F2-02DAS-2
0–5, 0–10V 2-Channel Isolated Output

13

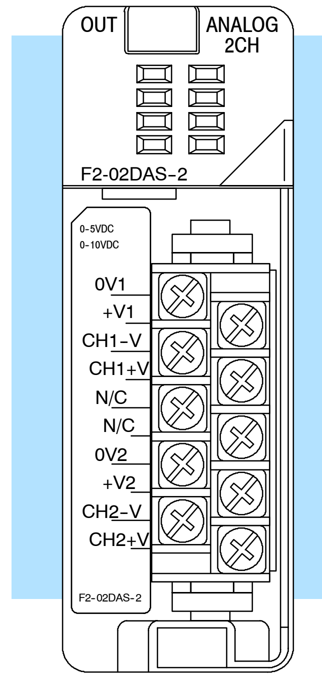
In This Chapter. . . .

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

Module Specifications

The F2-02DAS-2 Analog Output module provides several hardware features:

- Supports D2-230, D2-240, DL250-1 and D2-260 CPUs (see firmware requirements below).
- Analog outputs are isolated from channel to channel and channel to PLC logic.
- The module has a removable terminal block so the module can be easily removed or changed without disconnecting the wiring.
- Can update both channels in one scan (D2-240/D2-250-1/260 only)
- Outputs are sourced through external loop supply



F2-02DAS-2

Firmware Requirements:

To use this module, D2-230 CPUs must have firmware version 2.7 or later. To use the pointer method of writing values, D2-240 CPUs require firmware version 3.0 or later and D2-250 CPUs require firmware version 1.33 or later.

The following tables provide the specifications for the F2-02DAS-2 Isolated Analog Output Module. Review these specifications to make sure the module meets your application requirements.

Output Specifications

Number of Channels	2, isolated
Output Range	0–5VDC, 0–10VDC
Resolution	16 bit (1 in 65536)
Isolation Voltage	±750V continuous, channel to channel, channel to logic
Load Impedance	2KΩ Min
Linearity Error (end to end)	±10 counts (±0.015% of full scale) maximum
Conversion Settling time	3ms to 0.1% of full scale
Full Scale Calibration Error	±32 counts (±0.05%)
Offset Calibration Error	±13 counts (±0.02%)
Maximum Inaccuracy	±0.07% @ 25°C (77°F) ±0.18% @ 0 to 60°C (32 to 140°F)

General Specifications

PLC Update Rate	1 channel per scan maximum (Multiplexing) 2 channels per scan maximum (Pointer [DL240/DL250–1/DL260 only])
Digital outputs Output points required	16 binary data bits, 2 channel ID bits; 32 point (Y) output module
Power Budget Requirement	60 mA @ 5 VDC (supplied by base)
External Power Requirement	21.6–26.4 VDC @ 60 mA
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

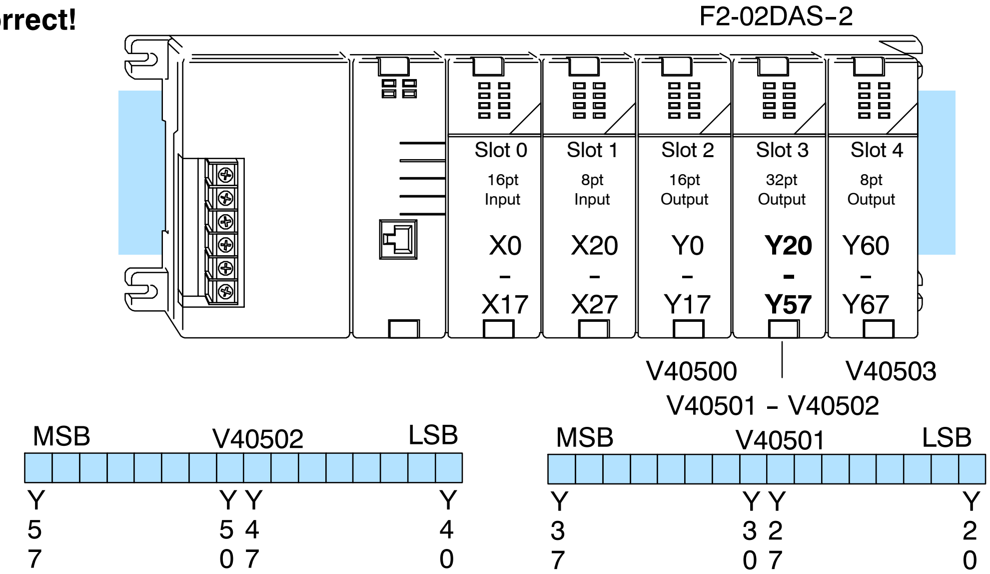
One count in the specification table is equal to one least significant bit of the analog data value (1 in 65536).

