F4–04DA–2
4-Channel Analog
Voltage Output

In This Chapter. . . .

Module Specifications

The F4–04DA–2 Analog Voltage Output Module provides several features and benefits.

- It is a direct replacement for the popular F4–04DA module in applications set for voltage outputs on all channels.

- Analog outputs 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 four analog outputs may be set in one CPU scan (DL440 and DL450 CPUs only).

- It provides four channels of voltage outputs, independently configured for the following voltage ranges:

    1) 0–5 VDC
    2) 0–10 VDC
    3) ± 5 VDC
    4) ± 10 VDC
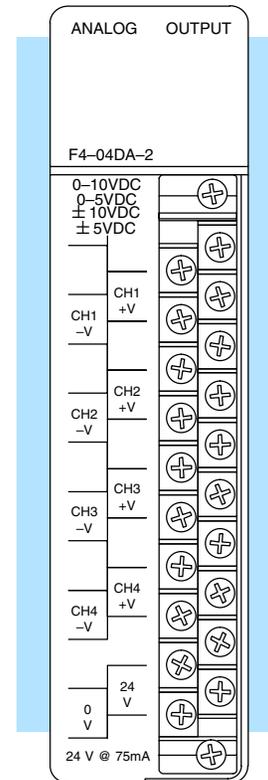
Analog Output Configuration Requirements

The F4–04DA–2 Analog Output requires 16 discrete output points in 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 table provides the specifications for the F4–04DA–2 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

| Number of Channels	4, single ended (one common)
Output Ranges	0–5, 0–10, ±5 , ±10 VDC
Resolution	12 bit (1 in 4095)
Load Impedance	2KΩ minimum
Load Capacitance	0.01 uF maximum
Voltage Output Current	5.0 mA sink or source
Short-circuit Current	15 mA typical
Linearity Error (end to end) and Relative Accuracy	±1 count (±0.025%) maximum
Offset Calibration Error	±3 counts maximum, unipolar ±4 counts maximum, bipolar
Full Scale Calibration Error	±8 counts maximum, (offset error included)
Maximum Inaccuracy	±0.2% @ 25°C (77°F) ±0.4% @ 0 to 60° C (32 to140°F)
Conversion Time	5 μs maximum, settling time 2.0 ms maximum, digital out to analog out

General Module Specifications

Digital Output Points Required	16 point (Y) outputs, 12 bits binary data, 4 active channel bits or 2 active channel bits and 1 sign bit for bipolar
Power Budget Requirement	90 mA @ 5 VDC (from base)
External Power Supply	21.6–26.4 VDC, 90 mA, class 2 (outputs fully loaded)
Accuracy vs. Temperature	±57 ppm / °C full scale calibration change (including maximum offset change, 2 counts)
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 user selectable options which are determined by jumper settings. At the rear of the module are two blocks of jumpers, accessible in the housing cutout. Use the upper block of four jumpers to select the data format for each channel. The lower jumper block sets the voltage ranges individually for each of the four channels.
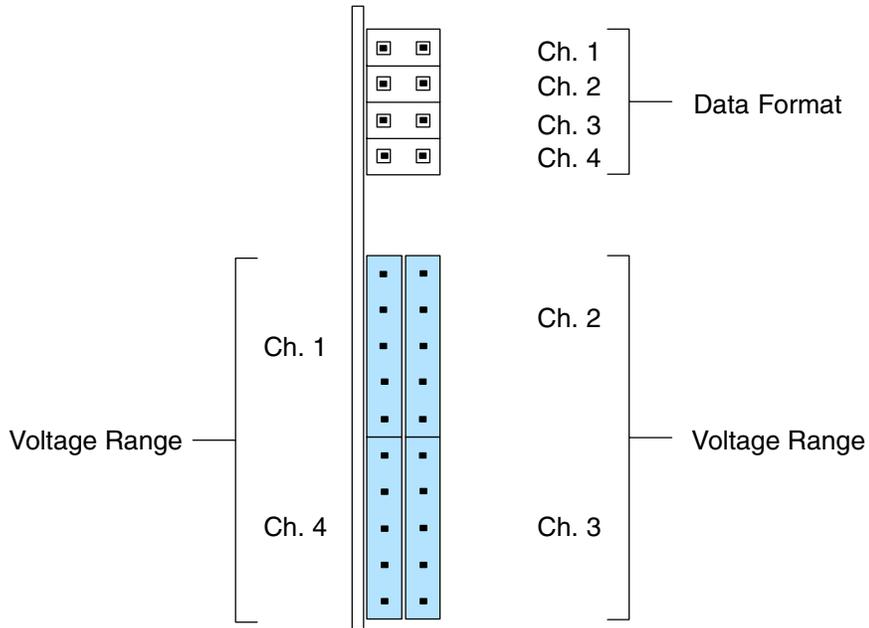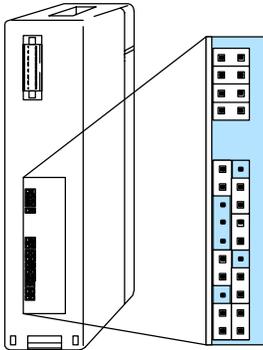
NOTE: If you are replacing an F4–04DA module with this one in an existing application, skip to the special section at the end of this chapter "Configuration Cross-Reference F4–04DA to F4–04DA–2".

Factory Default Settings

The factory default jumper settings has all four data format jumpers installed, and the output voltage range selection for each channel is set for 0–5 VDC. With all four data format jumpers installed, the module accepts numbers from the CPU from 0 to 4095 for each channel. These numbers convert to the output voltage range of 0–5 VDC.

Voltage Range Selection

The lower jumper block is a 2-by-10 array of posts. Functionally, the jumper posts are separated into four 1-by-5 columns, as shown to the right. Each group corresponds to a particular analog channel, as labeled. The five posts can receive one or two jumpers, selecting one of four voltage output ranges. In fact, it's possible to set each channel for a different range. The following table shows how to select the voltage range for a single channel.
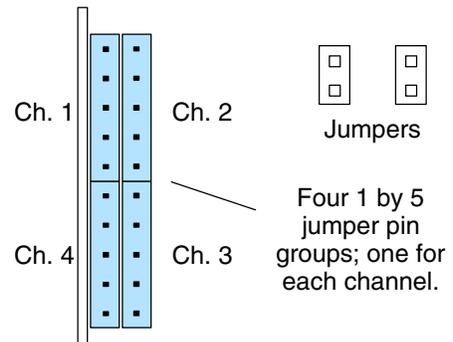
Unipolar Ranges	Jumper Settings	Bipolar Ranges	Jumper Settings
0 VDC to 5 VDC		−5 VDC to +5 VDC	
0 VDC to 10 VDC		−10 VDC to +10 VDC	

The table above shows how to set any output channel for one of four voltage ranges. Since each channel may be set independently of the others, there are many possible combinations. The following examples show how the whole voltage range jumper block will appear after setting all four channels for the ranges listed.

Example 1	Jumper Settings	Example 2	Jumper Settings
Ch. 1 ± 10 VDC		Ch. 1 0 to 5 VDC	
Ch. 2 ± 5 VDC		Ch. 2 (same)	
Ch. 3 0 to 5 VDC		Ch. 3 (same)	
Ch. 4 0 to 10VDC		Ch. 4 (same)	

Example 3	Jumper Settings	Example 4	Jumper Settings
Ch. 1 ± 10 VDC		Ch. 1 ± 10 VDC	
Ch. 2 (same)		Ch. 2 0 to 5 VDC	
Ch. 3 (same)		Ch. 3 ± 5 VDC	
Ch. 4 (same)		Ch. 4 (same)	
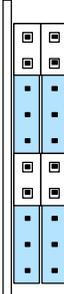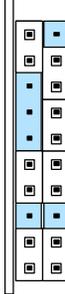
Data Format Selection

The upper jumper block consists of four pairs of jumper posts. Functionally, the jumper posts are separated into four horizontal pairs, as shown to the right. Each jumper site corresponds to a particular channel as labeled.

Installing a jumper sets the corresponding channel for 0 to 4095 data format conversion range. This is typically used with unipolar voltage range selections. The sign bit **is not** active in this data format.

Removing a jumper sets the corresponding channel for –2047 to +2047 data format conversion range. This is typically used with bipolar voltage range selections. The sign bit **is** active in this data format.

Ch 1

Ch 2

Ch 3

Ch 4

Four jumper sites
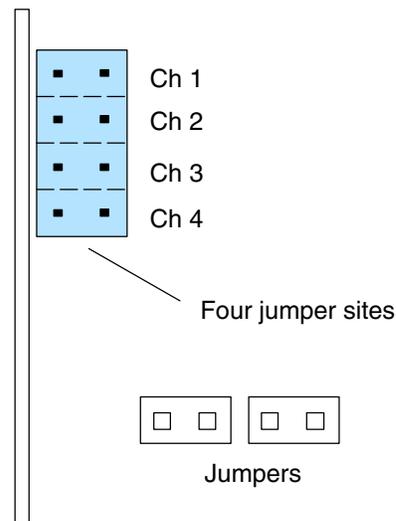
Jumpers

Note that you may set the data format independently for each channel, and the data format selection for all channels can be set independently from the voltage range selection. The tables below illustrate jumper settings to select the data format, along with example settings.

