DL405 Analog
I/O Modules

D4–ANLG–M

WARNING

Thank you for purchasing automation equipment from ***Automationdirect.com***™, doing business as **AutomationDirect**. We want your new automation equipment to operate safely. Anyone who installs or uses this equipment should read this publication (and any other relevant publications) before installing or operating the equipment.

To minimize the risk of potential safety problems, you should follow all applicable local and national codes that regulate the installation and operation of your equipment. These codes vary from area to area and usually change with time. It is your responsibility to determine which codes should be followed, and to verify that the equipment, installation, and operation are in compliance with the latest revision of these codes.

At a minimum, you should follow all applicable sections of the National Fire Code, National Electrical Code, and the codes of the National Electrical Manufacturer's Association (NEMA). There may be local regulatory or government offices that can also help determine which codes and standards are necessary for safe installation and operation.

Equipment damage or serious injury to personnel can result from the failure to follow all applicable codes and standards. We do not guarantee the products described in this publication are suitable for your particular application, nor do we assume any responsibility for your product design, installation, or operation.

Our products are not fault–tolerant and are not designed, manufactured or intended for use or resale as on–line control equipment in hazardous environments requiring fail–safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the product could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). **AutomationDirect** *specifically disclaims any expressed or implied warranty of fitness for High Risk Activities.*

For additional warranty and safety information, see the Terms and Conditions section of our Desk Reference. If you have any questions concerning the installation or operation of this equipment, or if you need additional information, please call us at 770–844–4200.

This publication is based on information that was available at the time it was printed. At **AutomationDirect** we constantly strive to improve our products and services, so we reserve the right to make changes to the products and/or publications at any time without notice and without any obligation. This publication may also discuss features that may not be available in certain revisions of the product.

Trademarks

This publication may contain references to products produced and/or offered by other companies. The product and company names may be trademarked and are the sole property of their respective owners. **AutomationDirect** disclaims any proprietary interest in the marks and names of others.

Copyright 2004, *Automationdirect.com*™ Incorporated
All Rights Reserved

No part of this manual shall be copied, reproduced, or transmitted in any way without the prior, written consent of ***Automationdirect.com***™ Incorporated. **AutomationDirect** retains the exclusive rights to all information included in this document.

AVERTISSEMENT

Nous vous remercions d'avoir acheté l'équipement d'automatisation de **Automationdirect.com^{MC}** en faisant des affaires comme **AutomationDirect**. Nous tenons à ce que votre nouvel équipement d'automatisation fonctionne en toute sécurité. Toute personne qui installe ou utilise cet équipement doit lire la présente publication (et toutes les autres publications pertinentes) avant de l'installer ou de l'utiliser.

Afin de réduire au minimum le risque d'éventuels problèmes de sécurité, vous devez respecter tous les codes locaux et nationaux applicables régissant l'installation et le fonctionnement de votre équipement. Ces codes diffèrent d'une région à l'autre et, habituellement, évoluent au fil du temps. Il vous incombe de déterminer les codes à respecter et de vous assurer que l'équipement, l'installation et le fonctionnement sont conformes aux exigences de la version la plus récente de ces codes.

Vous devez, à tout le moins, respecter toutes les sections applicables du Code national de prévention des incendies, du Code national de l'électricité et des codes de la National Electrical Manufacturer's Association (NEMA). Des organismes de réglementation ou des services gouvernementaux locaux peuvent également vous aider à déterminer les codes ainsi que les normes à respecter pour assurer une installation et un fonctionnement sûrs.

L'omission de respecter la totalité des codes et des normes applicables peut entraîner des dommages à l'équipement ou causer de graves blessures au personnel. Nous ne garantissons pas que les produits décrits dans cette publication conviennent à votre application particulière et nous n'assumons aucune responsabilité à l'égard de la conception, de l'installation ou du fonctionnement de votre produit.

Nos produits ne sont pas insensibles aux défaillances et ne sont ni conçus ni fabriqués pour l'utilisation ou la revente en tant qu'équipement de commande en ligne dans des environnements dangereux nécessitant une sécurité absolue, par exemple, l'exploitation d'installations nucléaires, les systèmes de navigation aérienne ou de communication, le contrôle de la circulation aérienne, les équipements de survie ou les systèmes d'armes, pour lesquels la défaillance du produit peut provoquer la mort, des blessures corporelles ou de graves dommages matériels ou environnementaux ("activités à risque élevé"). La société **AutomationDirect** nie toute garantie expresse ou implicite d'aptitude à l'emploi en ce qui a trait aux activités à risque élevé.

Pour des renseignements additionnels touchant la garantie et la sécurité, veuillez consulter la section Modalités et conditions de notre documentation. Si vous avez des questions au sujet de l'installation ou du fonctionnement de cet équipement, ou encore si vous avez besoin de renseignements supplémentaires, n'hésitez pas à nous téléphoner au 770–844–4200.

Cette publication s'appuie sur l'information qui était disponible au moment de l'impression. À la société **AutomationDirect**, nous nous efforçons constamment d'améliorer nos produits et services. C'est pourquoi nous nous réservons le droit d'apporter des modifications aux produits ou aux publications en tout temps, sans préavis ni quelque obligation que ce soit. La présente publication peut aussi porter sur des caractéristiques susceptibles de ne pas être offertes dans certaines versions révisées du produit.

Marques de commerce

La présente publication peut contenir des références à des produits fabriqués ou offerts par d'autres entreprises. Les désignations des produits et des entreprises peuvent être des marques de commerce et appartiennent exclusivement à leurs propriétaires respectifs. **AutomationDirect^{MC}** nie tout intérêt dans les autres marques et désignations.

Copyright 2004, Automationdirect.com^{MC} Incorporated

Tous droits réservés

Nulle partie de ce manuel ne doit être copiée, reproduite ou transmise de quelque façon que ce soit sans le consentement préalable écrit de la société **Automationdirect.com^{MC}** Incorporated. **AutomationDirect** conserve les droits exclusifs à l'égard de tous les renseignements contenus dans le présent document.

Table of Contents

Chapter 1: Getting Started

Introduction 1–2
The Purpose of this Manual 1–2
Supplemental Manuals 1–2
Technical Support 1–2
Conventions Used 1–3
Key Topics for Each Chapter 1–3
Analog Input Module Terminology 1–4
Channels per Module 1–4
Input Ranges 1–4
Resolution 1–4
Input Type 1–4
Input Impedance 1–4
Conversion Method 1–4
PLC Update Rate 1–4
Linearity Error 1–4
Maximum Inaccuracy 1–4
Accuracy vs. Temperature 1–4
I/O Points Required 1–4
External Power Source 1–4
Base Power Required 1–4
Operating Temperature 1–4
Relative Humidity 1–4
Step Response 1–4
Analog Output Module Terminology 1–5
Channels per Module 1–5
Output Ranges 1–5
Resolution 1–5
Output Current 1–5
Output Impedance 1–5
Load Impedance 1–5
PLC Update Rate 1–5
Linearity Error 1–5
Maximum Inaccuracy 1–5
Accuracy vs. Temperature 1–5
External Power Source 1–5
Base Power Required 1–5
Operating Temperature 1–5
Relative Humidity 1–5
I/O Points Required 1–5
Selecting the Appropriate Module 1–6
Wide Variety of Modules 1–6
Diagnostic Features 1–6

Analog Input Modules 1–7
Special Input Modules 1–8
Analog Output Modules 1–9
Physical Characteristics 1–10
Frequently Asked Questions 1–10
Analog Made Easy – Four Simple Steps 1–11

Chapter 2: Installation and Wiring

Safety Guidelines 2–2
Plan for Safety 2–2
Safety Techniques 2–2
Orderly System Shutdown 2–3
System Power Disconnect 2–3
I/O Wiring Strategies 2–4
Wiring Guidelines 2–4
I/O Module Placement and Wiring 2–5
Module Placement 2–5
I/O Module Status Indicators 2–6
Color Coding of I/O Modules 2–6
Wiring a Module with a Terminal Block 2–7

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

Module Specifications 3–2
Analog Input Configuration Requirements 3–2
Input Specifications 3–3
General Specifications 3–3
Setting the Module Jumpers 3–4
Jumper Locations 3–4
Factory Default Settings 3–4
Selecting the Number of Active Channels 3–4
Selecting 16 Input or 32 Input Modes 3–5
Operating Range Selection 16 Input Mode 3–6
Operating Range Selection 32 Input Mode 3–7
Units Selection for 32 Input Mode 3–8
Connecting the Field Wiring 3–9
Wiring Guidelines 3–9
User Power Supply Requirements 3–9
Using Current or Voltage Wiring 3–9
Current Loop Transmitter Impedance 3–11
Wiring Diagram 3–11

Module Operation 3–12
DL430 Special Requirements 3–12
Channel Scanning Sequence 3–13
Displaying Diagnostic Data 3–13
Displaying Channel Data 3–14
Input Assignments for 16 Input Mode 3–14
Active Channel Indicator Inputs, 16 Input Mode 3–15
Analog Data Bits, 16 Input Mode 3–15
Sign Bit, 16 Input Mode 3–15
Broken Transmitter Bit, 16 Input Mode 3–15
Input Assignments for 32 Input Mode 3–16
Active Channel Indicator Inputs, 32 Input Mode 3–16
Analog Data Bits, 32 Input Mode 3–17
Sign Bits, 32 Input Mode 3–18
Broken Transmitter bits, 32 Input Mode 3–18
12-Bit Magnitude Plus Sign Format, (All Modes) 3–19
Two’s Complement Format, 13-Bit 3–19
Two’s Complement Format, 12-Bit 3–21
Unipolar Resolution 3–22
Bipolar Resolution 3–22

Writing the Control Program, 16 Input Mode 3–23
Multiple Active Channels 3–23
Reading Values, DL440/450 3–23
Optional Method, DL440/450 3–24
Reading Values, DL430 3–24
Single Active Channel 3–25
Reading Four Channels in One Scan, DL440/450 3–25
Reading Values With Sign Bits, DL440/450 3–26
Broken Transmitter Detection 3–26

Writing the Control Program, 32 Input Mode 3–27
Multiple Active Channels 3–27
Reading Values, DL440/450 3–28
Reading Values, DL430 3–28
Single Active Channel 3–29
Reading Four Channels in One Scan, DL440/450 3–29
Reading Values With Sign Bits 3–30
Broken Transmitter Detection 3–30

Scaling and Converting the Input Data 3–30
Scaling the Input Data 3–30
16 Input Mode Example 3–31
32 Input Mode Example 3–31
Analog and Digital Value Conversions 3–32

Configuration Cross-Reference, D4-04AD to F4-04AD 3–33
Step 1: 32-Input Mode 3–33
Step 2: Range Selection 3–34
Step 3: Units Selection 3–34
Step 4: Number of Active Channels Selection 3–35

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

Module Specifications 4–2
Analog Input Configuration Requirements 4–2
Input Specifications 4–3
General Specifications 4–3
Setting the Module Jumpers 4–4
Jumper Locations 4–4
Factory Default Settings 4–4
Selecting the Number of Channels 4–5
Example Settings 4–5
Connecting the Field Wiring 4–7
Wiring Guidelines 4–7
User Power Supply Requirements 4–7
Custom Input Ranges 4–7
Current Loop Transmitter Impedance 4–8
Removable Connector 4–8
Wiring Diagram 4–8
Module Operation 4–9
DL430 Special Requirements 4–9
Channel Scanning Sequence 4–10
Input Bit Assignments 4–11
Active Channel Indicator Inputs 4–11
Analog Data Bits 4–12
Writing the Control Program 4–13
Multiple Channels Selected 4–13
Reading Values, DL440/450 4–13
Reading Values, DL430 4–14
Single Channel Selected 4–14
Reading Four Channels in One Scan, DL440/450 Only 4–15
Scaling the Input Data 4–15
Analog and Digital Value Conversions 4–17

Chapter 5: F4-08AD 8-Channel Analog Input

Module Specifications 5–2
Analog Input Configuration Requirements 5–2
Input Specifications 5–3
General Specifications 5–3
Setting the Module Jumpers 5–4
Jumper Locations 5–4
Selecting the Number of Channels 5–4
Selecting Current or Voltage 5–5
Selecting Input Signal and Ranges 5–5
Connecting the Field Wiring 5–6
Wiring Guidelines 5–6
User Power Supply Requirements 5–6
Custom Input Ranges 5–6

Current Loop Transmitter Impedance 5–7
Removable Connector 5–7
Wiring Diagram 5–8
Module Operation 5–9
DL430 Special Requirements 5–9
Channel Scanning Sequence 5–10
Input Bit Assignments 5–11
Active Channel Indicator Inputs 5–11
Analog Data Bits 5–12
Unusable MSB Bit 5–12
Writing the Control Program 5–13
Multiple Channels Selected 5–13
Reading Values, DL430 CPU 5–14
Single Channel Selected 5–15
Reading Values, DL440/450 5–15
Reading Eight Channels in One Scan, DL440/450 5–16
Scaling the Input Data 5–16
Analog and Digital Value Conversions 5–18

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

Module Specifications 6–2
Analog Input Configuration Requirements 6–2
Input Specifications 6–3
General Specifications 6–3
Setting the Module Jumpers 6–4
Jumper Locations 6–4
Selecting the Number of Channels 6–5
Selecting Input Ranges 6–5
Connecting the Field Wiring 6–6
Wiring Guidelines 6–6
User Power Supply Requirements 6–6
Current Loop Transmitter Impedance 6–7
Removable Connector 6–8
Wiring Diagram 6–8
Module Operation 6–9
DL430 Special Requirements 6–9
Channel Scanning Sequence 6–10
Input Bit Assignments 6–11
Active Channel Indicator Inputs 6–11
Analog Data Bits 6–12
Writing the Control Program 6–13
Multiple Channels Selected 6–13
Storing the Channel Data to V–Memory 6–13
Reading Values, DL430 CPU 6–14
Single Channel Selected 6–15
Reading Values 6–15
Reading sixteen Channels in One Scan 6–16

Scaling the Input Data 6–16
Analog and Digital Value Conversions 6–18

Chapter 7: F4–16AD–2 8-Channel Analog Input

Module Specifications 7–2
Analog Input Configuration Requirements 7–2
Input Specifications 7–3
General Specifications 7–3
Setting the Module Jumpers 7–4
Jumper Locations 7–4
Selecting the Number of Channels 7–5
Selecting Input Signal Ranges 7–5
Connecting the Field Wiring 7–6
Wiring Guidelines 7–6
User Power Supply Requirements 7–6
Custom Input Ranges 7–7
Removable Connector 7–8
Wiring Diagram 7–8
Module Operation 7–9
D4–430 Special Requirements 7–9
Channel Scanning Sequence 7–10
Input Bit Assignments 7–11
Active Channel Indicator Inputs 7–11
Analog Data Bits 7–12
Writing the Control Program 7–13
Multiple Channels Selected 7–13
Storing the Channel Data to V–Memory 7–13
Reading Values 7–14
Single Channel Selected 7–15
Reading Values 7–15
Reading Sixteen Channels in One Scan 7–16
Scaling the Input Data 7–16
Analog and Digital Value Conversions 7–18

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

Module Specifications 8–2
Input Specifications 8–3
General Specifications 8–3
Module Calibration 8–4
Thermocouple Input Configuration Requirements 8–4
Setting the Module Jumpers 8–4
Jumper Locations 8–4
Factory Default Settings 8–4
Selecting the Number of Channels 8–5
Selecting Fahrenheit or Celsius 8–5
Selecting Temperature or Counts 8–6

Connecting the Field Wiring **8–6**
Wiring Guidelines 8–6
User Power Supply Requirements 8–6
Wiring Diagram 8–7
Module Operation **8–8**
DL430 Special Requirements 8–8
Channel Scanning Sequence 8–9
Identifying the Data Locations 8–10
Active Channel Indicator Inputs 8–10
Temperature Sign Bit 8–11
Analog Data Bits 8–11
Temperature Input Resolution 8–11
Millivolt Input Resolution 8–11
Writing the Control Program **8–12**
Multiple Channels Selected 8–12
Automatic Temperature Conversion 8–12
Reading Values, DL430 8–13
Single Channel Selected 8–14
Reading Values, DL440/450 8–14
Reading Eight Channels in One Scan, DL440/450 8–15
Using the Sign Bit, DL440/450 8–16
Scaling the Input Data 8–17
Temperature and Digital Value Conversions 8–18
Millivolt and Digital Value Conversions 8–18

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

Module Specifications **9–2**
Module Calibration 9–2
RTD Input Configuration Requirements 9–2
Input Specifications 9–3
General Specifications 9–3
Setting the Module Jumpers **9–4**
Jumper Locations 9–4
Factory Default Settings 9–4
Selecting Fahrenheit or Celsius 9–4
Selecting RTD Type 9–5
Selecting the Number of Channels 9–6
Connecting the Field Wiring **9–7**
Wiring Guidelines 9–7
RTD – Resistance Temperature Detector 9–7
Ambient Variations in Temperature 9–7
Wiring Diagram 9–8
Module Operation **9–9**
DL430 Special Requirements 9–9
Channel Scanning Sequence 9–10
Identifying the Data Locations 9–11

Writing the Control Program 9–12
Multiple Active Channels 9–12
Analog Data and Sign Bits 9–12
Active Channel Bits 9–13
Broken Transmitter Bits 9–13
Reading Values, DL430 9–13
Reading Values, DL440/450 9–14
Reading Eight Channels in One Scan, DL440/DL450 9–15
Using Bipolar Ranges (Magnitude Plus Sign) 9–16
Using Bipolar Ranges Example Continued 9–17
Reading the Input Data 9–17

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

Module Specifications 10–2
General Specifications 10–2
Thermocouple Specifications 10–3
Voltage Specifications 10–3
Module Calibration 10–3
Thermocouple Input Configuration Requirements 10–3
Setting the Module Jumpers 10–4
Jumper Locations 10–4
Calibrate Enable 10–4
Selecting the Number of Channels 10–5
Setting Input Type 10–6
Selecting the Conversion Units 10–6
Thermocouple Conversion Units 10–6
Voltage Conversion Units 10–7
Connecting the Field Wiring 10–8
Wiring Guidelines 10–8
User Power Supply Requirements 10–8
Thermocouples 10–9
Ambient Variations in Temperature 10–9
Wiring Diagram 10–10
Module Operation 10–11
DL430 Special Requirements 10–11
Channel Scanning Sequence 10–12
Identifying the Data Locations 10–13
Writing the Control Program 10–14
Multiple Active Channels 10–14
Analog Data and Sign Bits 10–14
Active Channel Bits 10–15
Broken Transmitter Bits 10–15
Reading Values, DL430 10–15
Reading Values, DL440/450 10–16
Using Bipolar Ranges (Magnitude Plus Sign) 10–17
Using Bipolar Ranges Example Continued 10–18
Reading the Input Data 10–18
Module Resolution16-Bit (Unipolar Voltage Input) 10–19

Module Resolution 15-Bit Plus Sign (Bipolar Voltage Input) 10–19
Analog and Digital Value Conversions 10–20

Chapter 11: D4-02DA 2-Channel Analog Output

Module Specifications 11–2
Analog Output Configuration Requirements 11–2
Output Specifications 11–3
General Module Specifications 11–3
Connecting the Field Wiring 11–4
Wiring Guidelines 11–4
User Power Supply Requirements 11–4
Load Requirements 11–4
Using the 1–5 VDC Range 11–4
Wiring Diagram 11–5
Module Operation 11–6
DL430 Special Requirements 11–6
Channel Scanning Sequence 11–7
Output Bit Assignments 11–8
Analog Data Bits 11–9
Writing the Control Program 11–10
Update Either Channel 11–10
Calculating the Digital Value 11–10
Sending the Data to the Module 11–11
Analog and Digital Value Conversions 11–12

Chapter 12: F4–04DA 4-Channel Analog Output

Module Specifications 12–2
Analog Output Configuration Requirements 12–2
Output Specifications 12–3
General Module Specifications 12–3
Setting the Module Jumpers 12–4
Jumper Locations 12–4
Selecting Common Output Signal Ranges 12–4
Selecting Special Signal Ranges 12–5
Connecting the Field Wiring 12–6
Wiring Guidelines 12–6
User Power Supply Requirements 12–6
Load Requirements 12–7
Removable Connector 12–7
Wiring Diagram 12–7
Module Operation 12–8
DL430 Special Requirements 12–8
Channel Scanning Sequence 12–9
Output Bit Assignments 12–10
Channel Select Bits 12–10
Analog Data Bits 12–11

Writing the Control Program **12–12**
Update Any Channel 12–12
Calculating the Digital Value 12–12
Engineering Unit Conversion 12–13
Sending Data to One Channel 12–14
Sending the Same Data to All Channels 12–15
Sequencing the Channel Updates 12–16
Sequencing Example 1, DL440/450 12–17
Sequencing Example 2, DL430 12–18
Sequencing Example 3, DL440/450 12–19
Example 3 Continued 12–20
SequencingExample 4, DL430 12–21
Example 4 Continued 12–22
Updating All Channels in a Single Scan, DL440/450 12–23
Analog and Digital Value Conversions 12–24

Chapter 13: F4-04DA-1 4-Channel Analog Current Output

Module Specifications **13–2**
Output Specifications 13–2
General Module Specifications 13–3
Setting the Module Jumpers **13–3**
Connecting the Field Wiring **13–4**
Wiring Guidelines 13–4
User Power Supply Requirements 13–4
Load Requirements 13–4
Removable Connector 13–5
Wiring Diagram 13–5
Module Operation **13–6**
DL430 Special Requirements 13–6
Analog Output Configuration Requirements 13–7
Channel Update Sequence 13–7
Output Bit Assignments 13–8
Channel Select Bits 13–8
Output Enable Bit 13–9
Analog Data Bits 13–9
Module Resolution 13–9
Writing the Control Program **13–10**
Update Any Channel 13–10
Calculating the Digital Value 13–10
V-Memory Registers 13–11
Sending Data to One Channel 13–12
Sequencing the Channel Updates 13–13
Sequencing Example 1, DL440/450 13–14
Sequencing Example 2, DL430 13–15
Sequencing Example 3, DL440/DL450 13–16
SequencingExample 4, DL430 13–18
Updating all Channels in a Single Scan 13–20
Analog and Digital Value Conversions 13–21

Chapter 14: F4–04DA–2 4-Channel Analog Voltage Output

Module Specifications 14–2
Analog Output Configuration Requirements 14–2
Output Specifications 14–3
General Module Specifications 14–3
Setting the Module Jumpers 14–4
Jumper Locations 14–4
Factory Default Settings 14–4
Voltage Range Selection 14–4
Data Format Selection 14–6
Connecting the Field Wiring 14–7
Wiring Guidelines 14–7
User Power Supply Requirements 14–7
Load Requirements 14–8
Removable Connector 14–8
Wiring Diagram 14–8
Module Operation 14–9
DL430 Special Requirements 14–9
Channel Scanning Sequence 14–10
Output Bit Assignments 14–11
Channel Select Bits 14–11
Analog Data Bits 14–12
Signal Sign Bit 14–12
Bipolar Output Data 14–12
Output Enable Bit 14–13
Module Resolution 14–13
Writing the Control Program 14–14
Update Any Channel 14–14
Calculating the Digital Value 14–14
V-Memory Registers 14–15
Sending Data to One Channel 14–16
Sequencing the Channel Updates 14–17
Sequencing Example 1, DL440/450 14–18
Sequencing Example 2, DL430 14–19
Sequencing Example 3, DL440/DL450 14–20
Sequencing Example 4, DL430 14–22
Sequencing Example 5, Bipolar Data 14–24
Updating all Channels in a Single Scan, DL440/450 14–26
Analog and Digital Value Conversions 14–27
Configuration Cross-Reference, F4–04DA to F4–04DA–2 14–28
Step 1: Select the Compatibility Mode 14–28
Step 2: Voltage Range Selection 14–29

Chapter 15: F4–08DA–1 8-Channel Analog Current Output

Module Specifications 15–2
Analog Output Configuration Requirements 15–2
Output Specifications 15–3
General Module Specifications 15–3
Connecting the Field Wiring 15–4
Wiring Guidelines 15–4
User Power Supply Requirements 15–4
Load Requirements 15–4
Removable Connector 15–5
Wiring Diagram 15–5
Module Operation 15–6
DL430 Special Requirements 15–6
Channel Scanning Sequence 15–7
Output Bit Assignments 15–8
Channel Select Bits 15–8
Output Enable Bit 15–9
Analog Data Bits 15–9
Module Resolution 15–9
Writing the Control Program 15–10
Update Any Channel 15–10
Calculating the Digital Value 15–10
V-Memory Registers 15–11
Sending Data to One Channel 15–12
Sequencing the Channel Updates 15–13
Sequencing Example 1, DL440/450 15–14
Sequencing Example 2, DL430 15–17
Sequencing Example 3, DL440/450 15–18
Sequencing Example 4, DL430 15–20
Updating all Channels in a Single Scan, DL440/450 15–22
Updating a Single Channel During a Scan, DL440/450 15–24
Analog and Digital Value Conversions 15–24

Chapter 16: F4–16DA–1 16-Channel Analog Current Output

Module Specifications 16–2
Analog Output Configuration Requirements 16–2
Output Specifications 16–3
General Module Specifications 16–3
Connecting the Field Wiring 16–4
Wiring Guidelines 16–4
User Power Supply Requirements 16–4
Load Requirements 16–4
Removable Connector 16–5
Wiring Diagram 16–5
Module Operation 16–6
DL430 Special Requirements 16–6

Channel Scanning Sequence 16–7
Output Bit Assignments 16–8
Channel Select Bits 16–9
Output Enable Bits 16–9
Analog Data Bits 16–10
Module Resolution 16–10

Writing the Control Program 16–11

Update Any Channel 16–11
Calculating the Digital Value 16–11
V-Memory Registers 16–12
Sending Data to One Channel, DL440/450 16–13
Sequencing the Channel Updates 16–14
Sequencing Example 1, DL440/450 16–15
Sequencing Example 2, DL430 16–18
Sequencing Example 3, DL440/450 16–19
Sequencing Example 4, DL430 16–22
Updating All Channels in a Single Scan, DL440/450 16–25
Updating a Pair of Channels in a Single Scan, DL440/450 16–27
Analog and Digital Value Conversions 16–28

Chapter 17: F4–04DAS–1 4-Channel Iso. 4–20mA Output

Module Specifications 17–2

Analog Output Configuration Requirements 17–2
Output Specifications 17–3
General Module Specifications 17–3

Connecting the Field Wiring 17–4

Wiring Guidelines 17–4
Removable Connector 17–4
Wiring Diagram 17–4

Module Operation 17–5

DL430 Special Requirements 17–5
Channel Scanning Sequence 17–6
Output Bit Assignments 17–7
Channel Select Bits 17–7
Analog Data Bits 17–8
Output Enable Bit 17–8
Module Resolution 17–8

Writing the Control Program 17–9

Update Any Channel 17–9
Calculating the Digital Value 17–9
Engineering Unit Conversion 17–10
V-Memory Registers 17–10
Sending Data to One Channel 17–11
Sequencing the Channel Updates 17–12
Sequencing Example 1, DL430/440/450 17–13
Sequencing Example 2, DL430/440/450 17–14
Updating all Channels in a Single Scan, DL440/450 17–15
Analog and Digital Value Conversions 17–16

Chapter 18: F4–04DAS–2 4-Channel Isolated Volt Output

Module Specifications 18–2
Analog Output Configuration Requirements 18–2
Output Specifications 18–3
General Module Specifications 18–3
Setting the Module Jumpers 18–4
Output Range Selection 18–4
Connecting the Field Wiring 18–5
Wiring Guidelines 18–5
Removable Connector 18–5
Wiring Diagram 18–5
Module Operation 18–6
D4–430 Special Requirements 18–6
Channel Scanning Sequence 18–7
Output Bit Assignments 18–7
Channel Select Bits 18–8
Analog Data Bits 18–9
Output Enable Bit 18–9
Module Resolution 18–9
Writing the Control Program 18–10
Update Any Channel 18–10
Calculating the Digital Value 18–10
Engineering Unit Conversion 18–11
V-Memory Registers 18–11
Sending Data to One Channel 18–12
Sequencing the Channel Updates 18–13
Sequencing Example 1 18–14
Sequencing Example 2 18–15
Updating all Channels in a Single Scan 18–16
Analog and Digital Value Conversions 18–17

Chapter 19: F4–08DA–2 8-Channel Analog Voltage Output

Module Specifications 19–2
Analog Output Configuration Requirements 19–2
Output Specifications 19–3
General Module Specifications 19–3
Setting the Module Jumper 19–4
Connecting the Field Wiring 19–4
Wiring Guidelines 19–4
User Power Supply Requirements 19–4
Load Requirements 19–4
Removable Connector 19–5
Wiring Diagram 19–5
Module Operation 19–6
D4–430 Special Requirements 19–6

Channel Scanning Sequence 19–7
Output Bit Assignments 19–8
Channel Select Bits 19–8
Output Enable Bit 19–9
Analog Data Bits 19–9
Module Resolution 19–9

Writing the Control Program 19–10
Update Any Channel 19–10
Calculating the Digital Value 19–10
V-Memory Registers 19–11
Sending Data to One Channel 19–12
Sequencing the Channel Updates 19–13
SequencingExample 1, D4–440/450 19–14
Sequencing Example 2, D4–430 19–17
Sequencing Example 3, D4–440/450 19–18
Sequencing Example 4, D4–430 19–20
Updating all Channels in a Single Scan, D4–440/450 19–22
Updating a Single Channel During a Scan, D4–440/450 19–24
Analog and Digital Value Conversions 19–24

Chapter 20: F4–16DA–2 16-Channel Analog Voltage Output

Module Specifications 20–2
Analog Output Configuration Requirements 20–2
Output Specifications 20–3
General Module Specifications 20–3

Setting the Module Jumpers 20–4

Connecting the Field Wiring 20–4
Wiring Guidelines 20–4
User Power Supply Requirements 20–4
Load Requirements 20–5
Removable Connector 20–5
Wiring Diagram 20–5

Module Operation 20–6
D4–430 Special Requirements 20–6
Channel Scanning Sequence 20–7
Output Bit Assignments 20–8
Channel Select Bits 20–9
Output Enable Bits 20–9
Analog Data Bits 20–10
Module Resolution 20–10

Writing the Control Program 20–11
Update Any Channel 20–11
Calculating the Digital Value 20–11
V-Memory Registers 20–12
Sending Data to One Channel, D4–440/450 20–13
Sequencing the Channel Updates 20–14
Sequencing Example 1, D4–440/450 20–15
Sequencing Example 2, D4–430 20–18

Sequencing Example 3, D4–440/450 20–19
Sequencing Example 4, D4–430 20–22
Updating All Channels in a Single Scan, D4–440/450 20–25
Updating a Pair of Channels in a Single Scan, D4–440/450 20–27
Analog and Digital Value Conversions 20–28

Appendix A: DL405 Discrete I/O Memory Map

X Input / Y Output Bit Map A–2
Control Relay Bit Map A–4
Remote I/O Bit Map A–8
Stage Control / Status Bit Map A–11

Manual Revisions

If you contact us in reference to this manual, be sure to include the edition or revision number.

Title: DL405 Analog I/O Manual
Manual Number: D4–ANLG–M

Edition/Rev	Date	Description of Changes
Original	1/94	original issue
2nd Edition	7/95	Added Chapters to cover new modules
3rd Edition	12/97	Added Chapters to cover new modules
4th Edition	5/99	Added Chapters to cover new modules
4th Edition, Rev A	7/99	minor corrections
4th Edition, Rev B	9/99	Added Chapters to cover new modules
5th Edition	5/00	Added Chapters to cover new modules
5th Edition, Rev. A	7/04	Made several corrections

1

Getting Started

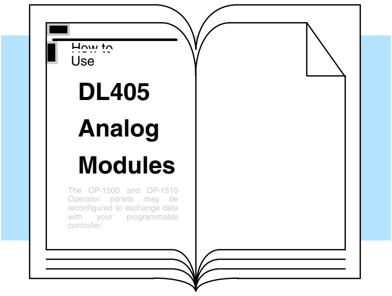
In This Chapter. . . .

— Introduction
— Conventions Used
— Physical Characteristics
— Analog Input Module Terminology
— Analog Output Module Terminology
— Selecting the Appropriate Module
— Frequently Asked Questions
— Analog Made Easy - Four Simple Steps

Introduction

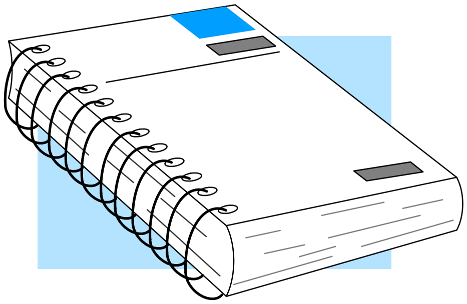
The Purpose of this Manual

This manual will show you how to select and install analog input and analog output modules. It also shows several ways to use the analog data in your PLC program. If you understand the DL405 instruction set and system setup requirements, this manual will provide the information you need to install and use the analog modules. This manual *is not* intended to be a tutorial on analog signal theory, but rather a user reference manual for the DL405 Analog I/O modules.


How to
Use
DL405
Analog
Modules
The OP-1500 and OP-1510 Operator panels may be reconfigured to exchange data with your programmable controller.

Supplemental Manuals

You may also want to have a copy of the DL405 User Manual (D4–USER–M) at hand when you are working with the analog modules. The DL405 User Manual is not absolutely necessary, but it does provide detailed descriptions of the instructions used to acquire the analog data. The User Manual also provides a more thorough description of how the I/O points are assigned to the module. Now, you have the material necessary to quickly understand the DL405 Analog I/O modules. So, let's get started!



Technical Support

We realize that even though we strive to be the best, we may have arranged our information in such a way you cannot find what you are looking for. First, check these resources for help in locating the information:

- **Table of Contents** – chapter and section listing of contents, in the front of this manual
- **Quick Guide to Contents** – chapter summary listing on the next page
- **Appendices** – reference material for key topics, near the end of this manual

You can also check our online resources for the latest product support information:

- **Internet** – Our Web site is http://www.automationdirect.com

If you still need assistance, please call us at 770–844–4200. Our technical support group is glad to work with you in answering your questions. They are available Monday through Friday from 9:00 A.M. to 6:00 P.M. Eastern Standard Time. If you have a comment or question about any of our products, services, or manuals, please fill out and return the 'Suggestions' card that was shipped with this manual.

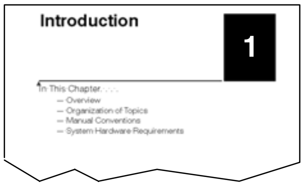
Conventions Used

When you see the “notepad” icon in the left-hand margin, the paragraph to its immediate right will be a **special note**.
The word **NOTE:** in boldface will mark the beginning of the text.

When you see the “exclamation mark” icon in the left-hand margin, the paragraph to its immediate right will be a **warning**. This information could prevent injury, loss of property, or even death (in extreme cases).
The word **WARNING:** in boldface will mark the beginning of the text.

Key Topics for Each Chapter

The beginning of each chapter will list the key topics that can be found in that chapter.



Introduction

1

In This Chapter. . . .
— Overview
— Organization of Topics
— Manual Conventions
— System Hardware Requirements

Analog Input Module Terminology

We use several different terms throughout the rest of this manual. You do not have to be an expert on analog terms to use the products, but it may help make it easier to select the appropriate modules if you take a few minutes to review these definitions.

Channels per Module	The total number of analog signals the module receives from field devices.
Input Ranges	The minimum to maximum spans in voltage or current the module will successfully convert to digital values.
Resolution	The number of binary weighted bits available on the digital side of the module for use in converting the analog value to a digital value.
Input Type	Specifies if the module accepts single ended, or differential input signals.
Input Impedance	The resistive load of the module as seen by a voltage or current input signal.
Conversion Method	The method the module uses to convert the analog signal to a digital value.
PLC Update Rate	Speed at which the analog signals are digitized and acknowledged in the PLC.
Linearity Error	The relative accuracy of the digital representation over the entire input range.
Maximum Inaccuracy	Maximum absolute error of the digital representation of the signal over the entire input range. Factors which contribute to maximum inaccuracy are also specified separately. These factors are full-scale calibration error, offset calibration error, and accuracy vs. temperature.
Accuracy vs. Temperature	The variations in the module's conversion accuracy with temperature over the module's operating temperature range.
I/O Points Required	The number of I/O points the CPU must dedicate to the module.
External Power Source	Some modules require a separate 24VDC power source. The 24VDC output supply at the local base can be used as long as you do not exceed the current rating of 400mA.
Base Power Required	The amount of base current required by the module. Use this value in your power budget calculations.
Operating Temperature	The minimum and maximum temperatures the module will operate within.
Relative Humidity	The minimum and maximum humidity the module will operate within.
Step Response	The time required for an analog input to reach 95% of its final value at the converter following a step change in the input signal level.

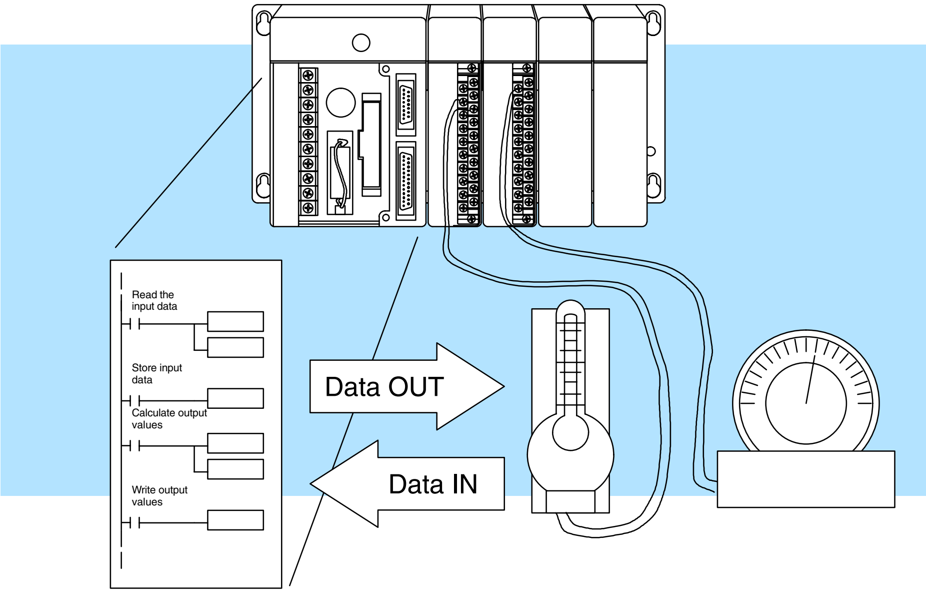
Analog Output Module Terminology

Channels per Module	The total number of analog signals the module sends to field devices.
Output Ranges	The minimum to maximum spans in voltage or current the module outputs, converted from digital values.
Resolution	The number of binary weighted bits available on the digital side of the module for use in converting the digital value to an analog signal.
Output Current	The maximum current the module will drive using a voltage output signal.
Output Impedance	The output impedance of the module using a voltage output signal.
Load Impedance	The minimum and maximum resistance the module can drive, specified for current and voltage output signals.
PLC Update Rate	The speed at which digital values in the PLC are converted to analog output signals.
Linearity Error	The relative accuracy of the digital representation over the entire output range.
Maximum Inaccuracy	Maximum absolute error of the digital representation of the signal over the entire output range. Factors which contribute to maximum inaccuracy are also specified separately. These factors are full-scale calibration error, offset calibration error, and accuracy vs temperature.
Accuracy vs. Temperature	The variations in the module's conversion accuracy with temperature over the module's operating temperature range.
External Power Source	Some output modules contain circuitry which is optically isolated from PLC-side logic. That circuitry requires field-side power from a separate 24VDC power source. The 24VDC output supply at the local base can be used as long as you do not exceed the current rating of 400mA.
Base Power Required	The amount of base current required by the module. Use this value in your power budget calculations.
Operating Temperature	The minimum and maximum temperatures the module will operate within.
Relative Humidity	The range of air humidity over which the module will operate properly.
I/O Points Required	The number of I/O points the CPU must dedicate to the module.

Selecting the Appropriate Module

Wide Variety of Modules

There are a wide variety of Analog I/O modules available for use with the DL405 family of automation products. These modules are well suited for monitoring and controlling various types of analog signals such as pressure, temperature, and so forth. No complex programming or module setup software is required. Simply install the module, add a few lines to your RLL program, and you're ready!



Analog input, temperature input and analog output modules are available. Most of these modules are designed and manufactured by FACTS Engineering. FACTS has been producing feature-packed products for the ***Direct***LOGIC families (and compatible products) for years! These modules are readily identifiable by their F4– prefix in the part number.

Diagnostic Features

The DL405 Analog Modules use an on-board microcontroller that automatically monitors module diagnostics. You can easily detect missing field-side supply 24 VDC voltage or a loose terminal block.

The following tables provide a condensed version of the information you need to select the appropriate module. The most important thing is to simply determine the number of channels required and the signal ranges that must be supported. Once you've determined these parameters, look in the specific chapter for the selected module to determine the installation and operation requirements.