Analog Output Configuration Requirements

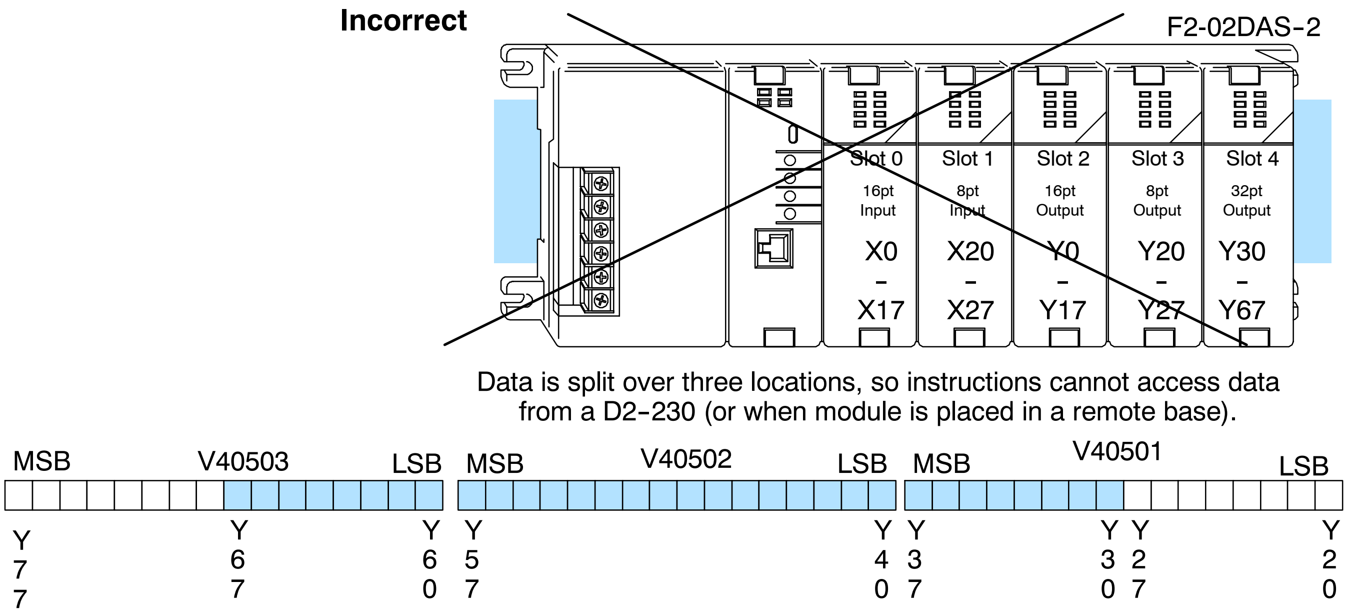
The F2-02DAS-2 analog output requires 32 discrete output points. The module can be installed in any slot of a DL205 system, but the available power budget and discrete I/O points can be limiting factors. Check the user manual for your particular model of CPU and I/O base for more information regarding power budget and number of local, local expanison or remote I/O points.

Special Placement Requirements (DL230 and Remote I/O Bases) Even though the module can be placed in any slot, it is important to examine the configuration if you are using multiplexing ladder. As you can see in the section on writing the program, you use V-memory locations to send the analog data. If you place the module so that the output points do not start on a V-memory boundary, the instructions cannot access the data. This also applies when module is placed in remote base (D2–RSSS in CPU slot).

Correct!



Incorrect



Data is split over three locations, so instructions cannot access data from a D2–230 (or when module is placed in a remote base).

To use the required V-memory references, the *first* output address assigned to the module must be one of the following Y locations. The table also shows the V-memory addresses that correspond to these Y locations.