Output Data Format	Jumper Setting	Data Format	Jumper Setting
0 to 4095	Install	–2047 to +2047	Remove
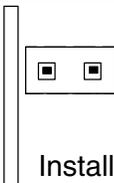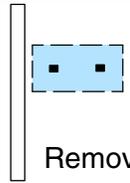
Example 1	Jumper Settings	Example 2	Jumper Settings
Ch 1 0 to 4095		Ch.= 1 0 to 4095	
Ch 2 0 to 4095		Ch 2 –2047 to +2047	
Ch 3 –2047 to +2047		Ch 3 0 to 4095	
Ch 4 0 to 4095		Ch 4 –2047 to +2047	
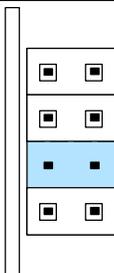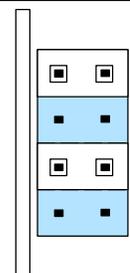
NOTE: The Data Format jumpers can be configured in a way (not shown above) for F4–04DA compatibility. If you choose to set the jumpers for F4–04DA compatibility, note that the ladder program examples in this chapter do not apply to that mode. Refer to "Configuration Cross-Reference, F4–04DA to F4–04DA–2" at the end of this chapter.

The Voltage Range jumpers and Data Format jumpers can be set independently for each of the four channels. Graphs of the data formats versus voltage ranges below show the eight possible combinations for each channel. The most commonly used selections are in the shaded graphs below. You will need to select the appropriate data format for the voltage range of each output channel in your application.

Unipolar Ranges
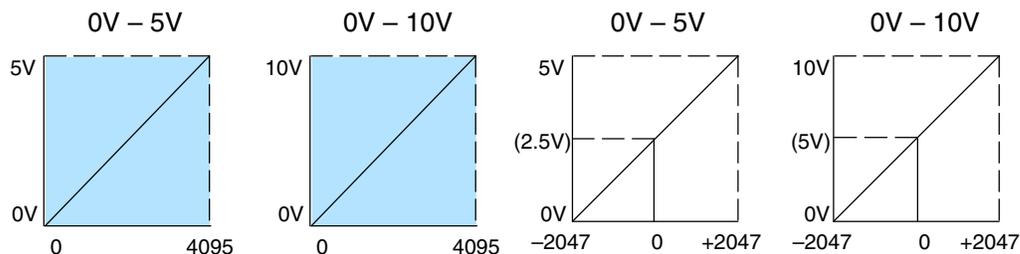
Bipolar Ranges
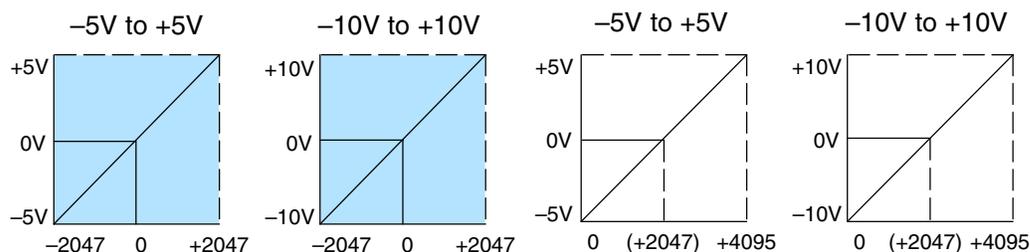
Connecting the Field Wiring

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 module or the power supply return (0V). *Do not* ground the shield at both the module and the transducer.
- 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 F4–04DA–2 requires a separate power supply. The DL430/440/450 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. Depending on the number of modules and types of output signals used, you can use this power source instead of a separate supply. If you would rather use a separate supply, choose one that meets the following requirements: 21.6 to 26.4 VDC, Class 2, 90 mA current.

Load Requirements   Each channel can be connected to a load with an impedance of 2k ohms or greater. Unused outputs should be left unconnected.

Removable Connector   The F4–04DA–2 module has a removable connector to make wiring easier. Simply loosen the retaining screws and gently pull the connector from the module.

Wiring Diagram

NOTE 1: Shields should be connected to the 0V terminal of the module or power supply.

NOTE 2: Unused voltage outputs should remain open (no connections).
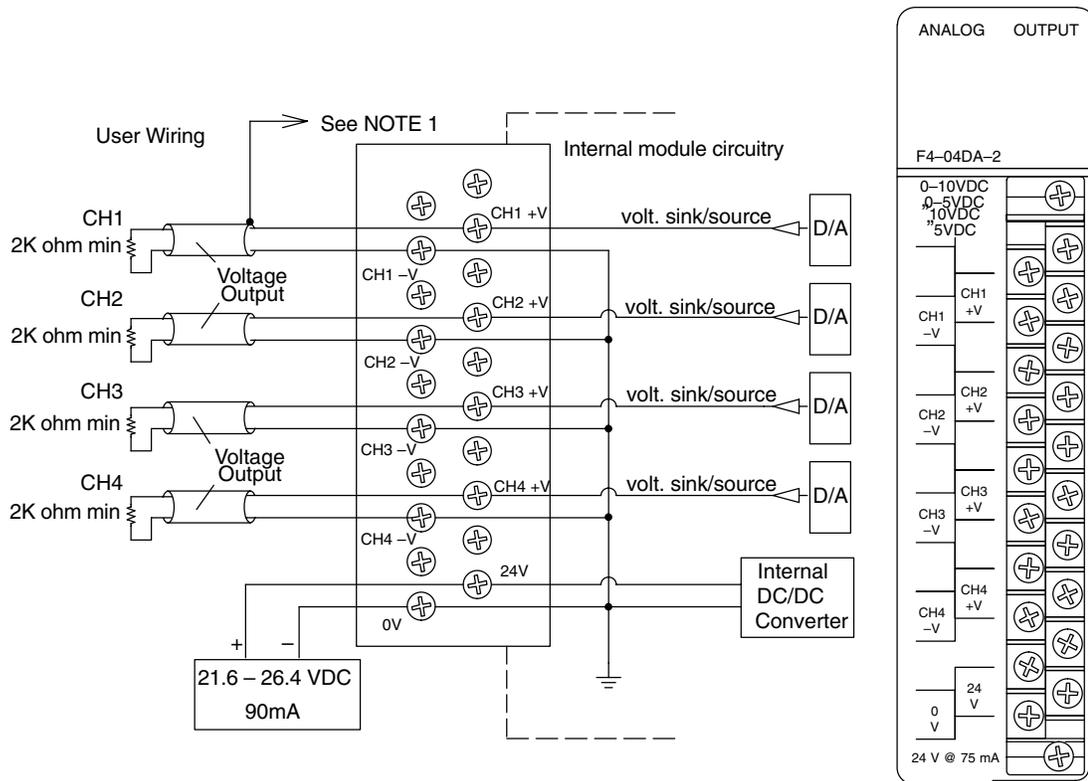
Module Operation

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 send the analog data. As shown in the following diagram, if you place the module so the output points do not start on a V-memory boundary, the instructions cannot access the data.

Correct!

F4–04DA–2

8pt Output	8pt Output	16pt Output	16pt Output	16pt Input	16pt Input
Y0 – Y7	Y10 – Y17	**Y20 – Y37**	Y40 – Y57		

V40500   V40502

V40501

MSB   LSB

Y 3 7   Y 3 0   Y 2 7   Y 2 0

Wrong!

F4–04DA–2

8pt Output	16pt Output	8pt Output	16pt Output	16pt Input	16pt Input
Y0 – Y7	Y10 – Y27	Y30 – Y37	Y40 – Y57		
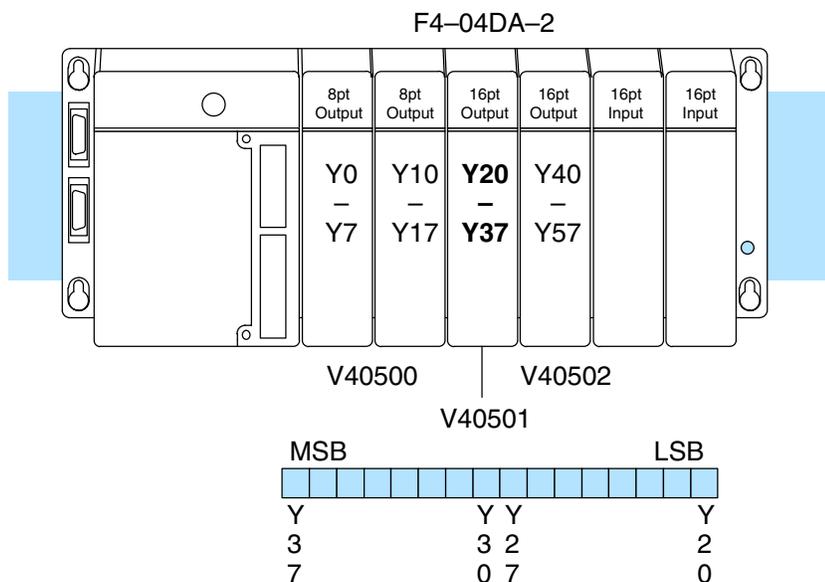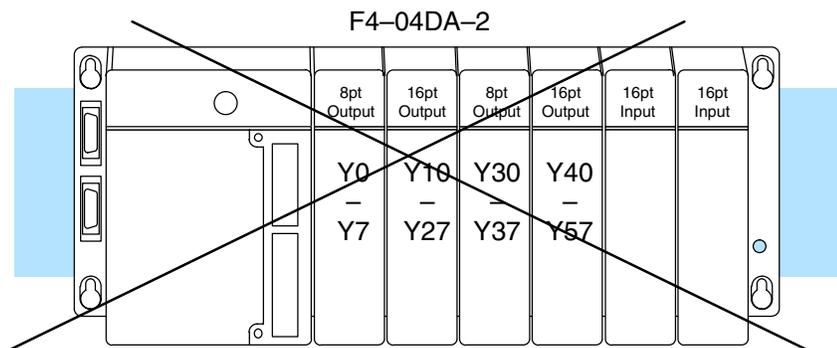
Data is split over two locations, so instructions cannot access data from a DL430.