Analog Input Modules

Specification	F4–04AD	F4–04ADS	F4–08AD
Channels	4	4	8
Input Ranges	0–20 mA, 4–20 mA, 1–5V, 0–5V, 0–10V, ±5V, ±10V	0–20 mA, 4–20 mA, 1–5V, 0–5V, 0–10V, ±5V, ±10V	0–20 mA, 4–20 mA, 1–5V, 0–5V, 0–10V, ±5V, ±10V
Resolution	12 bit (1 in 4096)	12 bit (1 in 4096)	12 bit (1 in 4096)
Input Type	Single ended	Isolated	Single ended
Maximum Inaccuracy	±0.4% at 25°C (77°F) ±0.55% at 0° to 60° C (32° to 140°F)	±0.4% at 25°C (77°F) ±0.7% at 0° to 60°C (32° to 140°F)	±0.3% at 25°C (77°F) ±0.5% at 0° to 60°C (32° to 140°F)
See Chapter	3	4	5
Specification	**F4–04AD**	**F4–04ADS**	**F4–08AD**
Channels	4	4	8
Input Ranges	0–20 mA, 4–20 mA, 1–5V, 0–5V, 0–10V, ±5V, ±10V	0–20 mA, 4–20 mA, 1–5V, 0–5V, 0–10V, ±5V, ±10V	0–20 mA, 4–20 mA, 1–5V, 0–5V, 0–10V, ±5V, ±10V
Resolution	12 bit (1 in 4096)	12 bit (1 in 4096)	12 bit (1 in 4096)
Input Type	Single ended	Isolated	Single ended
Maximum Inaccuracy	±0.4% at 25°C (77°F) ±0.55% at 0° to 60° C (32° to 140°F)	±0.4% at 25°C (77°F) ±0.7% at 0° to 60°C (32° to 140°F)	±0.3% at 25°C (77°F) ±0.5% at 0° to 60°C (32° to 140°F)
See Chapter	3	4	5
Specification	**F4–16AD–1**	**F4–16AD–2**	
Channels	16	4	
Input Ranges	0–20 mA, 4–20 mA	0–5V, 0–10V	
Resolution	12 bit (1 in 4096)	12 bit (1 in 4096)	
Input Type	Single ended	Single ended	
Maximum Inaccuracy	±0.4% at 25°C (77°F) ±0.55% at 0° to 60° C (32° to 140°F)	±0.4% at 25°C (77°F) ±0.7% at 0° to 60°C (32° to 140°F)	
See Chapter	6	7	

Special Input Modules

Specification	F4–08THM–n	F4–08RTD
Channels	8	8
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: 0/2320°C (–32/4208°F) Type B: 141/1820°C (286/3594°F) Type P: –99/1395°C (–146/2543°F) –1: 0 to 50mV –2: 0 to 100mV –3: 0 to 25mV	Pt100Ω: –200.0/850.0°C (–328/ 562°F) Pt1000Ω: –200.0/595.0°C (–328/1103°F) jPt100Ω: –38.0/450.0°C (–36/842°F) Cu. 25Ω, Cu. 10Ω: –200.0/260.0°C (–328/500°F)
Resolution	12 bit (1 in 4096)	15 bit (1 in 32768)
Maximum Inaccuracy	± 1° C type J,K,E,T thermocouples ± 3° C type R,S,B,C,P thermocouples	± 0.2% at 25°C (77°F)
See Chapter	8	9

Specification	F4–08THM
Channels	8
Input Ranges	Type J –190 to 760°C –310 to 1400°F Type E –210 to 1000°C –346 to1832°F Type K –150 to 1372°C –238 to 2502°F Type R 65 to 1768°C 149 to 3214°F Type S 65 to 1768°C 149 to 3214°F Type T –230 to 400°C –382 to 752°F Type B 529 to 1820°C 984 to 3308°F Type N –70 to 1300°C –94 to 2372°F Type C 65 to 2320°C 149 to 4208°F
Resolution	16 bit (1 in 65535)
Maximum Inaccuracy	± 3° C (excluding thermocouple error)
See Chapter	10

Analog Output Modules

Specification	D4–02DA	F4–04DA	F4–04DA–1
Channels	2	4	4
Output Ranges	4–20 mA, 1–5V, 0–10V	4–20 mA, 0–5V, 0–10V, ±5V, ±10V	4–20mA
Resolution	12 bit (1 in 4096)	12 bit (1 in 4096)	12 bit (1 in 4096)
Output Type	Independent	Single ended	Single ended
Maximum Inaccuracy	±0.2% at 25°C (77°F)	±0.5% at 60°C (unipo.) ±0.7% at 60°C (bipol.) ±0.8% at 60°C (curr.)	±0.1 % at 25°C (77°F) ±0.3 % at 0 to 60°C (32 to 140°F)
See Chapter	11	12	13

Specification	F4–04DA–2	F4–08DA–1	F4-16DA-1
Channels	4	8	16
Output Ranges	0–5V, 0–10V, ±5V, ±10V	4–20mA	4–20 mA
Resolution	12 bit (1 in 4096)	12 bit (1 in 4096)	12 bit (1 in 4096)
Output Type	Single ended	Single ended	Single ended
Maximum Inaccuracy	±0.2 % at 25°C (77°F) ± 0.4% at 0° to 60°C (32° to 140°F)	±0.2% at 25°C (77°F) ±0.4% at 0° to 60°C (32° to 140°F)	±0.2 % at 25°C (77°F) ±0.4 % at 0° to 60°C (32° to 140°F)
See Chapter	14	15	16

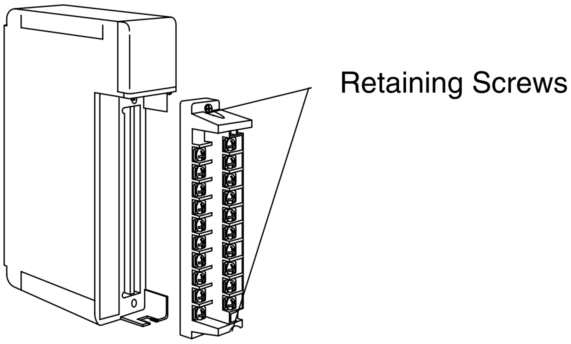
Specification	F4–04DAS–1	F4–08DA–2	F4-16DA-2
Channels	4, current sourcing	8	16
Output Ranges	4–20mA	0–5V, 0–10V	0–5V, 0–10V, combination of both
Resolution	16 bit (1 in 65536)	12 bit (1 in 4096)	12 bit (1 in 4096)
Output Type	Isolated	Single ended	Single ended
Maximum Inaccuracy	±0.07 % at 25° (77°F) ± 0.18% at 0° to 60°C 32° to 140°F)	±0.2 % at 25° (77°F) ± 0.4% at 0° to 60°C 32° to 140°F)	±0.2 % at 25° (77°F) ± 0.4% at 0° to 60°C 32° to 140°F)
See Chapter	17	18	19

Specification	F4–04DAS–2		
Channels	4		
Output Ranges	0–5V, 0–10V		
Resolution	16 bit (1 in 65536)		
Output Type	Isolated		
Maximum Inaccuracy	±0.07 % at 25° (77°F) ± 0.18% at 0° to 60°C 32° to 140°F)		
See Chapter	20		

Physical Characteristics

The DL405 Analog Modules provide many features that make the modules easy to use. With the exception of the Thermocouple module, the terminal blocks are removable, which makes wiring a simple task. All of the DL405 analog modules have normal screw terminal connectors. For ease of removal, there are captive screws on the top and bottom. To remove the terminal block, unscrew the screws at the ends to the terminal block and pull the terminal block away from the module.

WARNING: For some modules, field device power may still be present on the terminal block even though the PLC system is turned off. To minimize the risk of electrical shock, check all field device power *before* you remove the connector.



Frequently Asked Questions

Q. How many discrete I/O points does my analog module occupy?

A. Analog modules will occupy either or both X and Y points. You must pay particular attention to this information because it varies depending on which module is being used. Care must be taken to determine the proper placement within the base (see the specific analog module specifications to determine the proper placement).

Q. Does my module detect a broken transmitter or loose terminal block?

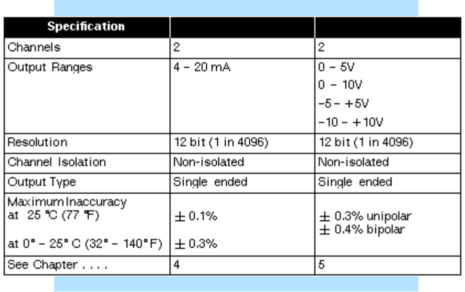
A. Yes, most of the analog modules have this function. The loose terminal block is indicated by the error code E201. See error code information in the DL405 User Manual for further information.

Q. Can I use a FOR/NEXT loop program to read all channels in one scan in a remote/slave arrangement?

A. No. A FOR/NEXT loop program will not work in a remote/slave arrangement. Use a program that reads one channel per scan. Remember, FOR/NEXT loops can only be used with DL440 and DL450 CPUs.

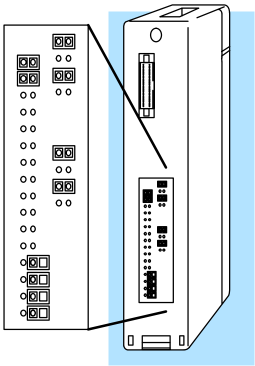
Analog Made Easy – Four Simple Steps

Once you've selected the appropriate module, use the chapter that describes the module and complete the following steps.



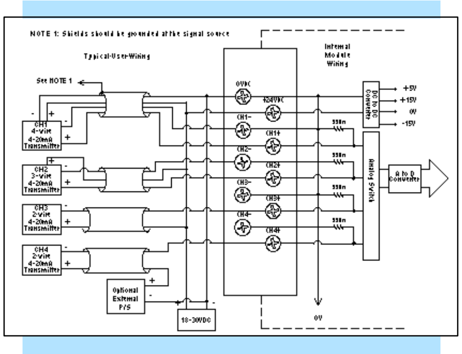
Specification		
Channels	2	2
Output Ranges	4 – 20 mA	0 – 5V 0 – 10V –5 – +5V –10 – +10V
Resolution	12 bit (1 in 4096)	12 bit (1 in 4096)
Channel Isolation	Non-isolated	Non-isolated
Output Type	Single ended	Single ended
Maximum Inaccuracy at 25 °C (77 °F)	± 0.1%	± 0.3% unipolar ± 0.4% bipolar
at 0° – 25° C (32° – 140° F)	± 0.3%	
See Chapter	4	5

Step 1 . Take a minute to review the detailed specifications to make sure the module meets your application requirements.

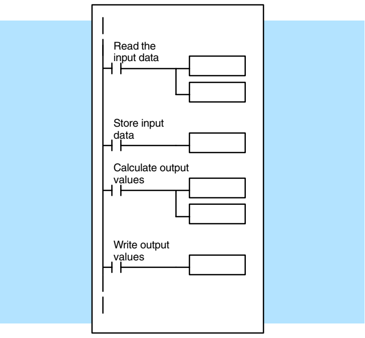


Step 2 . Set the module switches and/or jumpers to select:

- number of channels
- the operating ranges


NOTE 1: Shields should be grounded at the signal source
Typical User Wiring
See NOTE 1
Internal Module Wiring
CH1 4-wire 4-20mA Transmitter
CH2 3-wire 4-20mA Transmitter
CH3 2-wire 4-20mA Transmitter
CH4 2-wire 4-20mA Transmitter
Optional External P/S
18-30VDC
0VDC
+24VDC
CH1–
CH1+
CH2–
CH2+
CH3–
CH3+
CH4–
CH4+
DC to DC Converter
+5V
+15V
0V
-15V
250Ω
Analog Switch
A to D Converter
0V

Step 3 . Connect the field wiring to the module connector.


Read the input data
Store input data
Calculate output values
Write output values

Step 4 . Review the module operating characteristics and write the control program.

2

Installation and Wiring

In This Chapter. . . .

— Safety Guidelines
— I/O Wiring Strategies
— I/O Module Placement and Wiring

Safety Guidelines

WARNING: Providing a safe operating environment for personnel and equipment is your responsibility and should be your primary goal during system planning and installation. Automation systems can fail and may result in situations that can cause serious injury to personnel or damage to equipment. Do not rely on the automation system alone to provide a safe operating environment. You should use external electromechanical devices, such as relays or limit switches, that are independent of the PLC system to provide protection for any part of the system that may cause personal injury or damage.

Every automation application is different, so there may be special requirements for your particular application. Make sure you follow all National, State, and local government requirements for the proper installation and use of your equipment.

Plan for Safety

The best way to provide a safe operating environment is to make personnel and equipment safety part of the planning process. You should examine *every* aspect of the system to determine which areas are critical to operator or machine safety.

If you are not familiar with PLC system installation practices, or your company does not have established installation guidelines, you should obtain additional information from the following sources.

- NEMA — The National Electrical Manufacturers Association, located in Washington, D.C., publishes many different documents that discuss standards for industrial control systems. You can order these publications directly from NEMA. Some of these include:
  ICS 1, General Standards for Industrial Control and Systems
  ICS 3, Industrial Systems
  ICS 6, Enclosures for Industrial Control Systems
- NEC — The National Electrical Code provides regulations concerning the installation and use of various types of electrical equipment. Copies of the NEC Handbook can often be obtained from your local electrical equipment distributor or your local library.
- Local and State Agencies — many local governments and state governments have additional requirements above and beyond those described in the NEC Handbook. Check with your local Electrical Inspector or Fire Marshall office for information.

Safety Techniques

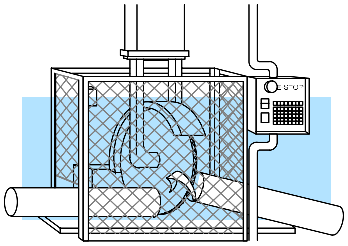
The publications mentioned provide many ideas and requirements for system safety. We recommend following these regulations as a minimum. Using the techniques listed below will further help reduce the risk of safety problems.

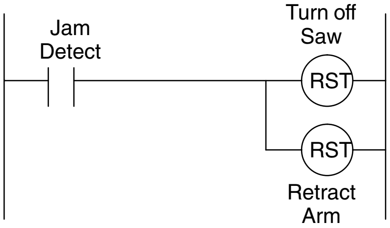
- Orderly system shutdown sequence in the PLC control program.
- System power disconnects (guard limits, emergency stop switches, etc.).

Orderly System Shutdown

The PLC control program can provide the first level of protection when used to identify machine problems. After analyzing your application, identify any shutdown sequences that must be performed. Typical problems include part jams, empty bins, etc. that do not pose a risk of personal injury or equipment damage.

WARNING: The control program *must not* be the only form of protection for any problems that may result in a risk of personal injury or equipment damage.



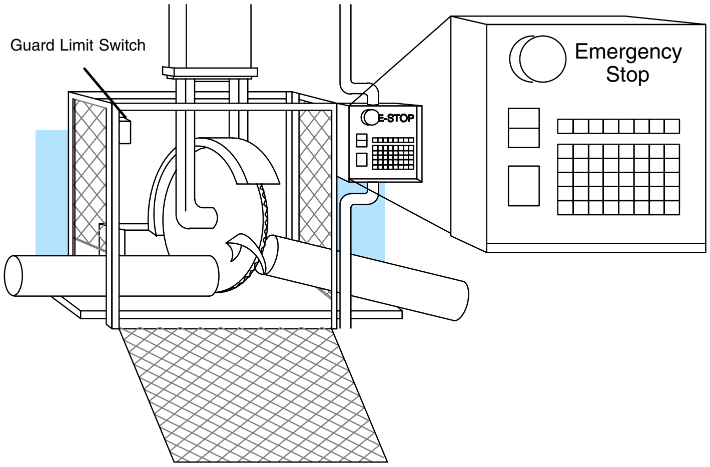

Jam Detect
Turn off Saw
RST
RST
Retract Arm

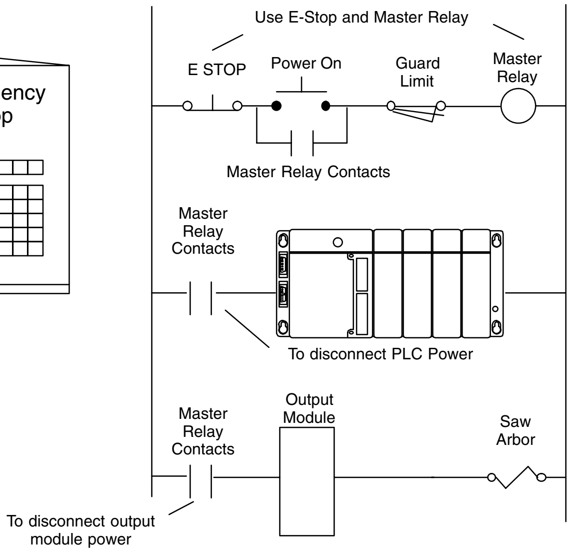
System Power Disconnect

By using electro-mechanical devices such as master control relays and/or limit switches, you can prevent accidental equipment startup. When installed properly, these devices will prevent *any* machine operation from occurring.

For example, if the machine has a jammed part the PLC control program can turn off the saw blade and retract the arbor. However, since the operator must open the guard to remove the part, you must include a disconnect switch that removes *all* system power any time the guard is opened.

The machine operator must also have a quick and easy method of manually disconnecting *all* system power. This is accomplished with a mechanical device that is clearly labeled as an **Emergency Stop** switch.


Guard Limit Switch
E-STOP
Emergency Stop


Use E-Stop and Master Relay
E STOP
Power On
Guard Limit
Master Relay
Master Relay Contacts
Master Relay Contacts
To disconnect PLC Power
Master Relay Contacts
Output Module
Saw Arbor
To disconnect output module power

After an Emergency shutdown or any other type of power interruption, there may be requirements that must be met before the PLC control program can be restarted. For example, there may be specific register values that must be established (or maintained from the state prior to the shutdown) before operations can resume. In this case, you may want to use retentive memory locations, or include constants in the control program to ensure a known starting point.

I/O Wiring Strategies

The DL405 PLC system is very flexible and will work in many different wiring configurations. By studying this section before actual installation, you can probably find the best wiring strategy for your application. This will help to lower system cost, reduce wiring errors, and avoid safety problems.

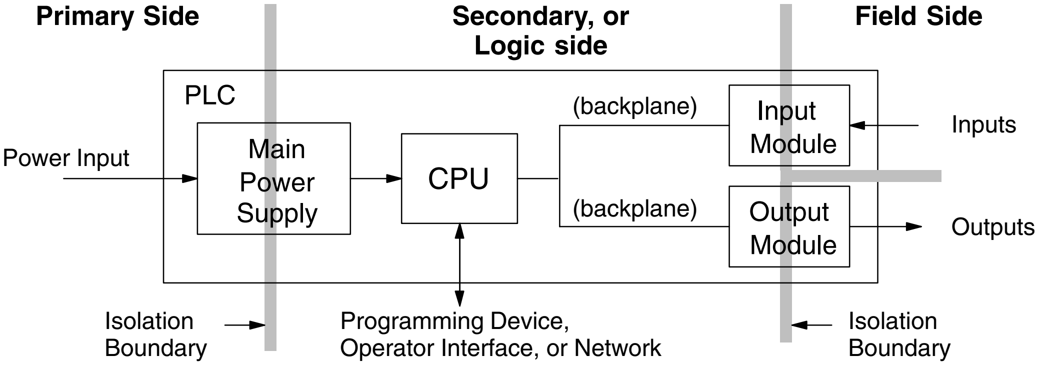
Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin installation. Here are some general things to consider.

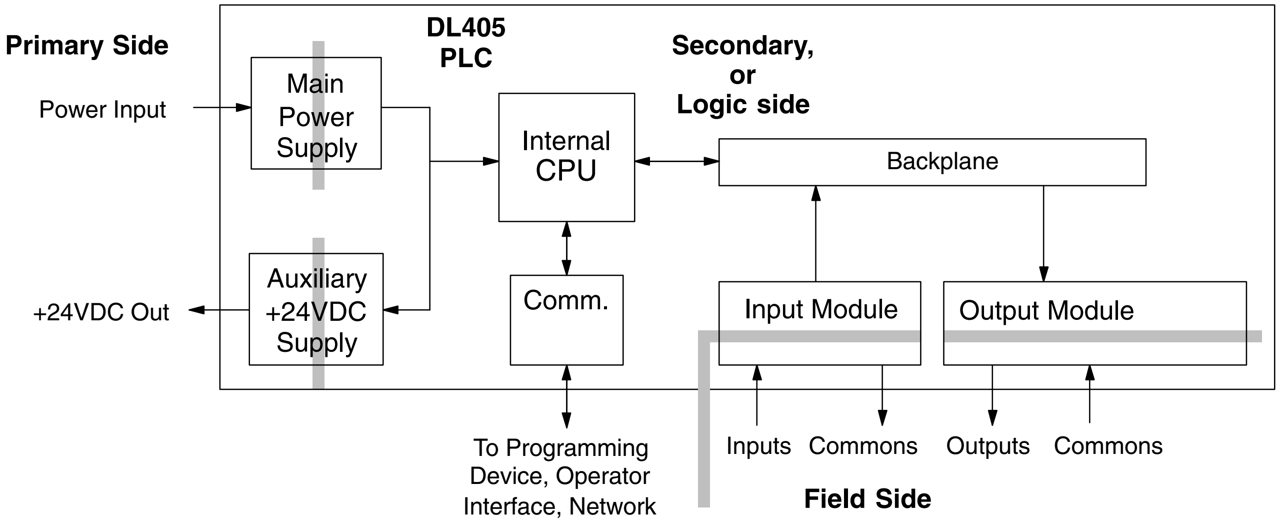
NOTE: See the DL405 User Manual for more detailed wiring information.

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route 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.
- When using thermocouple wire always use the same type of composition for thermocouple extensions. For example, do not use copper wire and expect the thermocouple to function correctly.
- When using RTDs, three-wire RTDs are preferred. The three wires connecting the RTD to the module must be the same type and length. Do not use the shield or drain wire for one of the connections.
- There are many different types of analog modules and there are slight wiring differences between modules. Please refer to your specific module type for the recommended sensors/sources wiring.

PLC circuitry is divided into three main regions separated by isolation boundaries, shown in the drawing below. Electrical isolation provides safety, so that a fault in one area does not damage another area.

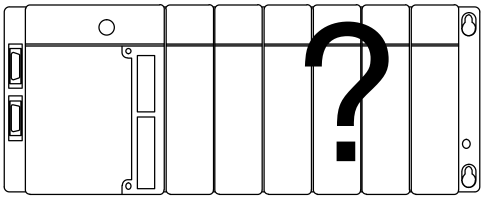

Primary Side
Secondary, or Logic side
Field Side
PLC
Power Input
Main Power Supply
CPU
(backplane)
(backplane)
Input Module
Output Module
Inputs
Outputs
Isolation Boundary
Programming Device, Operator Interface, or Network
Isolation Boundary

The next figure shows the physical layout of a DL405 PLC system as viewed from the front. In addition to the basic circuits, AC-powered CPUs include an auxiliary +24VDC power supply with its own isolation boundary. Since the supply output is isolated from the other three circuits, it can power input and/or output circuits.


Primary Side
Power Input
+24VDC Out
DL405 PLC
Main Power Supply
Auxiliary +24VDC Supply
Internal CPU
Comm.
Secondary, or Logic side
Backplane
Input Module
Output Module
To Programming Device, Operator Interface, Network
Inputs
Commons
Outputs
Commons
Field Side

I/O Module Placement and Wiring

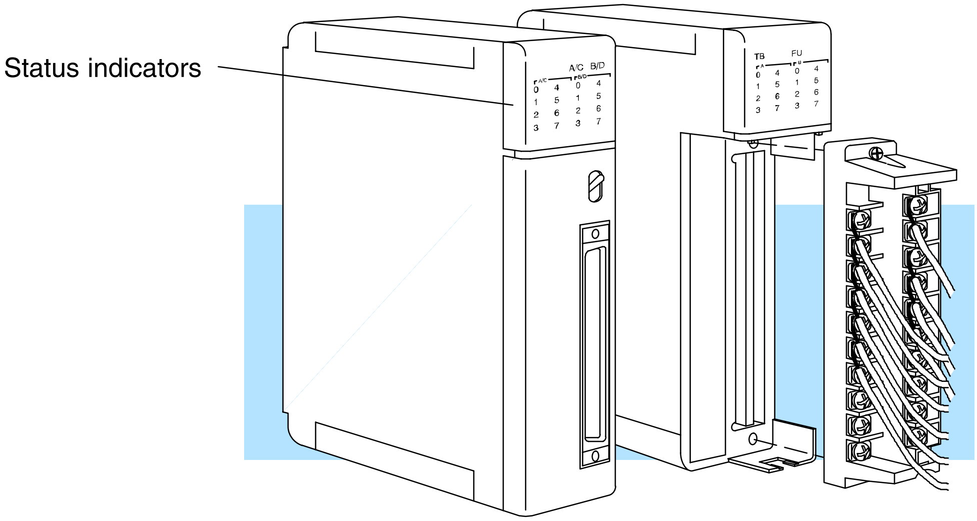
Module Placement Before wiring the I/O modules in your system to field devices, it's very important to make sure each I/O module is in the right slot and base in the system. Costly wiring errors may be avoided by doing the following:



- Do the power budget calculations for each base to verify the base power supply can power all the modules in the base. Information on how to do this is in the DL405 User Manual.
- Some specialty I/O modules may only be installed in particular slots or they will not function properly. Check the corresponding manuals before installation and wiring.
- Whenever possible, keep modules with high voltage and current wiring away from sensitive analog modules.

I/O Module Status Indicators

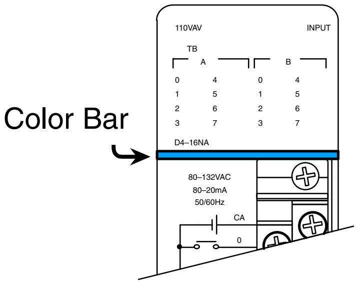
The diagram below shows the status indicator location for common I/O modules. The actual status indicators vary from one module to another, and some modules do not have status indicators.


Status indicators
A/C B/D
0 4 0 4
1 5 1 5
2 6 2 6
3 7 3 7
TB FU
0 4 0 4
1 5 1 5
2 6 2 6
3 7 3 7

Color Coding of I/O Modules

The DL405 family of I/O modules have a color-coded stripe on the front bezel to help identify whether the module type is input, output, or special module. The color code meaning is listed below:

Module Type	Color Code
Discrete/Analog Output	Red
Discrete/Analog Input	Blue
Other	White

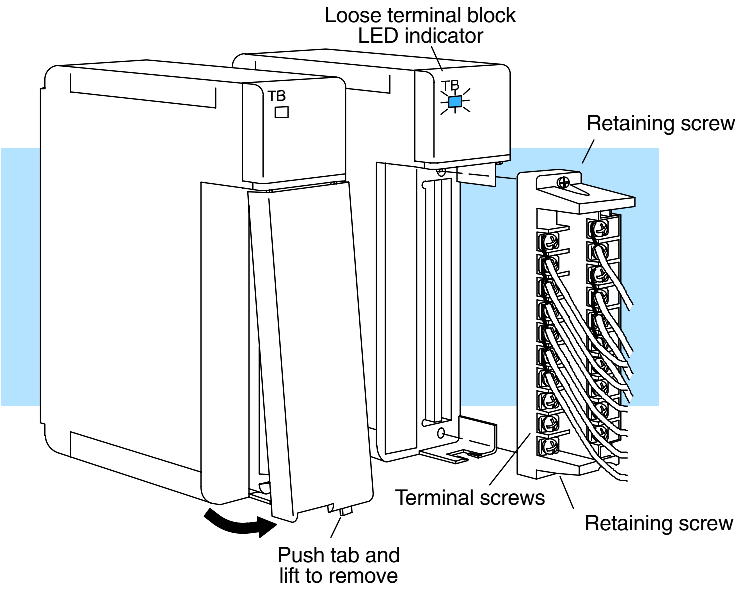

Color Bar
110VAV
INPUT
TB
A
B
0 4 0 4
1 5 1 5
2 6 2 6
3 7 3 7
D4–16NA
80–132VAC
80–20mA
50/60Hz
CA
0

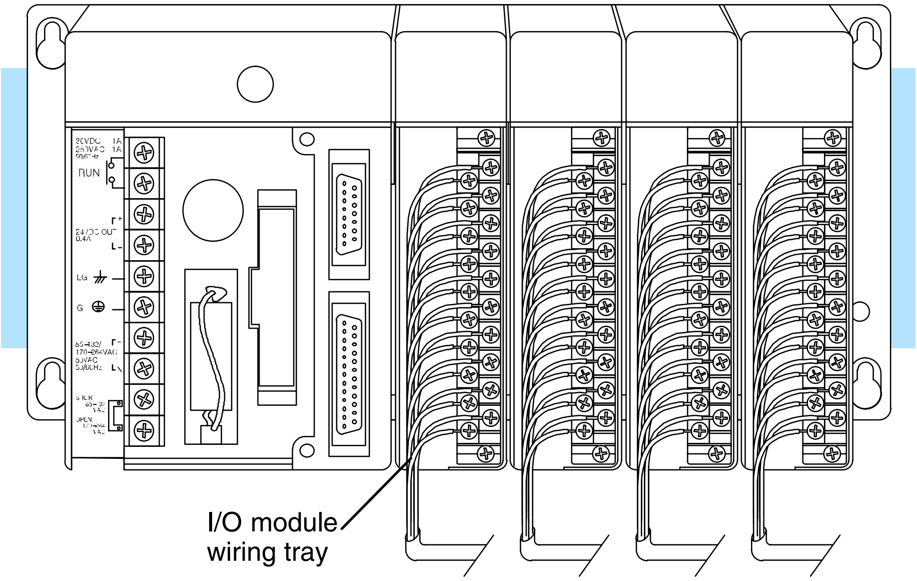
Wiring a Module with a Terminal Block

You must first remove the front cover of the module prior to wiring. To remove the cover press the bottom tab of the cover and tilt the cover up to loosen from the module.

All DL405 I/O module terminal blocks are removable for your convenience. To remove the terminal block loosen the retaining screws and and lift the terminal block away from the module. When you return the terminal block to the module make sure the terminal block is tightly seated. Be sure to tighten the retaining screws. If your module has a loose-terminal-block-LED (TB), you should also verify that it is off when system power is applied.

WARNING: For some modules, field device power may still be present on the terminal block even though the PLC system is turned off. To minimize the risk of electrical shock, check all field device power *before* you remove the connector.


Loose terminal block LED indicator
TB
TB
Retaining screw
Terminal screws
Retaining screw
Push tab and lift to remove


RUN
LG
G
I/O module wiring tray

F4–04AD
4-Channel Analog Input

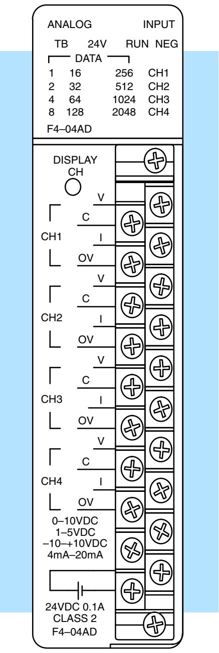
In This Chapter. . . .

— Module Specifications
— Setting the Module Jumpers
— Connecting the Field Wiring
— Module Operation
— Writing the Control Program, 16 Input Mode
— Writing the Control Program, 32 Input Mode
— Scaling and Converting the Input Data
— Configuration Cross-Reference, D4-04AD to F4-04AD

Module Specifications

The F4-04AD Analog Input module provides several features and benefits.

- It is a direct replacement for the popular D4-04AD module, when properly configured.
- It accepts four differential voltage or current inputs.
- Analog inputs are optically isolated from PLC logic components.
- The module has a removable terminal block, so the module can be easily removed or changed without disconnecting the wiring.
- All four analog inputs may be read in one CPU scan (DL440/450 CPUs only).
- Broken transmitter detection is provided for current inputs.


ANALOG
INPUT
TB
24V
RUN
NEG
DATA
1 16 256 CH1
2 32 512 CH2
4 64 1024 CH3
8 128 2048 CH4
F4–04AD
DISPLAY CH
CH1
CH2
CH3
CH4
V
C
I
OV
0–10VDC
1–5VDC
–10–+10VDC
4mA–20mA
24VDC 0.1A
CLASS 2
F4–04AD

NOTE: If you are replacing a D4–04AD with a F4–04AD in an existing application, make sure to read the last section in this chapter, “Configuration Cross-Reference, D4–04AD to F4–04AD.”

Analog Input Configuration Requirements

The F4–04AD Analog Input module requires either 16 or 32 discrete input points, depending on its operating mode (jumper selectable). 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 are the limiting factors.
- For remote I/O systems, the available power budget and number of remote I/O points are the limiting factors.

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

The following tables provide the specifications for the F4-04AD Analog Input Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

Number of Channels	4
Input Type	Single-ended or differential
Input Ranges	0–5, 1–5, 0–10, ± 5, ± 10 VDC, 0–20, 4–20 mA.
Resolution	12 bit (0 to 4095), unipolar 13 bit (± 4095), bipolar
Input Impedance	20 MΩ minimum, voltage input 250 Ω, 1/2W, ± 0.1%, 25 ppm/° C current input
Max. Continuous Overload	± 50 VDC, voltage input, ± 45 mA, current input
Recommended External Fuse	0.032A, Series 217 fast acting, current inputs
Common Mode Voltage Range	± 10V maximum
Linearity	± 0.025% of span (± 1 count maximum, unipolar)
Input Stability	± 1/2 count
Cross Talk	–80 dB, 1/2 count maximum
Full Scale Calibration Error	± 12 counts maximum, voltage input ± 16 counts maximum, at 20.000 mA current input
Offset Calibration Error	± 1 count maximum, voltage input ± 2 counts maximum, at 4.000 mA current input
Maximum Inaccuracy	0.4% maximum @ 25° C (77° F) 0.55% maximum @ 0 to 60° C (32 to 140° F)
Conversion Time	< 6 mS per selected channel
Noise Rejection Ratio	Normal mode: –3 dB @ 50 Hz, –6 dB / octave Common mode: –70 dB, DC to 12 kHz

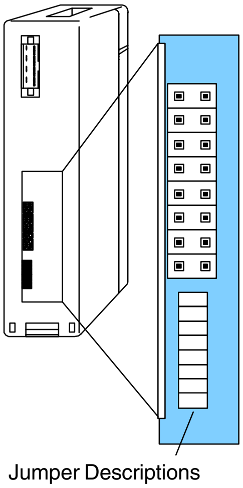
General Specifications

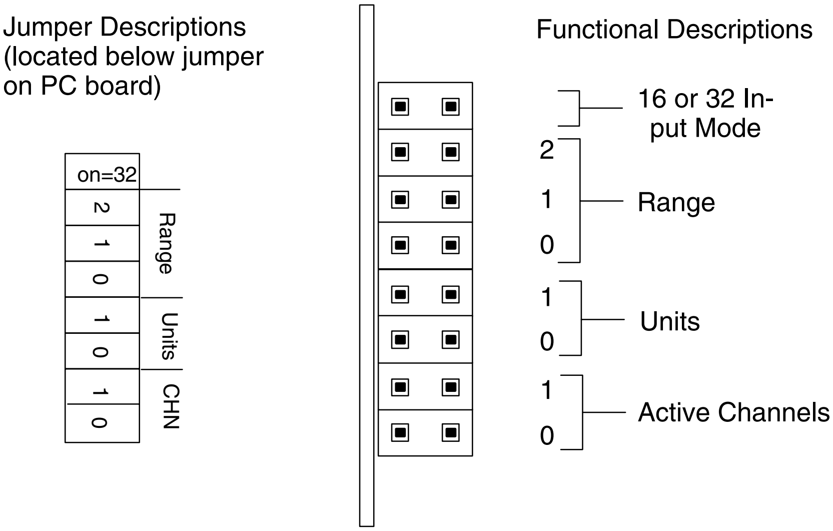
PLC Update Rate	4 channel per scan max.
Digital Input Points Required 16 or 32-bit mode	16 or 32 (X) input points 12 data bits, 4 bits optional for two's complement mode, 4 channel select bits, 12 bits unused in 32 bit mode
Power Budget Requirement	85 mA (power from base)
External Power Supply	24 VDC, ± 10%, 100 mA, class 2
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
Insulation Resistance	10 MΩ, 500 VDC
Noise Immunity	NEMA ICS3-304

Setting the Module Jumpers

Jumper Locations

The module has several options that you can select by installing or removing jumpers. At the rear of the module is a bank of eight jumpers. They may be configured to select either 16 Input Mode or 32 Input Mode operation, input range selection, units of measurement selection and the number of channels enabled.


Jumper Descriptions


Jumper Descriptions
(located below jumper
on PC board)
on=32
2
1
0
Range
1
0
Units
1
0
CHN
Functional Descriptions
16 or 32 Input Mode
2
1
0
Range
1
0
Units
1
0
Active Channels

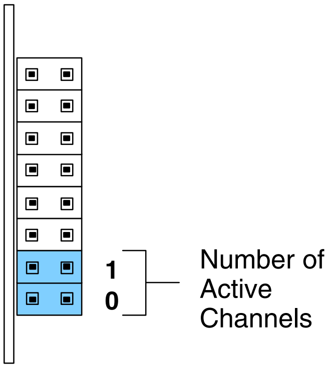
NOTE: If you are replacing a D4–04AD module with the F4–04AD in an existing application, skip to the special section at the end of this chapter, “Configuration Cross-Reference, D4–04AD to F4–04AD”.

Factory Default Settings

By default, the module arrives from the factory with all jumpers installed. With all jumpers installed, the module has four active channels, is in 32 Input Mode, has 4 to 20 mA. input range, and the units of the data are 12-bit (0 to 4095) BCD numbers.

Selecting the Number of Active Channels

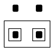
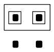
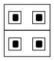
The F4-04AD module accepts from one to four analog inputs and converts the signal(s) to a desired format to send to the CPU. The bottom two jumpers (J7 and J8) select the number of channels enabled. The module only converts signals on channels that are enabled. If your application requires less than four signal inputs from this module, selecting fewer channels results in faster update times.


1
0
Number of Active Channels

Use the following table to set jumpers.

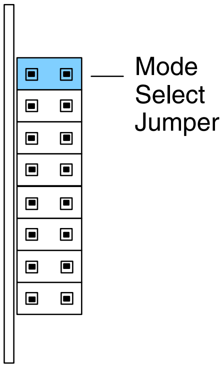
▪ ▪ = jumper removed

☐☐ = jumper installed

Channel(s) Selected	Jumper Settings
Channel 1	1 0
Channel 1, Channel 2	 1 0
Channel 1, Channel 2, Channel 3	 1 0
Channel 1, Channel 2, Channel 3, Channel 4	 1 0

Selecting 16 Input or 32 Input Modes

The top jumper selects either 16 Input (remove jumper) or 32 Input (install jumper) operating modes. This is the number of X inputs the module requires in the PLC memory map. The module can interface to the CPU in two different ways, depending on the setting of this jumper. Use 32 Input mode if you want to maintain compatibility with PLC software written for the D4-04AD, or to use features not available in 16 Input mode, and to simplify supporting ladder logic. However, use 16 Input Mode if you must consume fewer X inputs. The feature chart on the next page can help you choose the mode for your application.



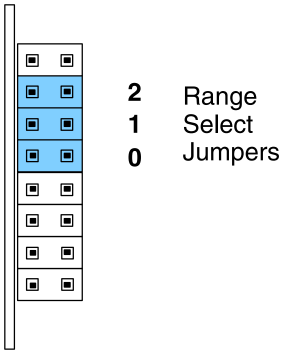
Mode	Jumper	I/O Points Consumed
16 Input	Remove	X0 – X17
32 Input	Install	X0 – X37

Features	16 Input Mode	32 Input Mode
Number of X Input Bits Required From CPU	16	32
Input Value, 12-Bit, Plus Sign Bit	Yes	Yes
Input Value, 2's Complement,12 Bits	No	Yes
Input Value, 2's Complement, 13 Bits	No	Yes
Input Value, 12-bit	Yes	Yes
Input Value, Binary-Coded Decimal, 16 bits (for bipolar voltage ranges only)	No	Yes
Active Channel Indicator Inputs	Yes	Yes
Broken Transmitter Detection	1 bit (combined)	4 bits (individual)
Sign Bit(s), indicates negative analog value	1 bit (combined)	4 bits (individual)

Based on this jumper selection, the module can behave as two different modules from the CPU point of view. This chapter covers both modes, so only the CPU program examples labeled for the mode you choose will apply.

Operating Range Selection 16 Input Mode

These three jumpers select the voltage or current range for all four input channels simultaneously. The type of input (voltage or current) is actually determined by user wiring to specific terminals on the front connector. Along with proper wiring, set these jumpers for the desired voltage or current signal range. The three jumpers are binary encoded to offer eight possible settings.



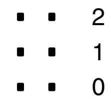
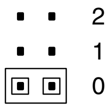
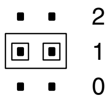
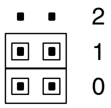
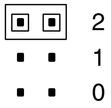
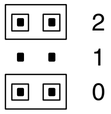
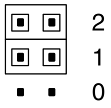
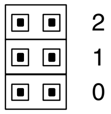
More input ranges are available for the module’s 32 Input mode than for 16 Input mode. The following tables list the ranges for each of the modes.

Input Range Selection, 16 Input Mode

Input Signal Range	Jumper Settings	Data Type and Range
(not used in 16 Input Mode)	▪ ▪ 2 ▪ ▪ 1 ▪ ▪ 0	
(not used in 16 Input Mode)	▪ ▪ 2 ▪ ▪ 1 ▣ ▣ 0	
–10 VDC to +10 VDC	▪ ▪ 2 ▣ ▣ 1 ▪ ▪ 0	12-Bit Magnitude Plus Sign Bit, (–4095 to +4095)
–5 VDC to +5 VDC	▪ ▪ 2 ▣ ▣ 1 ▣ ▣ 0	12-Bit Magnitude Plus Sign Bit, (–4095 to +4095)
0 VDC to +10 VDC	▣ ▣ 2 ▪ ▪ 1 ▪ ▪ 0	12-Bit Magnitude, (0 to 4095)
0 mA to 20 mA, or 0 VDC to +5 VDC	▣ ▣ 2 ▪ ▪ 1 ▣ ▣ 0	12-Bit Magnitude, (0 to 4095)
4 mA to 20 mA (*with* broken transmitter detection)	▣ ▣ 2 ▣ ▣ 1 ▪ ▪ 0	12-Bit Magnitude, (0 to 4095)
4 mA to 20 mA (*without* broken transmitter detection), or +1 VDC to +5 VDC	▣ ▣ 2 ▣ ▣ 1 ▣ ▣ 0	12-Bit Magnitude, (0 to 4095)

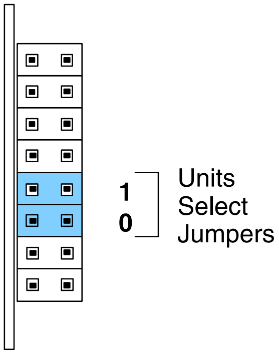
Operating Range Selection 32 Input Mode

The module's 32 Input mode provides eight possible input range and data type combinations. Two of the bipolar ranges are dedicated to BCD data type. The other six input signal ranges convert to various data types and ranges (selected by the units select jumpers).

Input Range Selection, 32 Input Mode		
Input Signal Range	**Jumper Settings**	**Data Type and Range**
–10 VDC to +10 VDC		Binary-Coded Decimal, (–9999 to +9999)
–5 VDC to +5 VDC		Binary-Coded Decimal, (–5000 to +5000)
–10 VDC to +10 VDC		Set by Units Select jumpers
–5 VDC to +5 VDC		Set by Units Select jumpers
0 VDC to +10 VDC		Set by Units Select jumpers
0 mA to 20 mA, or 0 VDC to +5 VDC		Set by Units Select jumpers
4 mA to 20 mA (*with* broken transmitter detection)		Set by Units Select jumpers
4 mA to 20 mA (*without* broken transmitter detection), or +1 VDC to +5 VDC		Set by Units Select jumpers

Units Selection for 32 Input Mode

The two jumpers for units selection determine the data format of the digital values of the channel inputs. They only apply to 32 Input mode operation, so the module ignores the position of these jumpers during 16 Input mode operation. The two jumpers are binary encoded to offer four possible settings. The units selection programmed by these jumpers applies simultaneously to all four input channels, and to all 32 Input Mode input signal ranges except the two bipolar BCD ranges. In those ranges, the module ignores the units select jumper settings.



The first two selections in the table offer more resolution than the last two selections, which are included for compatibility with previous application software. Accordingly, they are not recommended for new applications. After setting the configuration jumpers, you are ready to install the module in the base and connect the field wiring.

When you power up the module for the first time, if the jumper configuration is invalid the RUN light on the module's faceplate will NOT turn on and the Channel 1 LED will flash quickly. If this occurs, review this section and verify that the jumper settings are correct.