Y	Y0	Y20	Y40	Y60	Y100	Y120	Y140	Y160
V	V40500	V40501	V40502	V40503	V40504	V40505	V40506	V40507

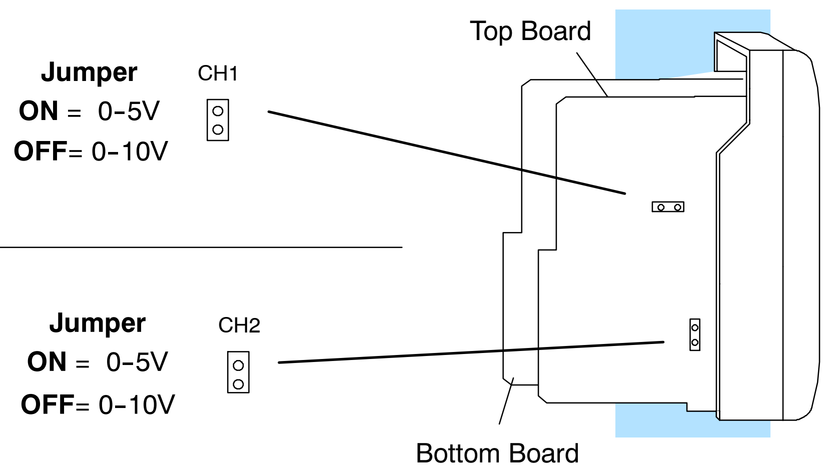
Setting the Module Jumpers

The F2-02DAS–2 Analog Output module uses jumpers for selecting the voltage range for each channel. The range of each channel can be independently set. The available operating ranges are 0–5V and 0–10V.

There is one jumper for each channel. They are on the top board near the user connector side. Install or remove these jumpers to select the desired range. Unused jumpers can be stored on a single pin so they do not get lost. The module comes from the factory set for the 0–5V range.

NOTE: It is important to set the module jumpers correctly. The module will not operate correctly if the jumpers are not properly set for the desired voltage range.

This figure shows the jumper locations.



Connecting the Field Wiring

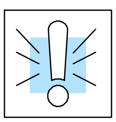
Wiring Guidelines

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 load.
- 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.

Transmitter Power Supply Requirements

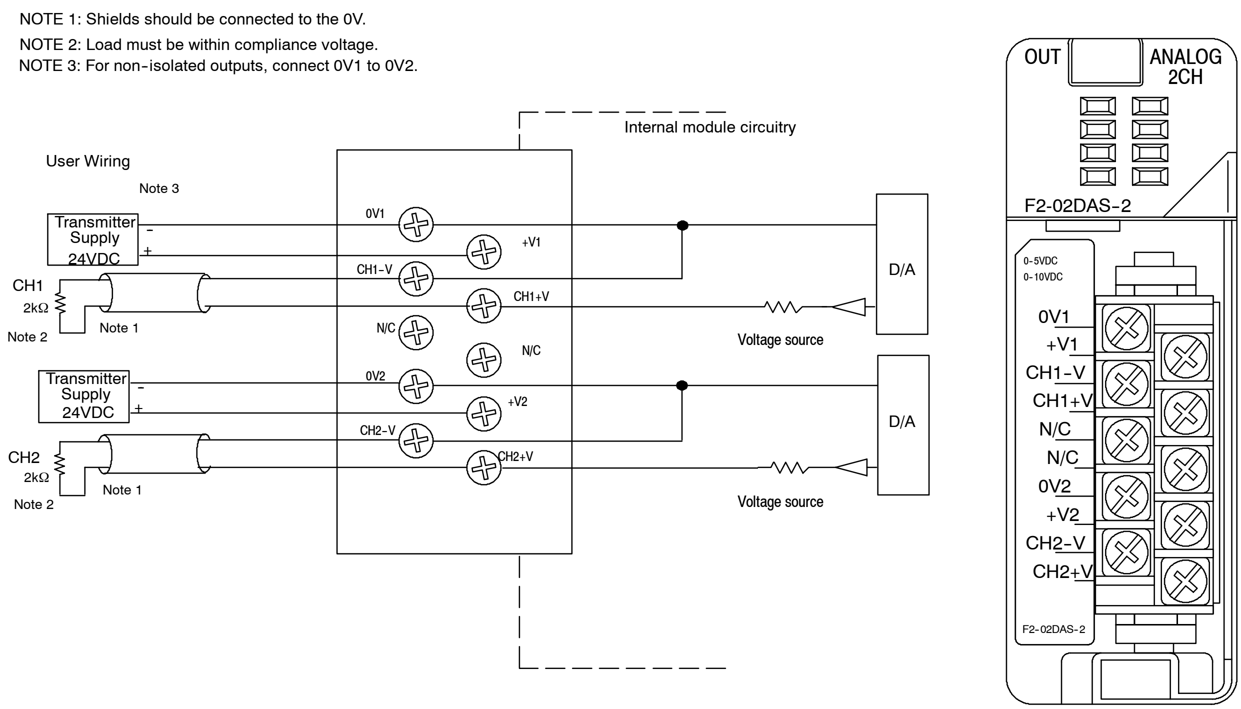
The F2-02DAS-2 requires a separate transmitter power supply. Each channel requires 21.6-26.4 VDC at 60 mA per channel.



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

Wiring Diagram

The F2-02DAS-2 module has a removable connector to make wiring easier. Simply squeeze the top and bottom retaining clips and gently pull the connector from the module. Use the following diagram to connect the field wiring.