MSB   V40501   LSB   MSB   V40500   LSB

3 7 7   Y 3 0 Y 2 7   Y 2 0   Y 1 7   Y 1 0 Y 7   Y 0

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.
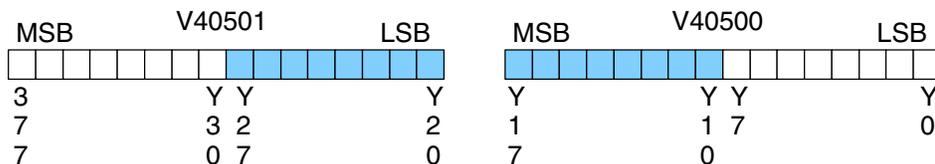
Channel Scanning Sequence

The F4–04DA–2 module allows you to update the channels in any order. Your control program determines which channel gets updated on any given scan by using two binary encoded output points. With a DL440 or DL450 CPU, you can use immediate instructions to update all four channels in the same scan (we'll show you how to do this later).
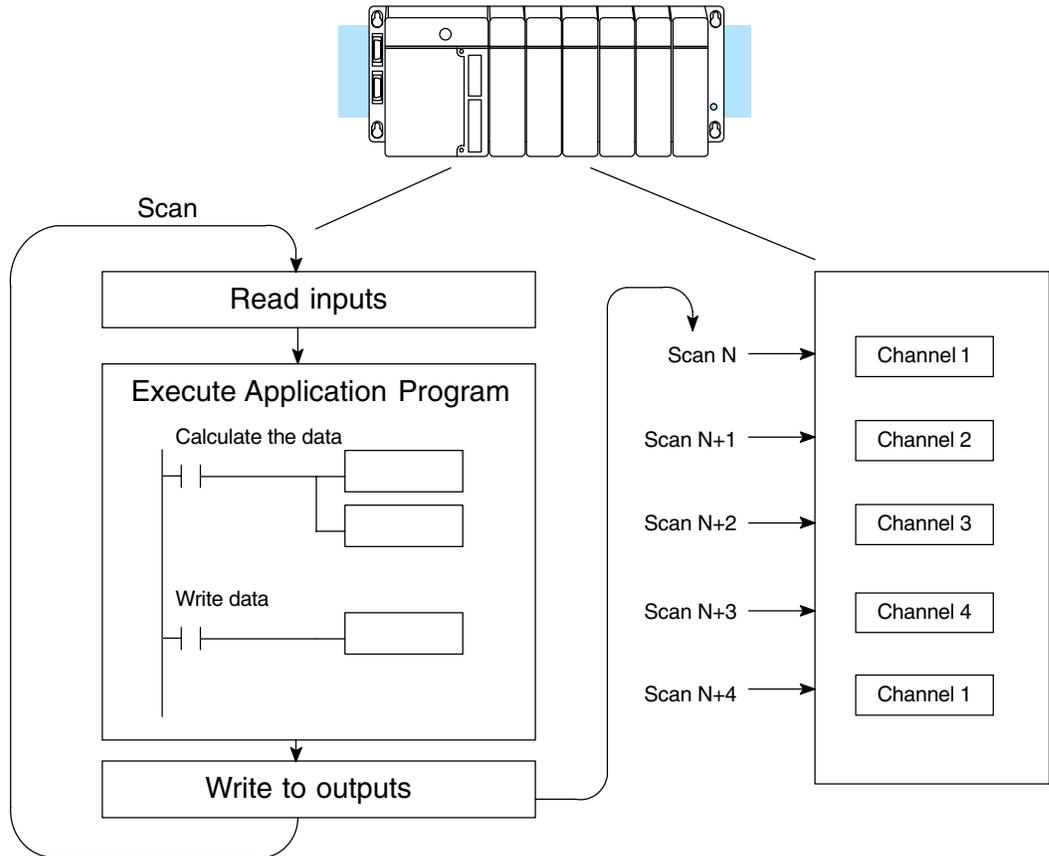
Output Bit Assignments

You may recall the F4–04DA–2 module requires 16 discrete output points from the CPU. These points provide:

- The digital representation of the analog signal.
- Identification of the channel that is to receive the data.

Since all output 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.

F4–04DA–2

8pt Output	8pt Output	16pt Output	16pt Output	16pt Input	16pt Input
Y0 – Y7	Y10 – Y17	**Y20 – Y37**	Y40 – Y57		

V40500   V40502

V40501

MSB   LSB

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

| Y 3 7 | | | | | | | Y 3 0 | Y 2 7 | | | | | | | Y 2 0 |
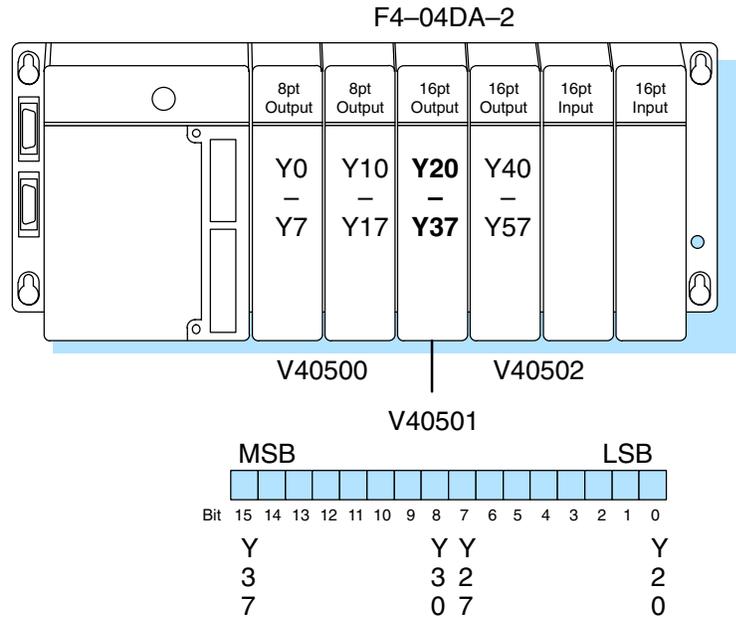
Within this V-memory location the individual bits represent specific information about the analog signal.

Channel Select Bits

The pair of output bits 12 and 13 are binary encoded to select the channel that will be updated with the data. The bits are assigned as follows.

Bit 13	Bit 12	Channel
Off	Off	1
Off	On	2
On	Off	3
On	On	4
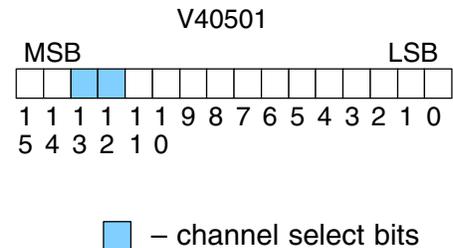
On each scan, the channel selected by these two bits will be updated. Unlike some modules, there is not a way to select *no* channel for updating on a particular scan.

V40501

MSB   LSB

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

■ – channel select bits

Analog Data Bits

The first twelve bits of the V-memory location 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

V40501

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
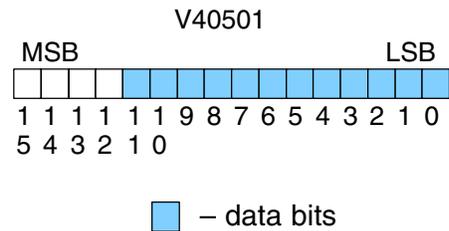
Signal Sign Bit

The most significant bit (bit 15) is the sign bit, when the data format for any channel is selected to be –2047 to +2047. By controlling this output (Y37 in our example), you can easily select positive or negative values.

Programming examples later show how easy it is to make sign selection part of your data value.

V40501

MSB ... LSB

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

☐ – signal sign output
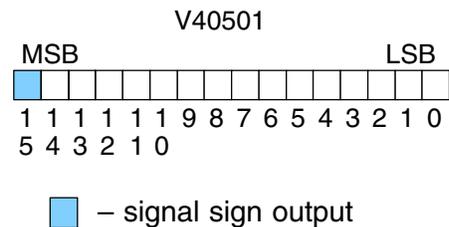
Bipolar Output Data

If an output channel is configured for an output format of –2047 to +2047, the maximum data valid value for the lower 12 bits is 2047. This means the 12th bit (bit 11) must always be "0".