NOTE: If you are replacing a D4-04AD module with the F4-04AD in an existing application, skip to the special section at the end of this chapter, "Configuration Cross-Reference, D4-04AD to F4-04AD".

Units Selection for 32 Input Mode	Jumper Settings	Notes
12-Bit Magnitude Plus Sign, 13 Bit Format, –4095 to +4095	▪ ▪ 1 ▪ ▪ 0	Recommended for most applications
2's Complement, 13-Bit Format	▪ ▪ 1 ▣ ▣ 0	Recommended two's complement format
2's Complement, 12-Bit Format	▣ ▣ 1 ▪ ▪ 0	Not recommended for new applications
12-Bit Magnitude, 0 to 4095	▣ ▣ 1 ▣ ▣ 0	Not recommended for new applications

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.
- Unused inputs must be shorted to help reduce the effects of electrical noise (see the wiring diagram for an example).

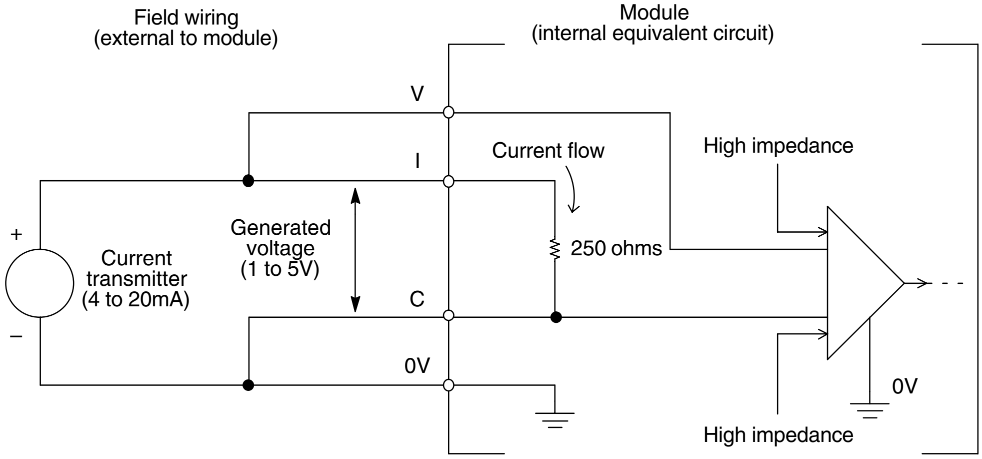
User Power Supply Requirements

The F4-04AD requires a separate power supply for the isolated (field) side of the module. The Series DL405 CPUs, D4-RS Remote I/O Controller, and D4-EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. If you only have a couple of analog modules, you can use this power source instead of a separate supply. If you have more than four analog modules, or you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ± 10%, Class 2, 100 mA current (per module).

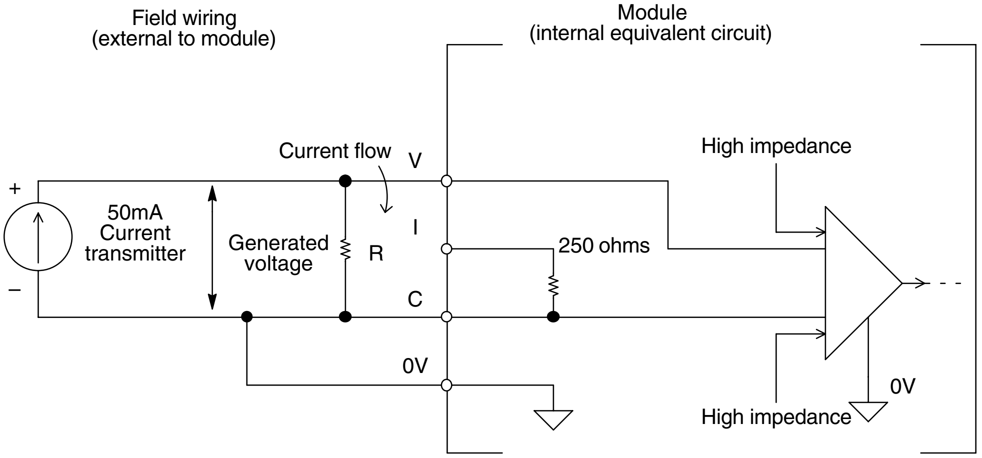
Using Current or Voltage Wiring

Even though you cannot select different ranges or units for each channel, you can still wire each individual channel for voltage or current signals. For example, even though you select a 1 to 5V range with the jumpers, you can still use a transmitter that provides a 4-20 mA signal.

The module uses a 250 ohm precision resistor to convert the current signals to voltage for you (4mA x 250 ohms = 1V, 20mA x 250 ohms = 5V). The following diagram shows how this works. Notice that the voltage (V) and (I) input terminals are connected together.



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


Field wiring
(external to module)
Module
(internal equivalent circuit)
Current flow
V
I
C
0V
+
–
50mA
Current
transmitter
Generated
voltage
R
250 ohms
High impedance
High impedance
0V

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

R = value of external resistor

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

I_{max} = maximum current supplied by the transmitter

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

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

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

Current Loop Transmitter Impedance

Standard 4 to 20 mA transmitters and transducers can operate from a wide variety of power supplies. Not all transmitters are alike and the manufacturers often specify a minimum loop or load resistance that must be used with the transmitter.

The F4-04AD provides 250 ohm resistance for each channel. If your transmitter requires a load resistance below 250 ohms, you do not have to make any adjustments. However, if your transmitter requires a load resistance higher than 250 ohms, you need to add a resistor in series with the module.

Consider the following example for a transmitter being operated from a 36 VDC supply with a recommended minimum load resistance of 750 ohms. Since the module has a 250 ohm resistor, you need to add an additional resistor.

$R = Tr - Mr$

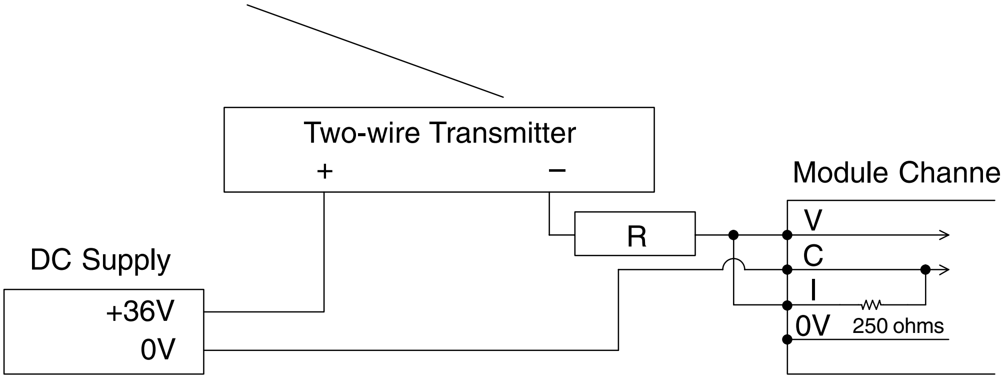
$R = 750 - 250$

$R \geq 500$

R – resistor to add

Tr – Transmitter Requirement

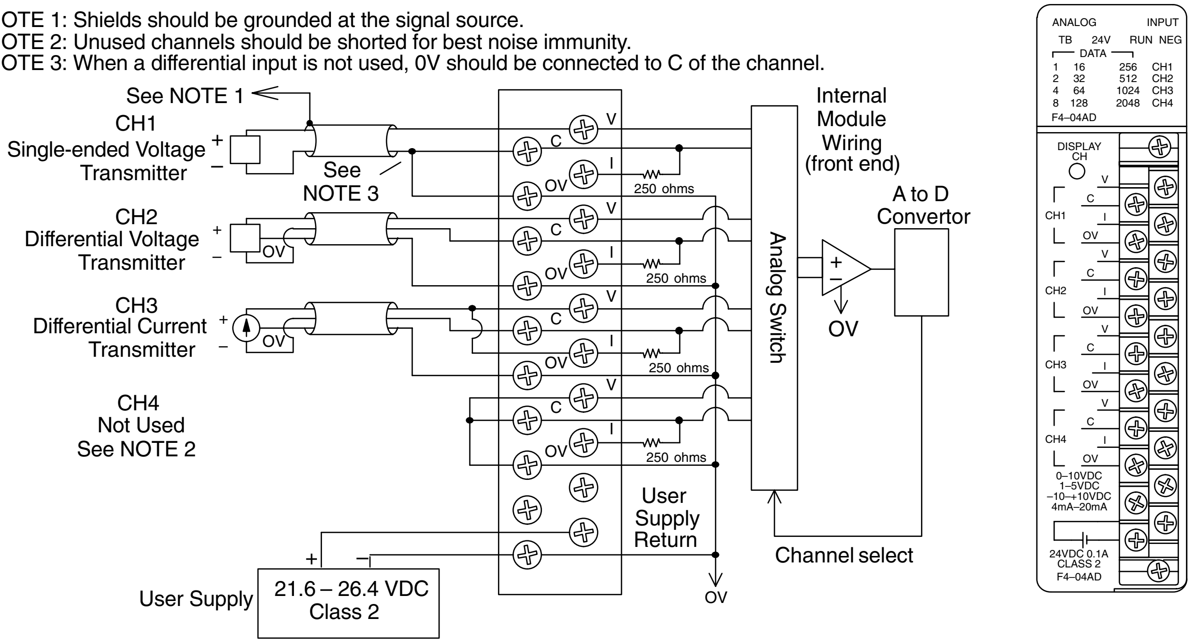
Mr – Module resistance (internal 250 ohms)


Two-wire Transmitter
+
–
DC Supply
+36V
0V
R
Module Channel 1
V
C
I
0V
250 ohms

Wiring Diagram

The F4-04AD module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

NOTE 1: Shields should be grounded at the signal source.
NOTE 2: Unused channels should be shorted for best noise immunity.
NOTE 3: When a differential input is not used, 0V should be connected to C of the channel.

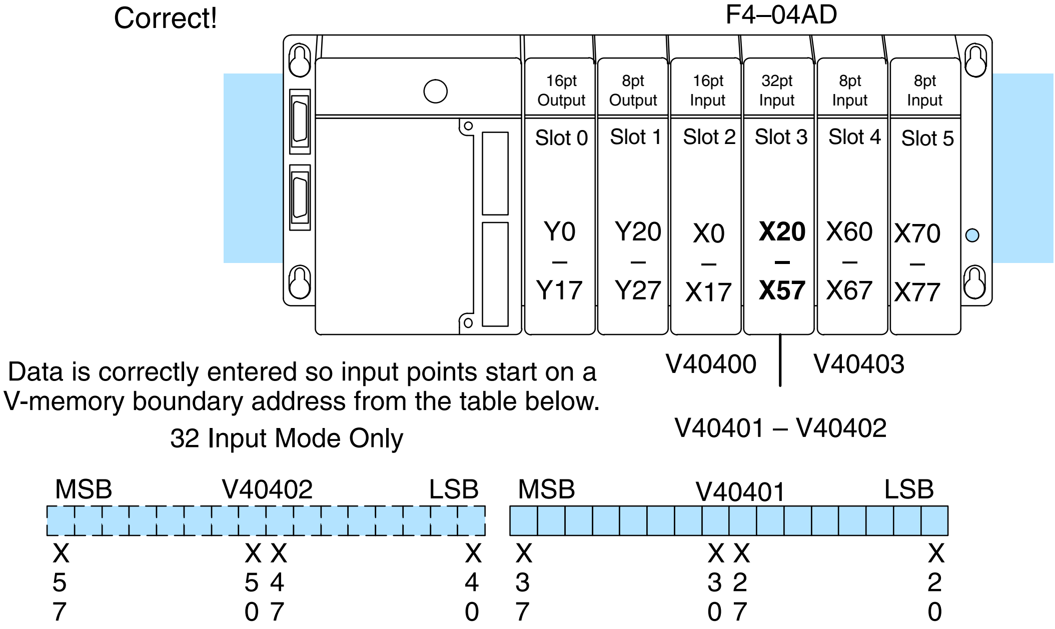

See NOTE 1
CH1
Single-ended Voltage Transmitter
See NOTE 3
CH2
Differential Voltage Transmitter
CH3
Differential Current Transmitter
CH4
Not Used
See NOTE 2
OV
V
C
I
OV
250 ohms
Internal Module Wiring (front end)
A to D Convertor
Analog Switch
User Supply Return
Channel select
User Supply
21.6 – 26.4 VDC Class 2
ANALOG INPUT
TB 24V RUN NEG
DATA
1 16 256 CH1
2 32 512 CH2
4 64 1024 CH3
8 128 2048 CH4
F4–04AD
DISPLAY CH
0–10VDC
1–5VDC
–10→+10VDC
4mA–20mA
24VDC 0.1A CLASS 2
F4–04AD

Module Operation

DL430 Special Requirements

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

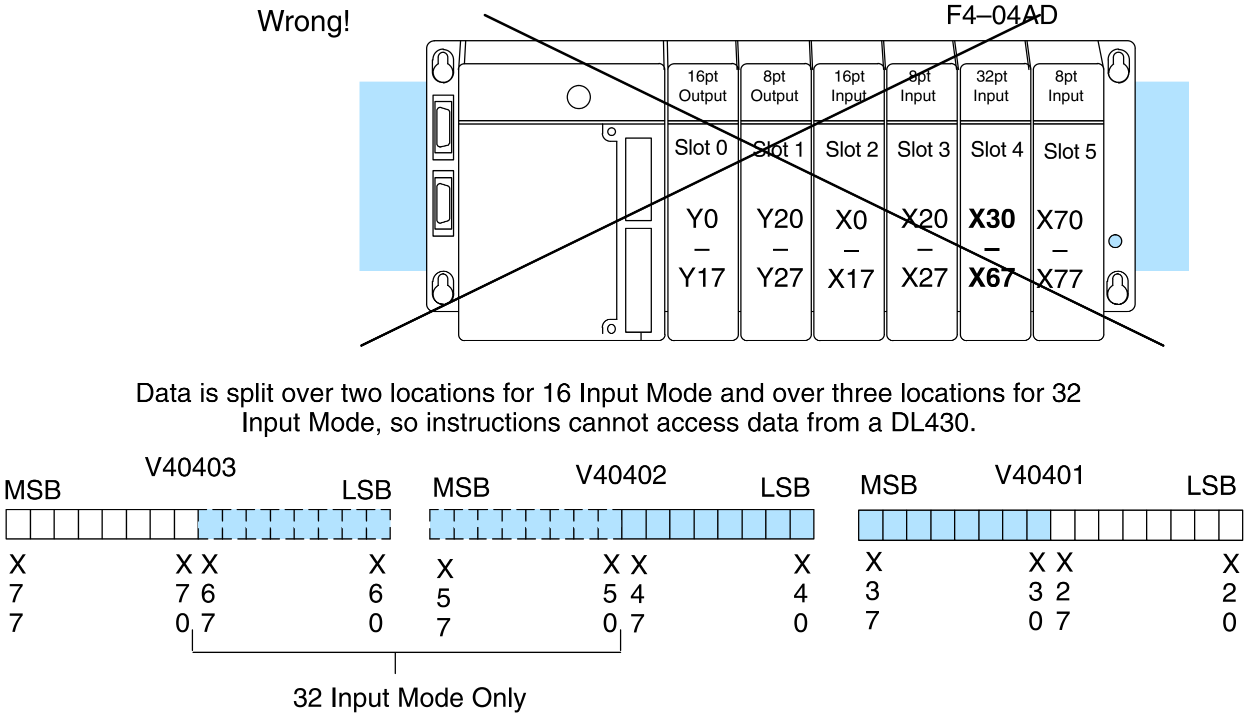
Correct!


F4–04AD
16pt Output
8pt Output
16pt Input
32pt Input
8pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y17
Y20 – Y27
X0 – X17
X20 – X57
X60 – X67
X70 – X77
V40400
V40403
V40401 – V40402
MSB
V40402
LSB
MSB
V40401
LSB
X57
X50
X47
X40
X37
X30
X27
X20

Data is correctly entered so input points start on a V-memory boundary address from the table below.

32 Input Mode Only

Wrong!


F4–04AD
16pt Output
8pt Output
16pt Input
8pt Input
32pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y17
Y20 – Y27
X0 – X17
X20 – X27
X30 – X67
X70 – X77
MSB
V40403
LSB
MSB
V40402
LSB
MSB
V40401
LSB
X77
X70
X67
X60
X57
X50
X47
X40
X37
X30
X27
X20

Data is split over two locations for 16 Input Mode and over three locations for 32 Input Mode, so instructions cannot access data from a DL430.

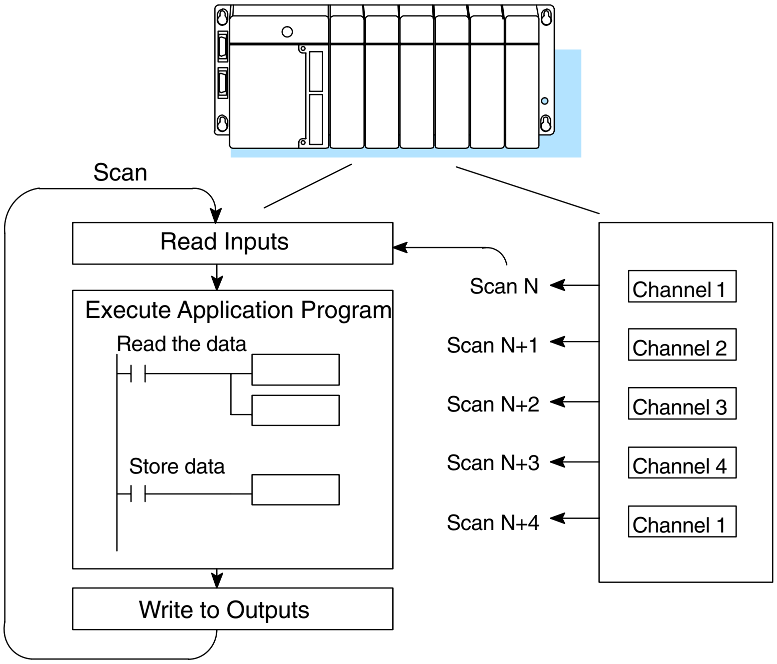
32 Input Mode Only

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 F4-04AD module supplies one channel of data per each CPU scan. This is true for both 16 Input and 32 Input Modes. Since there are four channels, it can take up to four scans to get data for all channels. Once all channels have been scanned the process starts over with channel 1.

Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan.



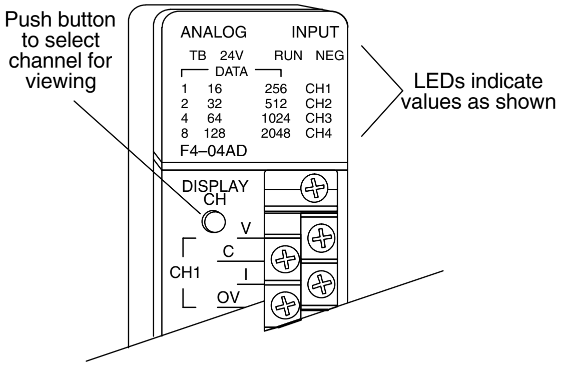
Even though channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

Displaying Diagnostic Data

At the top of the module's faceplate, LED indicators display information for the selected channel. The top row of LEDs display diagnostic information. The TB indicator turns on when the module senses a loose terminal block. The 24V indicator turns on when the external 24V supply voltage is low or not connected. The RUN LED flashes on and off only if the jumper configuration is valid, and the module's internal diagnostics have passed. If the jumper configuration is incorrect the RUN LED remains off. During normal operation, the RUN indicator flashes on and off continuously at approximately a one second rate. The NEG light turns on if the voltage or current input to the selected channel is negative.

Displaying Channel Data

By removing the connector cover you can access the push-button ”DISPLAY CH”, to select which channel’s data is currently being displayed. The CH1 through CH4 indicators correspond to the selected channel. The input value data corresponding to the channel is shown by the 12 data bit indicators. They are numbered from 1 to 2048 to indicate the binary weight. The bit is on (1) if the indicator is illuminated.


Push button to select channel for viewing
ANALOG
INPUT
TB
24V
RUN
NEG
DATA
1
2
4
8
16
32
64
128
256
512
1024
2048
CH1
CH2
CH3
CH4
F4–04AD
DISPLAY CH
V
C
CH1
I
OV
LEDs indicate values as shown

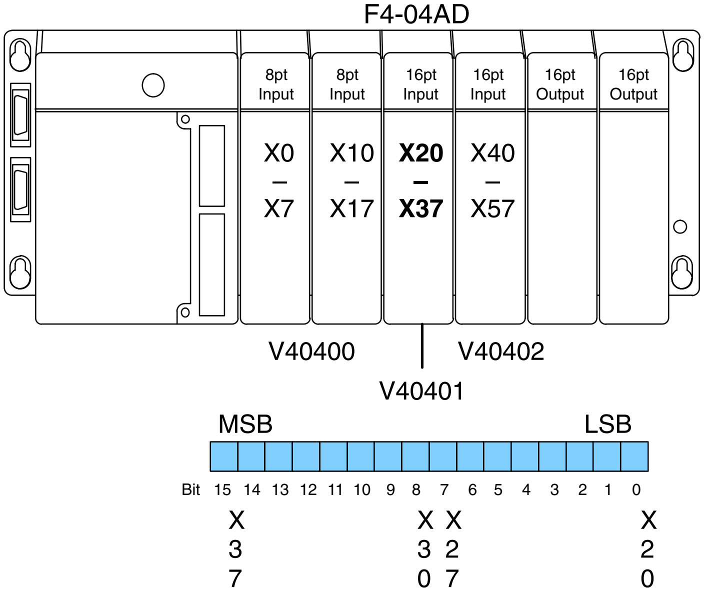
The next two sections describe the input bit assignments for both 16 Input and 32 Input operating modes. You need to read only the section that matches your selection in the jumper configuration.

Input Assignments for 16 Input Mode

In this mode, the F4-04AD module requires 16 discrete input points. These inputs provide:

- an indication of which channel is active.
- a digital representation of the analog signal (12 bit plus sign).
- broken transmitter detection for current signal inputs.

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.

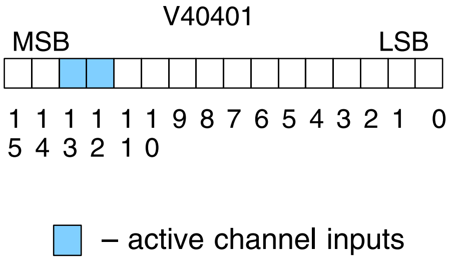

F4-04AD
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40401
V40402
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X30
X27
X20

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

Active Channel Indicator Inputs, 16 Input Mode

The two bits 12 and 13 (inputs) of the upper V-memory location indicate the active channel. They are binary encoded to indicate up to four active channels. Only the enabled channels are updated. The module automatically turns these inputs on and off to indicate the active channel for each scan.

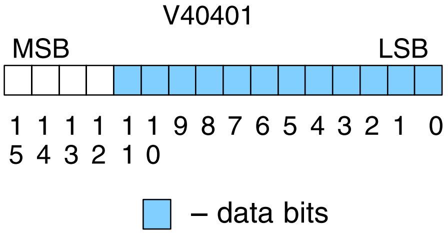
Scan	Bits 13	12	Active Channel
N	0	0	1
N+1	0	1	2
N+2	1	0	3
N+3	1	1	4
N+4	0	0	1


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
🟦 – active channel inputs

Analog Data Bits, 16 Input Mode

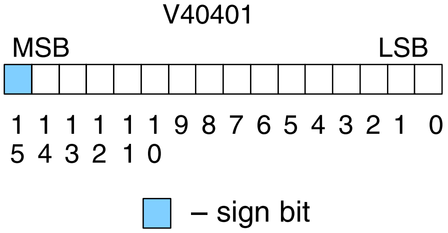
The first twelve bits of the first V-memory location represent the analog data in binary format. All input ranges use these bits.

Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
🟦 – data bits

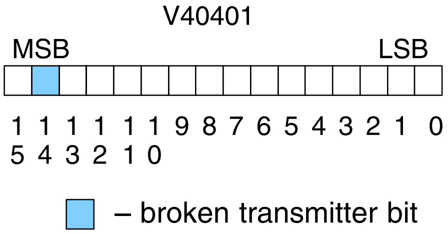
Sign Bit, 16 Input Mode

Bipolar input ranges use the twelve analog data bits as shown above, plus an additional sign bit. Bit 15 in the input word is the sign bit, and is a 1 when the polarity of the active channel is negative. If a unipolar mode is selected, the input value is assumed to be greater than or equal to zero, so this bit is always 0.


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
🟦 – sign bit

Broken Transmitter Bit, 16 Input Mode

One of the 4–20 mA current ranges features broken transmitter detection. Bit 14 in the input word is set to 1 if the current on the active channel is at 1.25 mA or less. This is useful for diagnostics or troubleshooting logic built in to your RLL program.

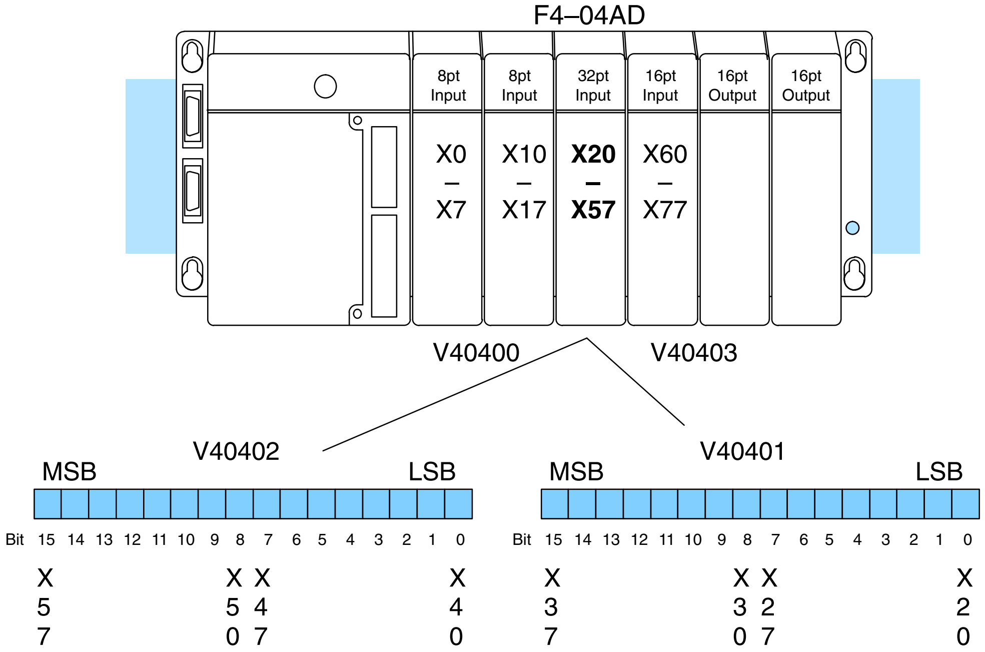

V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
🟦 – broken transmitter bit

Input Assignments for 32 Input Mode

In this mode, the F4–04AD module requires 32-point discrete input points. These inputs provide:

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

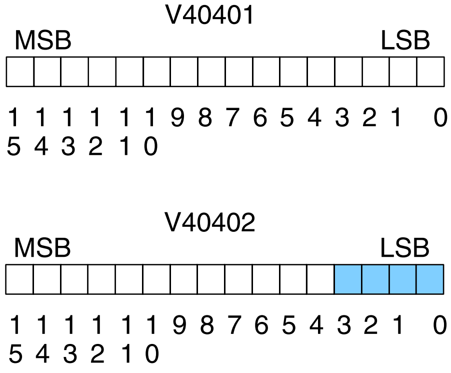
Since all input points are automatically mapped into V-memory, it is very easy to determine the location of the two data words that will be assigned to the module.


F4–04AD
8pt Input
8pt Input
32pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X57
X60 – X77
V40400
V40403
V40402
V40401
MSB
LSB
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X57
X50
X47
X40
X37
X30
X27
X20

Within these data word locations, the individual bits represent specific information about the analog signal.

Active Channel Indicator Inputs, 32 Input Mode

The first four input bits (0–3) of the upper V-memory location indicate the active channel. Each bit corresponds to a single channel to indicate four possible active channels. The module automatically turns these bits on and off each scan to indicate the active channel for that scan.


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
V40402
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

■ – active channel inputs

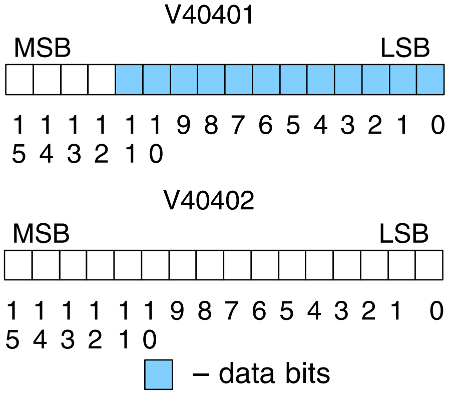
Scan	Bits 3	2	1	0	Active Channel
N	0	0	0	1	1
N+1	0	0	1	0	2
N+2	0	1	0	0	3
N+3	1	0	0	0	4
N+4	0	0	0	1	1

Analog Data Bits, 32 Input Mode

In 32 Input Mode the four possible data formats are 12-bit magnitude plus sign, two's complement 13-bit format, two's complement 12-bit format, and 12-bit magnitude. In the two 12-bit magnitude modes, the first twelve bits of the lower word represent the analog value's magnitude

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

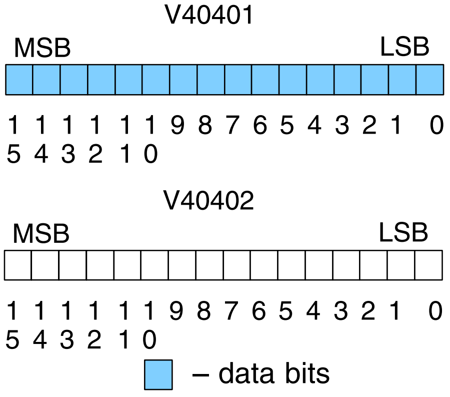
12-bit Magnitude Format


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
V40402
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– data bits

The two's complement formats are for bipolar inputs. Each range uses 16 data bits, and embeds the sign bit information in the data (no sign bit is required in these ranges). Each range is centered at 0, counting upward for positive numbers. Negative numbers start at 65535 (for count= –1), and count downward.

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

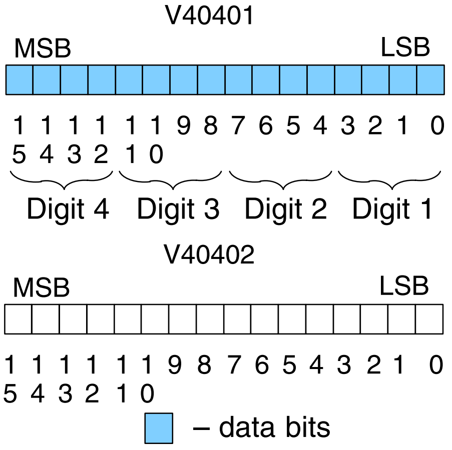
Two's Complement Format


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
V40402
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– data bits

The BCD formats use 16 bits of the lower word to represent four binary-coded decimal digits, from 0000 to 9999. Digit 1 is the LSD, Digit 4 is the MSD.

Bit	Value	Bit	Value
0	(digit 1), 1	8	(digit 3), 1
1	(digit 1), 2	9	(digit 3), 2
2	(digit 1), 4	10	(digit 3), 4
3	(digit 1), 8	11	(digit 3), 8
4	(digit 2), 1	12	(digit 4), 1
5	(digit 2), 2	13	(digit 4), 2
6	(digit 2), 4	14	(digit 4), 4
7	(digit 2), 8	15	(digit 4), 8

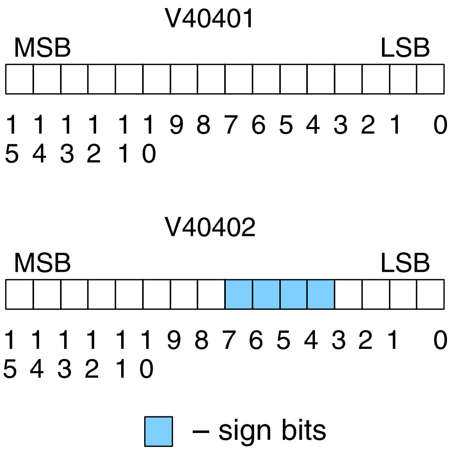
BCD Format


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Digit 4
Digit 3
Digit 2
Digit 1
V40402
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– data bits

Sign Bits, 32 Input Mode

Four bits (4 to 7) of the upper word are dedicated for use as sign bits. These are individually assigned to each of the four channels. When an input bit is on, the data for the corresponding channel represents a negative value. When the bit is off, the data is positive.

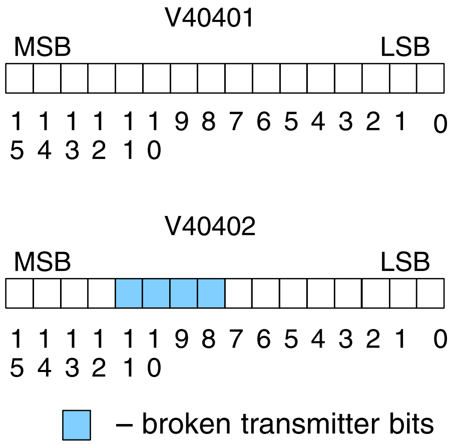
Bit	Channel
4	1
5	2
6	3
7	4


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
V40402
MSB
LSB
1 1 1 1 1 1 9 8 7 6 5 4 3 2 1 0
5 4 3 2 1 0
– sign bits

Broken Transmitter bits, 32 Input Mode

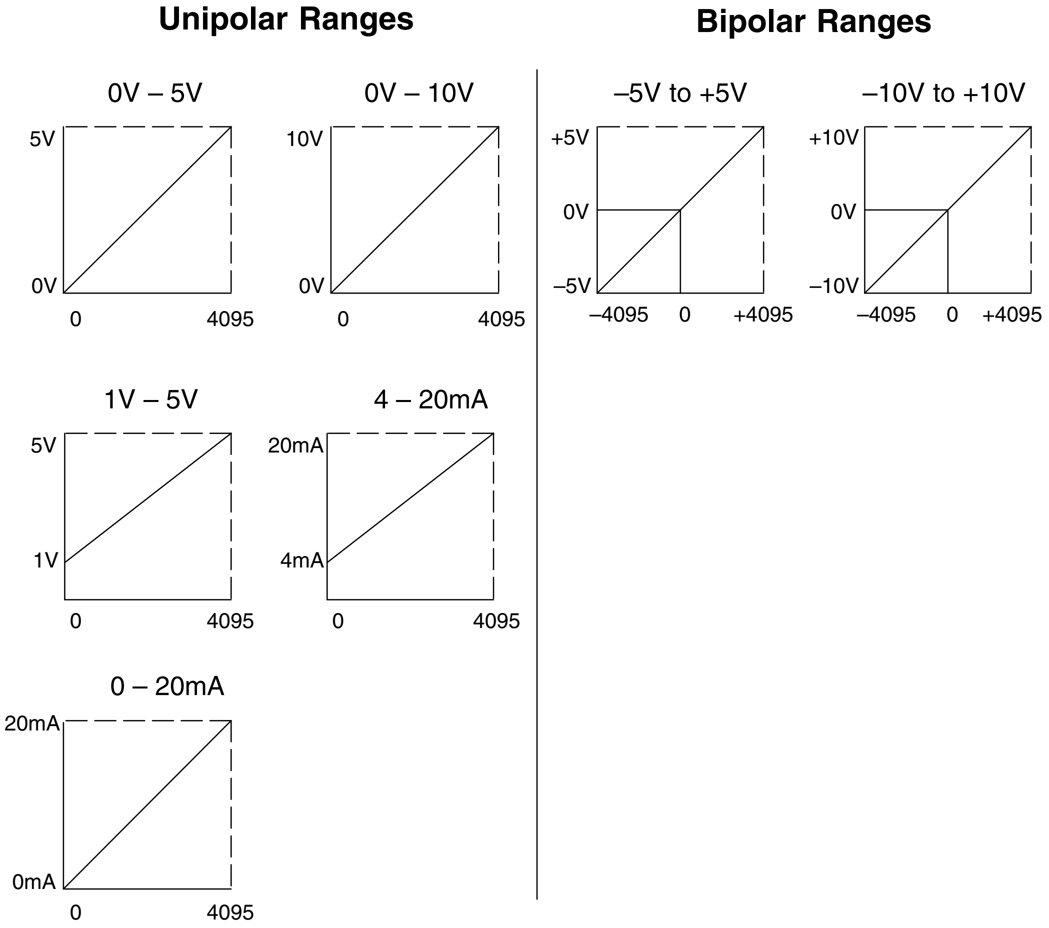
Four bits (8 to 11) of the upper word are dedicated for use as broken transmitter indications. They are only operational for the 4 to 20 mA. input range. When an input bit is on, the current for the corresponding channel is at or below 1.25 mA. When the condition ends, the bit automatically turns off.

Bit	Channel
8	1
9	2
10	3
11	4


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
V40402
MSB
LSB
1 1 1 1 1 1 9 8 7 6 5 4 3 2 1 0
5 4 3 2 1 0
– broken transmitter bits

12-Bit Magnitude Plus Sign Format, (All Modes)

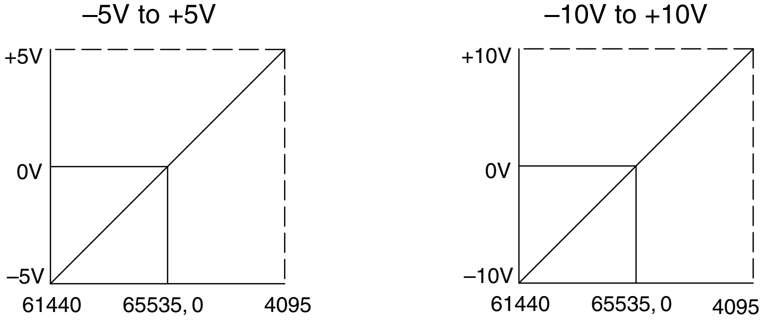
The 12-Bit Plus Sign conversion range is available in either 16 Input or 32 Input Modes, but it's the only data conversion format available in 16 Input mode. Unipolar signal ranges use 12-bit resolution. Bipolar ranges have 13-bit resolution because of the additional sign bit. The 12 data bits convert the analog signal to 4096 "pieces" ranging from 0 to 4095 (2^{12}). For example, with a 0 to 5V scale, a 0V signal would be 0, and a 5V signal would be 4095. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal.



Two's Complement Format, 13-Bit

The 32 Input Mode offers two's complement data formats in 12-bit and 13-bit ranges. The 13-bit range is recommended for new applications, while the 12-bit range is recommended only for compatibility with D4-04AD applications. The 13-bit format is for bipolar voltage input ranges only. Depending on your application, two's complement format can be very useful. Some operator interfaces or other peripheral devices may require two's complement format. If you need to add positive and negative values together (as in calculating an average), this format can simplify your RLL program. Two's complement representation imbeds the sign bit information in the data. It allows CPU instructions to add numbers together without specific logic to handle the sign bit for negative numbers. The 13-bit two's complement format actually uses 16 binary data bits. The following diagram shows how this works.

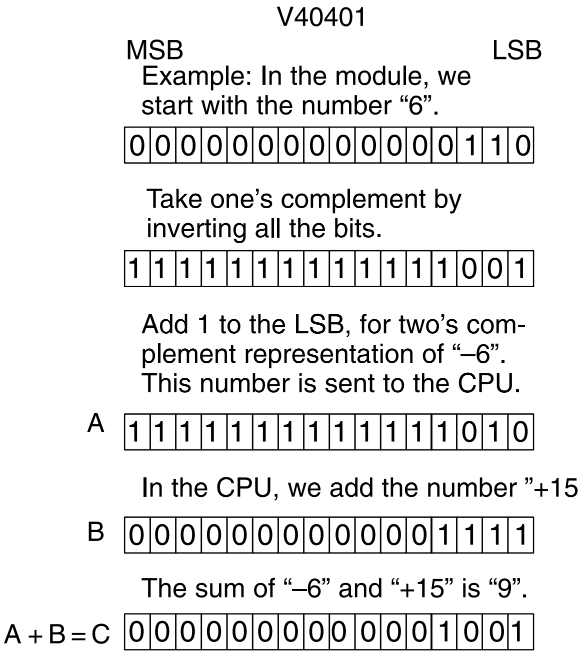
Two's Complement 13-bit Format



In the left graph above, zero volts converts to a count of zero. Positive voltages up to +5 volts convert to counts of up to decimal +4095. A few millivolts less than zero converts to 65535, the equivalent to -1 count. At -5V, the conversion is to 61440 counts. The conversion method translates positive polarity signals per normal binary scaling. It's negative values that include an additional step. In this case,we start at the top of the 16-bit binary range (65535), and count downward. With zero count point at mid-range,negative numbers transition to positive numbers.

As an example, suppose the module sends the counts of –6 and +15 in successive scans to the CPU. The RLL program is going to sum the input values. When the module is configured for two's complement format, negative numbers are specially formatted. It takes the –6 in binary and takes a one's complement by inverting all the bits. Then, it adds 1 to the LSB to get a two's complement representation. The 16-bit result the module sends to the CPU is decimal 65530, or FFFA hex, representing –6.

In the ladder program, you can add +15 to this number. By ignoring all end carries, we have the correct answer of +9. The ladder program is simplified because it does not need to examine a sign bit to do a subtract instruction.



Example: In the module, we start with the number "6".

0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0

Take one's complement by inverting all the bits.

1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1

Add 1 to the LSB, for two's complement representation of "–6". This number is sent to the CPU.

A 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0

In the CPU, we add the number "+15

B 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1

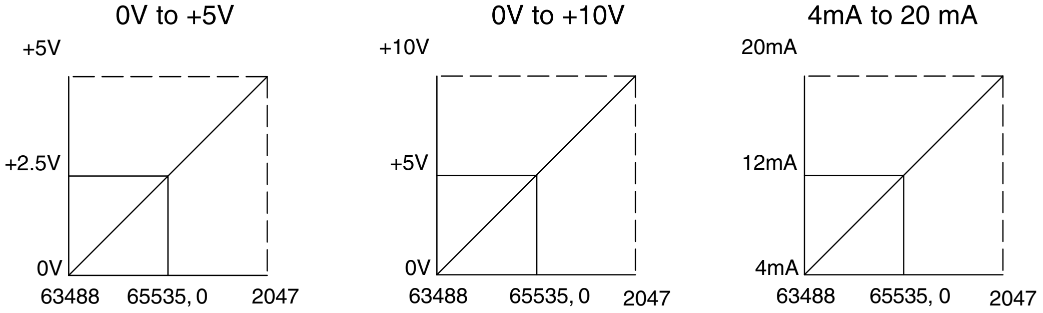
The sum of "–6" and "+15" is "9".