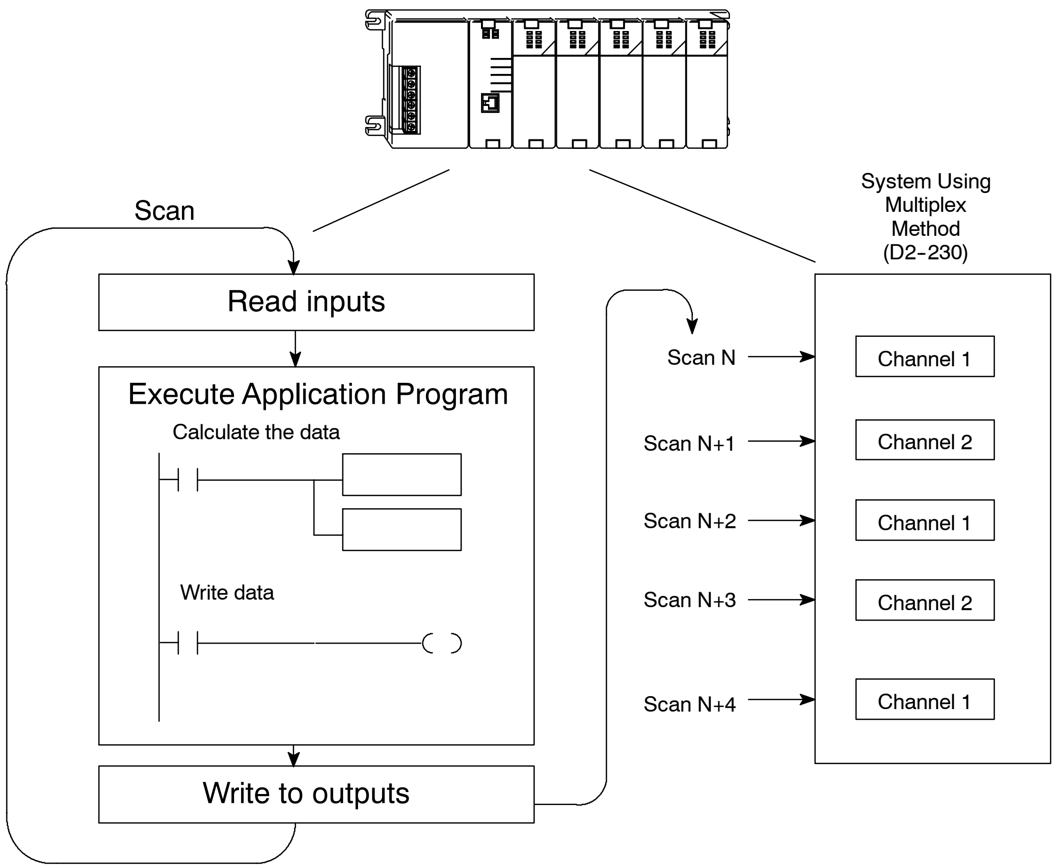


Module Operation

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.

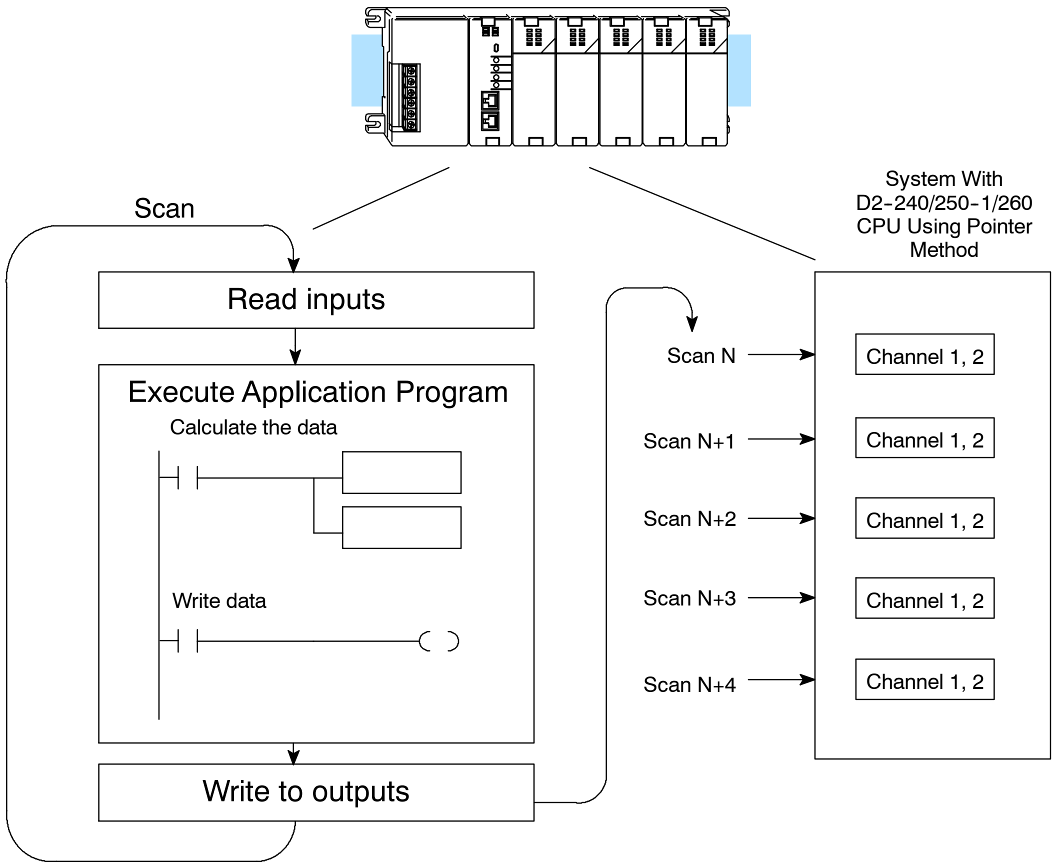
Channel Update Sequence for a DL230 CPU (Multiplexing)