V40501

MSB ... LSB

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

☐ – data bits
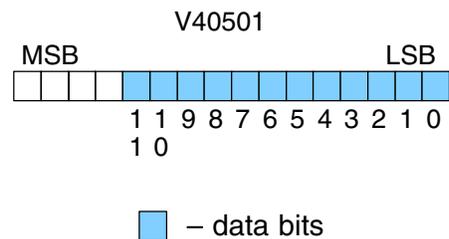
Bit 11 must be "0" for output format of –2047 to +2047.
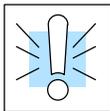
WARNING: If the data value exceeds 2047, the 12th bit becomes a "1", and the other eleven bits start over at "00000000000". At this point the module's channel output voltage also goes back to the bottom of its range and begins increasing again. The RLL program will be expecting a maximum output, but it will be minimum instead. This can have serious consequences in some applications, and may result in personal injury or damage to equipment. Therefore, in standard bipolar ranges (or whenever the output format is –2047 to +2047 in general), be sure that your RLL program does not create numbers with absolute values greater than 2047.

Output Enable Bit

The next to most significant bit (bit 14) is the Output Enable bit. Turning it on enables all four channels. Turning it off causes all output signal levels to go to zero volts, and clears the module's internal data registers for all channels.

V40501

MSB □ ■ □□□□□□□□□□□□□□□ LSB

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

■ – output enable bit

After an off-to-on transition of the Output Enable bit, each output stays at zero volts until the channel select bits select the channel and the CPU writes a non-zero value to it.
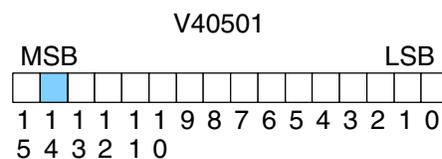
Module Resolution

Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0–4095. For output formats of –2047 to +2047, we still have 4095 counts. For example, with a 0 –10V scale and 0 to 4095 output format, you would send a 0 to get a 0V signal, and 4095 to get a 10V signal. This is equivalent to 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.

Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest change in signal level due to a value change of 1 LSB count.

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

H = high limit of the signal range

L = low limit of the signal range

Output Range	Signal Span (H – L)	Divide By	Smallest Change
0–5V	5V	4095	1.22 mV
0–10V	10V	4095	2.44 mV
± 5V	10V	4095	2.44 mV
± 10V	20V	4095	4.88 mV

Writing the Control Program

Update Any Channel

As mentioned earlier, you can update any channel per scan using regular I/O instructions, or any number of channels per scan using immediate I/O instructions. The following diagram shows the data locations for an example system. You use the channel selection outputs to determine which channel gets updated (more on this later).
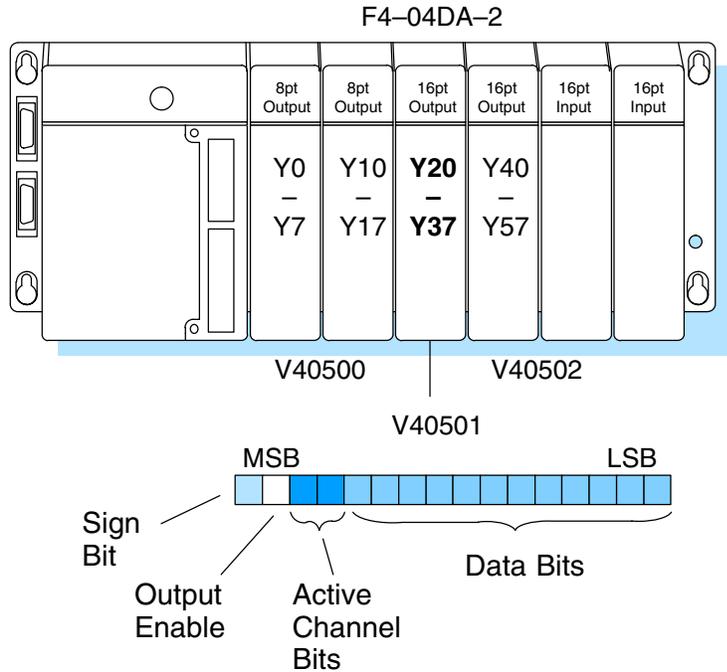
Calculating the Digital Value

Your program has to calculate the digital value to send to the analog module. There are many ways to do this, but most applications are understood more easily if you use measurements in engineering units. 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.

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

A = analog value (0 – 4095)

U = engineering units

H = high limit of the engineering unit range

L = low limit of the engineering unit range

Consider the following example which controls pressure from 0.0 to 99.9 PSI. By using the formula, you can easily determine the digital value that should be sent to the module. The example shows the conversion required to yield 49.4 PSI. Notice the formula uses a multiplier of 10. This is because the decimal portion of 49.4 cannot be loaded, so you adjust the formula to compensate for it.

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

✓ ✓ ✓
430 440 450

Here's how you would write the program to perform the engineering unit conversion. This example assumes you have calculated or loaded the engineering unit value and stored it in V3000. Also, you have to perform this for both channels if you're using different data for each channel.

NOTE: The DL405 offers various instructions that allow you to perform math operations using binary, BCD, etc. It's usually easier to perform any math calculations in BCD and then convert the value to binary before you send the data to the module. If you are using binary math, you do not have to include the BIN conversion.

X1

| LD V3000 | When X1 is on, the engineering units (stored in V3000) are loaded into the accumulator. This example assumes the numbers are BCD. |

| MUL K4095 | Multiply the accumulator by 4095 (to start the conversion). |

| DIV K1000 | Divide the accumulator by 1000 (because we used a multiplier of 10, we have to use 1000 instead of 100). |

| BIN | Convert the BCD number to binary. |

| OUT V3101 | Store the result in V3101. This is the digital value, in binary form, that should be sent to the module. |

V-Memory Registers

The ladder program examples that follow occasionally use certain V-memory register addresses in the CPU that correspond to 16-bit Y output modules. Use the table below to find the V-memory address for the particular location of your analog module. See Appendix A for additional addresses for DL450 CPUs.

V-Memory Register Addresses for 16-Point Output (Y) Locations									
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

F4–04DA–2
4-Ch. Analog Output

Sending Data to One Channel

The following programs show you how to update a single channel. Notice the DL430 CPU uses a slightly different program than the DL440 and DL450 CPUs. Since the DL430 does not support the OUTF instruction, the program must be modified to make sure the channel select bits are not accidentally changed by the data in the accumulator. The DL430 example will also work for DL440 and DL450 CPUs. This example assumes you already have the data loaded in V3001.

✕	✓	✓
430	**440**	**450**

DL440/450 Example
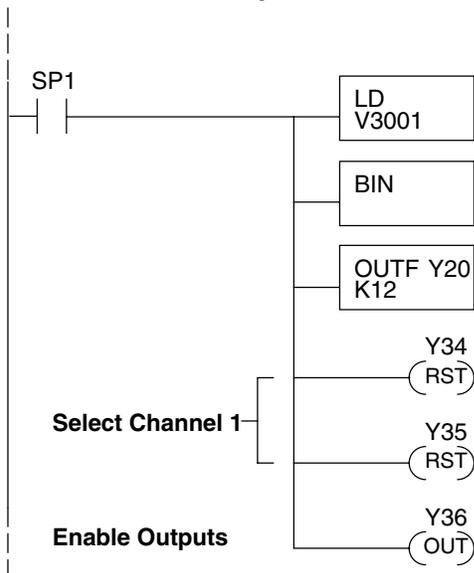
```
SP1
──┤ ├──────────────┬───  LD
                    │      V3001
                    │
                    ├───  BIN
                    │
                    ├───  OUTF Y20
                    │      K12
                    │
                    │             Y34
                    ├───────────(RST)
  Select Channel 1──┤
                    │             Y35
                    ├───────────(RST)
                    │
                    │             Y36
  Enable Outputs────┴───────────(OUT)
```

The LD instruction loads the data for channel 1 into the accumulator. Since SP1 is used, this rung automatically executes on every scan. You could also use an X, C, etc. permissive contact.

The BIN instruction converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere).

The OUTF sends the 12 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.

Select channel 1 for updating.

Y35	Y34	Channel
Off	Off	Ch. 1
Off	On	Ch. 2
On	Off	Ch. 3
On	On	Ch. 4

Turn on the output enable bit, to enable all output channels.

✓	✓	✓
430	**440**	**450**

DL430 Example

```
SP1
──┤ ├──────────────┬───  LD
                    │      V3001
                    │
                    ├───  BIN
                    │
                    ├───  ANDD
                    │      KFFF
                    │
                    ├───  OUT
                    │      V40501
                    │
                    │             Y34
                    ├───────────(RST)
  Select Channel 1──┤
                    │             Y35
                    ├───────────(RST)
                    │
                    │             Y36
  Enable Outputs────┴───────────(OUT)
```

The LD instruction loads the data for channel 1 into the accumulator. Since SP1 is used, this rung automatically executes every scan. You could also use an X, C, etc. permissive contact.