A + B = C 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1

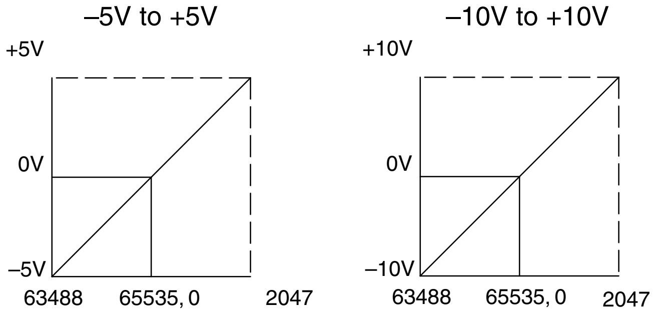
Two's Complement Format, 12-Bit

The module's 32 Input Mode offers two's complement data formats in 12-bit and 13-bit ranges. The 12-bit range shown here is recommended only for compatibility with existing D4–04AD applications. The 12-bit range may be used with both unipolar and bipolar input signal ranges. The 12 data bits convert the analog signal to 4096 "pieces" ranging from 0 to 4095 (2^{12}). For example, with a 0 to 5V scale, a 0V signal would be –2048, represented as 63488, and a 5V signal would be 2047. This is equivalent to a binary value of 1111 1000 0000 0000 to 0000 1111 1111 1111, or F800 to 0FFF hexadecimal. However, two's complement representation is more commonly used with bipolar input signal ranges.

Unipolar Ranges, Two's Complement 12-bit Format



Bipolar Ranges, Two's Complement 12-bit Format



Unipolar Resolution

Each count can also be expressed in terms of the signal level by using the equation shown. Unipolar ranges have 12 bits of resolution, which divides the signal span into 4095 counts. The following table shows the smallest signal change that will result in a single LSB change in the data value for each signal input range.

$$\text{Unipolar resolution} = \frac{H-L}{4095}$$

H = High limit of the input signal
L = Low limit of the input signal

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

Bipolar Resolution

Bipolar ranges have 13 bits of resolution, (the additional sign bit adds an additional bit of resolution). This divides the signal span into 8191 counts. The following table shows the smallest signal change that will result in a single LSB change in the data value for each signal input range.

$$\text{Bipolar resolution} = \frac{H-L}{8191}$$

H = High limit of the input signal
L = Low limit of the input signal

Range	Signal Span (H – L)	Divide By	Smallest Detectable Change
–5 to +5V	10 V	8191	1.22 mV
–10 to +10V	20 V	8191	2.44 mV

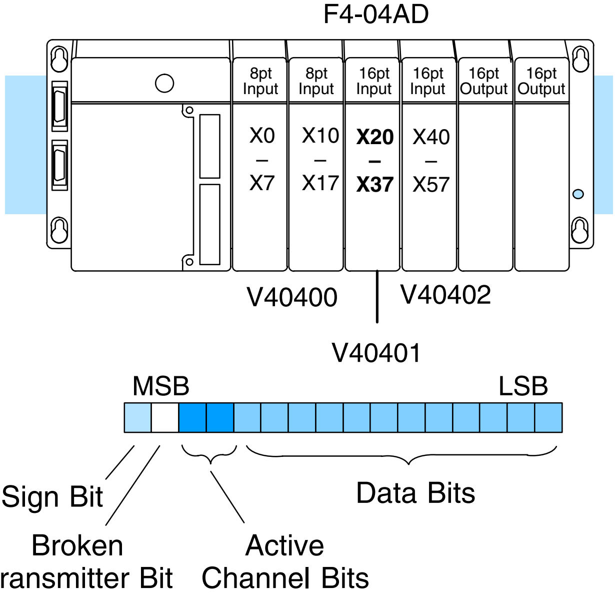
Now that you understand how the module and CPU work together to collect and store the information, you're ready to write the control program.

Writing the Control Program, 16 Input Mode

If you have configured the F4-04AD module for 16 Input mode, use the following examples to get started writing the control program. For modules configured in 32 Input mode, skip to the section titled “Writing the Control Program, 32 Input Mode”.

Multiple Active Channels

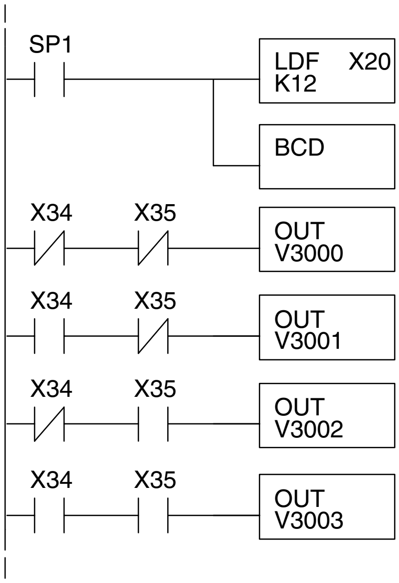
Since all channels are multiplexed into a single data word, the control program must determine which channel's data is being sent from the module during each scan. If you have enabled only one channel, then its data will be available on every scan. Two or more channels require multiplexing the data word. Since the module requires 16 input points from the CPU, it is very easy to use the active channel status bits to determine which channel is being monitored.


F4-04AD
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
Sign Bit
Data Bits
Broken Transmitter Bit
Active Channel Bits

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, etc.


SP1
LDF X20
K12
BCD
X34
X35
OUT V3000
X34
X35
OUT V3001
X34
X35
OUT V3002
X34
X35
OUT V3003

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

It's usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can omit this instruction if your application does not require it (such as PID loops).

When X34 and X35 are off, channel 1 data is being sent to the CPU. The OUT instruction moves the data from the accumulator to V3000.

When X34 is on and X35 is off, channel 2 data is stored in V3001.

When X34 is off and X35 is on, channel 3 data is stored in V3002.

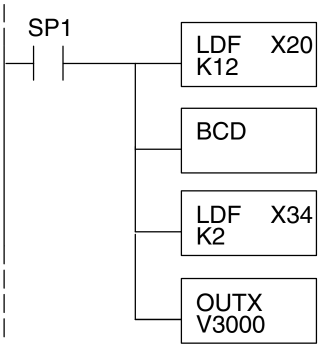
When X34 and X35 are on, channel 4 data is stored in V3003.

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

Optional Method, DL440/450

☒ ✓ ✓
430 440 450

The previous example used the OUT instruction to store channel data in V memory, requiring four ladder rungs. The OUTX (Out Indexed) instruction in the next example does much of that work for you. It uses the first stack location to temporarily hold the data to be stored at an address modified by an offset in the accumulator.



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

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

The OUTX (out indexed) instruction stores the channel data, currently the first item on stack, to an address that starts at V3000 plus the channel offset (0–3) located in the accumulator. For example, when channel 3 is read, the data is stored in V3002 (V3000 + 2).

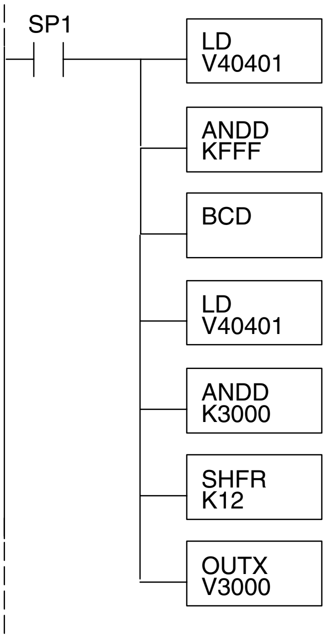
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	00	0	V3000
Channel 2	01	1	V3001
Channel 3	10	2	V3002
Channel 4	11	3	V3003

Reading Values, DL430

✓ ✓ ✓
430 440 450

The following program example shows how to read the analog data into V-memory locations with DL430 CPUs. Since the DL430 does not support the LDF instruction, you can use the LD instruction instead as shown. You can also use this method with DL440 and DL450 CPUs.



Loads the complete data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map.

Mask off active channel bits, etc. above the 12 bits of data.

It's usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can omit this instruction if your application does not require it (such as PID loops).

The load instruction reads the data into the accmulator again. This pushes the channel data onto a stack.

This instruction masks the analog data values, sign bit, and broken transmitter bit, to leave the active channel bits in the accumulator.

Now you have to shift the active channel bits to the right so the result has a value from 0 to 3 (inclusive) in binary format.

The OUTX (out indexed) instruction stores the channel data, currently the first item on stack, to an address that starts at V3000 plus the channel offset (0–3) located in the accumulator. For example, when channel 3 is read, the data is stored in V3002 (V3000 + 2).

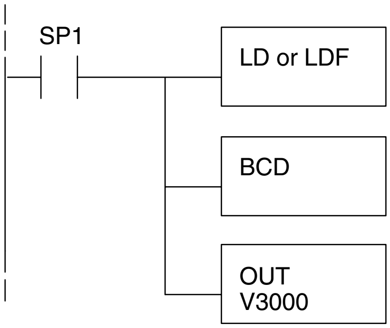
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	00	0	V3000
Channel 2	01	1	V3001
Channel 3	10	2	V3002
Channel 4	11	3	V3003

Single Active Channel

✓	✓	✓
430	440	450

If the module is configured for only one input channel, you can omit the channel selection logic which simplifies the program.



Channel 1 data is always being sent to the CPU. Use LD or LDF, depending on the type of CPU you are using.

The BCD instruction converts the data from binary to BCD. This instruction may be optional for your application (do not use with PID loops).

The OUT instruction stores the data in V3000.

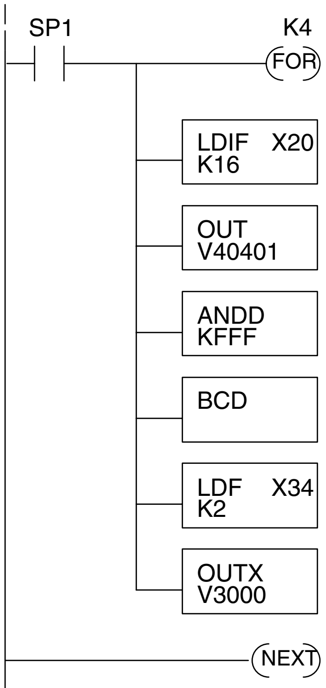
Note: This example uses SP1, which is always on. You can also use an X, C, etc. permissive contact. Also, the DL430 requires an additional instruction to mask off the most significant four bits that are brought in with the LD instruction, before the BCD instruction is executed. This method is shown in the previous example.

Reading Four Channels in One Scan, DL440/450

✗	✓	✓
430	440	450

The following program example shows how to read all four channels in one scan by using a FOR/NEXT loop. Before choosing this technique, do consider its impact on the scan time. Remember the FOR/NEXT routine shown here will add about 5 ms (1.25 ms/loop) to the overall scan time. If you don't need to read the analog data on every scan, change SP1 to a permissive contact (such as an X input, CR ,or stage bit) to only enable the FOR/NEXT loop when it is required.

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



Starts the FOR/NEXT loop. The constant (K4) specifies how many times the loop will execute, equal to the number of channels you are using. For example, enter K3 if you're using 3 channels.

Immediately loads all 16 bits of the data word into the accumulator. The LDIF instruction retreives the I/O points without waiting on the CPU to finish the scan.

Save the new input status which is in the accumulator to the image register (V memory). Remember, the FOR–NEXT loop will do this four times before the CPU's normal scan updates V40401 again.

This ANDD instruction masks off the upper four bits, leaving just the 12-bit analog value in the accumulator.

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 LDF instruction loads the two active channel bits into the accumulator. The OUT instruction above updated the V-memory image which makes this possible during a scan. X34 = X20 + 14.

The OUTX instruction stores the channel data to an address that starts at V3001 plus the channel offset (0–3). For example, if channel 3 was being read, the data would be stored in V3002 (V3000 + 2).

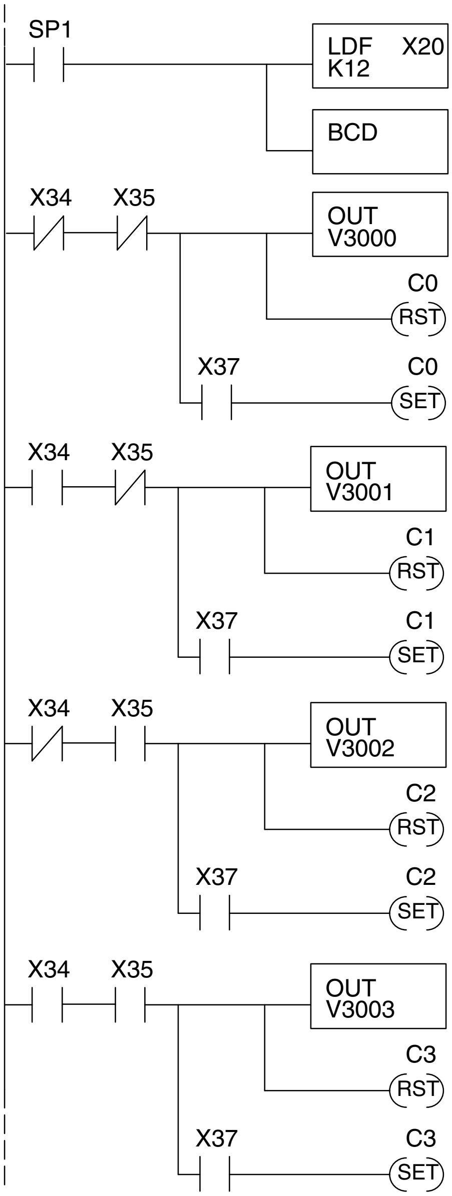
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003

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

Reading Values With Sign Bits, DL440/450

☒ ✓ ✓
430 440 450

In 16 Input Mode, the most significant bit (bit 15) is the sign bit for the active channel of the current scan. Because it is multiplexed (shared) among the four channels, you may need to separate it into four individual sign bits. The following example gives a method to do this, giving the resulting sign bits as internal contacts C0 to C3.



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

It's usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can omit this instruction if your application does not require it (such as PID loops).

Channel 1 data is being sent when X34 and X35 are off. The out instruction moves the data from the accumulator to V3000.

Turn off sign bit (C0) for channel 1. It will remain off for positive numbers.

When the module's sign bit (X37) is on the data is negative, C0 turns on.

Channel 2 data is being sent when X34 is on and X35 is off. The out instruction moves the data from the accumulator to V3001.

Turn off sign bit (C1) for channel 2. It will remain off for positive numbers.

When the module's sign bit (X37) is on the data is negative, C1 turns on.

Channel 3 data is being sent when X34 is off and X35 is on. The out instruction moves the data from the accumulator to V3002.

Turn off sign bit (C2) for channel 3. It will remain off for positive numbers.

When the module's sign bit (X37) is on the data is negative, C2 turns on.

Channel 4 data is being sent when X34 and X35 are on. The out instruction moves the data from the accumulator to V3003.

Turn off sign bit (C3) for channel 4. It will remain off for positive numbers.

When the module's sign bit (X37) is on the data is negative, C3 turns on.

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

Broken Transmitter Detection

When the 4–20 mA range is selected, the bit next to the most significant bit (bit 14) is on when the current for the active channel is less than 1.25 mA. You can use the method in the previous example to generate four independent broken transmitter bits. Just replace X37 with X36 in the example.

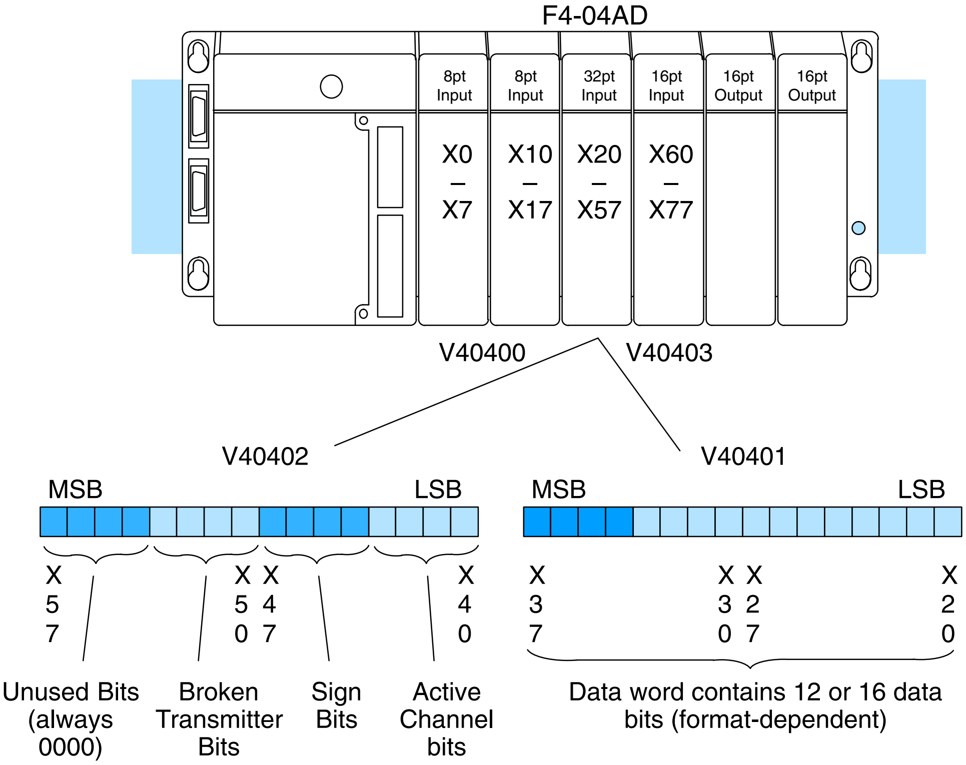
Writing the Control Program, 32 Input Mode

If you have configured the F4–04AD module for 32 Input mode, use the following examples to get started writing the control program (for modules configured in 16 Input mode, go back to the section titled “Writing the Control Program, 16 Input Mode”).

Multiple Active Channels

The analog data is multiplexed into the lower word. It is presented in either 12 or 16 bits, depending on the range and format selected. In the 12-bit format modes, the upper 4 bits are always 0000. The upper word contains three groups of bits that contain active channel status, sign bit information, and broken transmitter status. Each bit group contains one bit for each channel. The upper four bits are unused, and are always 0000.

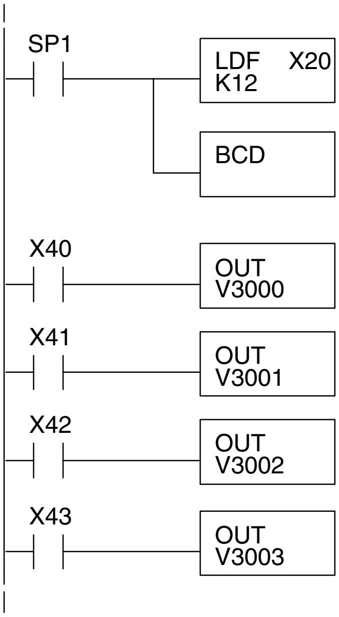
The control program must determine which channel’s data is being sent from the module. If you have enabled only one channel, its data will be available on every scan. Two or more channels require multiplexing the lower data word. Since the module communicates as X input points to the CPU, it is very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


F4-04AD
8pt Input
8pt Input
32pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X57
X60 – X77
V40400
V40403
V40402
V40401
MSB
LSB
MSB
LSB
X57
X50
X47
X40
X37
X30
X27
X20
Unused Bits (always 0000)
Broken Transmitter Bits
Sign Bits
Active Channel bits
Data word contains 12 or 16 data bits (format-dependent)

Reading Values, DL440/450

☒ ☑ ☑
430 440 450

The following program example shows how to read the analog data into V-memory locations with the DL440 and DL450 CPUs. Once the data is in V-memory, you can perform math on the data, compare the data against preset values, etc.



Loads the first 12 bits of the data word into the accumulator. The X address depends on the I/O configuration. If using a two's complement mode, use the constant K16 in the box.

It's usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. Note that you can configure the module to send bipolar voltage input data in BCD format, making this step unnecessary.

When X40 is on, channel 1 data is being sent to the CPU. The out instruction moves the data from the accumulator to V3000.

When X41 is on, channel 2 data is stored in V3001.

When X42 is on, channel 3 data is stored in V3002.

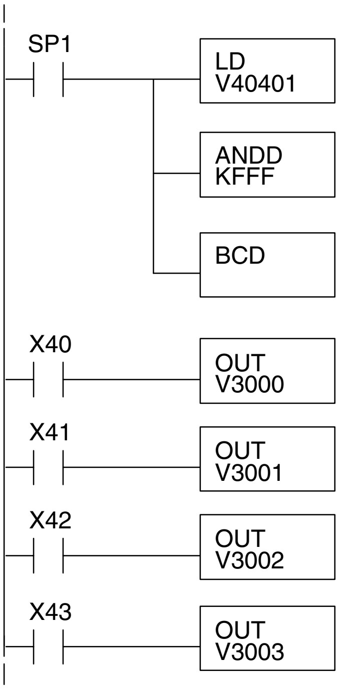
When X43 is on, channel 4 data is stored in V3003.

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

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. You can also use this method with DL440 and DL450 CPUs.



Loads all 16 bits of the data word into the accumulator. The X address depends on the I/O configuration. If the module is configured for one of the 12 bit ranges, the upper for bits are returned as 0000.

ANDDS the value in the accumulator with the constant KFFF, which masks the channel ID bits, and stores the value in the accumulator. Without this, the values will not be correct, so do not forget to include it.

It's usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. Note that you can configure the module to send bipolar voltage input data in BCD format, making this step unnecessary.

When X40 is on, channel 1 data is being sent to the CPU. The out instruction moves the data from the accumulator to V3000.

When X41 is on, channel 2 data is stored in V3001.

When X42 is on, channel 3 data is stored in V3002.

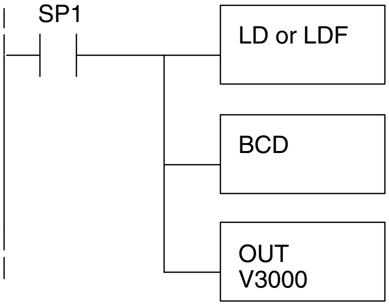
When X43 is on, channel 4 data is stored in V3003.

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

Single Active Channel

✓	✓	✓
430	440	450

If the module is configured for only one input channel you can omit the channel selection logic; this simplifies the program.



Channel 1 data is always being sent to the CPU. Use LD or LDF, depending on the type of CPU you are using.

The BCD instruction converts the data from binary to BCD. This instruction may be optional for your application. Do not use with PID loops.

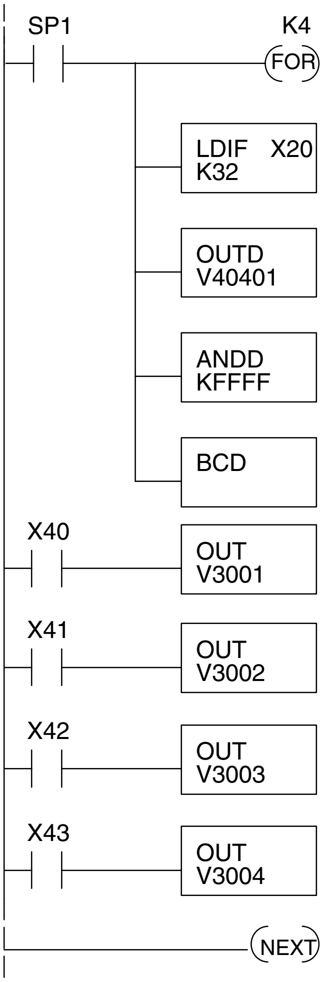
The OUT instruction stores the data in V3000.

Note: This example uses SP1, which is always on. You can also use an X, C, etc. permissive contact. Also, the DL430 requires an additional instruction to mask off the most significant four bits that are brought in with the LD instruction, before the BCD instruction is executed. This method is shown in the previous example using an ANDD instruction.

Reading Four Channels in One Scan, DL440/450

☒	✓	✓
430	440	450

The following program example shows how to read all four channels in one scan by using a FOR/NEXT loop. Remember, the FOR/NEXT routine shown here will add about 5 ms (1.25 ms/loop) to the overall scan time. If you don't need to read the analog data on every scan, change SP1 to a permissive contact (such as an X input, CR, or stage bit) to only enable the FOR/NEXT loop when it is required. This FOR/NEXT loop program will not work in a remote/slave arrangement; use one of the programs shown that reads one channel per scan.



Starts the FOR/NEXT loop. The constant (K4) specifies how many times the loop will execute, equal to the number of channels you are using. For example, enter K3 if you're using 3 channels.

Immediately loads all 32 bits of the data word into the accumulator. The LDIF instruction retreives the I/O points without waiting on the CPU to finish the scan.

Save the new input status which is in the accumulator to the image register (V memory). Remember, the FOR–NEXT loop will do this four times before the CPU's normal scan updates V40401 and 40402 again.

This ANDD instruction masks off the upper sixteen bits, leaving just the lower 16 bits which contain the 12 or 16 bit analog value in the accumulator.

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.

One of the four active channel bits will be on each time through the FOR–NEXT loop, indicating the active channel. The corresponding OUT instruction places the 12 or 16-bit value in the accumulator in the proper V memory location.

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

Reading Values With Sign Bits If the 13-bit magnitude plus sign format is selected, the sign bits (X44 to X47 in our example) will be on when the corresponding analog input channel(s) send negative data. The bits are always off (0000) in the 12-bit magnitude and two's complement formats.

Broken Transmitter Detection When the 4 to 20mA with broken transmitter detection range is selected, the four Broken Transmitter bits (X50 to X53 in our example) will be on when the current on the corresponding analog input channel(s) is below the normal 4 mA bottom end of the range. The threshold at which these bits turn on is +1.25 mA.

Scaling and Converting the Input Data

The following examples show you how to scale and convert the input data, for both 16 Input and 32 Input modes.

Scaling the Input Data Most applications usually require measurements in engineering units, which provide more meaningful data. This is accomplished by using the conversion formula shown.

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

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

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 wanted to measure pressure (PSI) from 0.0 to 99.9 then you would have to multiply the analog value by 10 in order to imply a decimal place when you view the value with the programming software or a handheld programmer. Notice how the calculations differ when you use the multiplier.

Analog Value of 2024, slightly less than half scale, should yield 49.4 PSI

Example without multiplier

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

$$\text{Units} = 2024\,\frac{100 - 0}{4095}$$

$$\text{Units} = 49$$

Handheld Display

```
   V 3101  V 3100
V MON   0000    0049
```

Example with multiplier

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

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

$$\text{Units} = 494$$

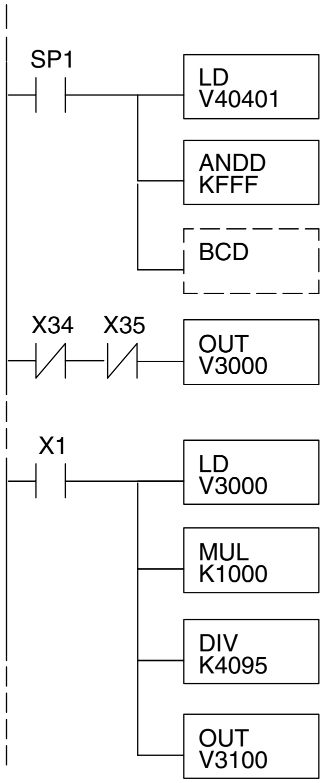
Handheld Display

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

This value is more accurate

Here's how you would write the program to perform the engineering unit conversion.

16 Input Mode Example



Loads the data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map. Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Mask off the upper four bits. If you have a DL440 or DL450 CPU you can use LDF with K12 for the first rung, making this instruction unnecessary.

Since we are going to perform some math operations in BCD, this instruction converts the data format. You may have already converted the data in the previous examples. If so, leave out this instruction.

Channel 1 data is being sent to the CPU when X34 and X35 are off. The OUT instruction moves the data from the accumulator to V3000.

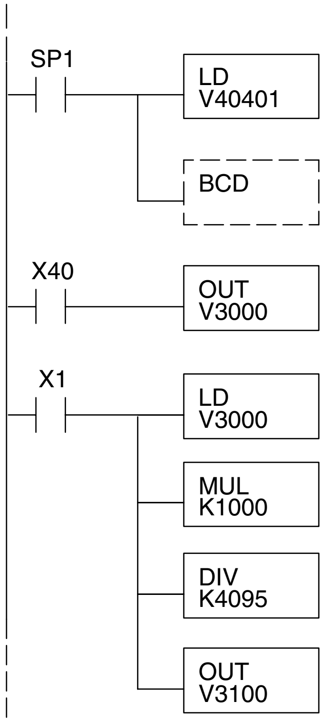
When X1 is on, load channel 1 data into the accumulator.

Multiply the accumulator by 1000 (to start the conversion). We have a range of 0 to 100, and also need to see tenths of a unit. So, 100 times 10 is 1000.

Divide the accumulator value by 4095.

Store the result in V3100.

32 Input Mode Example



Loads the data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map. Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Since we are going to perform some math operations in BCD, this instruction converts the data format. You may have already converted the data in the previous examples. If so, leave out this instruction.

Channel 1 data is being sent to the CPU when X40 is on. The OUT instruction moves the data from the accumulator to V3000.

When X1 is on, load channel 1 data to the accumulator.

Multiply the accumulator by 1000 (to start the conversion). We have a range of 0 to 100, and also need to see tenths of a unit. So, 100 times 10 is 1000.

Divide the accumulator by 4095.

Store the result in V3100.

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the signal levels and the digital values. This is especially useful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the analog signal level ...
0 to 5V	$A = \frac{5(D)}{4095}$	$D = \frac{4095}{5}(A)$
0 to 10V	$A = \frac{10(D)}{4095}$	$D = \frac{4095}{10}(A)$
1 to 5V	$A = \frac{4(D)}{4095} + 1$	$D = \frac{4095}{4}(A - 1)$
4 to 20mA	$A = \frac{16(D)}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$
0 to 20mA	$A = \frac{20(D)}{4095}$	$D = \frac{4095}{20}(A)$
±5V	$A = \frac{5(D)}{4095}$	$D = \frac{4095(A)}{5}$
± 10V	$A = \frac{10(D)}{4095}$	$D = \frac{4095(A)}{10}$

For example, suppose you are using the 4 to 20 mA input range. If you know the input signal measures 9 mA, just use the appropriate formula from the table. It will give you the the digital value the module sends to the CPU.

$$D = \frac{4095}{16}(9–4)$$

$$D = 256\ (5)$$

$$D = 1280 \text{ counts}$$

As a bipolar example, suppose you are using the ± 10V range. If you know the CPU receives 2893 counts and the sign bit is set, just insert –2893 into the appropriate formula from the table. It will give you the analog voltage that is present at the connector for the corresponding channel.

$$A = \frac{10\ (–2893)}{4095}$$

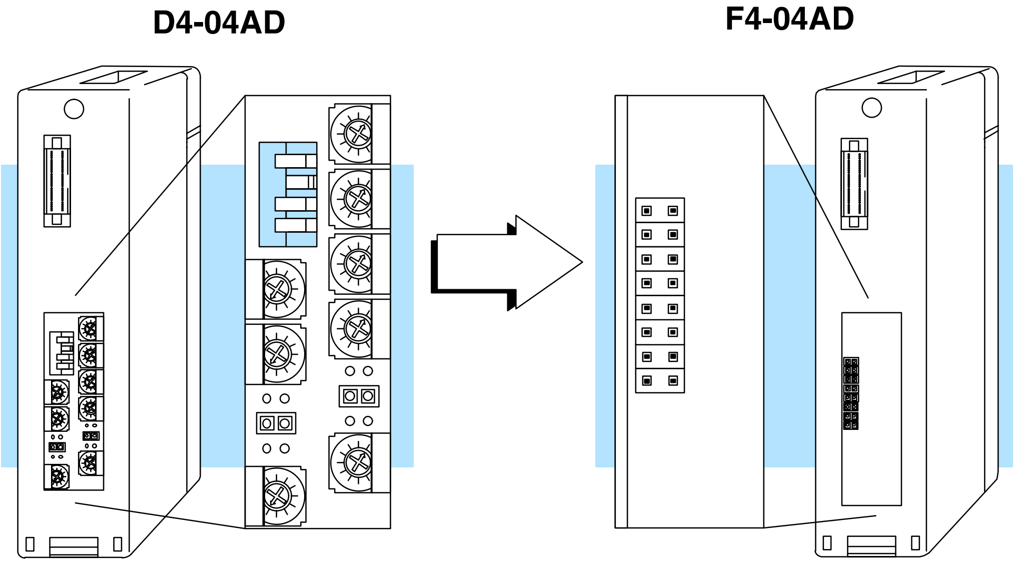
$$A = \frac{–28930}{4095}$$

$$A = –7.06 \text{ Volts}$$

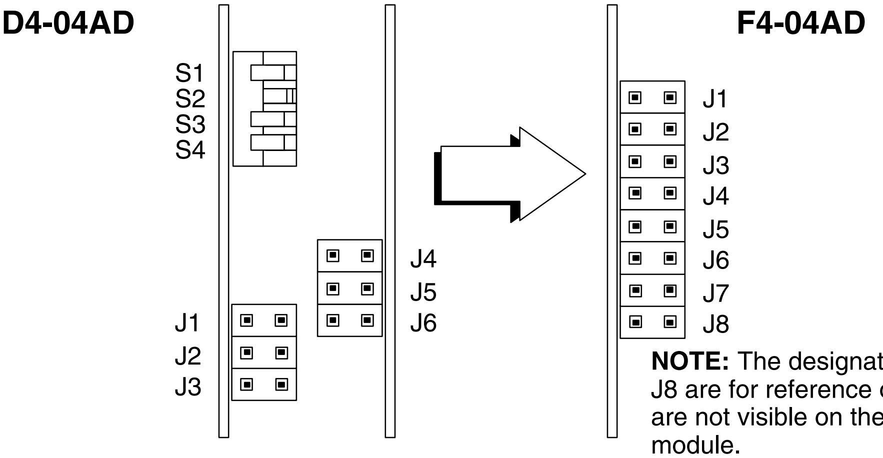
Configuration Cross-Reference, D4-04AD to F4-04AD

The new F4-04AD Analog Input Module replaces the existing D4-04AD Analog Input Module, plus adds new functionality. However, for existing installations this section shows how to configure the F4-04AD to directly replace a D4-04AD as presently configured. The RLL program that communicated with the D4-04AD needs *no modifications* to perform the same function with a properly configured F4-04AD!

NOTE: When the F4-04AD is configured for D4-04AD compatibility, the module's output word bit descriptions and ladder examples in this chapter do not apply.



In the following procedure, we will examine the present switch and jumper settings on the D4-04AD. Just follow the steps to translate that configuration into its equivalent on the F4-04AD module.



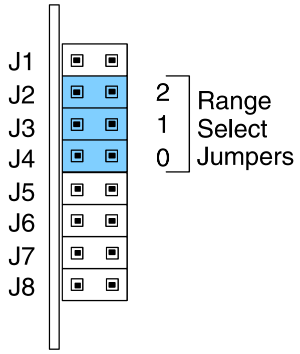
NOTE: The designations J1 thru J8 are for reference only; they are not visible on the actual module.

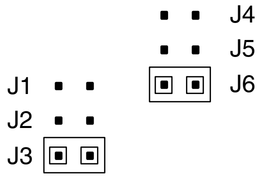
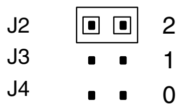
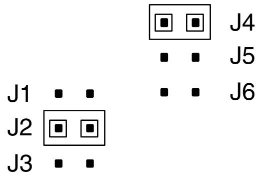
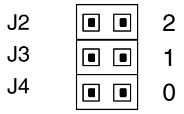
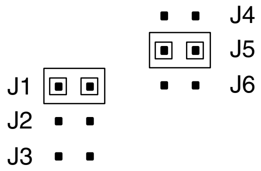
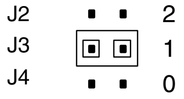
Step 1: 32-Input Mode

Install the top jumper J1, on the F4-04AD. This selects 32 Input Mode so the module has 32 X inputs to the CPU, as does the D4-04AD.

**Step 2:
Range Selection**

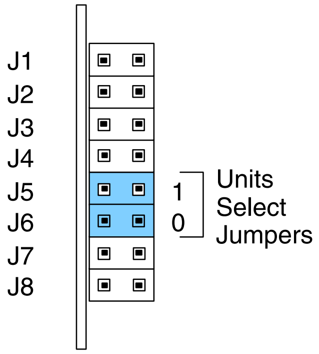
The jumpers on the D4-04AD correspond to the range select jumpers on the F4-04AD, The figure on the right shows these are a subset of the jumper block, and includes J2, J3, and J4. They select the voltage or current range on the F4-04AD for all four input channels simultaneously. Observe the present jumper settings on the D4-04AD. Then find the row in the table below that matches, and configure the F4-04AD jumpers to match.

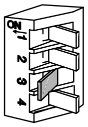
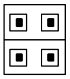
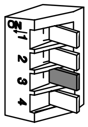
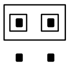

J1
J2
J3
J4
J5
J6
J7
J8
2
1
0
Range
Select
Jumpers

Input Range Selection Cross-Reference		
Input Signal Range	**D4-04AD Jumper Settings**	**F4-04AD Range Select Jumper Settings**
0 V to +10 VDC	 J4 J5 J6 J1 J2 J3	
+1 V to +5 VDC, or 4 to 20 mA	 J4 J5 J6 J1 J2 J3	
± 10 VDC	 J4 J5 J6 J1 J2 J3	

**Step 3:
Units Selection**

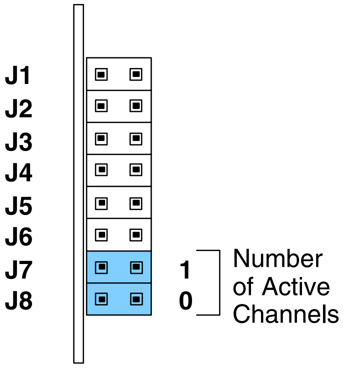
The DIP switch S3 on the D4-04AD corresponds to the units select jumpers on the F4-04AD. The figure on the right shows these are a subset of the jumper block, and include J5 and J6. They select the units for all four input channels simultaneously. Observe the present switch setting on the D4-04AD. Then find the row in the following table that matches, and configure the F4-04AD jumpers to match.

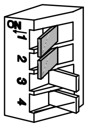
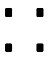
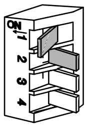
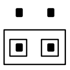
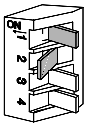
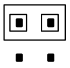
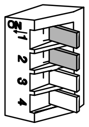
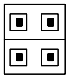

J1
J2
J3
J4
J5
J6
J7
J8
1
0
Units
Select
Jumpers

Units Select Cross-Reference		
Units or Format	**D4-04AD Switch Setting**	**F4-04AD Jumper Settings**
Standard Binary	 SW3=ON	 1 0
Two's Complement	 SW3=OFF	 1 0

Step 4: Number of Active Channels Selection

The DIP switches S1 and S2 on the D4-04AD correspond to the number of active channels jumpers on the F4-04AD. The figure on the right shows that these are a subset of the jumper block, and include J7 and J8. Observe the present switch setting on the D4-04AD. Then find the row in the table below that matches, and configure the F4-04AD jumpers to match.



Number of Active Channels Cross-Reference		
Channels Enabled	**D4-04AD Switch Settings**	**F4-04AD Jumper Settings**
Channel 1	 SW1=ON SW2=ON	 1 0
Channels 1 and 2	 SW1=ON SW2=OFF	 1 0
Channels 1, 2 and 3	 SW1=OFF SW2=ON	 1 0
Channels 1 ,2 ,3 and 4	 SW1=OFF SW2=OFF	 1 0

Now you have all the necessary information to get your analog module installed and operating correctly.

4

F4-04ADS 4-Channel Isolated Analog Input

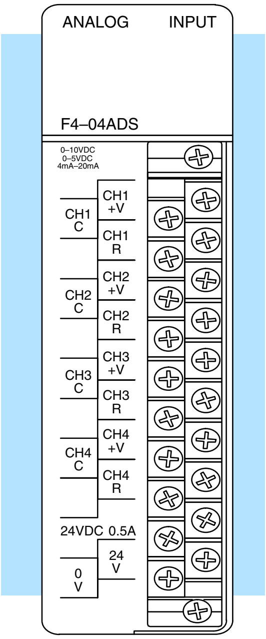
In This Chapter. . . .

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

Module Specifications

The F4-04ADS 4-Channel Isolated Analog Input module provides several features and benefits.

- It accepts four differential voltage or current inputs.
- Inputs have channel-to-channel isolation.
- Analog inputs are also 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.


ANALOG INPUT
F4–04ADS
0–10VDC
0–5VDC
4mA–20mA
CH1 +V
CH1 C
CH1 R
CH2 +V
CH2 C
CH2 R
CH3 +V
CH3 C
CH3 R
CH4 +V
CH4 C
CH4 R
24VDC 0.5A
24 V
0 V

Analog Input Configuration Requirements

The F4–04ADS Analog Input module requires 16 discrete input points from the CPU. The module can be installed in any slot of a DL405 system, including remote bases. The limitations on the number of analog modules are:

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

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

The following tables provide the specifications for the F4-04ADS Analog Input Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

Number of Channels	4
Input Ranges	0–5V, 0–10V, 1–5V, ± 5V, ± 10V, 0–20 mA, 4–20 mA
Resolution	12 bit (1 in 4096)
Conversion Method	Successive approximation
Input Type	Differential
Max. Common Mode Voltage	± 750V peak continuous transformer isolation
Noise Rejection Ratio	Common mode: –100 dB at 60Hz
Active Low-Pass Filtering	–3 dB at 20Hz, –12 dB per octave
Input Impedance	250Ω ± 0.1%, 1/2W current input 200KΩ voltage input
Absolute Maximum Ratings	± 45 mA, current input ± 100V, voltage input
Conversion Time	1 mS per selected channel
Linearity Error	± 1 count (0.025% of full scale) maximum
Full Scale Calibration Error	± 8 counts maximum (V_{in} = 20 mA)
Offset Calibration Error	± 8 counts maximum (V_{in} = 4 mA)

General Specifications

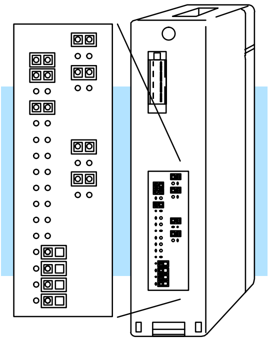
PLC Update Rate	4 channel per scan max.
Digital Input Points Required	12 binary data bits, 4 active channel indicator bits
Accuracy vs. Temperature	± 100 ppm / °C maximum full scale (including maximum offset)
Power Budget Requirement	270 mA @ 5 VDC (from base)
External Power Supply	24 VDC, ± 10%, 120 mA, class 2
Recommended Fuse	0.032 A, Series 217 fast-acting, current inputs
Operating Temperature	0 to 60°C (32 to 140°F)
Storage Temperature	–20 to 70°C (–4 to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

Setting the Module Jumpers

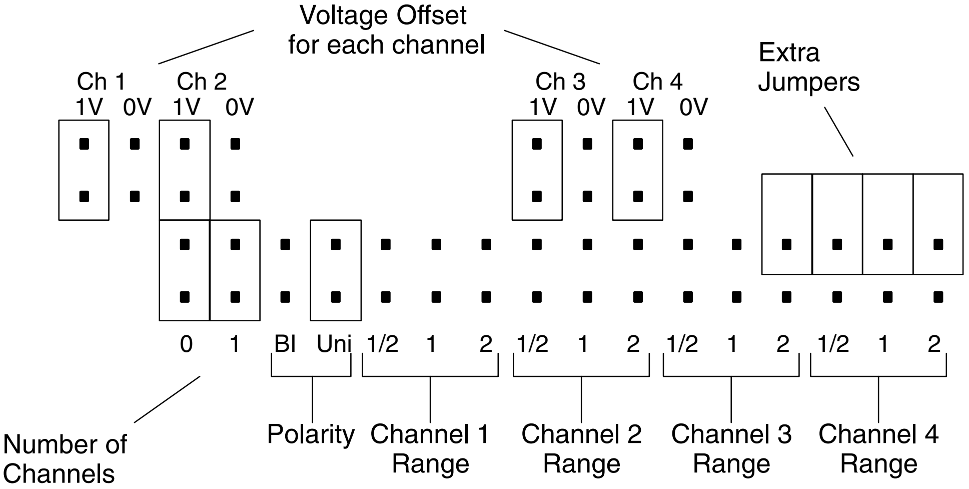
Jumper Locations

The module has several options that you can select by installing or removing jumpers. At the rear of the module are three banks of jumpers:

- One bank of 16 jumpers, which may be configured to select the number of channels enabled, channel range (for channels 1–4), and polarity.
- Two banks of four jumpers; one bank to set the offset voltage for channels 1 and 2, and the other bank to set the offset voltage for channels 3 and 4.