If you are using multiplexing ladder, you can send one channel of data to the output module on each scan. The module refreshes both field devices on each scan, but you can only get new data from the CPU at the rate of one channel per scan. Since there are two channels, it can take two scans to update both channels. However, if you are only using one channel, then you can update that channel on every scan. The multiplexing method can also be used for DL240/250–1/260 CPUs.



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

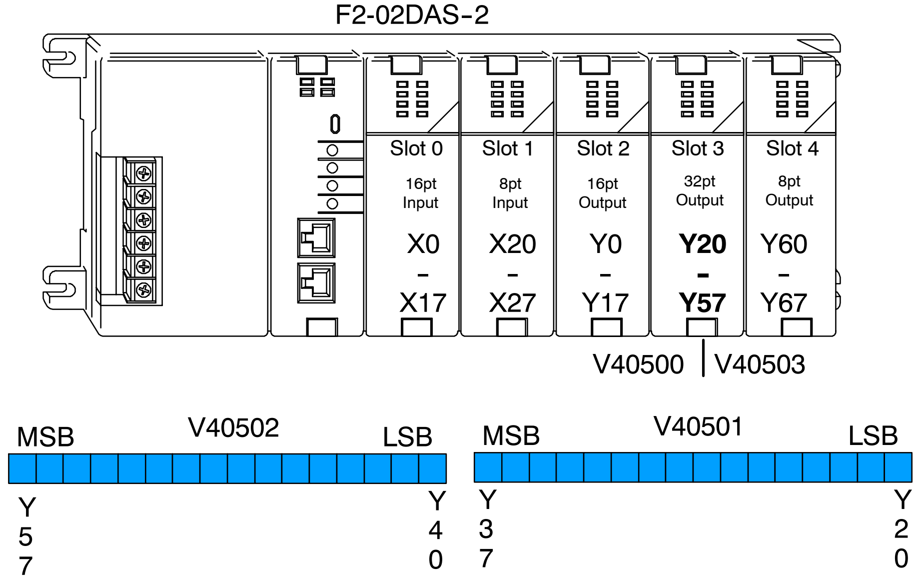
If you are using pointers (Pointer Method), you can update both channels on every scan. This is because the D2-240, D2-250-1 and D2-260 CPUs support special V-memory locations that are used to manage the data transfer. This is discussed in more detail in the section on Writing the Control Program.



Understanding the Output Assignments

You may recall the F2-02DAS-2 module appears to the CPU as a 32-point discrete output module. These points provide the data value and an indication of which channel to update. Note, if you are using a D2-240/250-1/260 CPU, you may never have to use these bits, but it may help you understand the data format.

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.

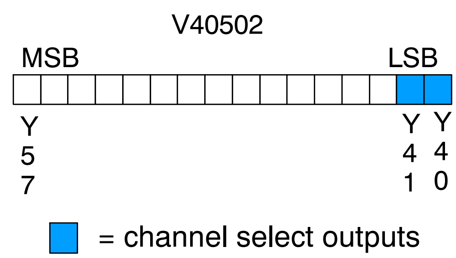


Within this word location, the individual bits represent specific information about the analog signal.

Channel Select Outputs

Two of the outputs select the active channel. Remember, the V-memory bits are mapped directly to discrete outputs. Turning a bit OFF selects its channel. By controlling these outputs, you can select which channel(s) gets updated.