The BIN instruction converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere).

The ANDD instruction masks off the channel select bits to prevent an accidental channel selection.

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

Select channel 1 for updating.

Y35	Y34	Channel
Off	Off	Ch. 1
Off	On	Ch. 2
On	Off	Ch. 3
On	On	Ch. 4

Turn on the output enable bit, to enable all output channels.

Sequencing the Channel Updates

The next six example programs show you how to send digital values to the module when you have more than one channel. These examples will automatically update all four channels over four scans.

The first two sequencing examples, examples 1 and 2, are fairly simple and will work in almost all situations. We recommend these for new users. They use control relays C1 through C4 as index numbers corresponding to the channel updated on any particular scan. At the end of each scan, only one control relay C1 through C4 is on. On each subsequent scan, the next control relay energizes. The channel sequencing automatically begins with channel 1 on the first scan, or after any disruption in the logic. You must use example 1 with DL430 CPUs. Either example will work with DL440 or DL450 CPUs.

The next two examples, 3 and 4, are slightly more complex. However, they do not depend on the use of control relays to provide channel sequencing. Instead, they use function boxes to increment a channel pointer value in V-memory. Then, other instructions perform bit manipulations to position the channel select bits properly in the output word to the module. You must use example 3 with DL430 CPUs. Either example will work with DL440 or DL450 CPUs.

Example 5 is similar to example 2, but is modified for use with bipolar outputs. This example works with DL430, DL440 and DL450 CPUs.

In the last example, we show you how you can update all four channels in the same scan with DL440 and DL450 CPUs. However, this can increase the scan time and you may not always need to update all four channels on every scan.

Sequencing Example 1, DL440/450

✕	✓	✓
430	440	450

The following program example shows how to send digital values to the module when you have more than one channel. This example assumes you already have the data loaded in V3001, V3002, V3003, and V3004 for channels 1 – 4 respectively. It is important to use the rungs in the order shown for the program to work. This example will not work with DL430 CPUs.

Ch4. Done
C4 ——| |—————————————————————(C0 OUT)

When channel 4 has been updated, C0 restarts the update sequence.

Ch3. Done
C3 ——| |———————————[LD V3004]
—————————————————————————(C4 OUT)

When channel 3 has been updated, this rung loads the data for channel 4 into the accumulator. By turning on C4, this triggers the channel update (see the channel select rungs).

Ch2. Done
C2 ——| |———————————[LD V3003]
—————————————————————————(C3 OUT)

When channel 2 has been updated, this rung loads the data for channel 3 into the accumulator. By turning on C3, this triggers the channel update (see the channel select rungs).

Ch1. Done
C1 ——| |———————————[LD V3002]
—————————————————————————(C2 OUT)

When channel 1 has been updated, this rung loads the data for channel 2 into the accumulator. By turning on C2, this triggers the channel update (see the channel select rungs below).

Restart
C0 ——| |———————————[LD V3001]

C1 C2 C3 C4
——|/|—|/|—|/|—|/|————————————(C1 OUT)

This rung loads the data for channel 1 into the accumulator. C0 restarts the sequence after channel 4 is done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C1 through C4 are off. In this case, we also start the sequence with channel 1.

SP1 ——| |———————————[BIN]

—————————————————[OUTF Y20 K12]

This rung converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere). It also loads the data to the appropriate bits of the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.
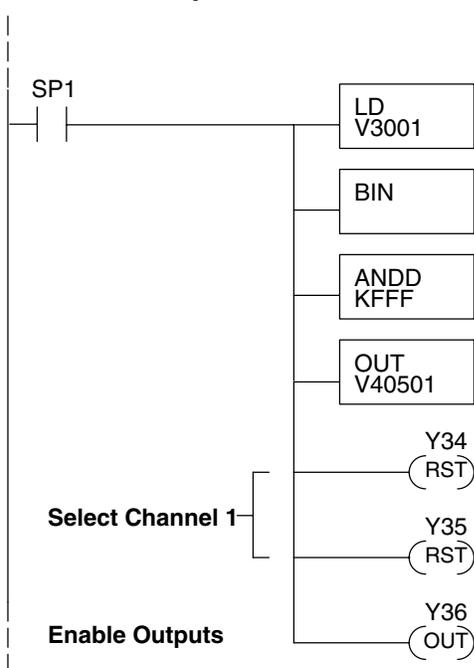
Select Channel, Binary Encoded
C3 ——| |—————————————————————(Y35 OUT)
C4 ——| |
C2 ——| |—————————————————————(Y34 OUT)
C4 ——| |

Set Y35 and Y34 to select the output channel, based on the control relay status.

CR(on)	Y35	Y34	Channel
C1	Off	Off	Ch. 1
C2	Off	On	Ch. 2
C3	On	Off	Ch. 3
C4	On	On	Ch. 4

SP1 **Enable Outputs** ——| |————————(Y36 OUT)

Enables all four output channels. SP1 is always on.

Sequencing Example 2, DL430

✓ ✓ ✓
430 440 450

Since the DL430 does not support the OUTF instruction, the previous program must be modified to make sure the channel select bits are not accidentally changed by the data in the accumulator. It is important to use the rungs in the order shown for the program to work. This example will also work with DL440 and DL450 CPUs.

Ch4. Done
C4 —| |—————————————————(OUT) C0

When channel 4 has been updated, C0 restarts the update sequence.

Ch3. Done
C3 —| |———————[LD V3004]
—————————————(OUT) C4

When channel 3 has been updated, this rung loads the data for channel 4 into the accumulator. By turning on C4, this triggers the channel update (see the channel select rungs).

Ch2. Done
C2 —| |———————[LD V3003]
—————————————(OUT) C3

When channel 2 has been updated, this rung loads the data for channel 3 into the accumulator. By turning on C3, this triggers the channel update (see the channel select rungs below).

Ch1. Done
C1 —| |———————[LD V3002]
—————————————(OUT) C2

When channel 1 has been updated, this rung loads the data for channel 2 into the accumulator. By turning on C2, this triggers the channel update (see the channel select rungs below).

Restart
C0 —| |——————————[LD V3001]
C1 C2 C3 C4
—|/|—|/|—|/|—|/|——(OUT) C1

This rung loads the data for channel 1 into the accumulator. C0 restarts the sequence after channel 4 is done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C1 through C4 are off. In this case, we also start the sequence with channel 1.

SP1 —| |——[BIN]
[ANDD KFFF]
[OUT V40501]

This rung converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere). The ANDD instruction masks off the channel select bits to prevent an accidental channel selection. The OUT instruction sends the data to the module. Our example starts with V40501, but the actual value depends on the location of the module in your application.

Select Channel, Binary Encoded
C3 —| |——————————————(OUT) Y35
C4 —| |—

Set Y35 and Y34 to select the output channel, based on the control relay status.

CR(on)	Y35	Y34	Channel
C1	Off	Off	Ch. 1
C2	Off	On	Ch. 2
C3	On	Off	Ch. 3
C4	On	On	Ch. 4

Select Channel, cont'd
C2 —| |——————————————(OUT) Y34
C4 —| |—

Enable Outputs
SP1 —| |——————————————(OUT) Y36

Enables all four output channels. SP1 is always on.

Sequencing Example 3, DL440/DL450

×	✓	✓
430	**440**	**450**

The following program example shows how to send digital values to the module when you have more than one channel. This example will not work with DL430 CPUs. This example assumes you are using the following data locations.

- V3000 – channel 1 data   V3001 – channel 2 data
- V3002 – channel 3 data   V3004 – channel 4 data
- V1500 – channel to update: 0 = ch. 1, 1 = ch. 2, 2 = ch. 3, 3 = ch. 4

Always On

SP1

LD V1500

This loads the number of the channel to be updated into the accumulator. The channels are 1–4, but the values in V1500 range from 0–3 and correspond to the channels. We'll use channel 2 as an example.

V1500

Acc. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ← | 0 | 0 | 0 | 1 |

LDX V3000

Use the channel selection value, which is now on the data stack, as an offset from V3000 to load the channel data into the accumulator.

HEX Value in 1st stack location

Octal   Octal

V | 3 | 0 | 0 | 0 | + | 0 | 1 | = | V | 3 | 0 | 0 | 1 |

Acc. | 0 | 0 | 0 | 0 | 2 | 3 | 4 | 5 |

The value in V3001 is 2345, which is slightly over half scale.

BIN

Convert the BCD data to binary. Since the value can never be above 4095, only the 12 least significant bits of the accumulator are used.

| 0 | 0 | 0 | 0 | 2 | 3 | 4 | 5 |

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc. 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	1	0	0	1

BCD Value converted to binary now in accumulator.

ANDD KFFF

Mask off the upper four bits of the word, just in case the data value is out of range (greater than 4095).

LD V1500

Load the number of the channel to be updated back into the accumulator again (the channel data is moved to the first data stack location).

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc. 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

SHFL K12

This instruction moves the channel select bit(s) into the proper location. We'll use it later when we send the 16-bit data word to the module.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc. 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Program is continued on the next page.