Also included are four additional jumpers to use as needed; each jumper is stored over a single pin on the Channel 3 and Channel 4 ranges (this is a good way to store unused jumpers so they do not get lost).

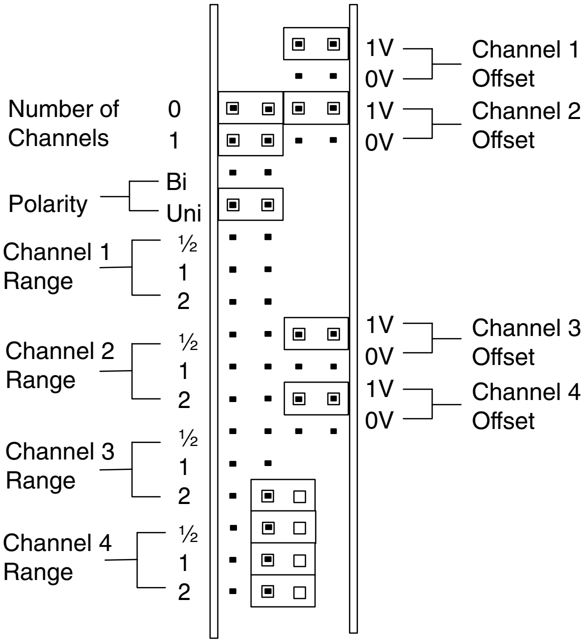

Voltage Offset for each channel
Ch 1
1V
0V
Ch 2
1V
0V
Ch 3
1V
0V
Ch 4
1V
0V
Extra Jumpers
0
1
BI
Uni
1/2
1
2
1/2
1
2
1/2
1
2
1/2
1
2
Number of Channels
Polarity
Channel 1 Range
Channel 2 Range
Channel 3 Range
Channel 4 Range

Factory Default Settings

By default, the module arrives from the factory with the jumpers installed or removed as shown here.

With these jumper settings the module is setup as follows:

- With four active channels.
- With each channel set to 1V signal offset.
- With Unipolar polarity mode (this setting will apply to all active channels).
- With 4–20mA signal range for each channel.

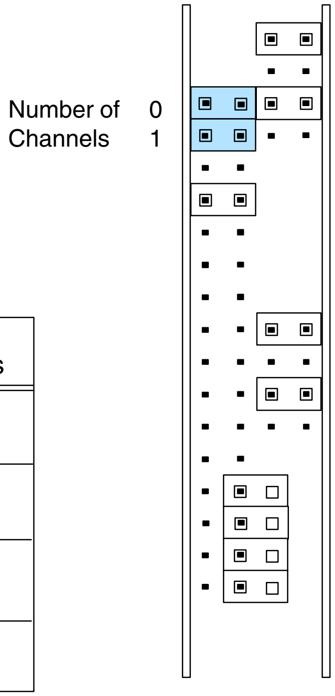

Number of Channels
0
1
Polarity
Bi
Uni
Channel 1 Range
½
1
2
Channel 2 Range
½
1
2
Channel 3 Range
½
1
2
Channel 4 Range
½
1
2
1V
0V
Channel 1 Offset
1V
0V
Channel 2 Offset
1V
0V
Channel 3 Offset
1V
0V
Channel 4 Offset

Selecting the Number of Channels

The jumpers labeled 0 and 1 are used to select the number of channels that will be used. The module is set from the factory for four-channel operation.

Any unused channels are not processed. For example, if you only select channels 1 thru 3, channel 4 will not be active. Use the following table to set the jumpers for your application.

Channels Selected	Jumper Settings
Channels 1	▪ ▪ 0 ▪ ▪ 1
Channels 1 and 2	▣ ▣ 0 ▪ ▪ 1
Channels 1, 2 and 3	▪ ▪ 0 ▣ ▣ 1
Channels 1, 2, 3 and 4	▣ ▣ 0 ▣ ▣ 1

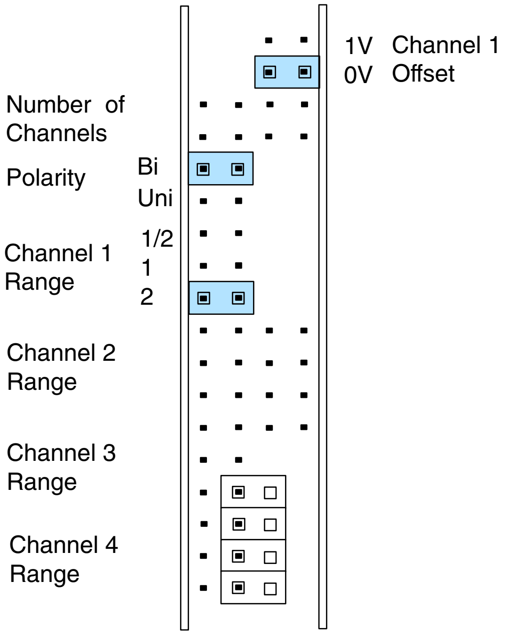

Number of Channels
0
1

Example Settings

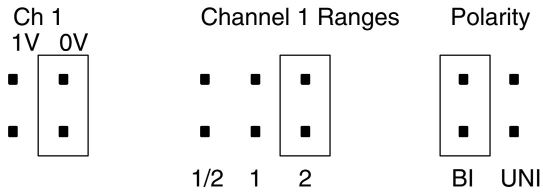
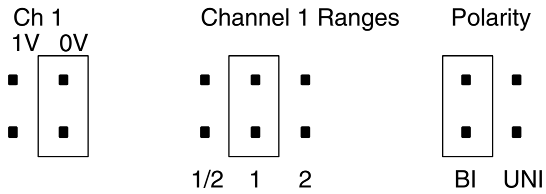
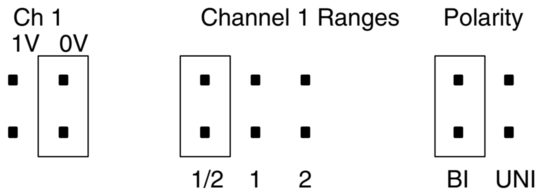
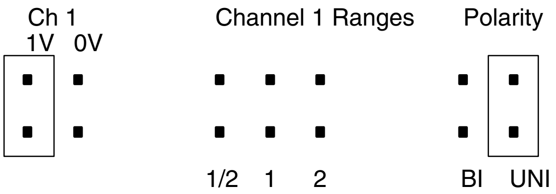
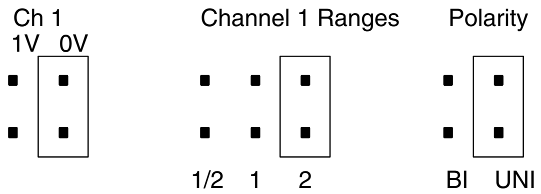
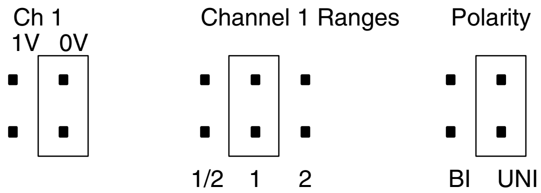
Once you select the number of channels, you must set the other parameters. Use this example to see how to set the jumpers. The example only shows settings for channel 1 operation, but the procedure is the same for the other channels.

An explanation of the example settings is as follows:

- *Number of Channels:* Both jumpers are removed for one-channel operation.
- *Polarity:* The jumper is set for Bipolar (**Bi**) signal range (**Uni** is the setting for unipolar range).
- *Channel 1 Offset:* The jumper is set for 0V offset.
- *Channel 1 Range:* The jumper is set to “2”, which is ±2.5 VDC (±10mA) when Bipolar signal range is selected (see the tables on the following page for more information).


1V
0V
Channel 1 Offset
Number of Channels
Polarity
Bi
Uni
Channel 1 Range
1/2
1
2
Channel 2 Range
Channel 3 Range
Channel 4 Range

The following tables show the jumper selections for the various ranges. Only channel 1 is used in the example, but all the channels must be set. You can have a combination of offsets and ranges but not polarities for each of the channels. For example, if the polarity is set for unipolar signal range, this setting will apply to all active channels.

Bipolar Signal Range	**Jumper Settings**
±2.5 VDC (±10 mA)	 Ch 1 1V 0V Channel 1 Ranges 1/2 1 2 Polarity BI UNI
±5 VD (±20 mA)	 Ch 1 1V 0V Channel 1 Ranges 1/2 1 2 Polarity BI UNI
±10 VDC	 Ch 1 1V 0V Channel 1 Ranges 1/2 1 2 Polarity BI UNI
Unipolar Signal Range	**Jumper Settings**
4 to 20 mA (1 VDC to 5 VDC)	 Ch 1 1V 0V Channel 1 Ranges 1/2 1 2 Polarity BI UNI
0 VDC to +5 VDC (0 to +20 mA)	 Ch 1 1V 0V Channel 1 Ranges 1/2 1 2 Polarity BI UNI
0 VDC to +10 VDC	 Ch 1 1V 0V Channel 1 Ranges 1/2 1 2 Polarity BI UNI

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

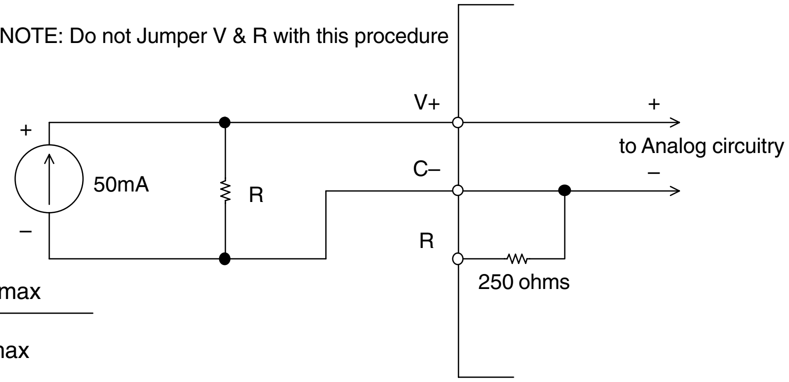
User Power Supply Requirements

The F4-04ADS requires a separate power supply. The DL430/440/450 CPU's, D4-RS Remote I/O Controller, and D4-EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. If you only have a few analog modules, you can use this power source instead of a separate supply. If you have already used the available current from this source, or if you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ±10%, Class 2, 120mA current.

Custom Input Ranges

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

NOTE: Do not Jumper V & R with this procedure



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

R = value of external resistor

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

I_{max} = maximum current supplied by the transmitter

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

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

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

Current Loop Transmitter Impedance

Standard 4 to 20 mA transmitters and transducers can operate from a wide variety of power supplies. Not all transmitters are alike and the manufacturers often specify a minimum loop or load resistance that must be used with the transmitter.

The F4-04ADS provides 250 ohms resistance for each channel. If your transmitter requires a load resistance below 250 ohms, then you do not have to make any adjustments. However, if your transmitter requires a load resistance higher than 250 ohms, then you need to add a resistor in series with the module.

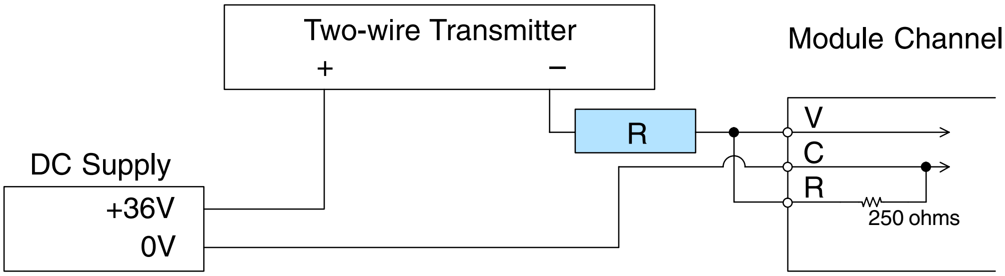
Consider the following example for a transmitter being operated from a 36 VDC supply with a recommended load resistance of 750 ohms. Since the module has a 250 ohm resistor, you need to add an additional resistor.

$R = Tr - Mr$

$R = 750 - 250$

$R \geq 500$

R – resistor to add
Tr – Transmitter Requirement
Mr – Module resistance (internal 250 ohms)


Two-wire Transmitter
+
–
Module Channel 1
R
V
C
R
250 ohms
DC Supply
+36V
0V

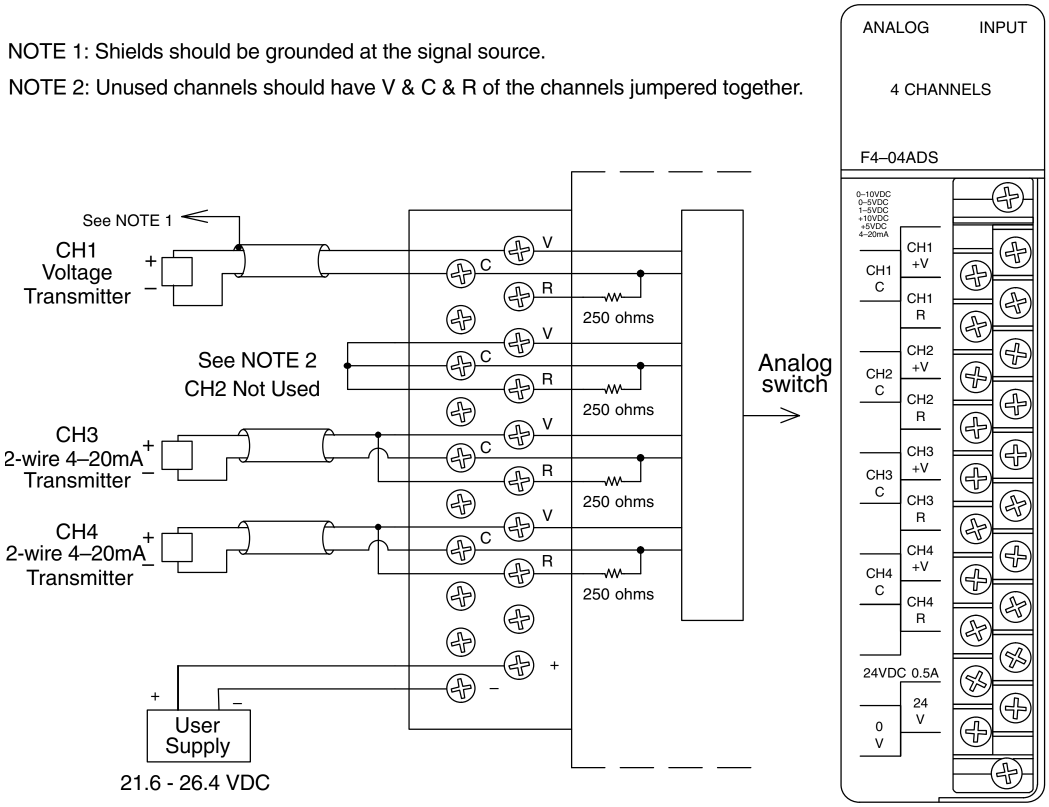
Removable Connector

The F4-04ADS module has a removeable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

Wiring Diagram

NOTE 1: Shields should be grounded at the signal source.

NOTE 2: Unused channels should have V & C & R of the channels jumpered together.

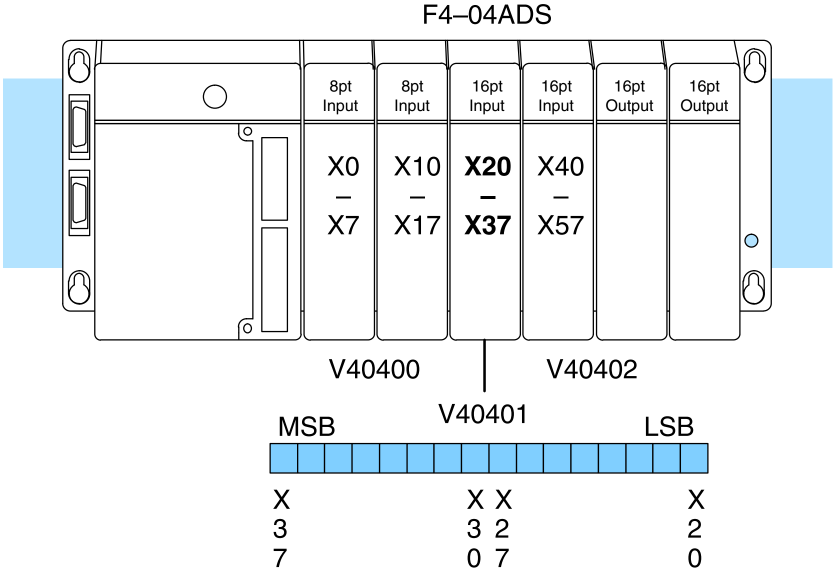

See NOTE 1
CH1
Voltage
Transmitter
+
–
See NOTE 2
CH2 Not Used
CH3
2-wire 4–20mA
Transmitter
CH4
2-wire 4–20mA
Transmitter
V
C
R
250 ohms
Analog
switch
User
Supply
21.6 - 26.4 VDC
ANALOG
INPUT
4 CHANNELS
F4–04ADS
0–10VDC
0–5VDC
1–5VDC
+10VDC
+5VDC
4–20mA
CH1
+V
CH1
C
CH1
R
CH2
+V
CH2
C
CH2
R
CH3
+V
CH3
C
CH3
R
CH4
+V
CH4
C
CH4
R
24VDC 0.5A
24
V
0
V

Module Operation

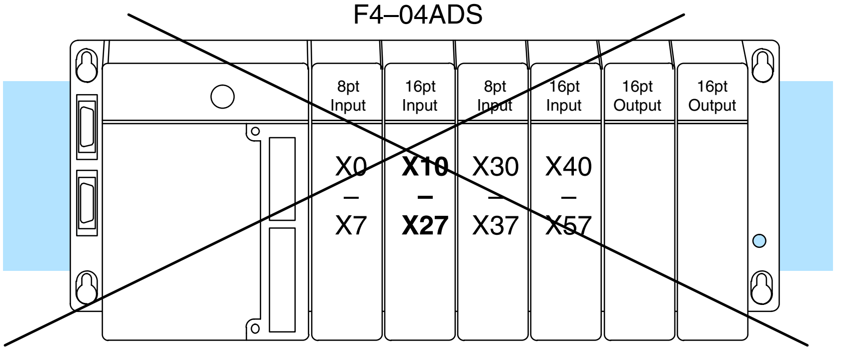
DL430 Special Requirements

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 can see in the section on writing the program, you use V-memory locations to extract the analog data. As shown in the following diagram, if you place the module so that the input points do not start on a V-memory boundary, the instructions cannot access the data.

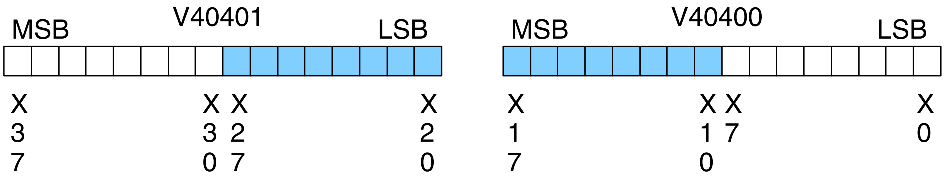
Correct!


F4–04ADS
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40401
V40402
MSB
LSB
X37
X30
X27
X20

Wrong!


F4–04ADS
8pt Input
16pt Input
8pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X27
X30 – X37
X40 – X57

Data is split over two locations, so instructions cannot access data from a DL430.

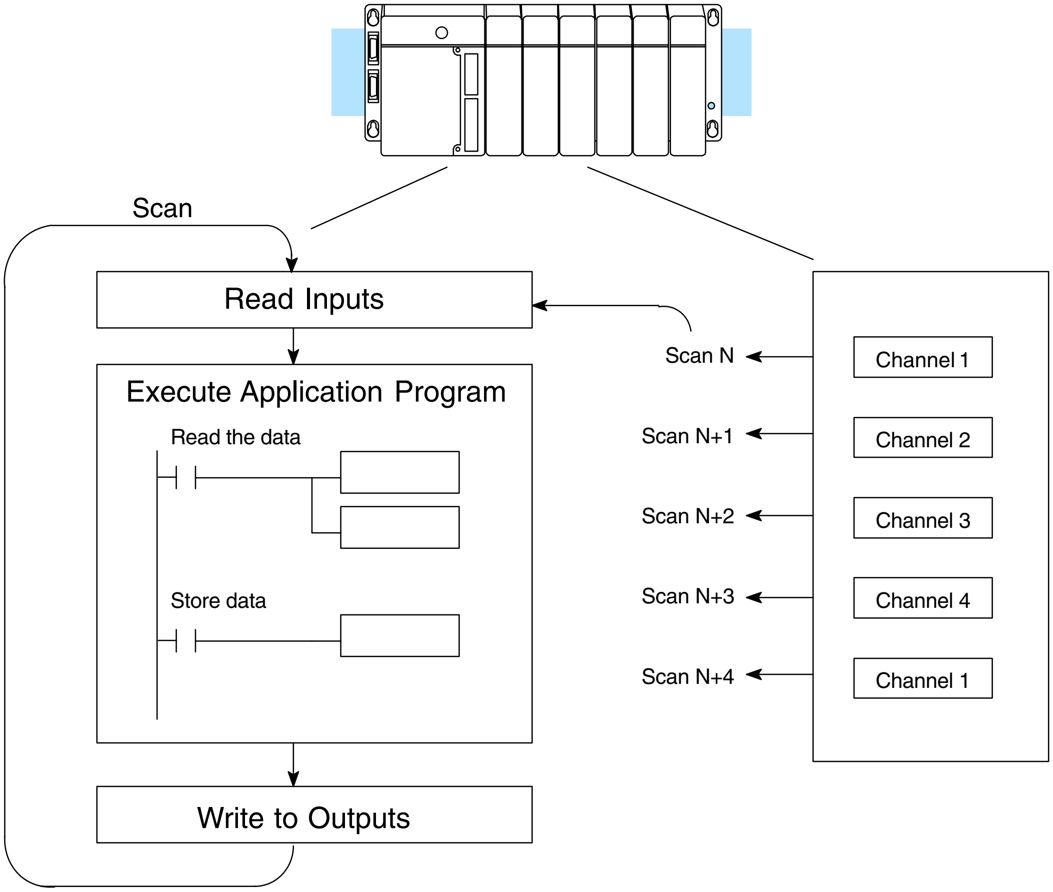

V40401
MSB
LSB
X37
X30
X27
X20
V40400
MSB
LSB
X17
X10
X7
X0

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

Channel Scanning Sequence

The F4-04ADS module supplies one channel of data per each CPU scan. Since there are four channels, it can take up to four scans to get data for all channels. Once all channels have been scanned the process starts over with channel 1.

Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan.


Scan
Read Inputs
Execute Application Program
Read the data
Store data
Write to Outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 3
Channel 4
Channel 1

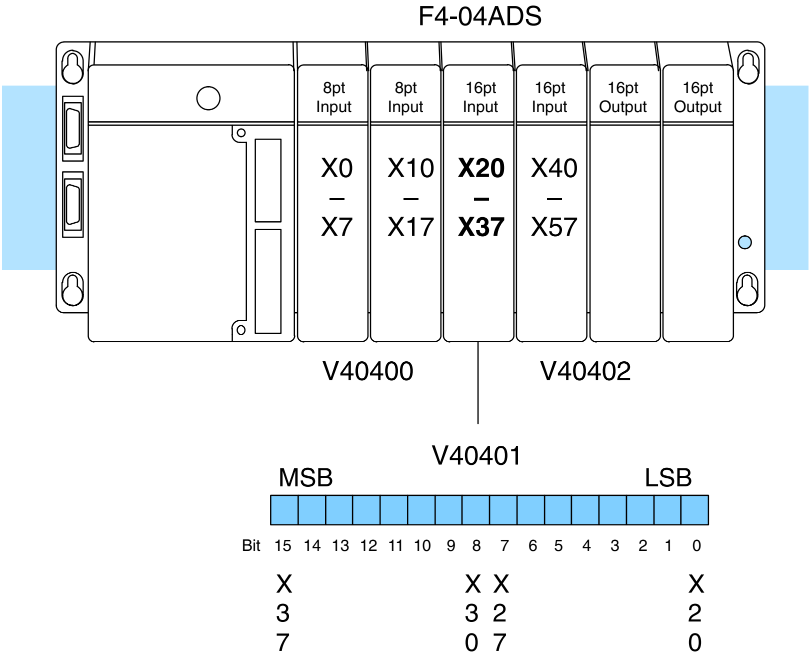
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

Input Bit Assignments

The F4-04ADS module requires 16 discrete input points from the CPU. These 16 points provide:

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

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

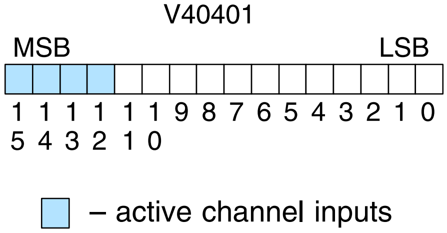

F4-04ADS
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X30
X27
X20

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

Active Channel Indicator Inputs

The last four bits (inputs) of the upper V-memory location indicate the active channel. The inputs are automatically turned on and off to indicate the current channel for each scan.

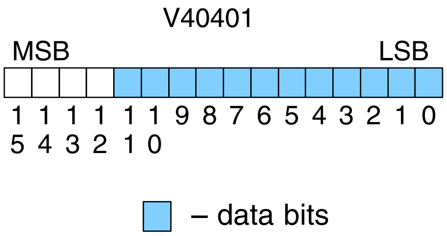
Scan	Channel Bits	Channel
N	0001	1
N+1	0010	2
N+2	0100	3
N+3	1000	4


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– active channel inputs

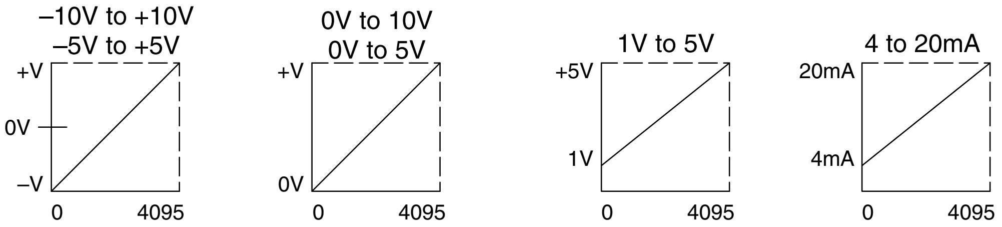
Analog Data Bits

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

Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



Since the module has 12-bit resolution, the analog signal is converted into 4096 counts ranging from 0 – 4095 (2^{12}). For example, with a 0 to 10V scale, a 0V signal would be 0, and a 10V signal would be 4095. This is equivalent to a 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 signal change that will result in a single LSB change in the data value for each signal input range.

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

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

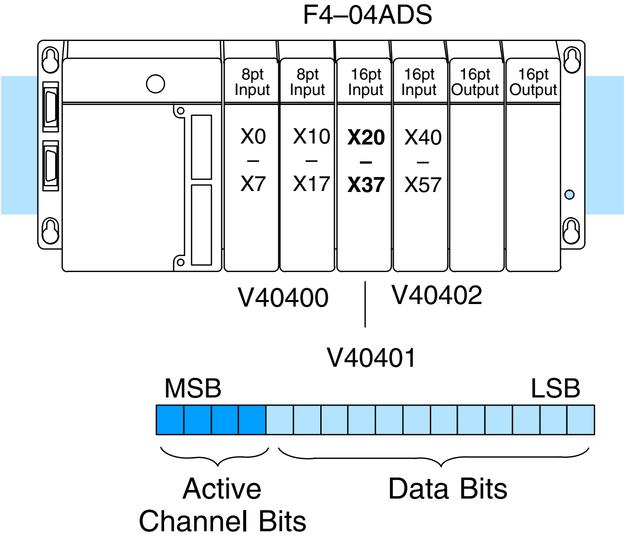
Range	Signal Span (H – L)	Divide By	Smallest Detectable Change
± 10V	20 V	4095	4.88 mV
± 5V	10 V	4095	2.44 mV
0 to 5V	5 V	4095	1.22 mV
0 to 10V	10 V	4095	2.44 mV
1 to 5V	4 V	4095	0.98 mV
4 to 20mA	16 mA	4095	3.91 µA

Writing the Control Program

Once you have configured the F4–04ADS module, use the following examples to get started writing the control program.

Multiple Channels Selected

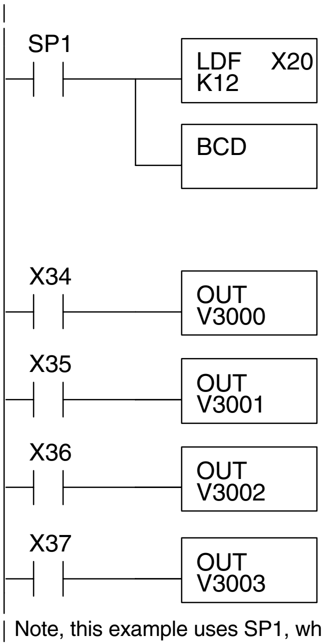
Since all channels are multiplexed into a single data word, the control program must be setup to determine which channel is being read. Since the module appears as X input points to the CPU, it is very easy to use the active channel status bits to determine which channel is being monitored.


F4–04ADS
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
Active Channel Bits
Data Bits

Reading Values, DL440/450

☒ ☑ ☑
430 440 450

This program example shows how to read the analog data into V-memory locations with DL440/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 reads one channel per scan, so it takes four scans to read all four channels.


SP1
LDF X20
K12
BCD
X34
OUT V3000
X35
OUT V3001
X36
OUT V3002
X37
OUT V3003

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

Converts the binary value in the accumulator to BCD and stores the result in the accumulator. It is usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it. Do not use 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.

When contact X34 is on, channel 1 data is being sent to the CPU. The OUT instruction moves the data from the accumulator to V3000.

When contact X35 is on, channel 2 data is stored in V3001.

When contact X36 is on, channel 3 data is stored in V3002.

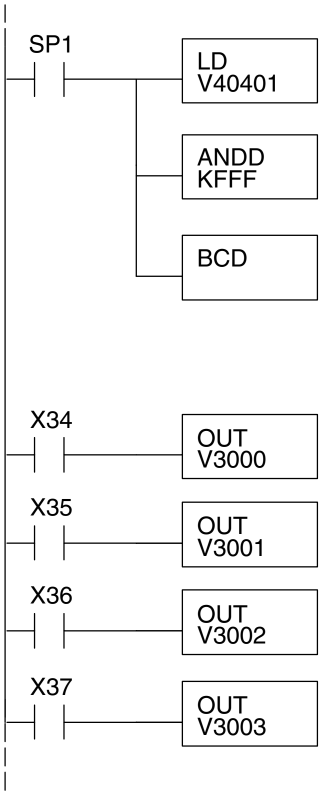
When contact X37 is on, channel 4 data is stored in V3003.

Note, this example uses SP1, which is always on and the inputs are continually being updated. You could also use an X, C, etc. permissive contact.

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 four scans to read all four channels.



Loads the complete data word into the accumulator. The V-memory location depends on the I/O configuration. See Appendix A for the memory map.

ANDs the value in the accumulator with the constant KFFF, which masks the channel identification bits, and stores the vaue in the accumulator. Without this, the values used will not be correct, so do not forget to include it.

Converts the binary value in the accumulator to BCD and stores the result in the accumulator. It is usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it. Do not use 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.

When contact X34 is on, channel 1 data is being sent to the CPU. The OUT instruction moves the data from the accumulator to V3000.

When contact X35 is on, channel 2 data is stored in V3001.

When contact X36 is on, channel 3 data is stored in V3002.

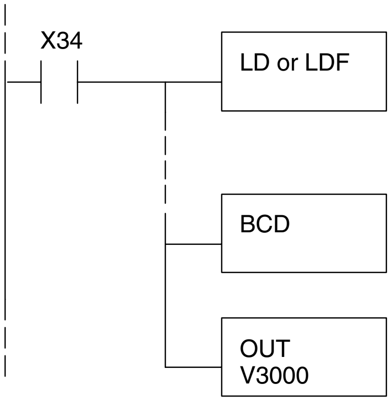
When contact X37 is on, channel 4 data is stored in V3003.

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

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.



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

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

The OUT instruction stores the data in V3000.

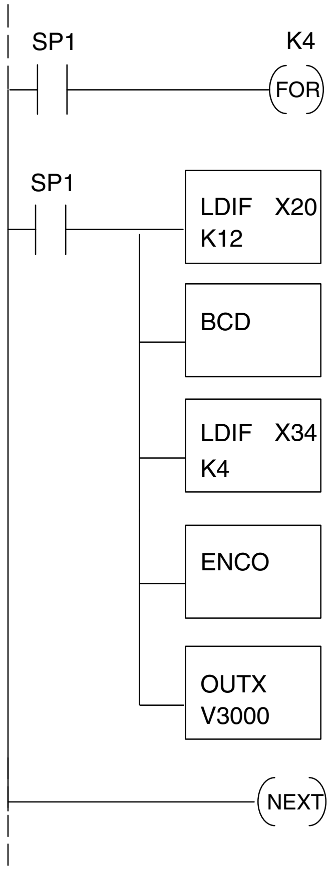
* Remember, before the BCD instruction is executed, the DL430 requires an additional instruction to mask out the first four bits that are brought in with the LD instruction. An example of how to do this using an ANDD instruction is shown in the previous section.

Reading Four Channels in One Scan, DL440/450 Only

☒ ✓ ✓
430 440 450

The following program shows you how to read all four channels in one scan by using a FOR/NEXT loop. Remember, this routine will lengthen the scan time. If you do not need to read the analog data on every scan, change the SP1 to a permissive contact (such as X input, CR, or stage bit) to only enable the 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 that reads one channel per scan.



Starts the FOR/NEXT loop. The constant (K4) specifies how many times the loop will execute, equal to the number of channels you are using. For example, enter K3 if you are using 3 channels.

Loads immediate 12 bits of the data word into the accumulator. The LDIF instruction retreives the I/O points without waiting for the CPU to finish the scan.

Changes the value in the accumulator to BCD. You can leave this out if it is not required (such as for PID loops).

Loads immediate 4 bits of the data word into the accumulator. The LDIF instruction retreives the I/O points without waiting for the CPU to finish the scan.

The ENCO instruction encodes the bit position in the accumulator having a value of 1, and returns the corresponding binary representation.

The OUTX instruction copies a 16 bit value from the accumulator to V3000.

One of the four active channel bits will be on each time through the FOR/NEXT loop, indicating the active channel. The corresponding OUT instruction places the 12 or 16-bit value in the accumulator in the proper V-memory location.

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

Scaling the Input Data

Most applications usually require measurements in engineering units, which provide more meaningful data. This is accomplished by using the conversion formula shown.

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

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

H = high limit of the EU range
L = low limit of the EU range
A = Analog value (0 – 4095)

For example, if you wanted to measure pressure (PSI) from 0.0 to 99.9 then you would have to multiply the analog value by 10 in order to imply a decimal place when you view the value with the programming software or a handheld programmer. Notice how the calculations differ when you use the multiplier.

Analog Value of 2024, slightly less than half scale, should yield 49.4 PSI

Example without multiplier	Example with multiplier
$\text{Units} = A\,\frac{H - L}{4095}$	$\text{Units} = 10\,A\,\frac{H - L}{4095}$
$\text{Units} = 2024\,\frac{100 - 0}{4095}$	$\text{Units} = 20240\,\frac{100 - 0}{4095}$
$\text{Units} = 49$	$\text{Units} = 494$

Handheld Display

```
     V 3101  V 3100
V MON   0000     0049
```

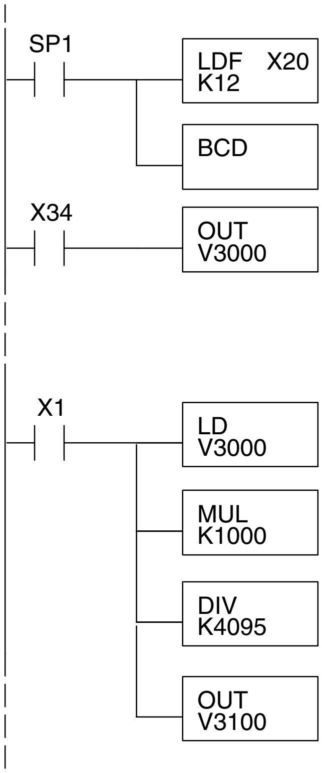
Handheld Display

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

This value is more accurate

Here is how you would write the program to perform the engineering unit conversion.

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



Loads the data word into the accumulator. The X address depends on the I/O configuration.

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

When X34 is on, channel 1 data is being sent to the CPU. The OUT instruction moves the data from the accumulator to V3000.

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

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

Divides the accumulator contents by 4095.

Stores the converted result in V3100.

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the signal levels and the digital values. This is especially useful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the signal level ...
±10V	$A = \frac{20D}{4095} - 10$	$D = \frac{4095}{20}(A + 10)$
±5V	$A = \frac{10D}{4095} - 5$	$D = \frac{4095}{10}(A + 5)$
0 to 5V	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}(A)$
0 to 10V	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}(A)$
1 to 5V	$A = \frac{4D}{4095} + 1$	$D = \frac{4095}{4}(A - 1)$
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

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

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

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

$$D = (204.75)\,(16)$$

$$D = 3276$$

5

F4-08AD 8-Channel Analog Input

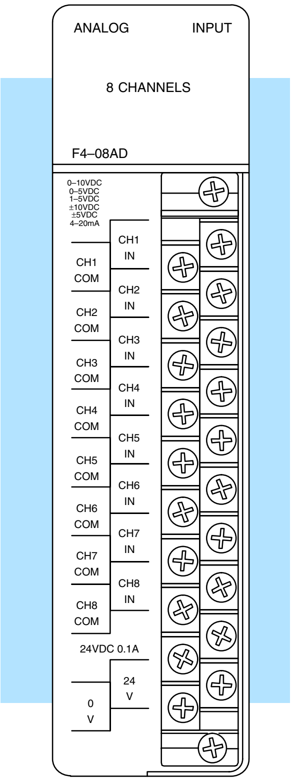
In This Chapter. . . .

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

Module Specifications

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

- It accepts eight single-ended voltage or current inputs.
- Analog inputs are optically isolated from PLC logic components.
- The module has a removable terminal block, so the module can be easily removed or changed without disconnecting the wiring.
- All eight analog inputs may be read in one CPU scan (DL440 and DL450 CPUs only).


ANALOG INPUT
8 CHANNELS
F4–08AD
0–10VDC
0–5VDC
1–5VDC
±10VDC
±5VDC
4–20mA
CH1 IN
CH1 COM
CH2 IN
CH2 COM
CH3 IN
CH3 COM
CH4 IN
CH4 COM
CH5 IN
CH5 COM
CH6 IN
CH6 COM
CH7 IN
CH7 COM
CH8 IN
CH8 COM
24VDC 0.1A
24 V
0 V

Analog Input Configuration Requirements

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

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

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

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

Input Specifications

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

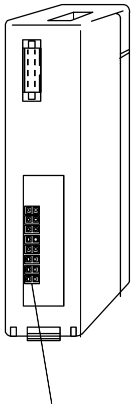
General Specifications

PLC Update Rate	8 Channel per scan max.
Digital Input Points Required	16 (X) input points total 12 binary data bits, 3 active channel bits,
Power Budget Requirement	75 mA (power from base)
External Power Supply	18–30 VDC, 120 mA, class 2
Recommended Fuse	0.032 A, Series 217 fast-acting, current inputs
Accuracy vs. Temperature	±50 ppm / °C maximum full scale (including maximum offset change of 2 counts)
Operating Temperature	0 to 60°C (32 to 140° F)
Storage Temperature	–20 to 70°C (–4° F to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental Air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

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

Setting the Module Jumpers

Jumper Locations

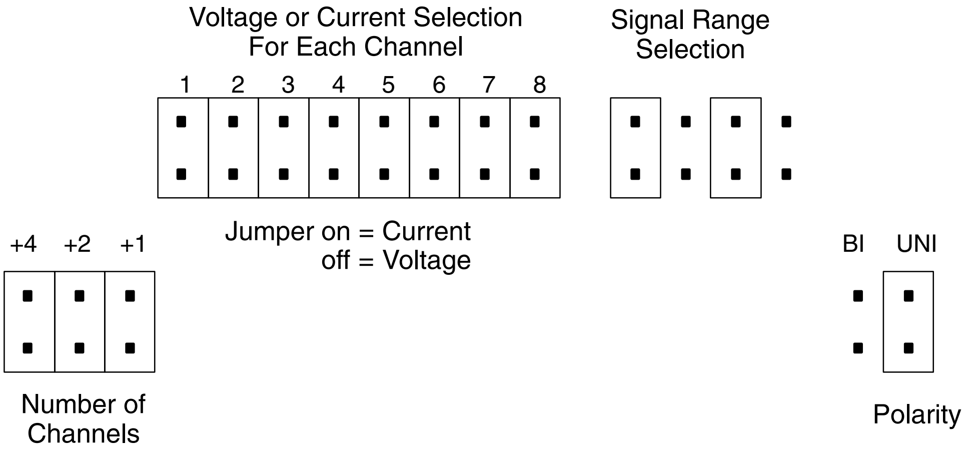


Jumper Locations

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

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

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


Voltage or Current Selection For Each Channel
1
2
3
4
5
6
7
8
Signal Range Selection
Jumper on = Current
off = Voltage
+4
+2
+1
Number of Channels
BI
UNI
Polarity

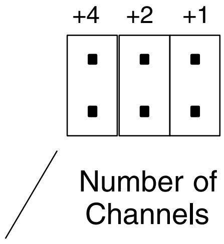
Selecting the Number of Channels

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

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

Yes = jumper installed
No = jumper removed

Number of Channels Selected	+4	+2	+1
1	No	No	No
2	No	No	Yes
3	No	Yes	No
4	No	Yes	Yes
5	Yes	No	No
6	Yes	No	Yes
7	Yes	Yes	No
8	Yes	Yes	Yes


+4
+2
+1
Number of Channels

Jumpers installed as shown selects 8-channel operation.