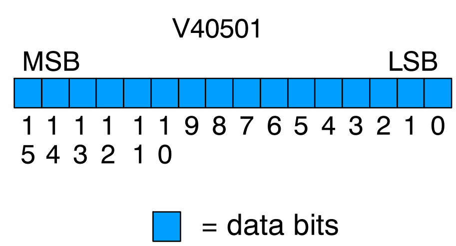
Y41	Y40	Channel
On	Off	1
Off	On	2
Off	Off	1 & 2 (same data to both channels)
On	On	none (both channels hold current values)



Analog Data Bits

The first sixteen bits represent the analog data in binary format.

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

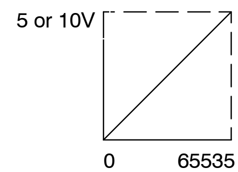


Module Resolution

Since the module has 16-bit resolution, the analog signal is converted into 65536 counts ranging from 0 – 65535 (2^{16}). For example, send a 0 to get a 0V signal and 65535 to get a 10V signal. This is equivalent to a binary value of 0000 0000 0000 0000 to 1111 1111 1111 1111, or 0000 to FFFF hexadecimal. The diagram shows how this relates to the signal range.

Each count can also be expressed in terms of the signal level by using the equation shown.

0–5V or 0–10V



$$\text{Resolution} = \frac{H - L}{65535}$$

H = high limit of the signal range

L = low limit of the signal range

Writing the Control Program

Calculating the Digital Value

Your program must 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.

$$A = U \frac{65535}{H - L}$$

A = Analog value (0 - 65535)

U = Engineering Units

H = high limit of the engineering unit range

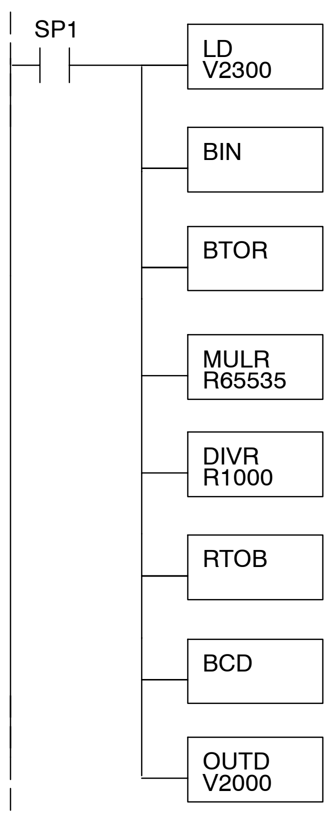
L = low limit of the engineering unit range

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.

$$A = 10U \frac{65535}{10(H - L)} \qquad A = 494 \frac{65535}{1000 - 0} \qquad A = 32374$$

Engineering Units Conversion

The example program shows how you would write the program to perform the engineering unit conversion to output data formats 0 - 65535 when using a D2-250 CPU. This example assumes you have calculated or loaded the engineering unit values in BCD format and stored it in V2300 for channel 1.



The LD instruction loads the engineering units used with channel 1 into the accumulator. This example assumes the numbers are BCD. Since SP1 is used, this rung automatically executes on every scan. You could also use an X, C, etc. permissive contact.

Convert BCD number to binary number.

Convert binary number to real number.

Multiply the accumumlator by 65535 to start the conversion.

Divide the accumulator by 1000 (1000 = 100.0%).

Convert the result to binary.

Convert the result to BCD.

Store the BCD double word result in V2000 / V2001.

Reading Values: Pointer Method and Multiplexing

There are two methods of reading values:

- The pointer method
- Multiplexing

You can use either method when using D2-240, D2-250-1 and D2-260 CPUs, but for ease of programming it is strongly recommended that you use the pointer method. You must use the multiplexing method with remote I/O modules (the pointer method will not work).

Pointer Method

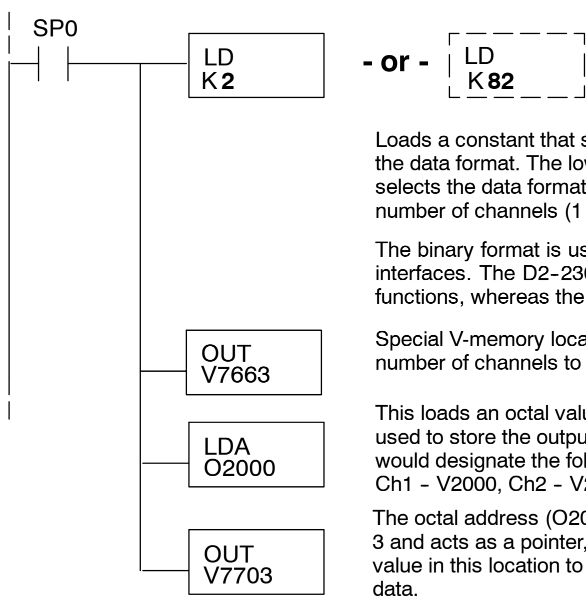
230	240	250-1	260
✕	✓	✓	✓

Once you have calculated the data values (shown previously) you have to enter the program that actually updates the module. The D2-240/250-1/260 has special V-memory locations assigned to each base slot that greatly simplify the programming requirements. By using these V-memory locations you can:

- specify the number of channels to update.
- specify where to obtain the output data.