Example 3 Continued

ORD
K4000

Set the Output Enable bit, by combining the value of 4000 hex with the accumulator value. This sets bit 14 to "1", enabling all channels.

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

ADDBS

Earlier in the program the data value was placed into the first data stack location. The ADDBS instruction adds the value currently in the accumulator with the value in the first data stack location.

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

+

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Stack	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	1	0	0	1

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

OUTF Y20
K15

Send the lower 15 bits stored in the accumulator to the analog module. The lowest 12 bits contain the analog data. Bits 12 and 13 are the channel selection bits. Bit 14 is the Output Enable bit.

INCB
V1500

Increment the channel selection value. This allows the logic to cycle through all four channels.

V1500 K4
—| = |—

LD
K0

When channel four has been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1).

OUT
V1500
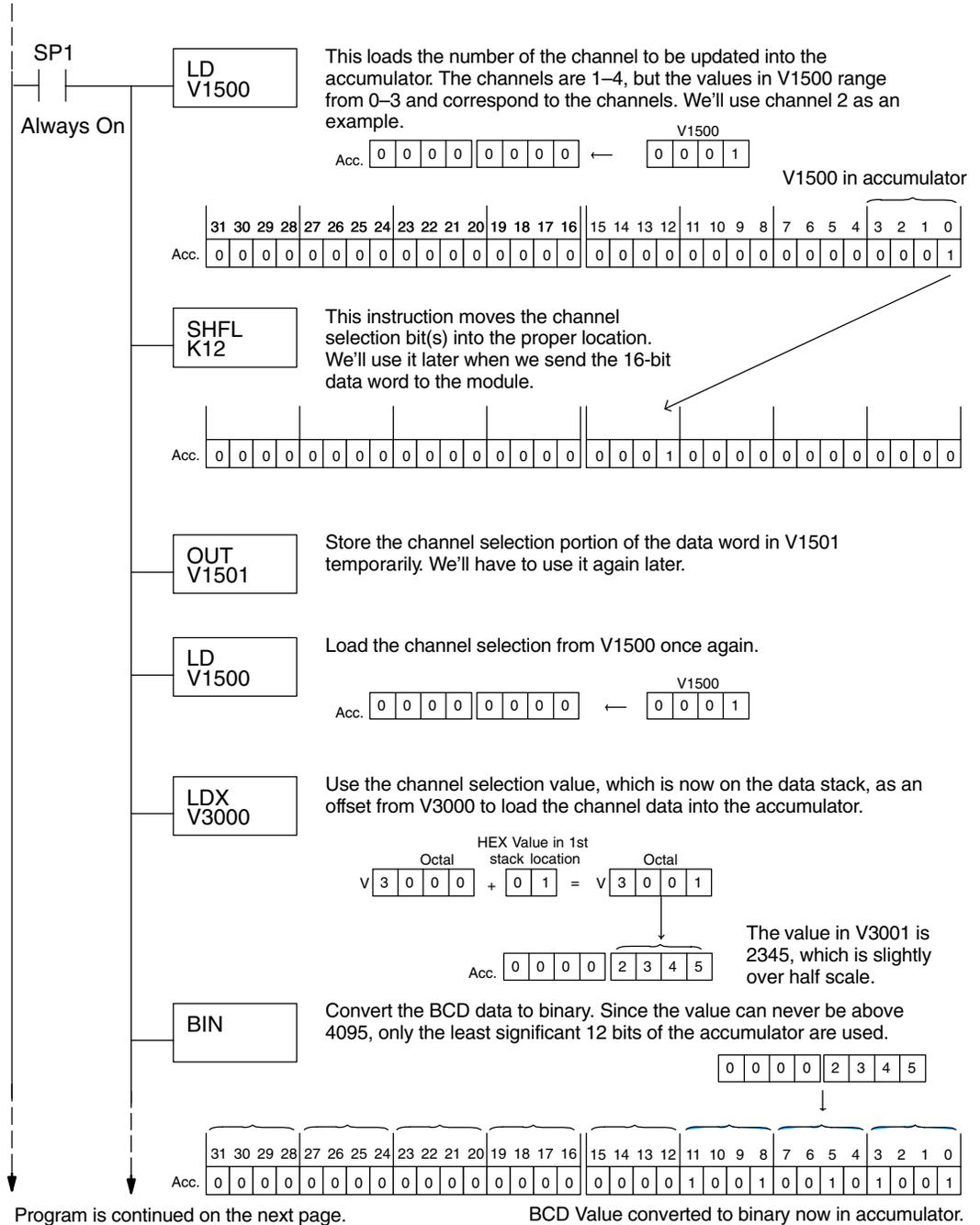
Sequencing Example 4, DL430

✓ ✓ ✓
430 440 450

The following program example shows how to send digital values to the module when you have more than one channel. This example will also work with DL440 and DL450 CPUs. This example assumes you are using the following data locations.

- V3000 – channel 1 data   V3001 – channel 2 data
- V3002 – channel 3 data   V3004 – channel 4 data
- V1500 – channel to update: 0 = ch. 1, 1 = ch. 2, 2 = ch. 3, 3 = ch. 4
- V1501 – temporary location for the channel selection

SP1
Always On

LD V1500

This loads the number of the channel to be updated into the accumulator. The channels are 1–4, but the values in V1500 range from 0–3 and correspond to the channels. We'll use channel 2 as an example.

Acc. | 0 0 0 0 | 0 0 0 0 | ← V1500 | 0 0 0 1 |

V1500 in accumulator

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Acc. | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 1 |

SHFL K12

This instruction moves the channel selection bit(s) into the proper location. We'll use it later when we send the 16-bit data word to the module.

Acc. | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 1 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 |

OUT V1501

Store the channel selection portion of the data word in V1501 temporarily. We'll have to use it again later.

LD V1500

Load the channel selection from V1500 once again.

Acc. | 0 0 0 0 | 0 0 0 0 | ← V1500 | 0 0 0 1 |

LDX V3000

Use the channel selection value, which is now on the data stack, as an offset from V3000 to load the channel data into the accumulator.

HEX Value in 1st stack location

V | 3 0 0 0 | Octal + | 0 1 | = V | 3 0 0 1 | Octal

Acc. | 0 0 0 0 | 2 3 4 5 |

The value in V3001 is 2345, which is slightly over half scale.

BIN

Convert the BCD data to binary. Since the value can never be above 4095, only the least significant 12 bits of the accumulator are used.

| 0 0 0 0 | 2 3 4 5 |

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
Acc. | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 0 0 0 0 | 1 0 0 1 | 0 0 1 0 | 1 0 0 1 |

BCD Value converted to binary now in accumulator.

Example 4
Continued

ANDD
KFFF

Mask off the upper four bits of the word, just in case the data value is out of range (greater than 4095).

OR
V1501

Earlier in the program the channel selection portion of the data word was created and stored in V1501. Now we can OR this location with the data word currently in the accumulator to get the final data word that is ready to be sent to the analog module.

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	1	0	0	1

+

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
V1501	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

ORD
K4000

Set the Output Enable bit, by combining the value of 4000 hex with the accumulator value. This sets bit 14 to "1", enabling all channels.

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	1	0	0	1	0	1	0	0	1

OUT
V40501

Send the data stored in the lower half of the accumulator to the analog module (the OUT instruction ignores the upper 16 bits of the accumulator). The most significant four bits of the analog word contain the channel selection bits. The remaining 12 bits contain the analog data.

