, if you have to know whether the temperature is +100 °F or −100 °F, an easy way to do this is to use the channel indicator inputs *and* the sign bit to set a control relay when the temperature is negative. For example, assume Channel 2 is the only channel you expect to receive both positive and negative temperatures. Notice we have added some logic for Channel 2 to set a control relay when the temperature is negative.

The example shown here uses the logic for a DL440 or DL450 CPU, but you could just as easily use the sign bit logic with any of the other methods.

SP1

| LDF   X20 |
| K12 |
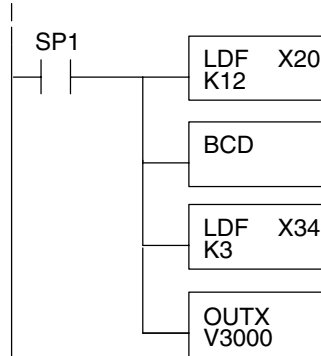
Loads the first 12 bits of the data word into the accumulator. The X address depends on the I/O configuration.

| BCD |

Since the module automatically converts the temperature reading to its binary equivalent, just convert the data to BCD to get the temperature.

| LDF   X34 |
| K3 |

This LDF instruction loads the three channel indicator bits into the accumulator. The channel data is pushed onto a stack.

| OUTX |
| V3000 |

The OUTX (out indexed) instruction is used to store the channel data, currently the first item on the stack, to an address that starts at V3000 plus the channel offset (0–7) which is located in the accumulator. For example, if channel two was being read, the data would be stored in V3001 (V3000 + 1).

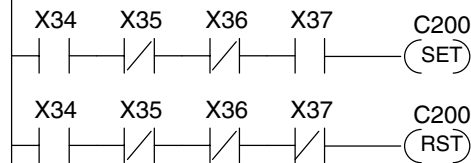Note, this example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

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

X34 X35 X36 X37   C200
                          (SET)

If X37 is on, then the temperature on channel 2 is negative.

X34 X35 X36 X37   C200
                          (RST)

If X37 is off, then the temperature on channel 2 is positive.

Scaling the Input Data

The Thermocouple Input module automatically converts the temperature readings into the digital equivalent, so as long as you are using the module to monitor temperatures you never have to perform any scaling. However, there are two situations where you will probably want to understand how to scale the data.

- When you use the –1 (50mV), –2 (100mV), or –3 (25mV) versions, the millivolt signals are represented by digital values between 0 and 4095. These values may actually represent pressure, position, etc.

- When you use the CNTS (counts) option instead of temperature, the temperature range is converted into a digital value between 0 and 4095. This is especially useful when you use this module in conjunction with PID control loops.

The scaling 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 were using the millivolt input version and you wanted to measure pressure (PSI) from 0.0 to 99.9, 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

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

$$\text{Units} = 20240 \frac{100 - 0}{4095}$$
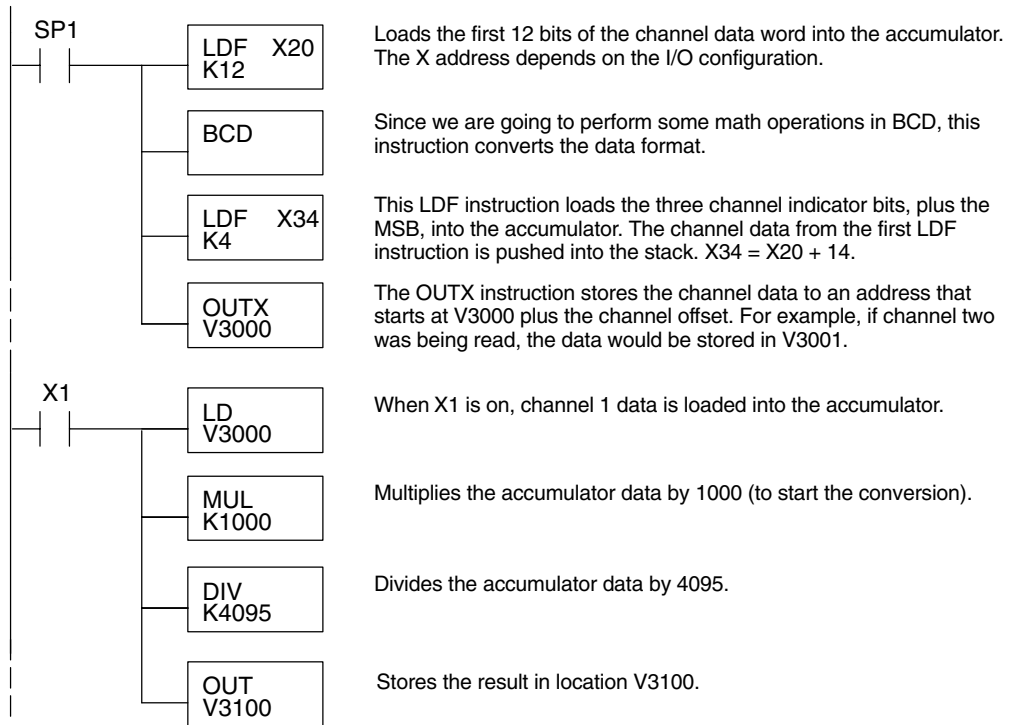
$$\text{Units} = 494$$

Handheld Display

```
   V 3101  V 3100
V MON   0000   0494*
```

*Value is more accurate

The following example shows 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 │   Loads the first 12 bits of the channel data word into the accumulator.
                  │ K12       │   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 │   This LDF instruction loads the three channel indicator bits, plus the
                  │ K4        │   MSB, into the accumulator. The channel data from the first LDF
                  │           │   instruction is pushed into the stack. X34 = X20 + 14.

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

  X1
 ┤ ├──────┬─────┤ LD         │   When X1 is on, channel 1 data is loaded into the accumulator.
                  │ V3000     │

                  ┤ MUL       │   Multiplies the accumulator data by 1000 (to start the conversion).
                  │ K1000     │

                  ┤ DIV       │   Divides the accumulator data by 4095.
                  │ K4095     │

                  ┤ OUT       │   Stores the result in location V3100.
                  │ V3100     │
```

Temperature and Digital Value Conversions

Since the thermocouple devices are non-linear, it is much easier to rely on published standards for conversion information. The National Institute of Standards and Technology (NIST) publishes conversion tables that show how each temperature corresponds to an equivalent signal level.

Millivolt 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.

mV Range	If you know the digital value ...	If you know the analog signal level ...
0 to 25 mV	$A = \dfrac{25D}{4095}$	$D = \dfrac{4095}{25}A$
0 to 50 mV	$A = \dfrac{50D}{4095}$	$D = \dfrac{4095}{50}A$
0 to 100 mV	$A = \dfrac{100D}{4095}$	$D = \dfrac{4095}{100}A$

For example, if you are using a –2 (100mV) version and you have measured the signal as 30 mV, you would use the following formula to determine the digital value that should be stored in the register location that contains the data.

$$D = \frac{4095}{100}A$$

$$D = \frac{4095}{100}(30)$$

$$D = (40.95)(30)$$

$$D = 1229$$