NOTE: D2-240 CPUs with firmware version 3.0 and D2-250 CPUs with version 1.33 or later support this method.

The following program example shows how to set up these locations. Place this rung anywhere in the ladder program, or in the initial stage when using stage programming. In this example we are using V2000 and V2002 to store the calculated values, and the analog module is installed in slot 3. You should use the appropriate memory locations for your application. The pointer method automatically converts values to binary.



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

The binary format is used for displaying data on some operator interfaces. The D2-230/240 CPUs do not support binary math functions, whereas the D2-250 does.

Special V-memory location assigned to slot 3 that contains the number of channels to scan.

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

The octal address (O2000) is stored here. V7703 is assigned to slot 3 and acts as a pointer, which means the CPU will use the octal value in this location to determine exactly where to store the output data.

The tables below show the special V-memory locations used by the DL240, DL250–1 and DL260 for the CPU base and local expansion base I/O slots. Slot 0 (zero) is the module next to the CPU or D2–CM module. Slot 1 is the module two places from the CPU or D2–CM, and so on. Remember, the CPU only examines the pointer values at these locations after a mode transition. Also, if you use the DL230 (multiplexing) method, verify that these addresses in the CPU are zero.

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

CPU Base: Analog Output Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V7660	V7661	V7662	V7663	V7664	V7665	V7666	V7667
Storage Pointer	V7700	V7701	V7702	V7703	V7704	V7705	V7706	V7707

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

Expansion Base D2–CM #1: Analog Output Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36000	V36001	V36002	V36003	V36004	V36005	V36006	V36007
Storage Pointer	V36020	V36021	V36022	V36023	V36024	V36025	V36026	V36027

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

Expansion Base D2–CM #2: Analog Output Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36100	V36101	V36102	V36103	V36104	V36105	V36106	V36107
Storage Pointer	V36120	V36121	V36122	V36123	V36124	V36125	V36126	V36127

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

Expansion Base D2–CM #3: Analog Output Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36200	V36201	V36202	V36203	V36204	V36205	V36206	V36207
Storage Pointer	V36220	V36221	V36222	V36223	V36224	V36225	V36226	V36227

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

Expansion Base D2–CM #4: Analog Output Module Slot-Dependent V-memory Locations								
Slot	0	1	2	3	4	5	6	7
No. of Channels	V36300	V36301	V36302	V36303	V36304	V36305	V36306	V36307
Storage Pointer	V36320	V36321	V36322	V36323	V36324	V36325	V36326	V36327

Writing Data (Multiplexing)

230	240	250-1	260
✓	✓	✓	✓

Since all channels are multiplexed into a single data word, the control program can be setup to determine which channel to write. Since the module appears as Y output points to the CPU, it is very easy to use the channel selection outputs to determine which channel to update.

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

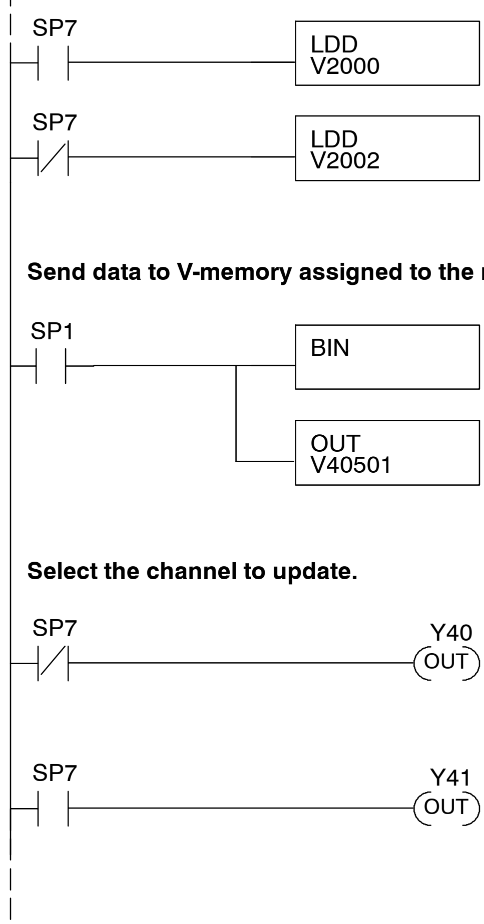
This example is a two-channel multiplexer that updates each channel on alternate scans. Relay SP7 is a special relay that is on for one scan, then off for one scan.