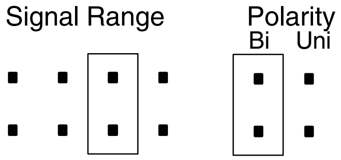
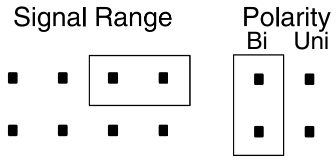
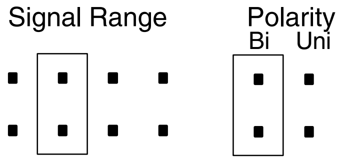
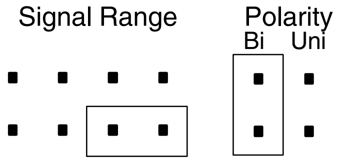
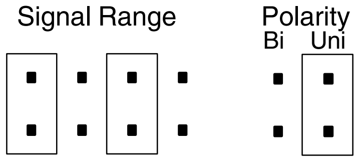
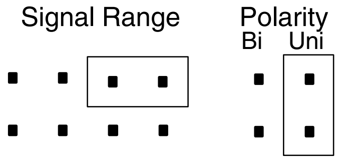
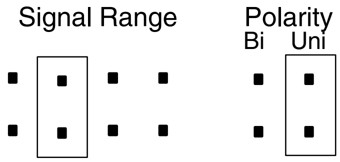
For example: To select 3-channel operation, remove the +4 and +1 jumpers and install the +2 jumper.

Selecting Current or Voltage

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

Selecting Input Signal and Ranges

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

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

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

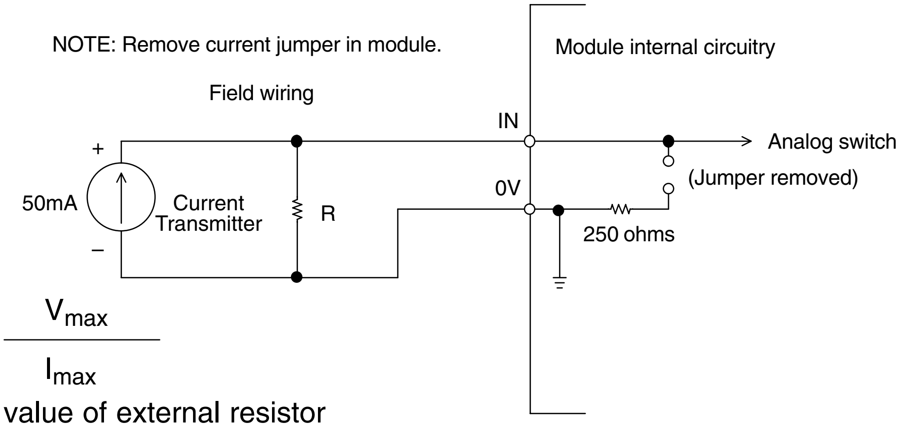
- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

User Power Supply Requirements

The F4-08AD module requires a separate power supply. The Series DL405 CPUs, D4-RS Remote I/O Controller, and D4-EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. If you only have a couple analog modules, you can use this power source instead of a separate supply. If you have more than four analog modules, or you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ±10%, Class 2, 100 mA current (per module).

Custom Input Ranges

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



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

R = value of external resistor

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

I_{max} = maximum current supplied by the transmitter

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

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

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

Current Loop Transmitter Impedance

Standard 4 to 20 mA transmitters and transducers can operate from a wide variety of power supplies. Not all transmitters are alike and the manufacturers often specify a minimum loop or load resistance that must be used with the transmitter.

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

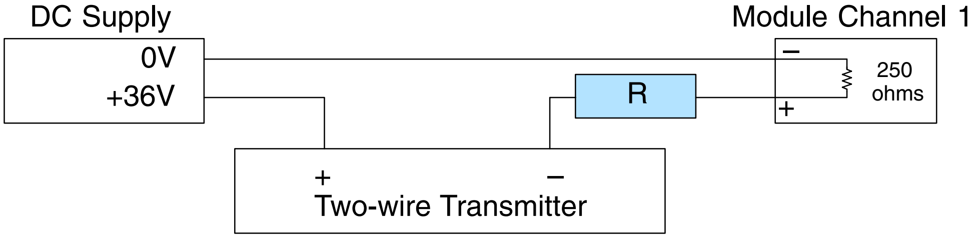
Consider the following example for a transmitter being operated from a 36 VDC supply with a recommended load resistance of 750 ohms. Since the module has a 250 ohm resistor, you need to add an additional resistor.

$R = Tr - Mr$

$R = 750 - 250$

$R \geq 500$

R – resistor to add
Tr – Transmitter Requirement
Mr – Module resistance (internal 250 ohms)

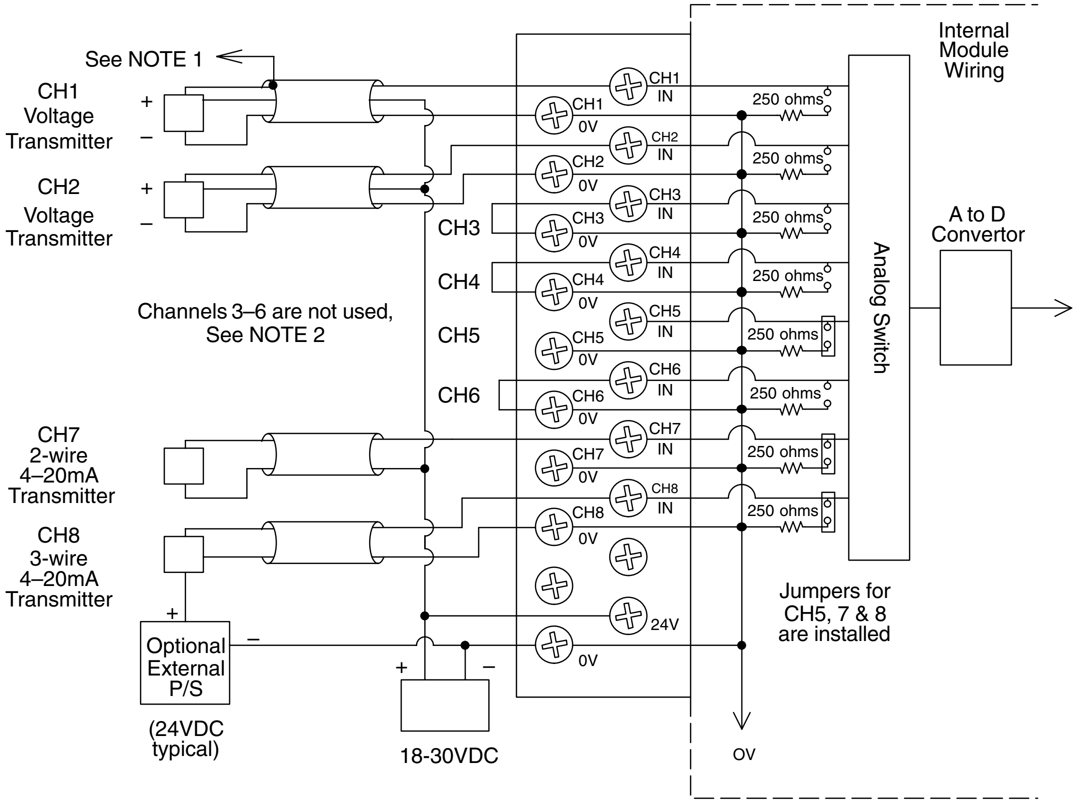


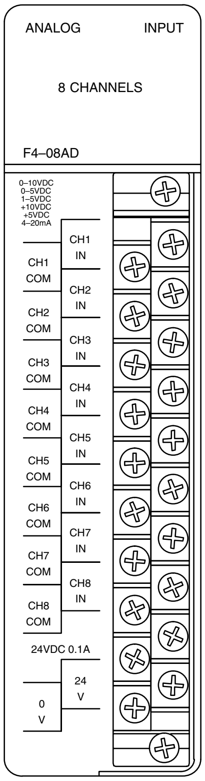
Removable Connector

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

Wiring Diagram

NOTE 1: Shields should be grounded at the signal source.
NOTE 2: Unused channels should be connected to 0V
or have current jumpers installed.


See NOTE 1
CH1 Voltage Transmitter
+
–
CH2 Voltage Transmitter
+
–
CH3
CH4
CH5
CH6
Channels 3–6 are not used, See NOTE 2
CH7 2-wire 4–20mA Transmitter
CH8 3-wire 4–20mA Transmitter
+
–
Optional External P/S
(24VDC typical)
+
–
18-30VDC
CH1 IN
CH1 0V
CH2 IN
CH2 0V
CH3 IN
CH3 0V
CH4 IN
CH4 0V
CH5 IN
CH5 0V
CH6 IN
CH6 0V
CH7 IN
CH7 0V
CH8 IN
CH8 0V
24V
0V
Internal Module Wiring
250 ohms
Analog Switch
A to D Convertor
Jumpers for CH5, 7 & 8 are installed
OV


ANALOG
INPUT
8 CHANNELS
F4–08AD
0–10VDC
0–5VDC
1–5VDC
+10VDC
+5VDC
4–20mA
CH1 IN
CH1 COM
CH2 IN
CH2 COM
CH3 IN
CH3 COM
CH4 IN
CH4 COM
CH5 IN
CH5 COM
CH6 IN
CH6 COM
CH7 IN
CH7 COM
CH8 IN
CH8 COM
24VDC 0.1A
24 V
0 V

More than one external power supply can be used (see channel 8).

If the power supply common of an external power supply is not connected to 0V on the module, then the output of the external transmitter must be isolated. To avoid “ground loop” errors, recommended 4–20mA transmitter types are:

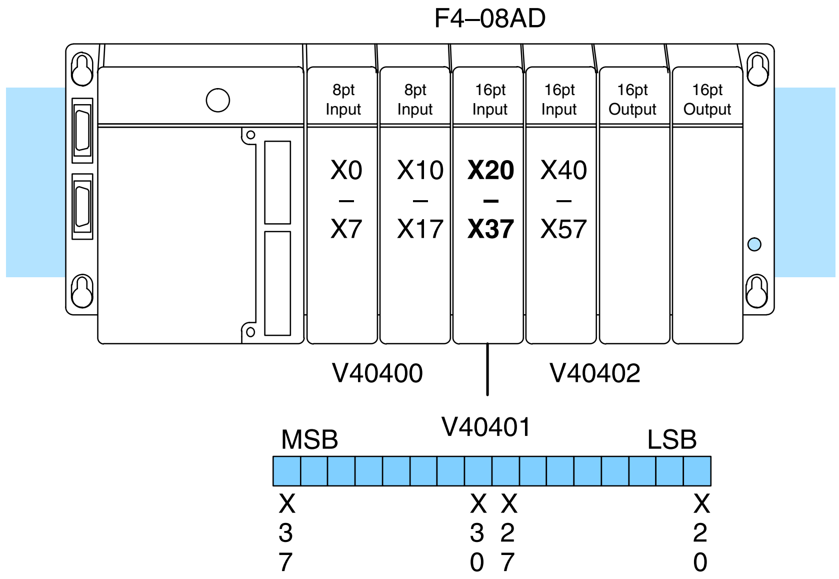
2 or 3 wire: Isolation between input signal and power supply.
4 wire: Isolation between input signal, power supply, and 4–20mA output.

Module Operation

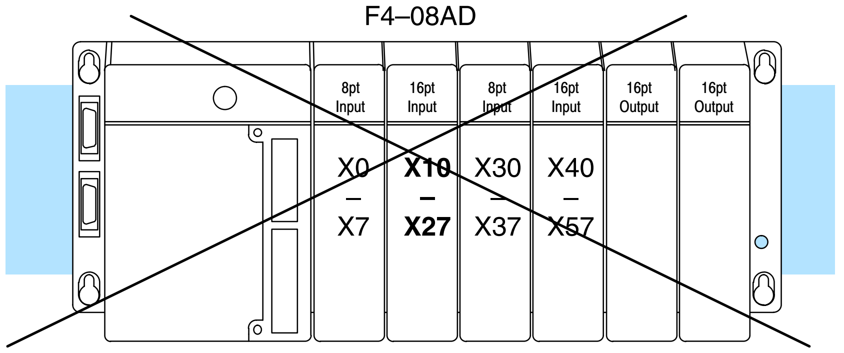
DL430 Special Requirements

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

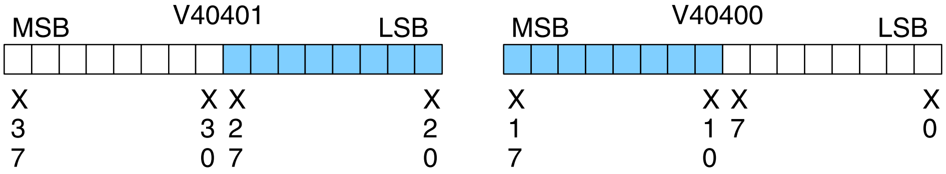
Correct!


F4–08AD
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
X37
X30
X27
X20

Wrong!


F4–08AD
8pt Input
16pt Input
8pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X27
X30 – X37
X40 – X57

Data is split over two locations, so instructions cannot access data from a DL430.

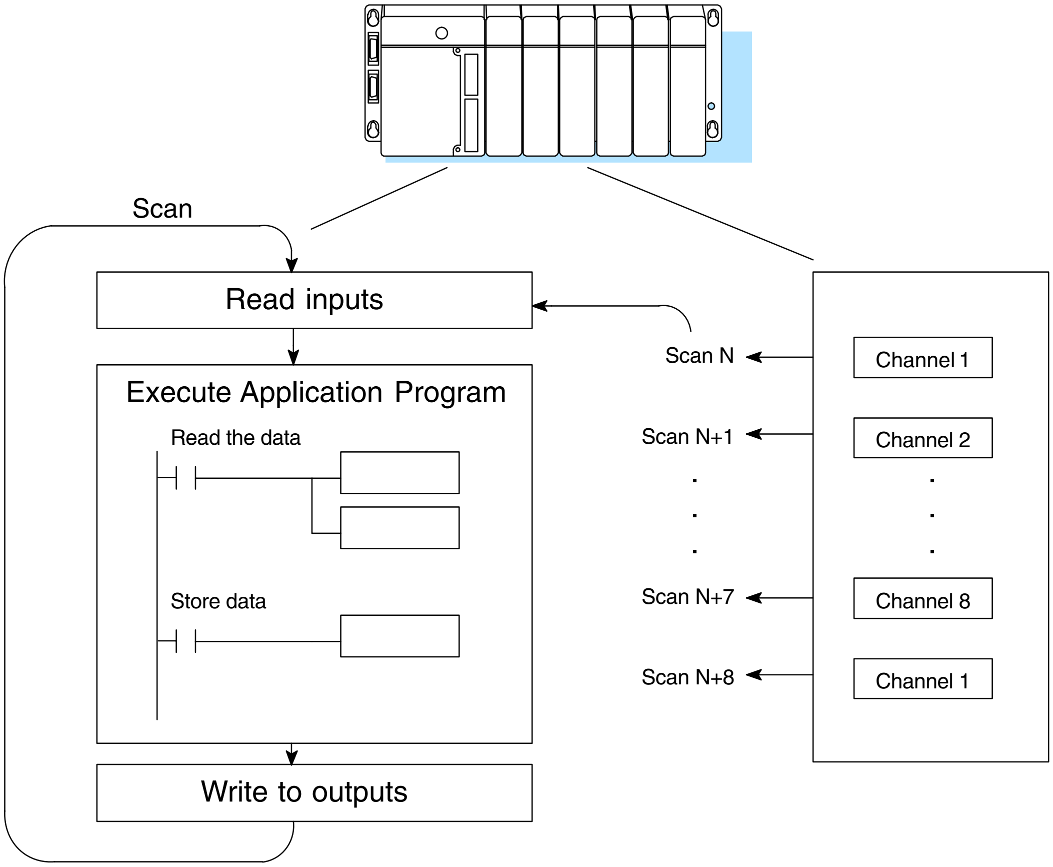

MSB
V40401
LSB
X37
X30
X27
X20
MSB
V40400
LSB
X17
X10
X7
X0

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

Channel Scanning Sequence

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

Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan.


Scan
Read inputs
Execute Application Program
Read the data
Store data
Write to outputs
Scan N
Scan N+1
Scan N+7
Scan N+8
Channel 1
Channel 2
Channel 8
Channel 1

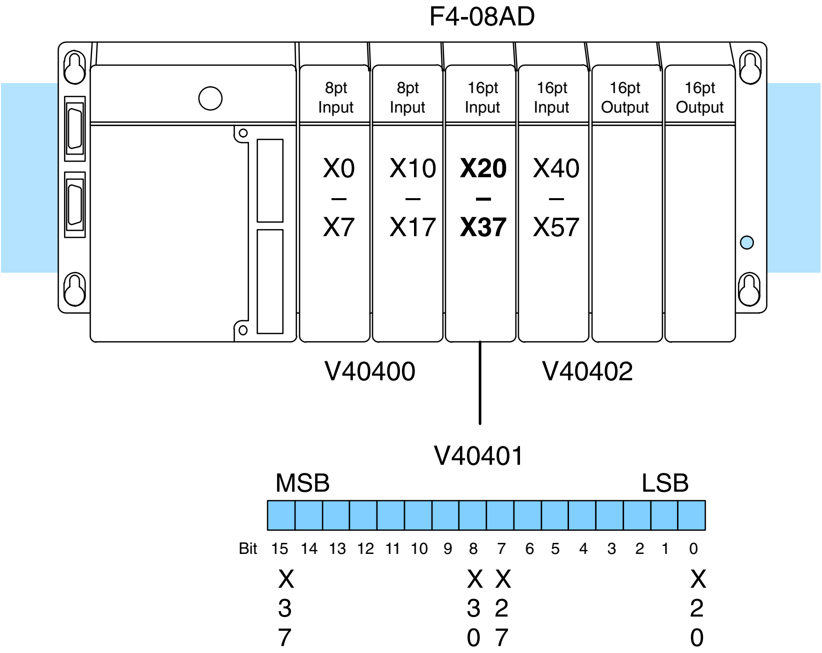
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

Input Bit Assignments

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

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

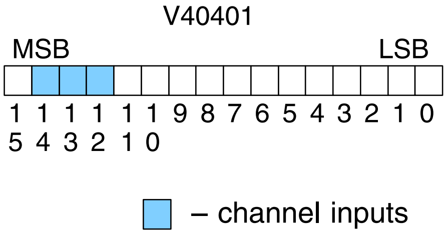
Since all input points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4-08AD
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X30
X27
X20

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

Active Channel Indicator Inputs

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

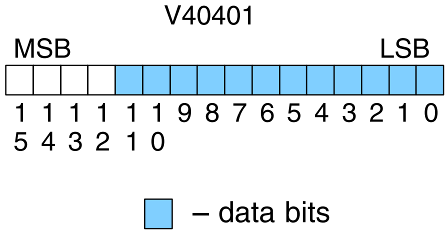

V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– channel inputs

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

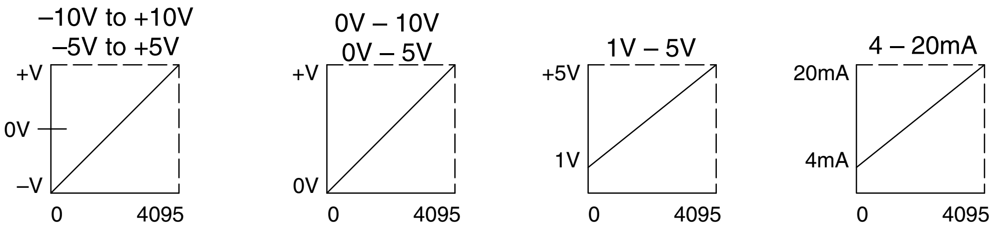
Analog Data Bits

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

Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



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



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

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

H = high limit of the signal range

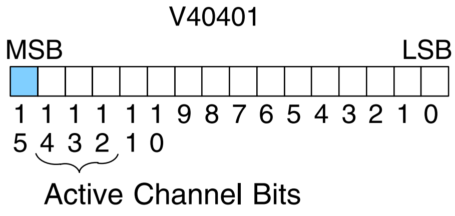
L = low limit of the signal range

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

Unusable MSB Bit

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

🟦 – unusable bit

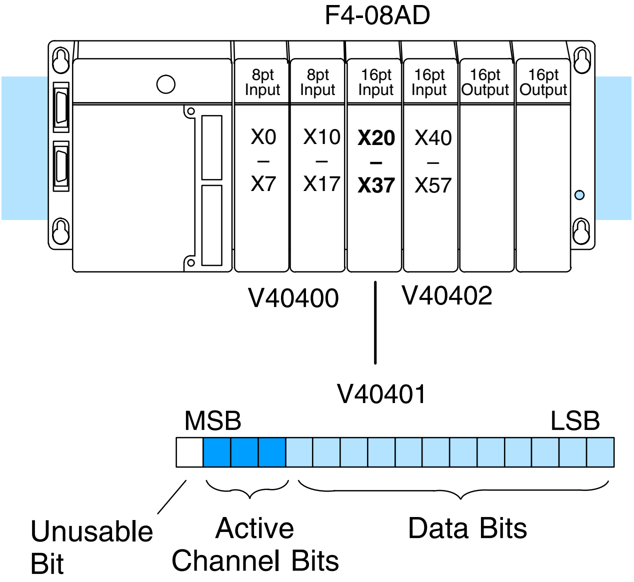


Writing the Control Program

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

Multiple Channels Selected

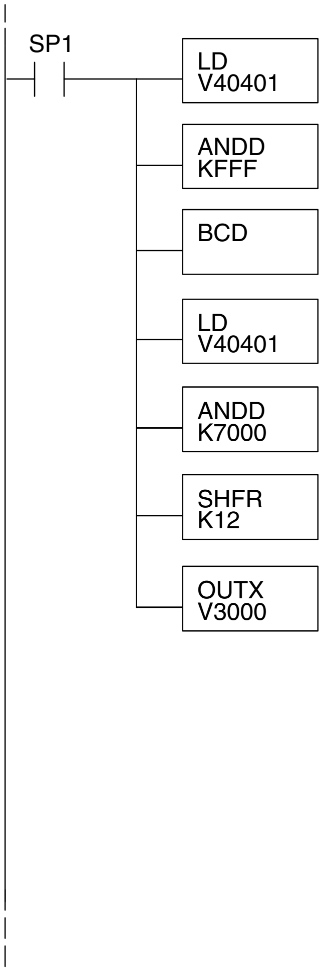
Since all channels are multiplexed into a single data word, the control program must be set up to determine which channel is being read. Since the module appears as X input points to the CPU, it is very easy to use the active channel status bits to determine which channel is being monitored.


F4-08AD
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
Unusable Bit
Active Channel Bits
Data Bits

Reading Values, DL430 CPU

✓ ✓ ✓
430 440 450

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



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.

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

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).

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

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

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 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.

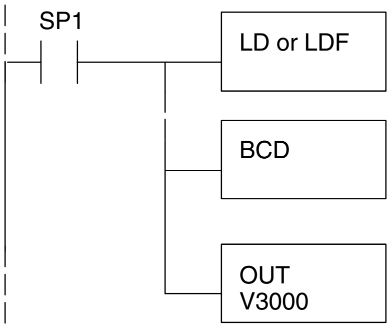
Module Reading Stored in ...	Acc. Bits	Offset	Data
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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

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.



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

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

The OUT instruction stores the data in V3000.

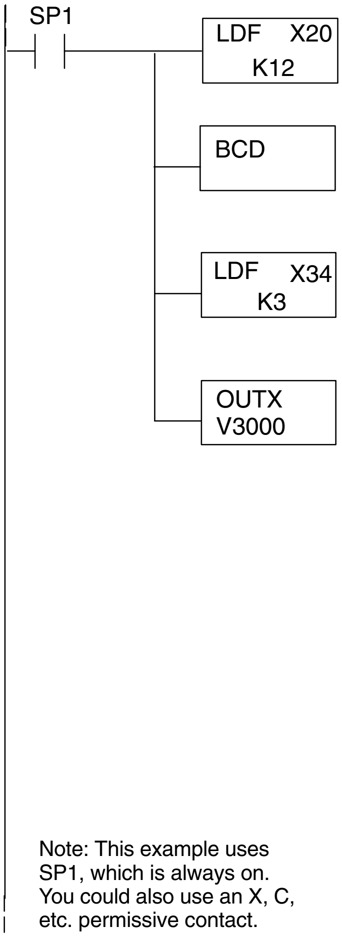
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

* 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.



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

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.

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

OUTX copies the 16 bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (which is the offset) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0) and if the binary value is 6, then the channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

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

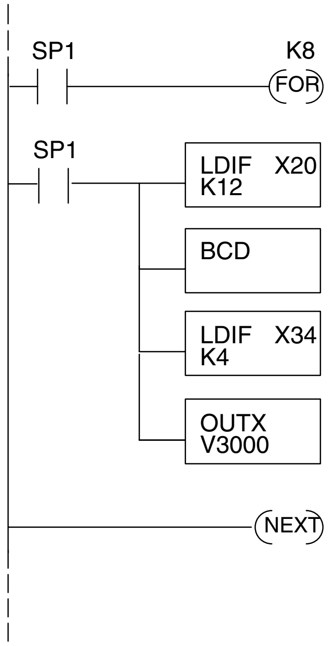
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Reading Eight Channels in One Scan, DL440/450

✗	✓	✓
430	440	450

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

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



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

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, that's why the 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. For example, if channel 3 was being read, the data would be stored in V3002 (V3000 + 2).

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.

Scaling the Input Data

Most applications usually require measurements in engineering units, which provide more meaningful data. This is accomplished by using the conversion formula shown.

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

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

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 wanted to measure pressure (PSI) from 0.0 to 99.9 then you would have to multiply the analog value by 10 in order to imply a decimal place when you view the value with the programming software or a handheld programmer. Notice how the calculations differ when you use the multiplier.

Analog Value of 2024, slightly less than half scale, should yield 49.4 PSI

Example without multiplier

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

$$\text{Units} = 2024\,\frac{100 - 0}{4095}$$

$$\text{Units} = 49$$

Handheld Display

```
V 3101  V 3100
V MON   0000    0049
```

Example with multiplier

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

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

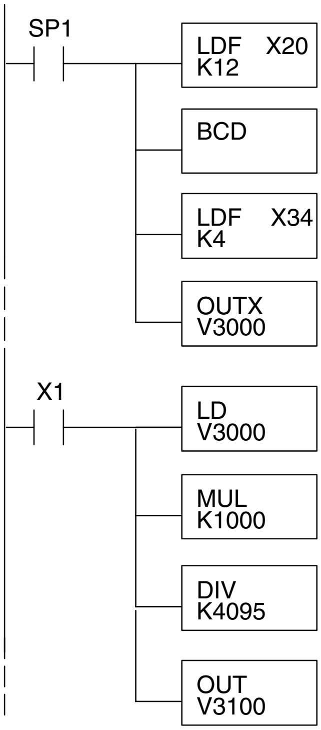
$$\text{Units} = 494$$

Handheld Display

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

*Value is more accurate

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



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

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

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

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

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

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

Divides the accumulator data by 4095.

Stores the result in location V3100.

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the signal levels and the digital values. This is especially useful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the signal level ...
–10V to + 10V	$A = \frac{20D}{4095} - 10$	$D = \frac{4095}{20}(A + 10)$
–5V to + 5V	$A = \frac{10D}{4095} - 5$	$D = \frac{4095}{10}(A + 5)$
0 to 5V	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}\ (A)$
0 to 10V	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}\ (A)$
1 to 5V	$A = \frac{4D}{4095} + 1$	$D = \frac{4095}{4}(A - 1)$
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

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

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

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

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

$$D = 3276$$

6

F4–16AD–1 16-Channel Analog Input

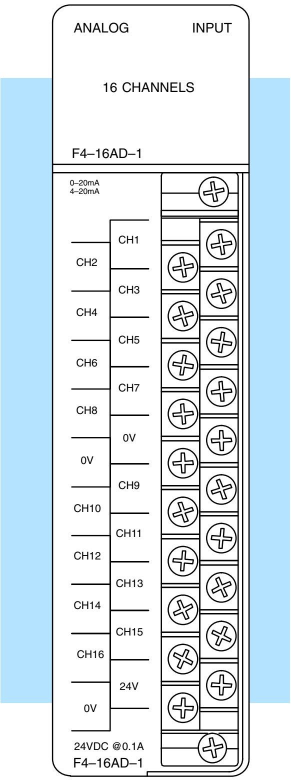
In This Chapter. . . .

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

Module Specifications

The F4-16AD–1 Analog Input module provides several features and benefits.

- It accepts 16 single-ended current inputs.
- Analog inputs are optically isolated from PLC logic components.
- The module has a removable terminal block, so the module can be easily removed or changed without disconnecting the wiring.
- All 16 analog inputs may be read in one CPU scan (D4–440 and D4–450 CPUs only).


ANALOG
INPUT
16 CHANNELS
F4–16AD–1
0–20mA
4–20mA
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
0V
0V
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
24V
0V
24VDC @0.1A
F4–16AD–1

Analog Input Configuration Requirements

The F4–16AD–1 Analog Input module requires 16 discrete input points. The module can be installed in any slot of a DL405 system, including remote bases. The limitations on the number of analog modules are:

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

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

The following table provides the specifications for the F4–16AD–1 Analog Input Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

Number of Channels	16, single ended (one common)
Input Ranges	0–20 mA, 4–20 mA
Resolution	12 bit (1 in 4096)
Active Low-pass Filtering	–3 dB at 20Hz, –6 dB per octave
Input Impedance	250 ohms ± 0.1%, 1/2W current input
Absolute Maximum Ratings	± 45 mA, current input
Conversion Time	2.0ms per channel (module conversion)
Linearity Error (End to End)	± 2 count maximum
Input Stability	± 1 count
Full Scale Calibration Error (Offset error not included)	± 12 counts maximum, @ 20mA current input
Offset Calibration Error	± 3 counts maximum, 4 mA current input

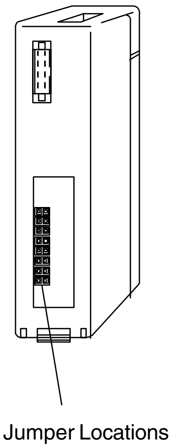
General Specifications

PLC Update Rate	16 Channel per scan max.
Digital Input Points Required	16 (X) input points total 12 binary data bits, 4 active channel bits,
Power Budget Requirement	100 mA @ 5 VDC
External Power Supply	21.6–26.4 VDC, 100 mA, class 2
Recommended Fuse	0.032 A, Series 217 fast-acting, current inputs
Accuracy vs. Temperature	± 50 ppm / °C maximum full scale (including maximum offset change of 3 counts)
Operating Temperature	0 to 60°C (32 to 140° F)
Storage Temperature	–20 to 70°C (–4° F to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental Air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

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

Setting the Module Jumpers

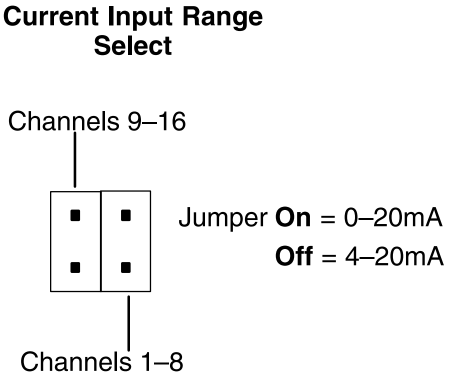
Jumper Locations

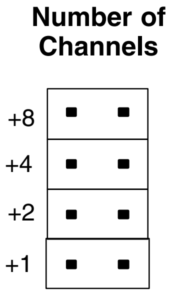

Jumper Locations

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

- A bank of two jumpers sets the current input range for the 16 channels (channels 1–8 and 9–16).
- A bank of four jumpers to select the number of channels used.

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

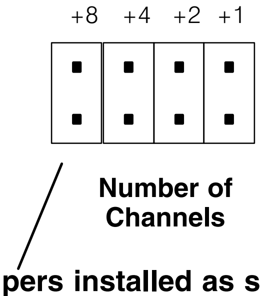

Current Input Range Select
Channels 9–16
Jumper On = 0–20mA
Off = 4–20mA
Channels 1–8


Number of Channels
+8
+4
+2
+1

Selecting the Number of Channels

The jumpers labeled +1, +2, +4 and +8 are used to select the number of channels that will be used.

Any unused channels are not processed so if you only select channels 1–8, then the last eight channels will not be active. The following table shows which jumpers to install.



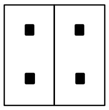
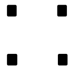
Number of Channels

Jumpers installed as shown selects 16-channel operation

Channel(s)	Jumper			
	+8	+4	+2	+1
1	No	No	No	No
1 2	No	No	No	Yes
1 2 3	No	No	Yes	No
1 2 3 4	No	No	Yes	Yes
1 2 3 4 5	No	Yes	No	No
1 2 3 4 5 6	No	Yes	No	Yes
1 2 3 4 5 6 7	No	Yes	Yes	No
1 2 3 4 5 6 7 8	No	Yes	Yes	Yes
1 2 3 4 5 6 7 8 9	Yes	No	No	No
1 2 3 4 5 6 7 8 9 10	Yes	No	No	Yes
1 2 3 4 5 6 7 8 9 10 11	Yes	No	Yes	No
1 2 3 4 5 6 7 8 9 10 11 12	Yes	No	Yes	Yes
1 2 3 4 5 6 7 8 9 10 11 12 13	Yes	Yes	No	No
1 2 3 4 5 6 7 8 9 10 11 12 13 14	Yes	Yes	No	Yes
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	Yes	Yes	Yes	No
1 2 3 4 5 6 7 8 9 10 11 12 13141516	Yes	Yes	Yes	Yes

Selecting Input Ranges

The following table shows the jumper selections for Current Input Range (0–20mA or 4–20mA). See note below.

Current Input Range	Jumper Settings
0 to 20 mA	Channels 1–8 Jumper  Channels 9–16 Jumper
4 to 20 mA	Channels 1–8 Jumper  Channels 9–16 Jumper

NOTE: The jumpers do not have to both be set On or Off (e.g. Channels 1–8 can be selected for 0–20mA and Channels 9–16 can be selected for 4–20mA operation).

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

User Power Supply Requirements

The F4–16AD–1 module requires a separate power supply. The Series DL405 CPUs, D4-RS Remote I/O Controller, and D4-EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. If you only have a couple analog modules, you can use this power source instead of a separate supply. If you have more than four analog modules, or you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ±10%, Class 2, 100 mA current (per module).

Current Loop Transmitter Impedance

Standard 4 to 20 mA transmitters and transducers can operate from a wide variety of power supplies. Not all transmitters are alike and manufacturers often specify a minimum loop or load resistance that must be used with the transmitter.

The F4–16AD–1 provides 250 ohm resistance for each channel. If your transmitter requires a load resistance below 250 ohms, then you do not have to make any adjustments. However, if your transmitter requires a load resistance higher than 250 ohms, then you need to add a resistor in series with the module.

Consider the following example for a transmitter being operated from a 36 VDC supply with a recommended load resistance of 750 ohms. Since the module has a 250 ohm resistor, you need to add an additional resistor.

$R = Tr - Mr$

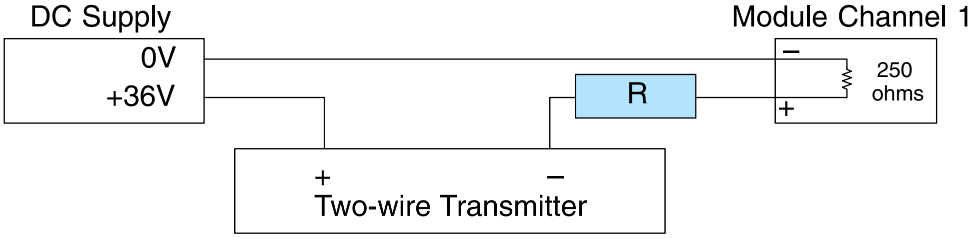
$R = 750 - 250$

$R \geq 500$

R – resistor to add

Tr – Transmitter Requirement

Mr – Module resistance (internal 250 ohms)



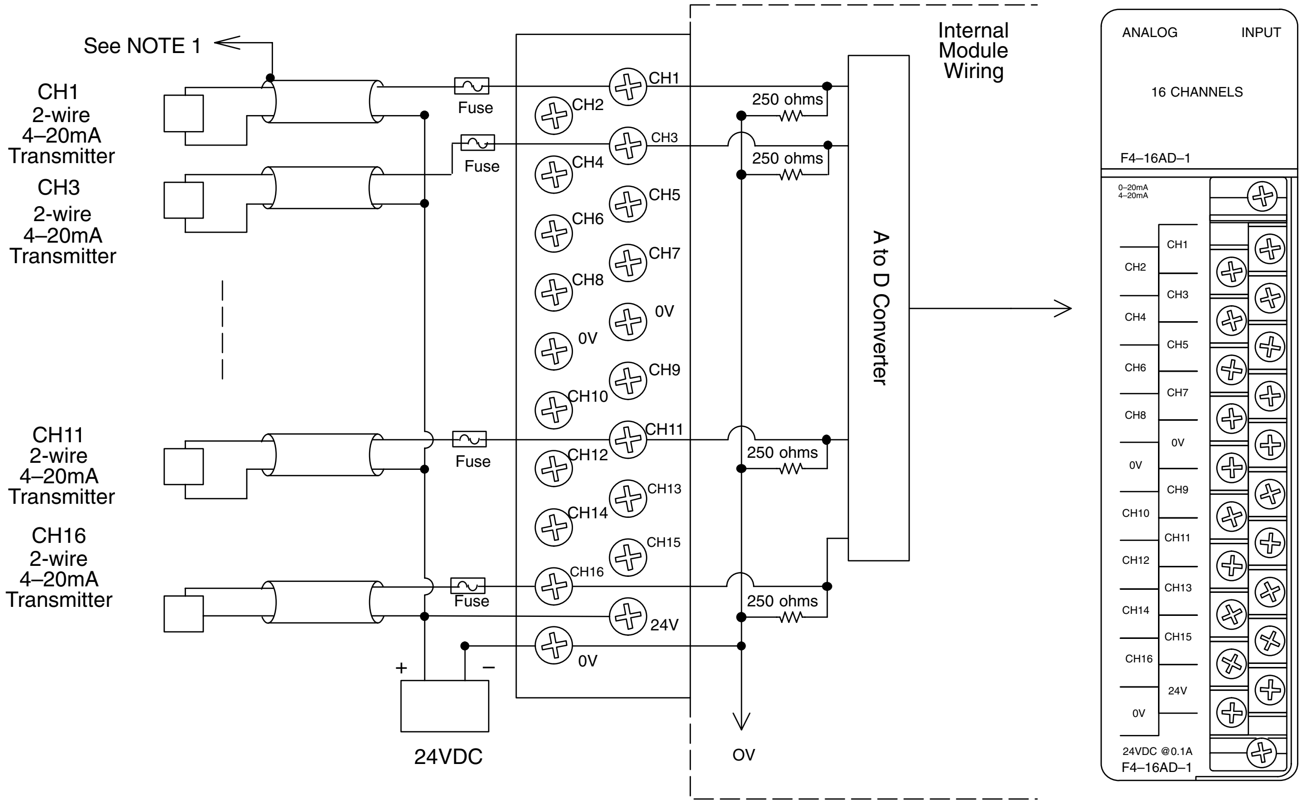
NOTE: Make sure the power dissipated in the resistor does not exceed its power rating. (P=VI) In the example above, P=36 * 0.02 = 0.72, or 3/4 Watts.

Removable Connector

The F4–16AD–1 module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

Wiring Diagram

NOTE 1: Shields should be grounded at the signal source.


See NOTE 1
CH1
2-wire
4–20mA
Transmitter
CH3
2-wire
4–20mA
Transmitter
CH11
2-wire
4–20mA
Transmitter
CH16
2-wire
4–20mA
Transmitter
Fuse
Fuse
Fuse
Fuse
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
0V
0V
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
24V
0V
+
–
24VDC
Internal
Module
Wiring
250 ohms
250 ohms
250 ohms
250 ohms
A to D Converter
OV
ANALOG
INPUT
16 CHANNELS
F4–16AD–1
0–20mA
4–20mA
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
0V
0V
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
24V
0V
24VDC @0.1A
F4–16AD–1

A series 217, 0.032A fast acting fuse is recommended for current inputs

If the power supply common of an external power supply is not connected to 0V on the module, then the output of the external transmitter must be isolated. To avoid "ground loop" errors, recommended 4–20mA transmitter types are:

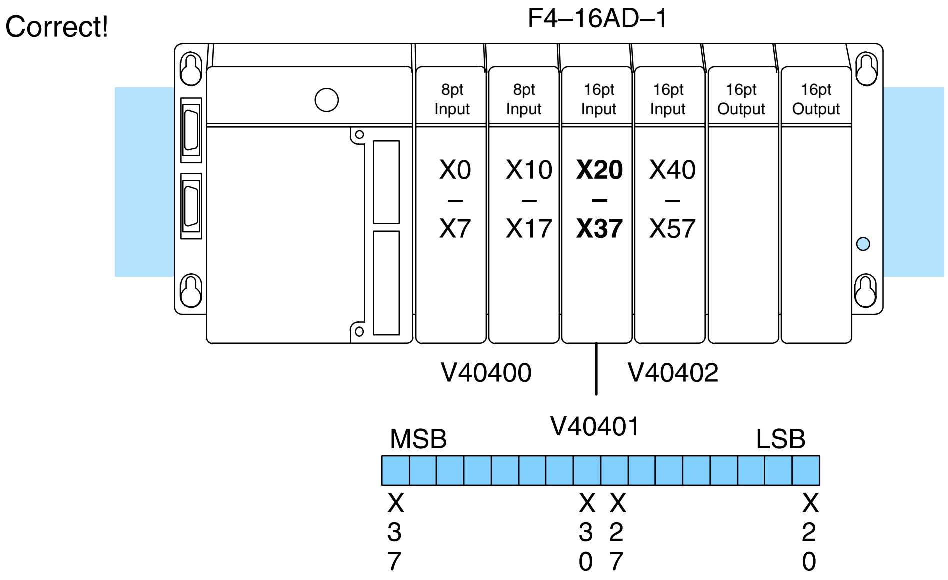
2 or 3 wire: Isolation between input signal and power supply.