INCB
V1500

Increment the channel selection value. This allows the logic to cycle through all four channels.
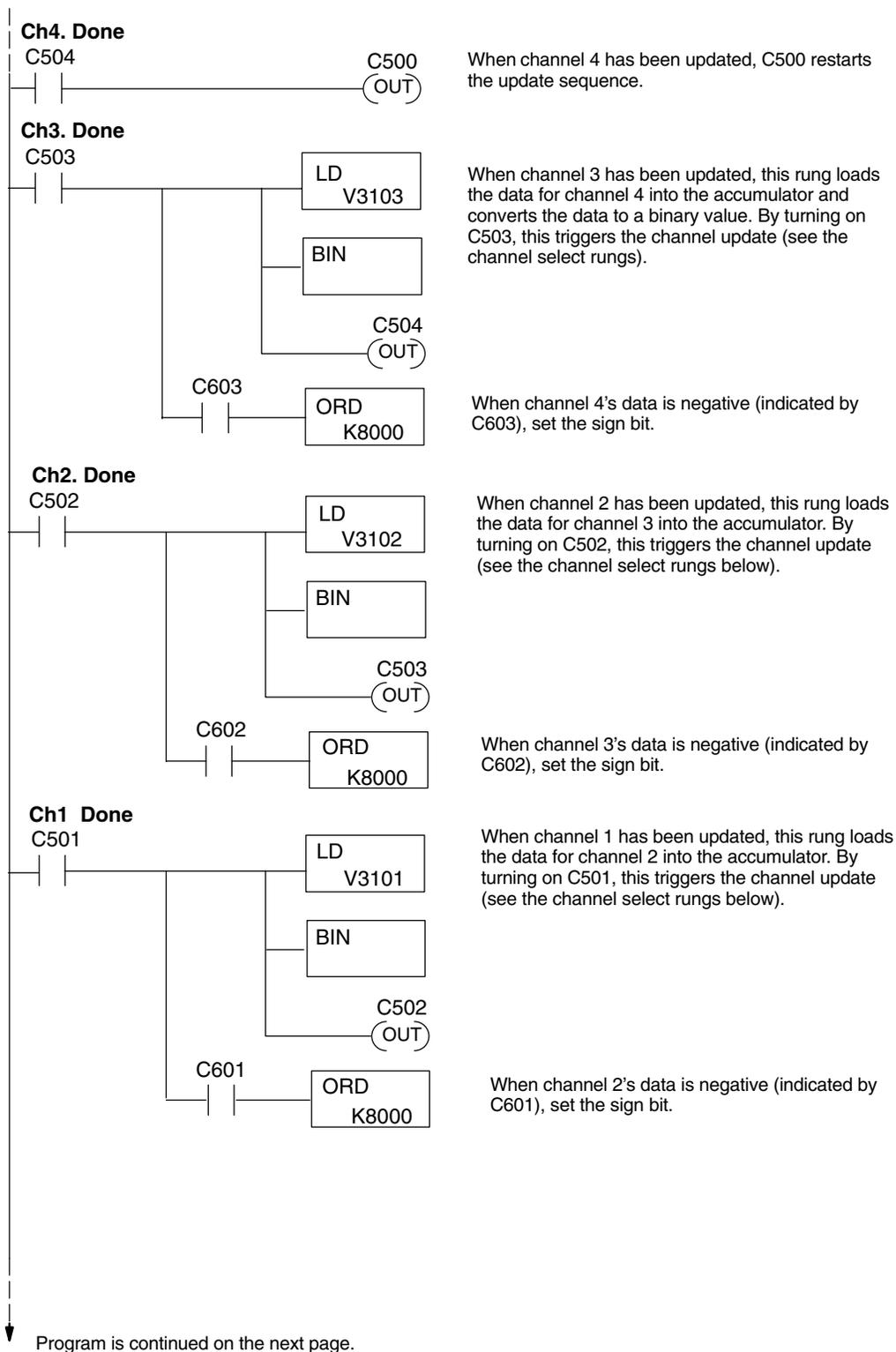
V1500 K4
──┤ = ├──

LD
K0

When channel four has been updated, this instruction resets the channel selection memory location to 0 (remember, 0 represents channel 1).
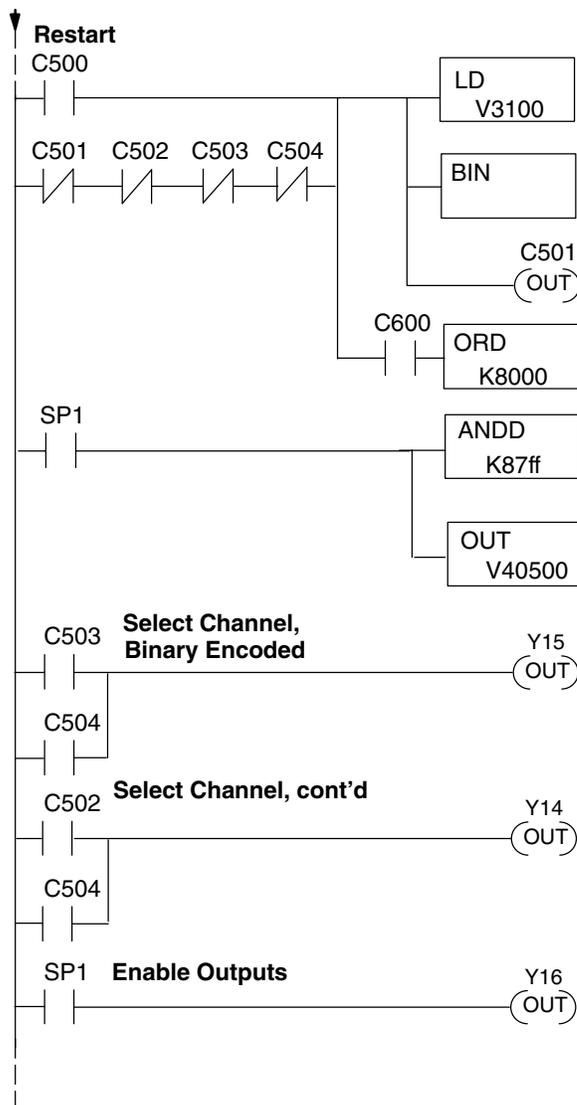
OUT
V1500

Sequencing Example 5 Bipolar Data

✓ ✓ ✓
430 440 450

The following example program is for use with bipolar outputs. All this does is add a sign bit output which the ladder must handle. This example works with all DL405 CPUs.

Ch4. Done
C504 ———| |——————————————————(C500 OUT)

When channel 4 has been updated, C500 restarts the update sequence.

Ch3. Done
C503 ———| |———┬——— [LD V3103]
              │
              ├——— [BIN]
              │
              │   (C504 OUT)
              │
     C603   │
    ——| |——————[ORD K8000]

When channel 3 has been updated, this rung loads the data for channel 4 into the accumulator and converts the data to a binary value. By turning on C503, this triggers the channel update (see the channel select rungs).

When channel 4's data is negative (indicated by C603), set the sign bit.

Ch2. Done
C502 ———| |———┬——— [LD V3102]
              │
              ├——— [BIN]
              │
              │   (C503 OUT)
              │
     C602   │
    ——| |——————[ORD K8000]

When channel 2 has been updated, this rung loads the data for channel 3 into the accumulator. By turning on C502, this triggers the channel update (see the channel select rungs below).

When channel 3's data is negative (indicated by C602), set the sign bit.

Ch1 Done
C501 ———| |———┬——— [LD V3101]
              │
              ├——— [BIN]
              │
              │   (C502 OUT)
              │
     C601   │
    ——| |——————[ORD K8000]

When channel 1 has been updated, this rung loads the data for channel 2 into the accumulator. By turning on C501, this triggers the channel update (see the channel select rungs below).

When channel 2's data is negative (indicated by C601), set the sign bit.

Program is continued on the next page.

**Example 5
Continued**

Restart
C500

LD
V3100

C501 C502 C503 C504

BIN

C501
(OUT)

This rung loads the data for channel 1 into the accumulator. C500 restarts the sequence after channel 4 is done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C501 through C504 are off. In this case, the sequence starts with channel 1.

C600
ORD
K8000

When channel 1's data is negative (indicated by C600), set the sign bit.

SP1
ANDD
K87ff

The ANDD instruction keeps data from exceeding 2047, and masks off the channel select bits to prevent an accidental channel selection.

OUT
V40500

The OUT instruction sends the data to the module. The example starts with V40500, but the actual value depends on the location of the module in your application.

C503 **Select Channel, Binary Encoded**
Y15
(OUT)

C504

Set Y15 and Y14 to select the output channel, based on the control relay status.

C502 **Select Channel, cont'd**
Y14
(OUT)

C504

CR(on)	Y15	Y14	Channel
C501	Off	Off	Ch. 1
C502	Off	On	Ch. 2
C503	On	Off	Ch. 3
C504	On	On	Ch. 4

SP1 **Enable Outputs**
Y16
(OUT)

Enables all four output channels. SP1 is always on.

Updating all Channels in a Single Scan, DL440/450

X	✓	✓
430	**440**	**450**

By using the Immediate instructions found in the DL440 and DL450 CPUs, you can easily update all four channels in a single scan. Before choosing this method, remember the Immediate instructions slow the CPU scan time. To minimize this impact, change the SP1 (Always On) contact to an X, C, etc. permissive contact that only updates the channels as required. This example assumes you already have the data loaded in V3001, V3002, V3003, and V3004 for channels 1 – 4 respectively. This example will not work with DL430 CPUs.

NOTE: This program will not work in a remote/slave arrangement. Use one of the programs shown that reads one channel per scan.

Channel 1 Example
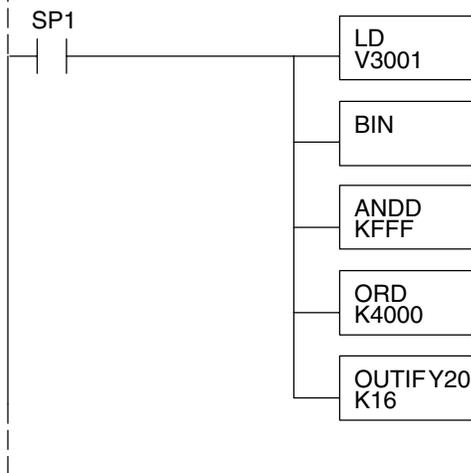
SP1

| LD V3001 | The LD instruction loads the data for channel 1 into the accumulator. |

| BIN | The BIN instruction converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere). |

| ANDD KFFF | The ANDD instruction masks off the channel select bits to prevent an accidental channel selection. |

| ORD K4000 | The ORD instruction (with K4000) sets the Output Enable bit. Outputs Y34 and Y35 are left off to select channel 1 for updating with the data. |

| OUTIF Y20 K16 | The OUTIF sends 16 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application. |

You have to send 16 bits with the OUTIF instruction. If you don't send all 16 bits, the module will ignore the data.