NOTE: You must send binary data to the module. If the data is already in binary format, you should not use the BIN instruction shown in this example.

Load data into the accumulator.

Send data to V-memory assigned to the module.

Select the channel to update.



Loads the data for channel 1 into the accumulator.
Note: Use LD if using binary, and use LDD if using BCD.

Loads the data for channel 2 into the accumulator.
Note: Use LD if using binary, and use LDD if using BCD.

Convert the data to binary (you must omit this step if you have converted the data elsewhere). SP1 is always on.

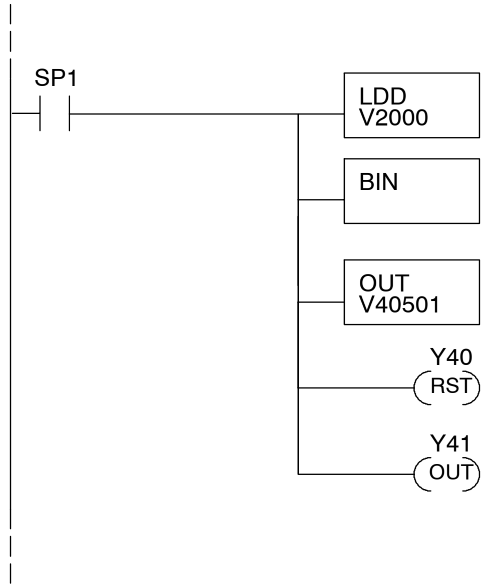
The OUT instruction sends the data to the module. Our example starts with V40501, but the actual value depends on the location of the module in your application.

Selects channel 2 for update when Y41 is OFF (Y40-ON deselects channel 1). Note, Y40 and Y41 are used as in the previous examples. If the module was installed in a different I/O arrangement the addresses would be different.

Selects channel 1 for update when Y40 is OFF (Y41-ON deselects channel 2). Note, Y40 and Y41 are used as in the previous examples. If the module was installed in a different I/O arrangement the addresses would be different.

Sending Data to One Channel

If you are not using both channels, or if you want to control the updates separately, use the following program.



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

Note: Use LD if using binary, and use LDD if using BCD.

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

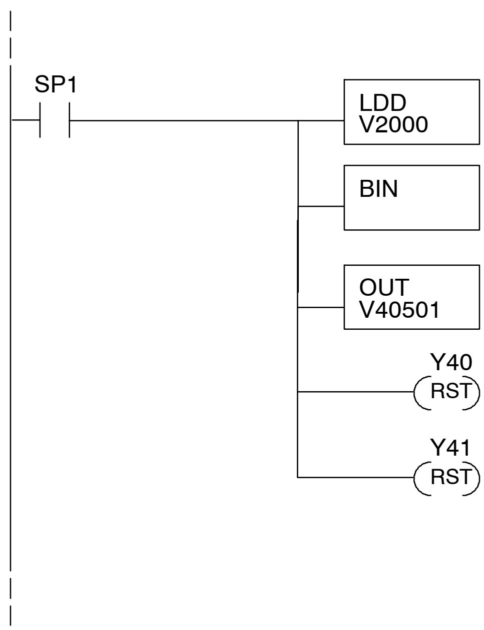
The OUT instruction sends the data to the module. Our example starts with V40501, but the actual value depends on the location of the module in your application.

Y40–OFF selects channel 1 for updating.

Y41–ON deselects channel 2 (do not update).

Sending the Same Data to Both Channels

If both channel selection outputs are off, both channels will be updated with the same data.



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

Note: Use LD if using binary, and use LDD if using BCD.

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

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.

Y40–OFF selects channel 1 for updating.

Y41–OFF selects channel 2 for updating.

Analog and Digital Value Conversions

Sometimes it is useful to be able to quickly convert between the signal levels and the digital values. This is especially helpful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier. Remember, if you imbed the sign information into the data value, you must adjust the formulas accordingly.

Range	If you know the digital value ...	If you know the signal level ...
0–5 VDC	$A = \frac{5D}{65535}$	$D = \frac{65535}{5}A$
0–10 VDC	$A = \frac{10D}{65535}$	$D = \frac{65535}{10}A$

For example, if you know you need a 4V signal to achieve the desired result, you can easily determine the digital value that should be used.

$$D = \frac{65535}{5}A$$

$$D = \frac{65535}{5}(4)$$

$$D = (13107)\ (4)$$

$$D = 52428(CCCC_h)$$