The remaining channels are updated with a similar program segment. The only changes are the location of the data for each channel (V3002, V3003, and V3004) and the ORD instruction. The constant loaded with the ORD instruction is different for each channel. The following example shows where these differences occur.

Changes for channels 2 – 4

SP1

| LD V3002 | The LD instruction loads the data for channel 2 into the accumulator. |
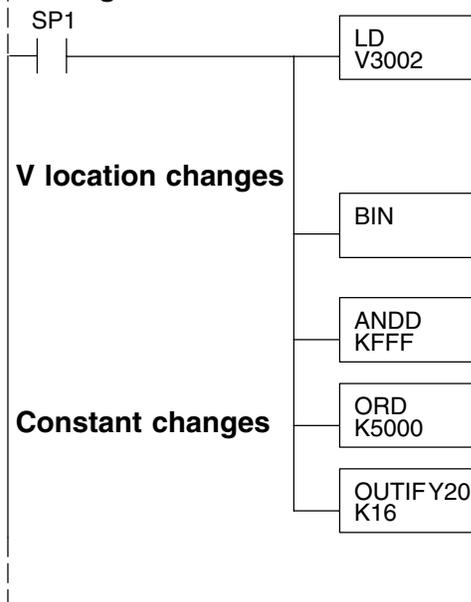
V location changes

Location	Channel
V3001	1
V3002	2
V3003	3
V3004	4

| BIN | The BIN instruction converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere). |

| ANDD KFFF | Mask off the upper four bits, so bad data cannot corrupt the channel select bits, output enable bit, or sign bit. |

Constant changes

| ORD K5000 | The ORD instruction with the constants as specified selects the appropriate channel to be updated, and sets the Output Enable bit. The following constants are used. |

| OUTIF Y20 K16 | |

Constant	Channel
K 4000	1
K 5000	2
K 6000	3
K 7000	4

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the voltage or current 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.

Voltage Range	Output Format	If you know the digital value ...	If you know the analog signal level ...
0 – 5V	0 to 4095	$A = \dfrac{5D}{4095}$	$D = \dfrac{4095}{5}(A)$
	−2047 to +2047	$A = \dfrac{5(D + 2047)}{4095}$	$D = \dfrac{4095(A)}{5} - 2047$
0 – 10V	0 to 4095	$A = \dfrac{10D}{4095}$	$D = \dfrac{4095}{10}(A)$
	−2047 to +2047	$A = \dfrac{10(D + 2047)}{4095}$	$D = \dfrac{4095(A)}{10} - 2047$
± 5V	0 to 4095	$A = \dfrac{10D}{4095} - 5$	$D = \dfrac{4095}{10}(A + 5)$
	−2047 to +2047	$A = \dfrac{5(D)}{2047}$	$D = \dfrac{2047(A)}{5}$
± 10V	0 to 4095	$A = \dfrac{20D}{4095} - 10$	$D = \dfrac{4095}{20}(A + 10)$
	−2047 to +2047	$A = \dfrac{10(D)}{2047}$	$D = \dfrac{2047(A)}{10}$

For example, suppose you are using the ± 10V range and an output data format of −2047 to +2047. If you know you need a −3V signal level, just insert these numbers into the appropriate formula from the table. It will give you the digital value to store in the V-memory location containing the output data.

$$D = \dfrac{2047(-3)}{10}$$

$$D = \dfrac{-6141}{10}$$

$$D = -614$$

Configuration Cross-Reference, F4–04DA to F4–04DA–2

The F4–04DA–2 Analog Output Module can replace the existing F4–04DA Analog Output Module when all four channels are used as voltage outputs for any of four ranges: 0 – 5VDC, 0 – 10VDC, ±5VDC, and ±10VDC. For existing voltage output installations this section shows how to configure the F4–04DA–2 to directly replace a F4–04DA as presently configured. The RLL program that communicated with the F4–04DA needs *no modifications* to perform the same function with a properly configured F4–04DA–2.

NOTE: When the F4–04DA–2 is configured for F4–04DA compatibility, the module's output word bit descriptions and ladder examples in this chapter do not apply. Please refer to the F4–04DA chapter for this information.
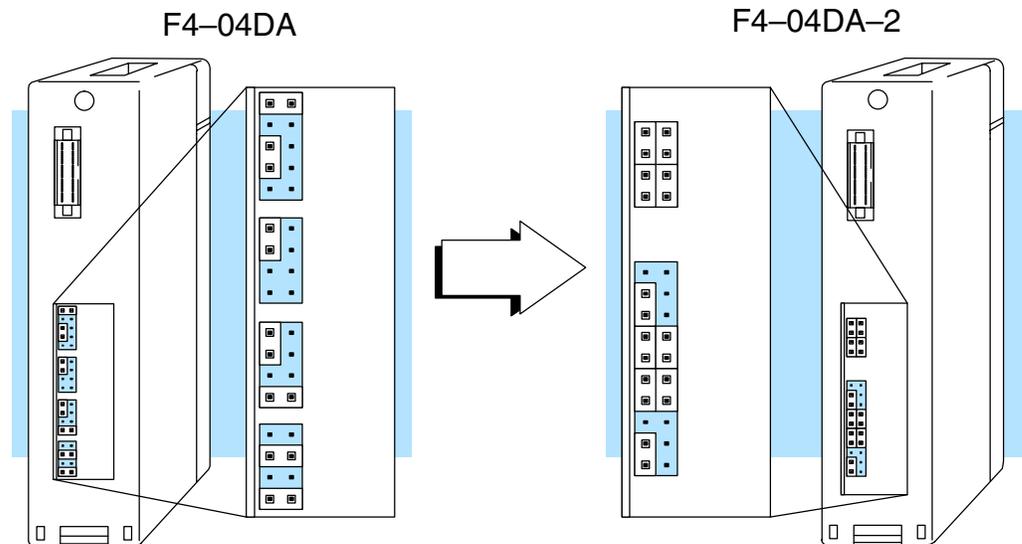
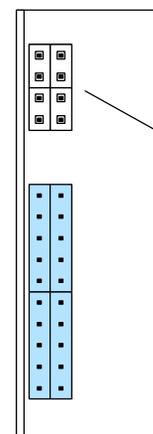F4–04DA   F4–04DA–2

In the following procedure, we will examine the present jumper settings on the F4–04DA. Just follow the steps to translate that configuration into its equivalent on the F4–04DA–2 module.

**Step 1:
Select the
Compatibility
Mode**

The upper jumper block on the F4–04DA–2 is the data format jumpers. They are normally installed individually and horizontally for each of the four channels. To select the F4–04DA compatibility mode, just install all four jumpers in the vertical orientation as shown to the right. Installed in this way, they do not correspond to individual channels anymore.

You may recall that the F4–04DA has only one data format (0 to 4095) for all channels. This data format is in effect now in the F4–04DA–2's compatibility mode.
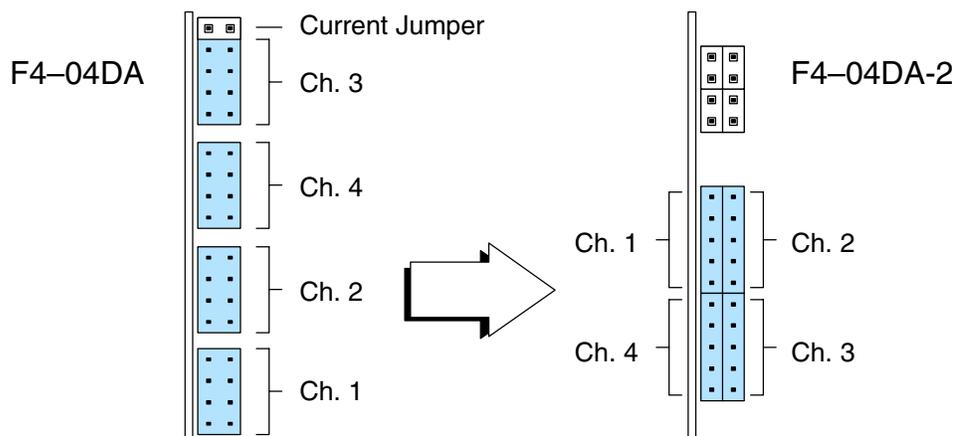
F4–04DA–2

Install all the data format jumpers **vertically** for F4–04DA compatibility.

Step 2:
Voltage Range
Selection

Now we set the F4–04DA–2 voltage range selection to match the F4–04DA settings. Both modules have four sets of jumper blocks corresponding to each of four channels, as shown in the following diagram. First you will need to see whether the F4–04DA's current jumper is installed. Then use the appropriate column in the table to translate each channel's jumper settings to their equivalent on the F4–04DA–2.

Output Range Selection Cross-Reference			
Output Voltage Range	F4–04DA, Current Jumper		F4–04DA–2
	Installed	Removed	
0 – 5VDC			
0 – 10 VDC			
± 5 VDC			
± 10 VDC	Not supported with current jumper installed		