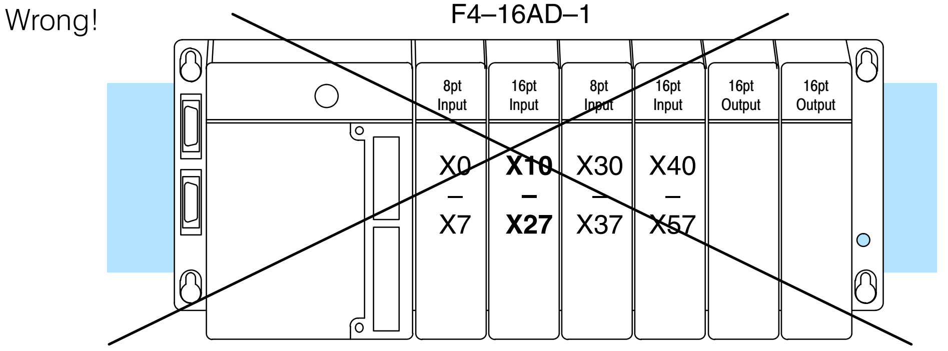
4 wire: Isolation between input signal, power supply, and 4–20mA output.

Module Operation

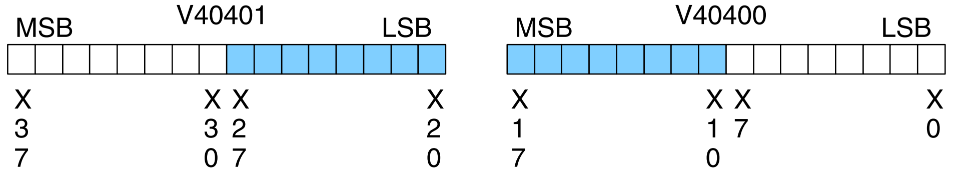
DL430 Special Requirements

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


Correct!
F4–16AD–1
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
X37
X30
X27
X20


Wrong!
F4–16AD–1
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.

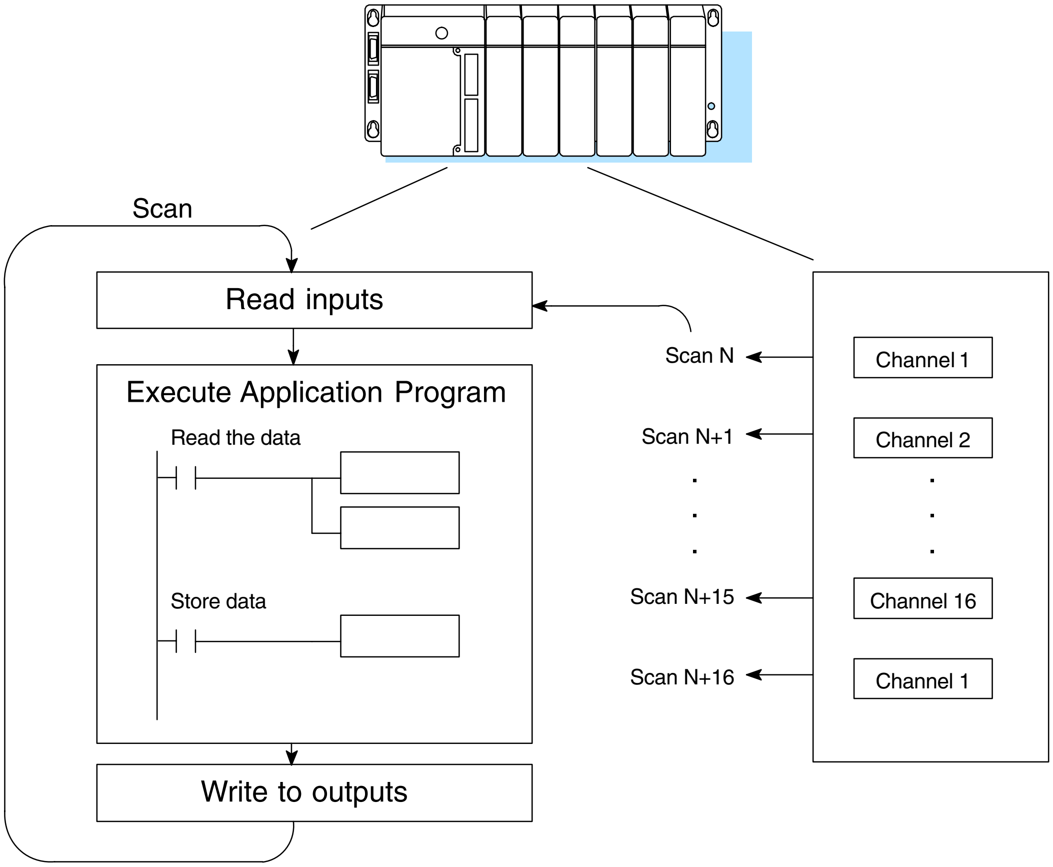

MSB
V40401
LSB
X37
X30
X27
X20
MSB
V40400
LSB
X17
X10
X7
X0

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

Channel Scanning Sequence

The F4–16AD–1 module supplies one channel of data per each CPU scan. Since there are 16 channels, it can take up to 16 scans to get data for all channels. Once all channels have been scanned, the process starts over with channel 1. There are ways around this. Later we'll show you how to write a program that will get all 16 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.


Scan
Read inputs
Execute Application Program
Read the data
Store data
Write to outputs
Scan N
Scan N+1
Scan N+15
Scan N+16
Channel 1
Channel 2
Channel 16
Channel 1

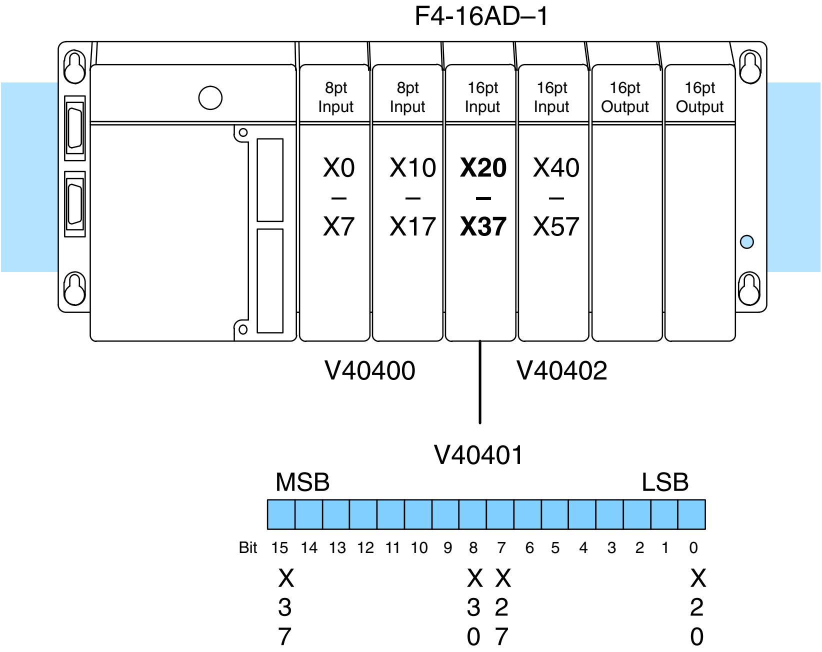
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

Input Bit Assignments

You may recall the F4–16AD–1 module requires 16 discrete input points from the CPU. These 16 points provide:

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

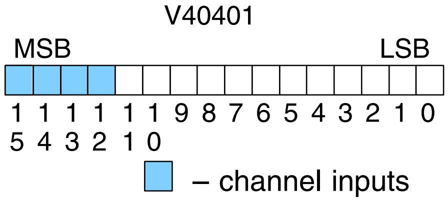
Since all input points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4-16AD–1
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X30
X27
X20

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

Active Channel Indicator Inputs

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


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– channel inputs

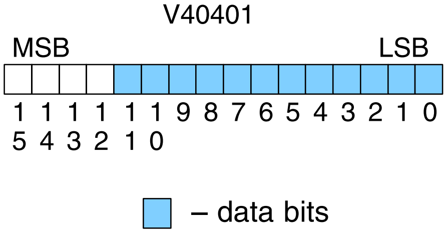
Scan	Channel Inputs	Channel
N	0000	1
N+1	0001	2
N+2	0010	3
N+3	0011	4
N+4	0100	5
N+5	0101	6
N+6	0110	7
N+7	0111	8

Scan	Channel Inputs	Channel
N+8	1000	9
N+9	1001	10
N+10	1010	11
N+11	1011	12
N+12	1100	13
N+13	1101	14
N+14	1110	15
N+15	1111	16

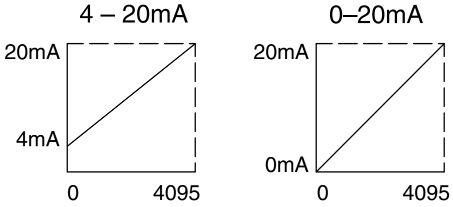
Analog Data Bits

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

Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



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



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

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

H = high limit of the signal range

L = low limit of the signal range

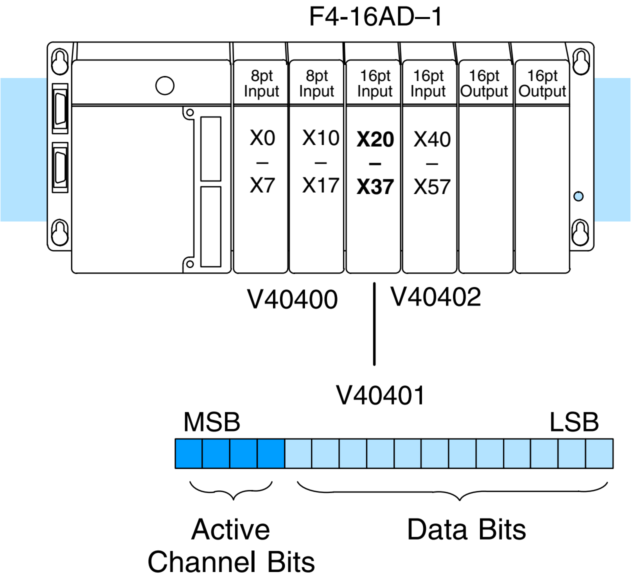
Range	Signal Span (H - L)	Divide By	Smallest Detectable Change
0 to 20mA	20 mA	4095	4.88 μA
4 to 20mA	16 mA	4095	3.91 μA

Writing the Control Program

If you have configured the F4–16AD–1 module, use the following examples to get started writing the control program.

Multiple Channels Selected

Since all channels are multiplexed into a single data word, the control program must be set up to determine which channel is being read. Since the module appears as X input points to the CPU, it is very easy to use the active channel status bits to determine which channel is being monitored.


F4-16AD–1
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
Active Channel Bits
Data Bits

Storing the Channel Data to V–Memory

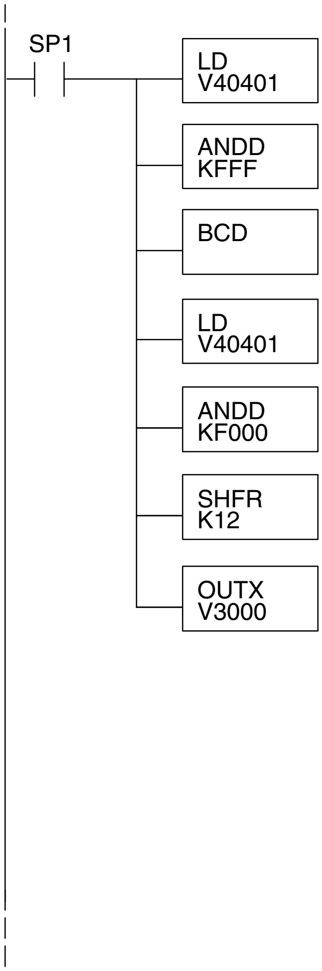
The OUTX instruction used the following programming examples that follow stores the channel data to an address that starts at V3000 plus the channel offset. For example, if channel 2 was being read, the data would be stored in V3002 (V3000 + 2).

Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	0000	0	V3000
Channel 2	0001	1	V3001
Channel 3	0010	2	V3002
Channel 4	0011	3	V3003
Channel 5	0100	4	V3004
Channel 6	0101	5	V3005
Channel 7	0110	6	V3006
Channel 8	0111	7	V3007
Channel 9	1000	8	V3010
Channel 10	1001	9	V3011
Channel 11	1010	10	V3012
Channel 12	1011	11	V3013
Channel 13	1100	12	V3014
Channel 14	1101	13	V3015
Channel 15	1110	14	V3016
Channel 16	1111	15	V3017

Reading Values, DL430 CPU

✓	✓	✓
430	440	450

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



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.

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

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).

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

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

Now you have to shift the accumulator bits so the channel ID bits will result in a value between 0 and 15 (binary format). This value is the offset and indicates which channel is being processed in that scan.

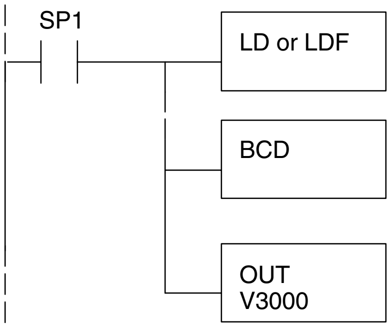
OUTX copies the value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–15) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel 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 table on page 6–13.

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

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.



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

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

The OUT instruction stores the data in V3000.

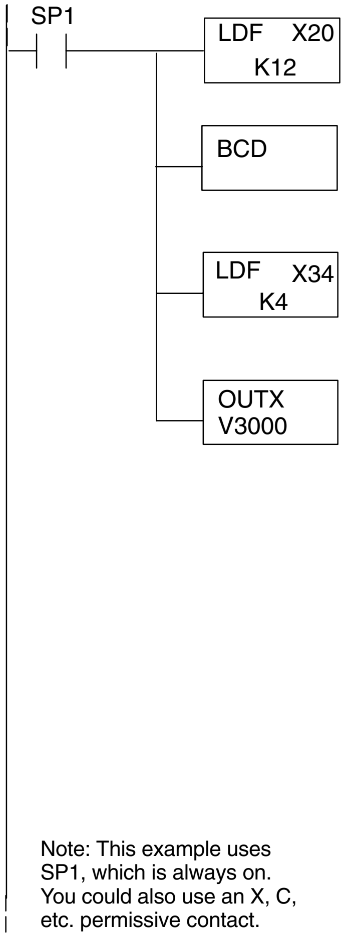
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

* 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

✗ ✓ ✓
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.



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

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.

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

OUTX 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 table on page 6–13.

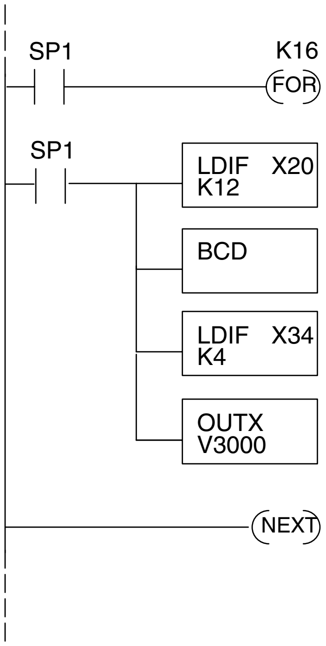
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Reading sixteen Channels in One Scan

☒ ✓ ✓
430 440 450

The following program example shows how to read all sixteen 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 32ms (2ms/loop) to the overall scan time. If you do not need to read the analog data on every scan, change SP1 to a permissive contact (such as an X input, CR, or stage bit) to only enable the FOR/NEXT loop when it is required.

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



Starts the FOR/NEXT loop. The constant (K16) specifies how many times the loop will execute. Enter a constant equal to the number of channels you are using. For example, enter K4 if you're using 4 channels.

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 four channel indicator bits into the accumulator.

The OUTX instruction stores the channel data to an address that starts at V3000 plus the channel offset. For example, if channel 2 was being read, the data would be stored in V3002 (V3000 + 2). See the table on page 6–13.

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

Scaling the Input Data

Most applications usually require measurements in engineering units, which provide more meaningful data. This is accomplished by using the conversion formula shown.

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

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

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 wanted to measure pressure (PSI) from 0.0 to 99.9 then you would have to multiply the analog value by 10 in order to imply a decimal place when you view the value with the programming software or a handheld programmer. Notice how the calculations differ when you use the multiplier.

Analog Value of 2024, slightly less than half scale, should yield 49.4 PSI

Example without multiplier

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

$$\text{Units} = 2024\,\frac{100 - 0}{4095}$$

$$\text{Units} = 49$$

Handheld Display

```
V 3101  V 3100
V MON   0000    0049
```

Example with multiplier

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

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

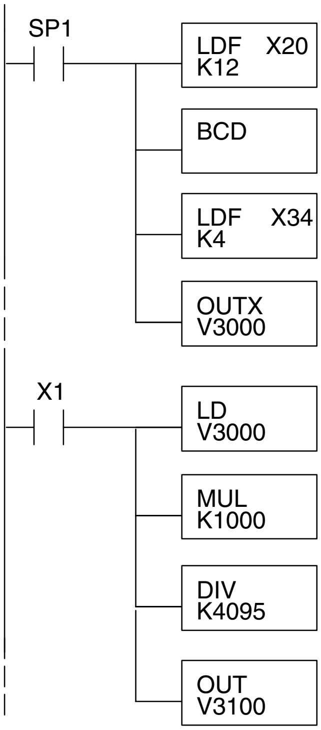
$$\text{Units} = 494$$

Handheld Display

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

*Value is more accurate

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



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

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

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

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

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

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

Divides the accumulator data by 4095.

Stores the result in location V3100.

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the signal levels and the digital values. This is especially useful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the signal level ...
0 to 20mA	$A = \frac{20D}{4095}$	$D = \frac{4095}{20}(A)$
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

For example, if you are using the 0 to 20mA range and you have measured the signal at 6mA, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

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

$$D = \frac{4095}{20}(6mA)$$

$$D = (204.75)\,(6)$$

$$D = 1228$$

F4–16AD–2 8-Channel Analog Input

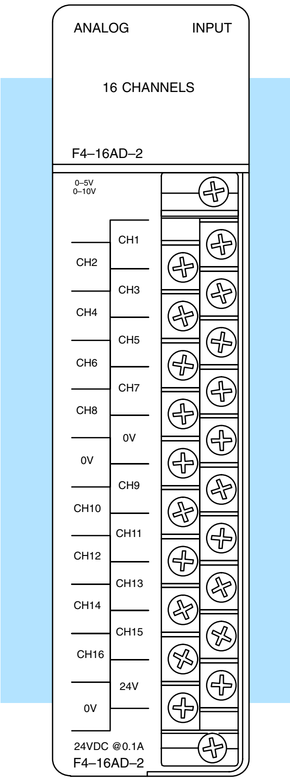
In This Chapter. . . .

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

Module Specifications

The F4-16AD–2 Analog Input module provides several features and benefits.

- It accepts 16 single-ended voltage inputs.
- Analog inputs are optically isolated from PLC logic components.
- The module has a removable terminal block, so the module can be easily removed or changed without disconnecting the wiring.
- All 16 analog inputs may be read in one CPU scan (D4–440 and D4–450 CPUs only).


ANALOG
INPUT
16 CHANNELS
F4–16AD–2
0–5V
0–10V
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
0V
0V
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
24V
0V
24VDC @0.1A
F4–16AD–2

Analog Input Configuration Requirements

The F4–16AD–2 Analog Input module requires 16 discrete input points. The module can be installed in any slot of a DL405 system, including remote bases. The limitations on the number of analog modules are:

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

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

The following table provides the specifications for the F4–16AD–2 Analog Input Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

Number of Channels	16, single ended (one common)
Input Ranges	0–5V, 0–10V
Resolution	12 bit (1 in 4096)
Active Low-pass Filtering	–3 dB at 20Hz, –6 dB per octave
Input Impedance	100K Ohm minimum
Absolute Maximum Ratings	130VAC / 100VDC
Conversion Time	0.4ms per channel (module conversion) 2 ms per selected channel minimum (CPU)
Linearity Error (End to End)	± 2 count (0.050% of full scale) maximum
Input Stability	± 1 count
Full Scale Calibration Error (Offset error not included)	± 12 counts maximum , voltage input
Offset Calibration Error	± 3 counts maximum, unipolar voltage input

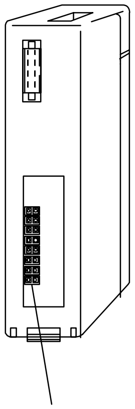
General Specifications

PLC Update Rate	16 Channel per scan max.
Digital Input Points Required	16 (X) input points total 12 binary data bits, 4 active channel bits,
Power Budget Requirement	75 mA (power from base)
External Power Supply	21.6–26.4 VDC, 100 mA, class 2
Accuracy vs. Temperature	± 50 ppm / °C maximum full scale (including maximum offset change of 2 counts)
Operating Temperature	0 to 60°C (32 to 140° F)
Storage Temperature	–20 to 70°C (–4° F to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental Air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

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

Setting the Module Jumpers

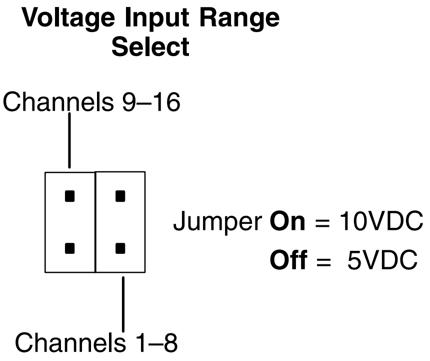
Jumper Locations

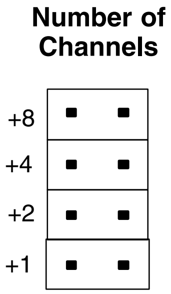

Jumper Locations

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

- A bank of two jumpers to set voltage input range for each bank of 8 channels.
- A bank of four jumpers to select the number of channels used.

The module is set at the factory for a 0–10 VDC signal range on all sixteen channels. The following diagram shows how the jumpers are set at the factory and describes the function of each jumper. When removing a jumper, store it by placing it on a single pin to prevent losing it.

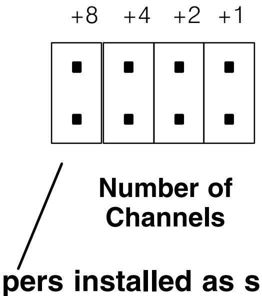

Voltage Input Range Select
Channels 9–16
Channels 1–8
Jumper **On** = 10VDC
Off = 5VDC


Number of Channels
+8
+4
+2
+1

Selecting the Number of Channels

The jumpers labeled +1, +2, +4 and +8 are used to select the number of channels that will be used.

Any unused channels are not processed so if you only select channels 1–8, then the last eight channels will not be active. The following table shows which jumpers to install.

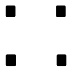
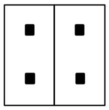


Jumpers installed as shown selects 16-channel operation

Channel(s)	Jumper			
	+8	+4	+2	+1
1	No	No	No	No
1 2	No	No	No	Yes
1 2 3	No	No	Yes	No
1 2 3 4	No	No	Yes	Yes
1 2 3 4 5	No	Yes	No	No
1 2 3 4 5 6	No	Yes	No	Yes
1 2 3 4 5 6 7	No	Yes	Yes	No
1 2 3 4 5 6 7 8	No	Yes	Yes	Yes
1 2 3 4 5 6 7 8 9	Yes	No	No	No
1 2 3 4 5 6 7 8 9 10	Yes	No	No	Yes
1 2 3 4 5 6 7 8 9 10 11	Yes	No	Yes	No
1 2 3 4 5 6 7 8 9 10 11 12	Yes	No	Yes	Yes
1 2 3 4 5 6 7 8 9 10 11 12 13	Yes	Yes	No	No
1 2 3 4 5 6 7 8 9 10 11 12 13 14	Yes	Yes	No	Yes
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	Yes	Yes	Yes	No
1 2 3 4 5 6 7 8 9 10 11 12 13141516	Yes	Yes	Yes	Yes

Selecting Input Signal Ranges

The following table shows the jumper selections for the 5V and 10V ranges. The module comes from the factory set for the 10V operation (jumpers installed). See note below.

Signal Range	Jumper Settings
0 to +5 VDC Jumpers OFF	Channels 1–8 Jumper  Channels 9–16 Jumper
0 to +10 VDC Jumpers ON	Channels 1–8 Jumper  Channels 9–16 Jumper

NOTE: The jumpers do not have to both be set On or Off (e.g. Channels 1–8 can be selected for 5V and Channels 9–16 can be selected for 10V operation).

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

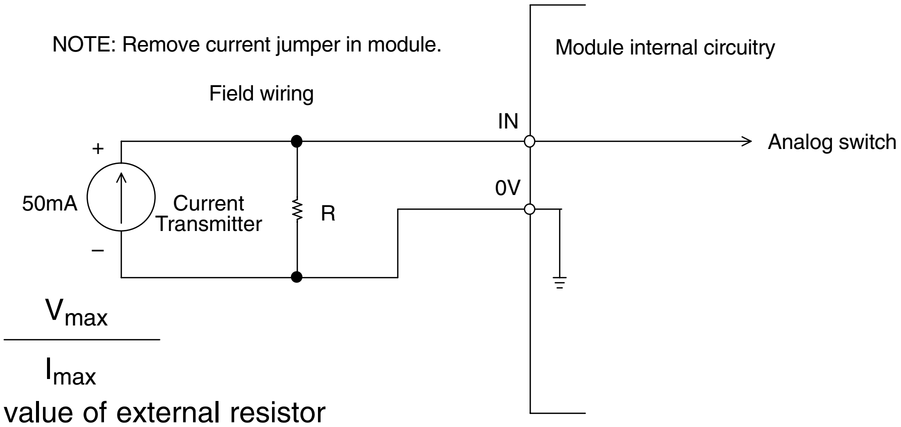
- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

User Power Supply Requirements

The F4–16AD–2 module requires a separate power supply. The Series DL405 CPUs, D4-RS Remote I/O Controller, and D4-EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. If you only have a couple analog modules, you can use this power source instead of a separate supply. If you have more than four analog modules, or you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ±10%, Class 2, 100 mA current (per module).

Custom Input Ranges

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


NOTE: Remove current jumper in module.
Module internal circuitry
Field wiring
IN
Analog switch
+
50mA
Current Transmitter
–
R
0V

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

R = value of external resistor

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

I_{max} = maximum current supplied by the transmitter

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

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

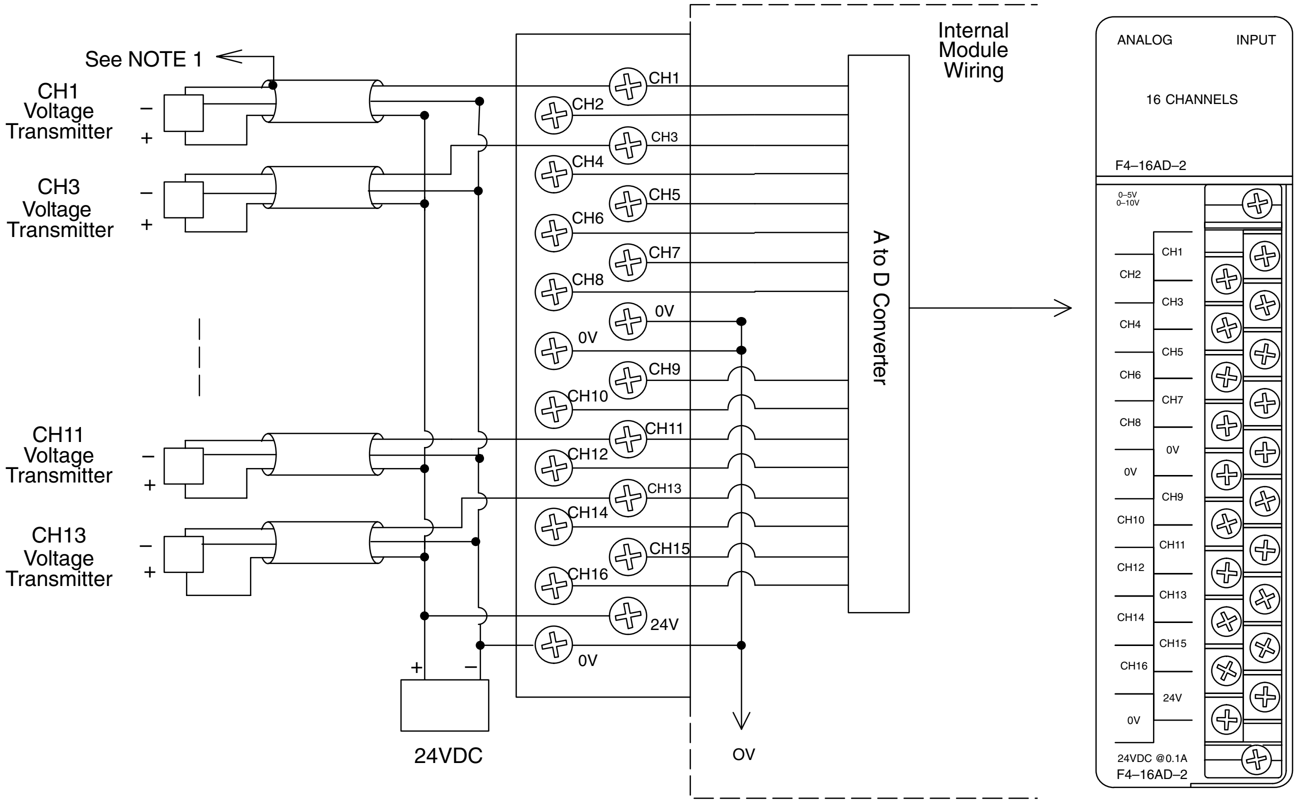
NOTE: Your choice of resistor can affect the accuracy of the module. A resistor that has ±0.1% tolerance and a ±50ppm / °C temperature coefficient is recommended.

Removable Connector

The F4–16AD–2 module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

Wiring Diagram

NOTE 1: Shields should be grounded at the signal source.


See NOTE 1
CH1 Voltage Transmitter
CH3 Voltage Transmitter
CH11 Voltage Transmitter
CH13 Voltage Transmitter
–
+
24VDC
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
0V
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
24V
0V
Internal Module Wiring
A to D Converter
OV
ANALOG
INPUT
16 CHANNELS
F4–16AD–2
0–5V
0–10V
24VDC @0.1A

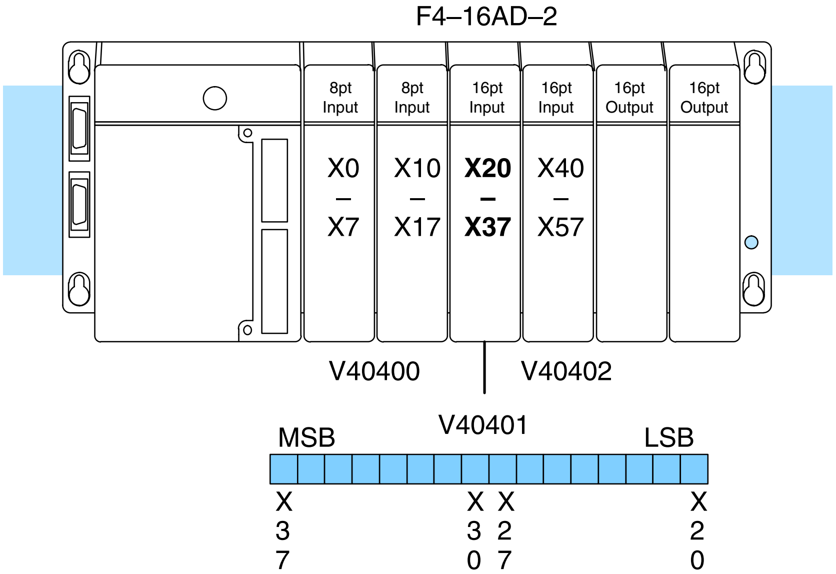
If the power supply common of an external power supply is not connected to 0V on the module, then the output of the external transmitter must be isolated.

Module Operation

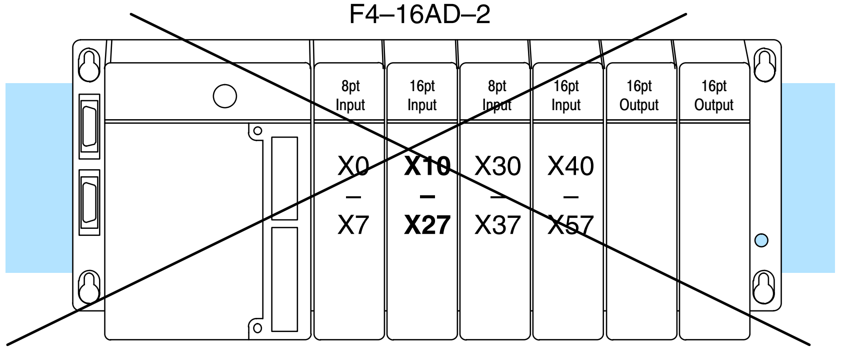
D4–430 Special Requirements

Even though the module can be placed in any slot, it is important to examine the configuration if you are using a D4–430 CPU. As you will see in the section on writing the program, you use V-memory locations to extract the analog data. As shown in the following diagram, if you place the module so the input points do not start on a V-memory boundary, the instructions cannot access the data.

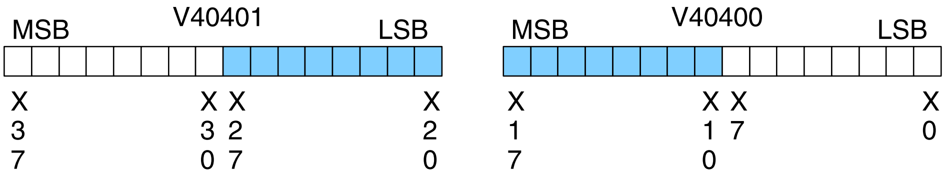
Correct!


F4–16AD–2
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
MSB
V40401
LSB
X37
X30
X27
X20

Wrong!


F4–16AD–2
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 D4–430.

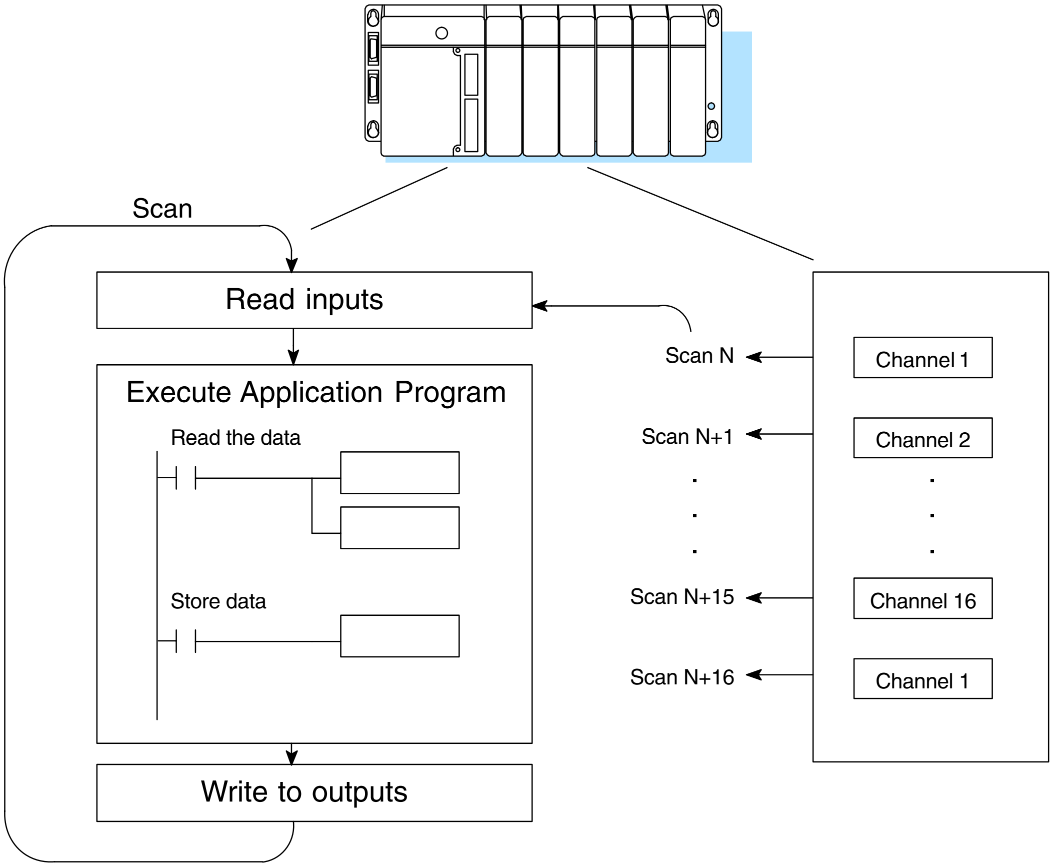

MSB
V40401
LSB
X37
X30
X27
X20
MSB
V40400
LSB
X17
X10
X7
X0

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

Channel Scanning Sequence

The F4–16AD–2 module supplies one channel of data per each CPU scan. Since there are sixteen channels, it can take up to sixteen scans to get data for all channels. Once all channels have been scanned the process starts over with channel 1. There are ways around this. Later we'll show you how to write a program that will get all sixteen 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.


Scan
Read inputs
Execute Application Program
Read the data
Store data
Write to outputs
Scan N
Scan N+1
Scan N+15
Scan N+16
Channel 1
Channel 2
Channel 16
Channel 1

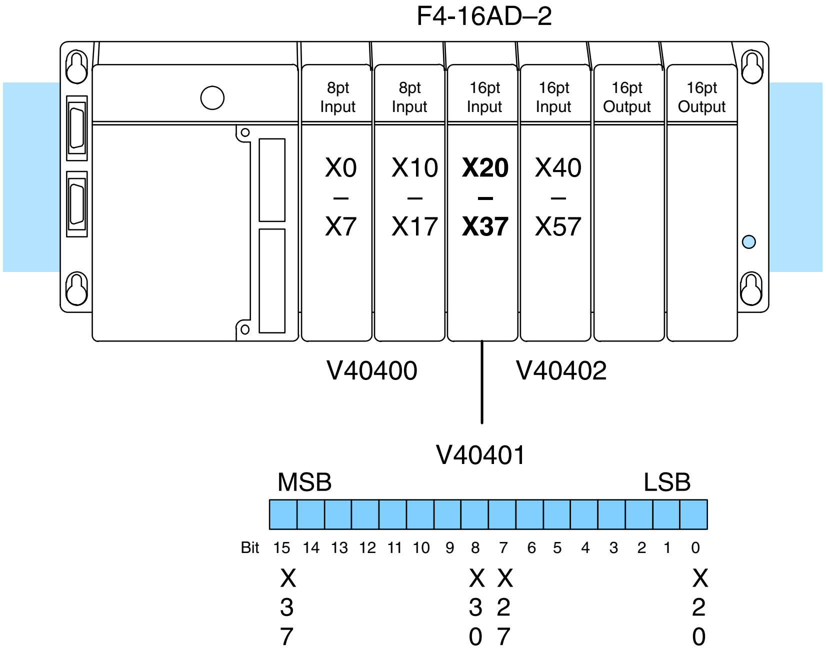
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the analog transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control logic in the RLL program.

Input Bit Assignments

You may recall the F4–16AD–2 module requires 16 discrete input points from the CPU. These 16 points provide:

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

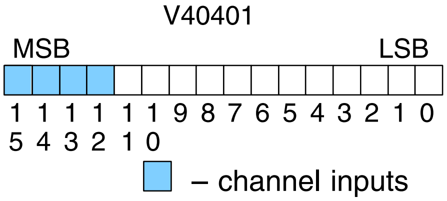
Since all input points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4-16AD–2
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X30
X27
X20

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

Active Channel Indicator Inputs

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


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– channel inputs

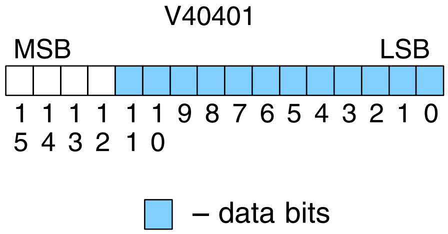
Scan	Channel Inputs	Channel
N	0000	1
N+1	0001	2
N+2	0010	3
N+3	0011	4
N+4	0100	5
N+5	0101	6
N+6	0110	7
N+7	0111	8

Scan	Channel Inputs	Channel
N+8	1000	9
N+9	1001	10
N+10	1010	11
N+11	1011	12
N+12	1100	13
N+13	1101	14
N+14	1110	15
N+15	1111	16

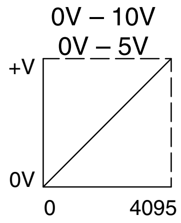
Analog Data Bits

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

Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



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



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

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

H = high limit of the signal range

L = low limit of the signal range

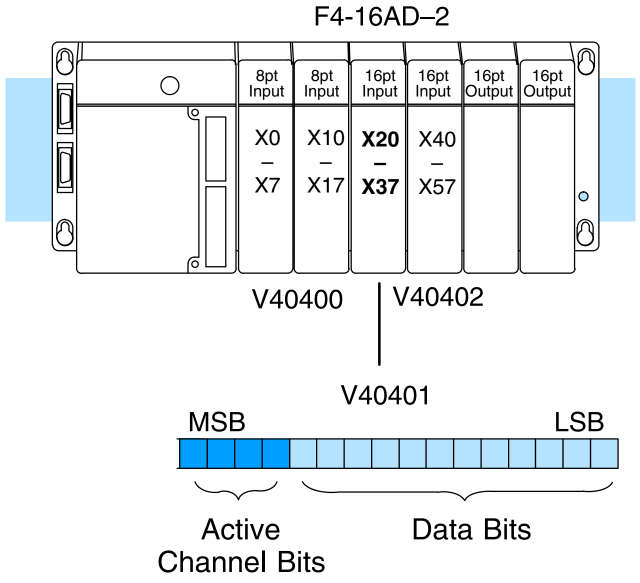
Range	Signal Span (H - L)	Divide By	Smallest Detectable Change
0 to 5V	5 V	4095	1.22 mV
0 to 10V	10 V	4095	2.44 mV

Writing the Control Program

If you have configured the F4–16AD–2 module, use the following examples to get started writing the control program.

Multiple Channels Selected

Since all channels are multiplexed into a single data word, the control program must be set up to determine which channel is being read. Since the module appears as X input points to the CPU, it is very easy to use the active channel status bits to determine which channel is being monitored.


F4-16AD–2
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
Active Channel Bits
Data Bits

Storing the Channel Data to V–Memory

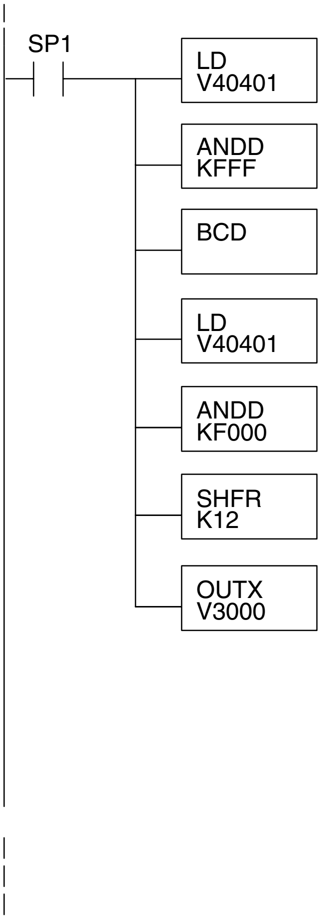
The OUTX instruction used the following programming examples that follow stores the channel data to an address that starts at V3000 plus the channel offset. For example, if channel 2 was being read, the data would be stored in V3002 (V3000 + 2).

Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	0000	0	V3000
Channel 2	0001	1	V3001
Channel 3	0010	2	V3002
Channel 4	0011	3	V3003
Channel 5	0100	4	V3004
Channel 6	0101	5	V3005
Channel 7	0110	6	V3006
Channel 8	0111	7	V3007
Channel 9	1000	8	V3010
Channel 10	1001	9	V3011
Channel 11	1010	10	V3012
Channel 12	1011	11	V3013
Channel 13	1100	12	V3014
Channel 14	1101	13	V3015
Channel 15	1110	14	V3016
Channel 16	1111	15	V3017

Reading Values

✓ ✓ ✓
430 440 450

The following program example shows how to read the analog data into V-memory locations with the D4–430 CPU. Since the D4–430 does not support the LDF instruction, you can use the LD instruction instead as shown. The example also works for D4–440 and D4–450 CPUs. This example will read one channel per scan, so it will take sixteen scans to read all sixteen channels. Contact SP1 is used in the example because the inputs are continually being updated.



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.

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

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).

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

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

Now you have to shift the accumulator bits so the channel ID bits will result in a value between 0 and 15 (binary format).

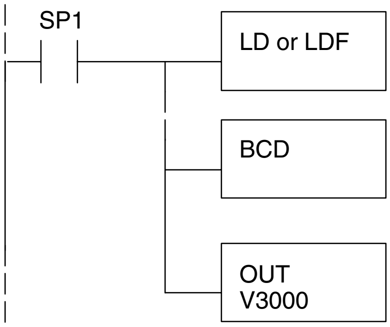
OUTX copies the value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–15) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel 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 table on page 7–13.

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

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.



When X34 is on, channel 1 data is being sent to the CPU. Use the LD instruction when using a D4–430 CPU.*

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

The OUT instruction stores the data in V3000.

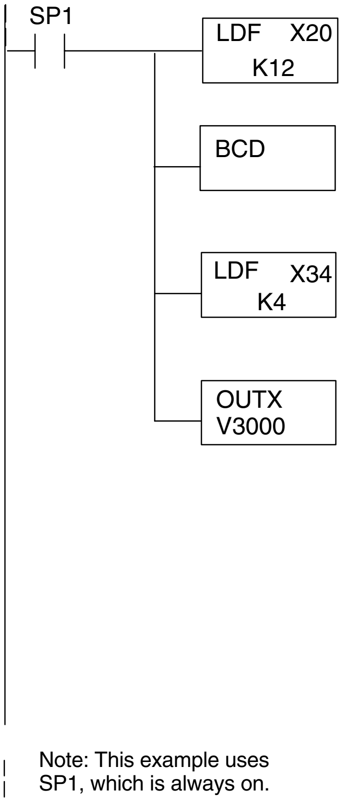
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

* Remember, before the BCD instruction is executed, the D4–430 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

✗ ✓ ✓
430 440 450

The following program example shows how to read the analog data into V-memory locations with D4–440 and D4–450 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 sixteen scans to read all sixteen channels.



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

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.

Loads the binary value of the four 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 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 table on page 7–13.

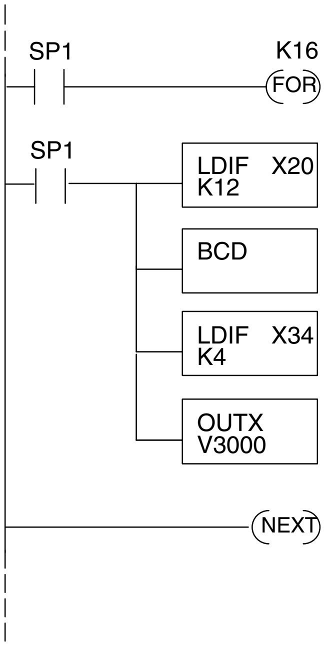
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Reading Sixteen Channels in One Scan

☒ ✓ ✓
430 440 450

The following program example shows how to read all sixteen 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 32ms (2ms/loop) to the overall scan time. If you do not need to read the analog data on every scan, change SP1 to a permissive contact (such as an X input, CR, or stage bit) to only enable the FOR/NEXT loop when it is required.

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



Starts the FOR/NEXT loop. The constant (K16) specifies how many times the loop will execute. Enter a constant equal to the number of channels you are using. For example, enter K4 if you're using 4 channels.

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 four channel indicator bits into the accumulator.

The OUTX instruction stores the channel data to an address that starts at V3000 plus the channel offset. For example, if channel 2 was being read, the data would be stored in V3002 (V3000 + 2). See the table on page 7–13.

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

Scaling the Input Data

Most applications usually require measurements in engineering units, which provide more meaningful data. This is accomplished by using the conversion formula shown.

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

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

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 wanted to measure pressure (PSI) from 0.0 to 99.9 then you would have to multiply the analog value by 10 in order to imply a decimal place when you view the value with the programming software or a handheld programmer. Notice how the calculations differ when you use the multiplier.

Analog Value of 2024, slightly less than half scale, should yield 49.4 PSI

Example without multiplier

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

$$\text{Units} = 2024\,\frac{100 - 0}{4095}$$

$$\text{Units} = 49$$

Handheld Display

```
    V 3101  V 3100
V MON   0000    0049
```

Example with multiplier

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

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

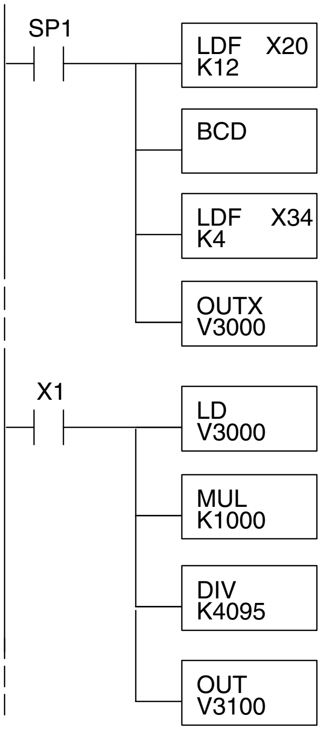
$$\text{Units} = 494$$

Handheld Display

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

*Value is more accurate

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



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

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

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

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

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

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

Divides the accumulator data by 4095.

Stores the result in location V3100.

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the signal levels and the digital values. This is especially useful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the signal level ...
0 to 5V	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}(A)$
0 to 10V	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}(A)$

For example, if you are using the 0V to +10V range and you have measured the signal at 6V, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

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

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

$$D = (409.5)(6)$$

$$D = 2457$$

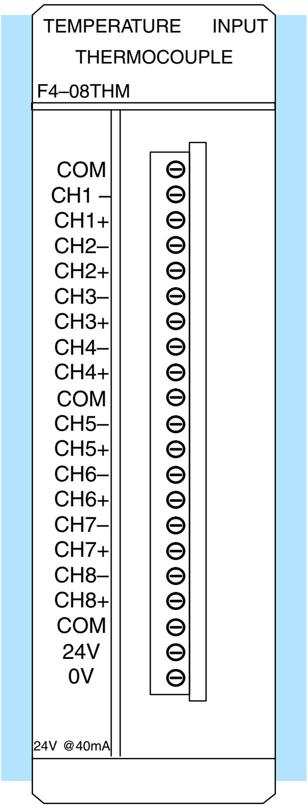
8

F4-08THM-n 8-Channel Thermocouple Input

In This Chapter. . . .

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

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.

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.

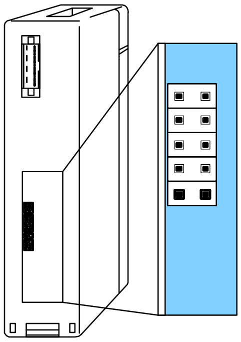
Check the user manual for your particular model of CPU for more information regarding power budget and number of local or remote I/O points.

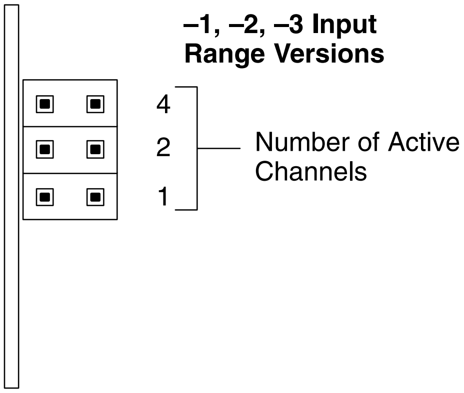
Setting the Module Jumpers

Jumper Locations

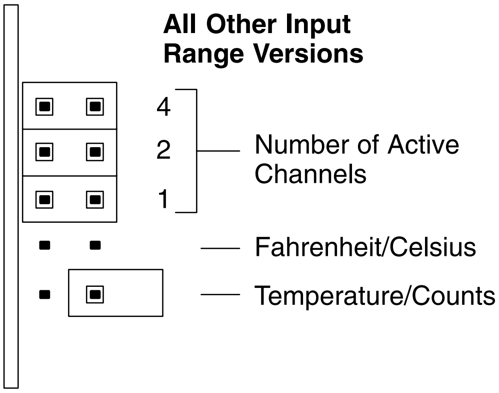
At the rear of the module is a bank of three or five jumpers, depending on the module version. The module has options that you can select by installing or removing these jumpers:

- 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



All Other Input Range Versions

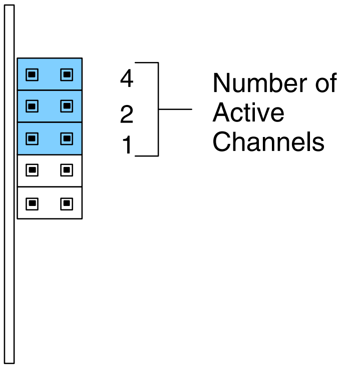
Factory Default Settings

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

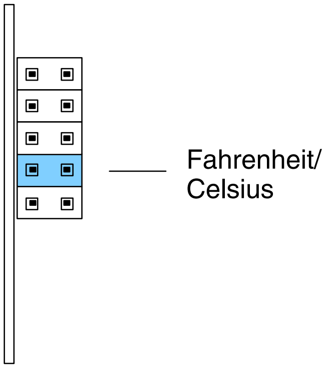
Number of Channels	Jumper Pins		
	4	2	1
1			
2			yes
3		yes	
4		yes	yes
5	yes		
6	yes		yes
7	yes	yes	
8	yes	yes	yes

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.

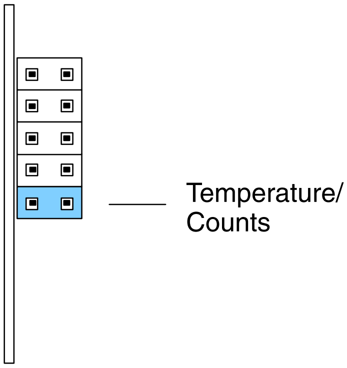
Selecting Temperature or Counts

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.



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.

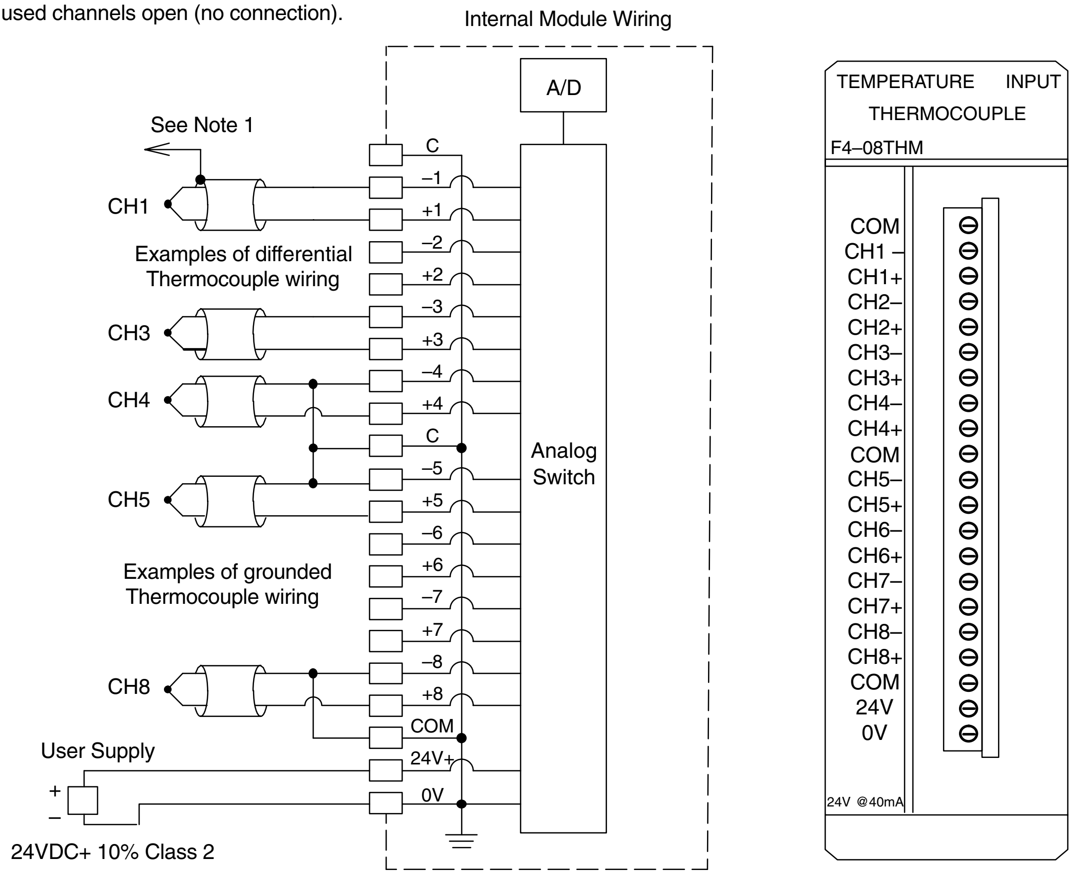
User Power Supply Requirements

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
CH1
Examples of differential
Thermocouple wiring
CH3
CH4
CH5
Examples of grounded
Thermocouple wiring
CH8
User Supply
+
–
24VDC+ 10% Class 2
C
–1
+1
–2
+2
–3
+3
–4
+4
C
–5
+5
–6
+6
–7
+7
–8
+8
COM
24V+
0V
Analog
Switch
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

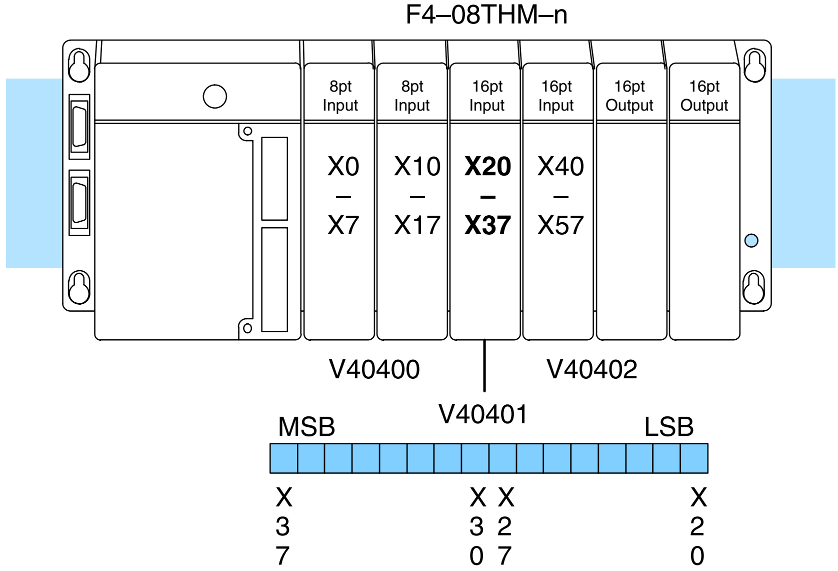
Module Operation

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.

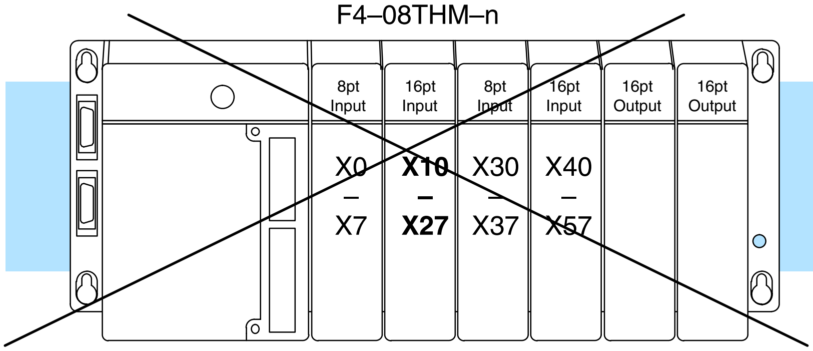
DL430 Special Requirements

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.

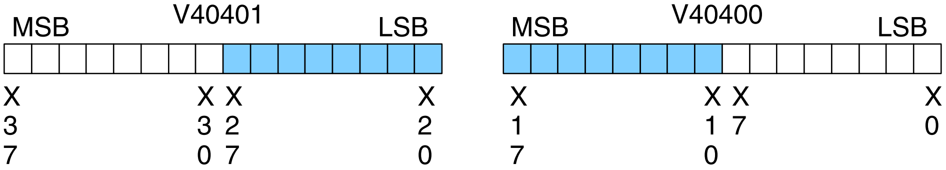
Correct!


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

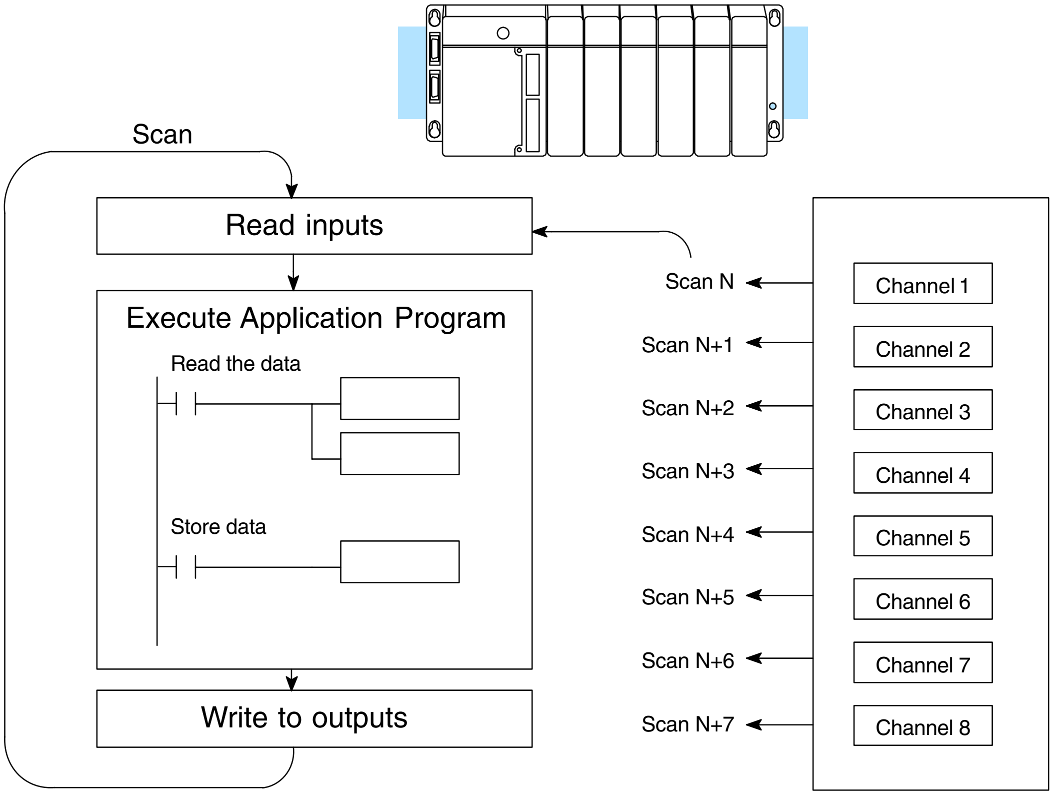
Data is split over two locations, so instructions cannot access data from a DL430.


MSB
V40401
LSB
X 3 7
X 3 0
X 2 7
X 2 0
MSB
V40400
LSB
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.


Scan
Read inputs
Execute Application Program
Read the data
Store data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Channel 1
Channel 2
Channel 3
Channel 4
Channel 5
Channel 6
Channel 7
Channel 8

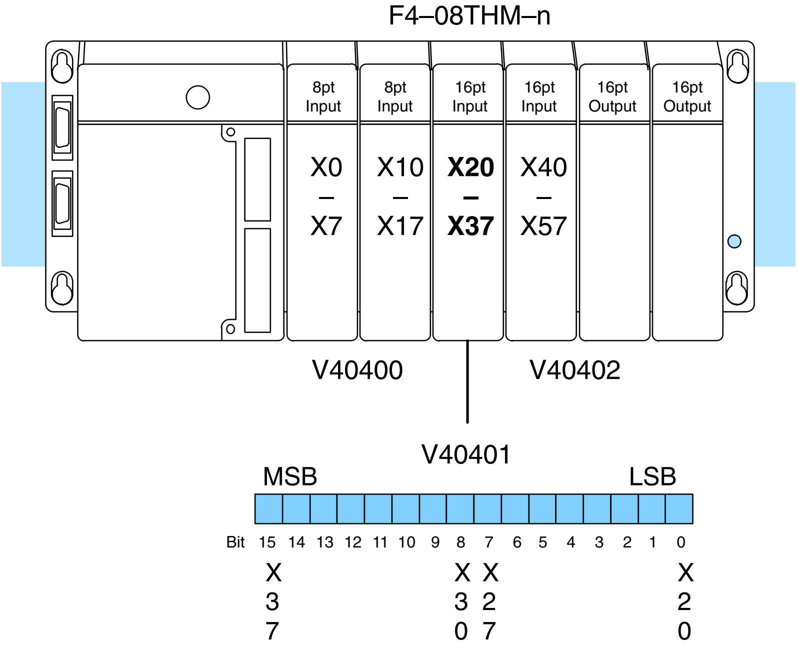
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the thermocouple transmitter signal and converts the signal to a 12-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the 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.

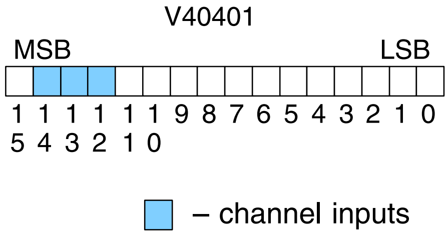
Since all input points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4–08THM–n
8pt Input
8pt Input
16pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X37
X40 – X57
V40400
V40402
V40401
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X30
X27
X20

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

Active Channel Indicator Inputs

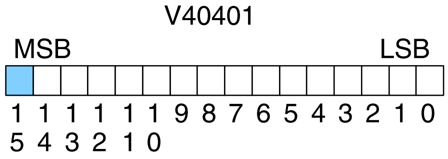
The bits (inputs) shown in the diagram indicate the active channel. The next-to-last three bits of the V-memory location indicate the active channel. The inputs are automatically turned on and off on each CPU scan to indicate the active channel.


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– channel inputs

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

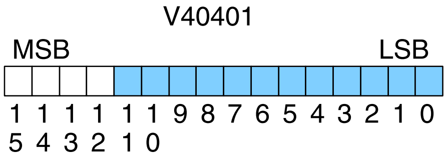


🟦 – temperature sign

Analog Data Bits

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

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



🟦 – data bits

Temperature Input Resolution

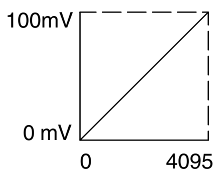
Typically, the F4–08THM–n resolution enables you to detect a 1°F change in temperature. The National Institute of Standards and Technology (NIST) publishes conversion tables that show how each temperature corresponds to an equivalent signal level.

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.

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.

0–100 mV Scale



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

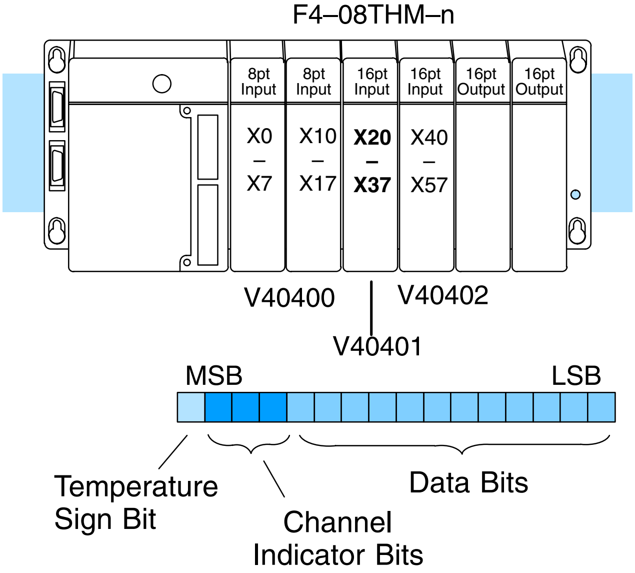
H = high limit of the signal range
L = low limit of the signal range

Range	Signal Span (H – L)	Divide By	Smallest Detectable Change
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

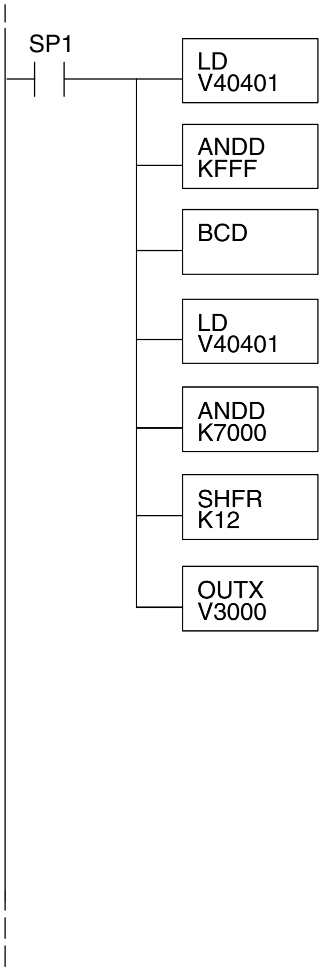
Automatic Temperature Conversion

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.



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.

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

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).

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

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

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 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.

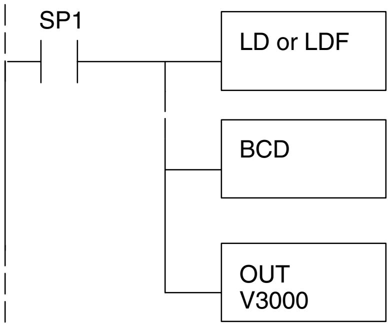
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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

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.



Channel 1 data is being sent to the CPU. Use the LD instruction when using a DL430 CPU.*

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

The OUT instruction stores the data in V3000.

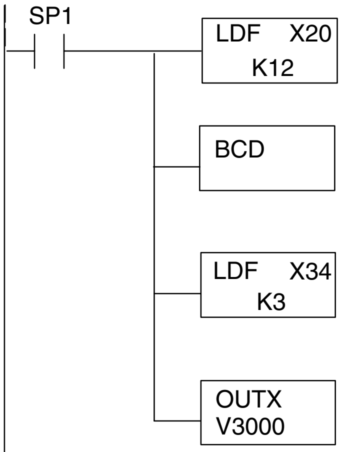
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

* 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.



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

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.

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 copies the 16 bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (which is the offset) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0) and if the binary value is 6, then the channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

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

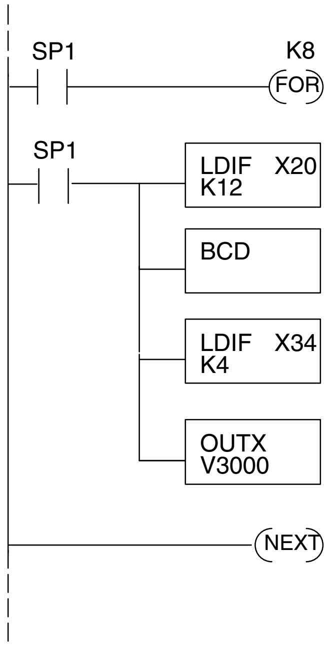
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Reading Eight Channels in One Scan, DL440/450

430	440	450
☒	✓	✓

The following program example shows how to read all eight channels in one scan by using a FOR/NEXT loop. Before you choose this method, do consider its impact on CPU scan time. The FOR/NEXT routine shown here will add about 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.



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

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.

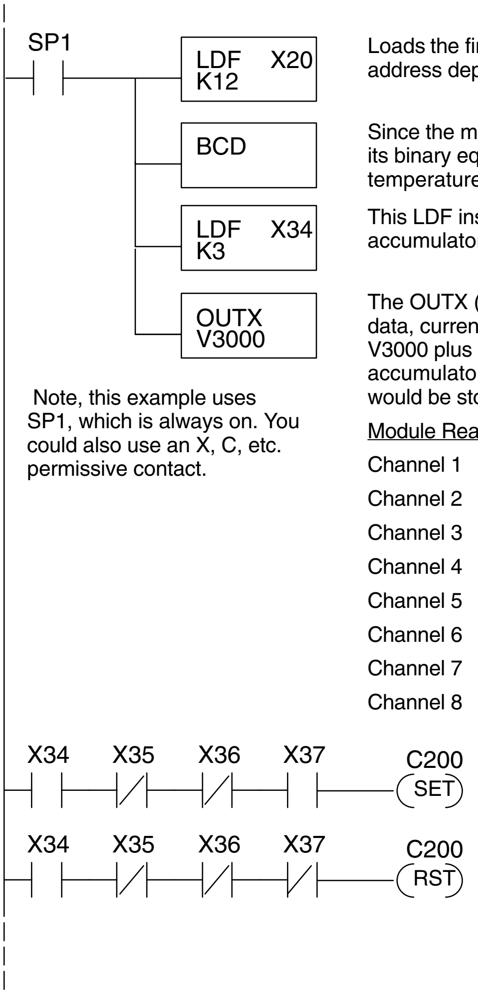
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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.



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

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

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

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).

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.

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

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

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

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

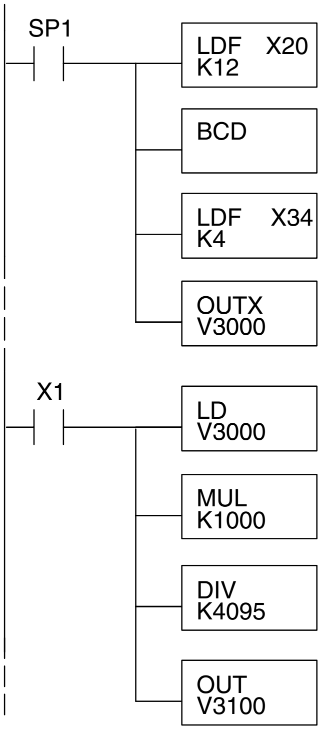
$$\text{Units} = 494$$

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.



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

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

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

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

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

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

Divides the accumulator data by 4095.

Stores the result in location 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	$A = \frac{25D}{4095}$	$D = \frac{4095}{25}A$
0 to 50 mV	$A = \frac{50D}{4095}$	$D = \frac{4095}{50}A$
0 to 100 mV	$A = \frac{100D}{4095}$	$D = \frac{4095}{100}A$

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.

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

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

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

$$D = 1229$$

9

F4-08RTD 8-Channel RTD Input

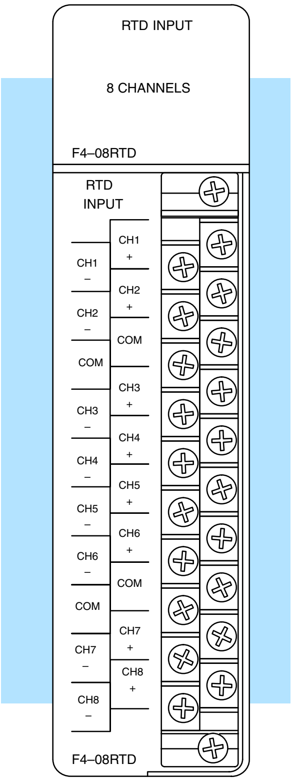
In This Chapter. . . .

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

Module Specifications

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

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



Module Calibration

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

RTD Input Configuration Requirements

The F4–08RTD Input Module requires 32 discrete input points from the CPU. The module can be installed in any slot of a DL405 system, including remote bases. The limitations on the number of analog modules are:

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

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

The following tables provide the specifications for the F4–08RTD Module. Review these specifications to ensure the module meets your application requirements.

Input Specifications

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

General Specifications

PLC Update Rate	8 Channels/Scan max. DL440/DL450 CPUs 1 Channel/Scan max. DL430 CPU
Digital Input Points Required	32 (X) input points, 16 binary data bits, 3 channel ID bits, 8 fault bits
Power Budget Requirement	80 mA @ 5 VDC (from base)
Operating Temperature	0° to 60°C (32° to 140°F)
Storage Temperature	–20° to 70°C (–4° to 158°F)
Relative Humidity	5 to 95% (non-condensing)
Environmental Air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

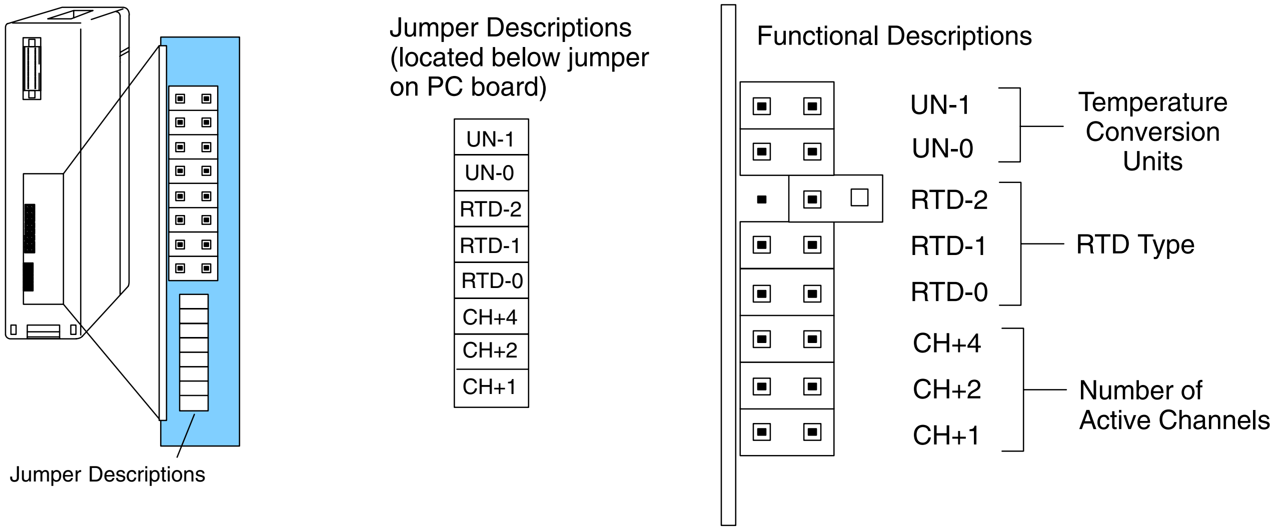
Setting the Module Jumpers

Jumper Locations

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

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

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

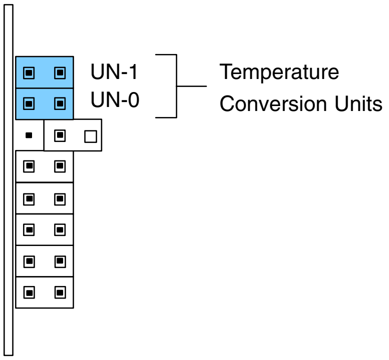

Jumper Descriptions
Jumper Descriptions (located below jumper on PC board)
UN-1
UN-0
RTD-2
RTD-1
RTD-0
CH+4
CH+2
CH+1
Functional Descriptions
UN-1
UN-0
Temperature Conversion Units
RTD-2
RTD-1
RTD-0
RTD Type
CH+4
CH+2
CH+1
Number of Active Channels

Factory Default Settings

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

Selecting Fahrenheit or Celsius

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


UN-1
UN-0
Temperature Conversion Units

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

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

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

The table shows how to arrange the jumpers.

X = jumper installed, empty space = jumper removed.

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

Selecting RTD Type

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

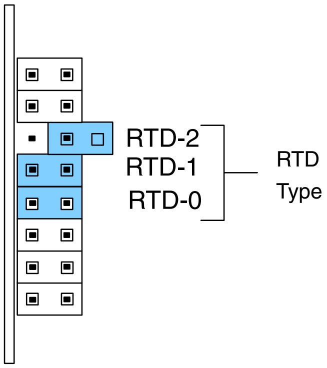
The default setting from the factory is Pt100 Ω (RTD-2 comes with the jumper removed). This selects the DIN 43760 European type RTD. European curve type RTDs are calibrated to DIN 43760, BS1905, or IEC751 specifications which is .00385 Ω /Ω / ° C (100° C = 138.5Ω).

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

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

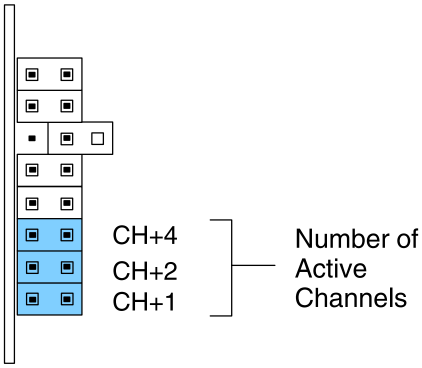
X = jumper installed, empty space = jumper removed.

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



Selecting the Number of Channels

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



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

Any unused channels are not processed. For example, if you only select the first four channels, then the last four channels will not be active. The following table shows how to arrange the jumpers. For example, to select channels 1 thru 4, remove jumper CH+4 and install jumpers CH+2 and CH+1.

X = jumper installed,

empty space = jumper removed.

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

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the signal source. *Do not* ground the shield at both the module and the source.
- Don't run the signal wiring next to large motors, high current switches, or transformers. This may cause noise problems.
- Route the wiring through an approved cable housing to minimize the risk of accidental damage. Check local and national codes to choose the correct method for your application.

RTD – Resistance Temperature Detector

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

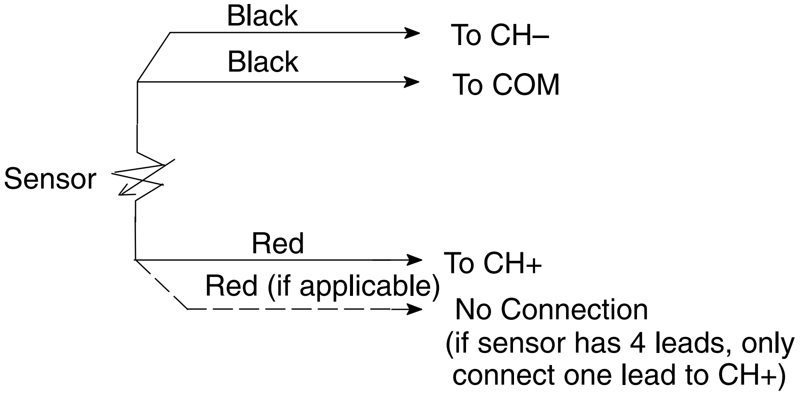
Lead Configuration for RTD Sensors

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

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

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

Wiring Connections For Typical RTD Sensor

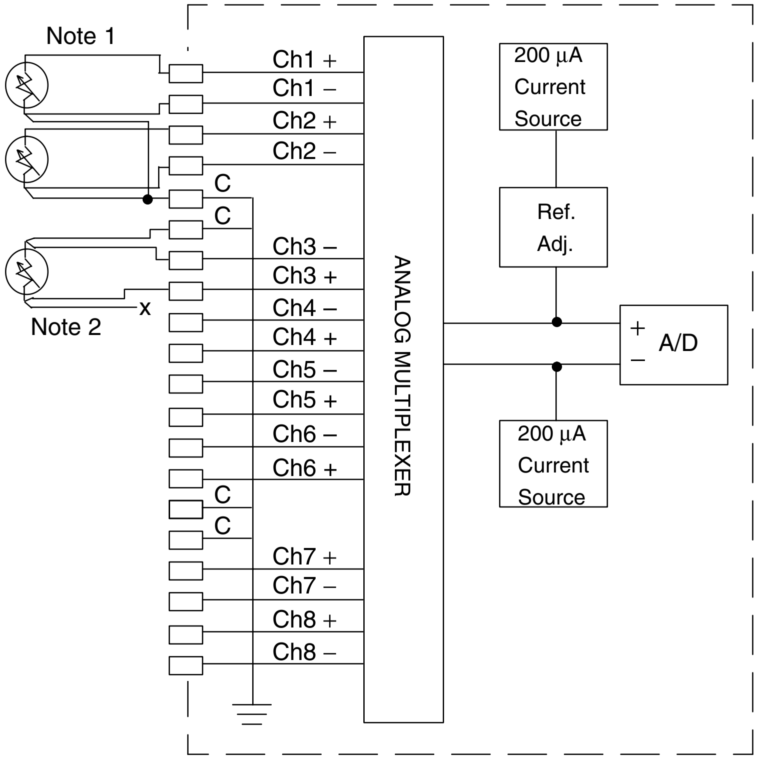


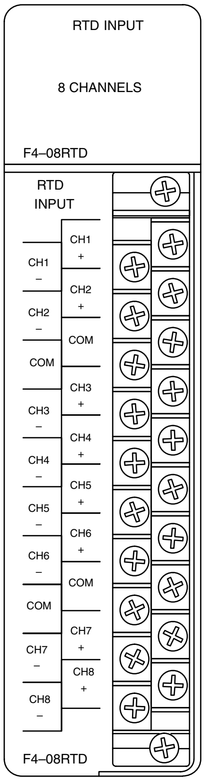
Ambient Variations in Temperature

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

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

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


Note 1
Note 2
x
Ch1 +
Ch1 –
Ch2 +
Ch2 –
C
C
Ch3 –
Ch3 +
Ch4 –
Ch4 +
Ch5 –
Ch5 +
Ch6 –
Ch6 +
C
C
Ch7 +
Ch7 –
Ch8 +
Ch8 –
ANALOG MULTIPLEXER
200 µA Current Source
Ref. Adj.
+
–
A/D
200 µA Current Source


RTD INPUT
8 CHANNELS
F4–08RTD
RTD INPUT
CH1 +
CH1 –
CH2 +
CH2 –
COM
COM
CH3 +
CH3 –
CH4 +
CH4 –
CH5 +
CH5 –
CH6 +
CH6 –
COM
COM
CH7 +
CH7 –
CH8 +
CH8 –
F4–08RTD

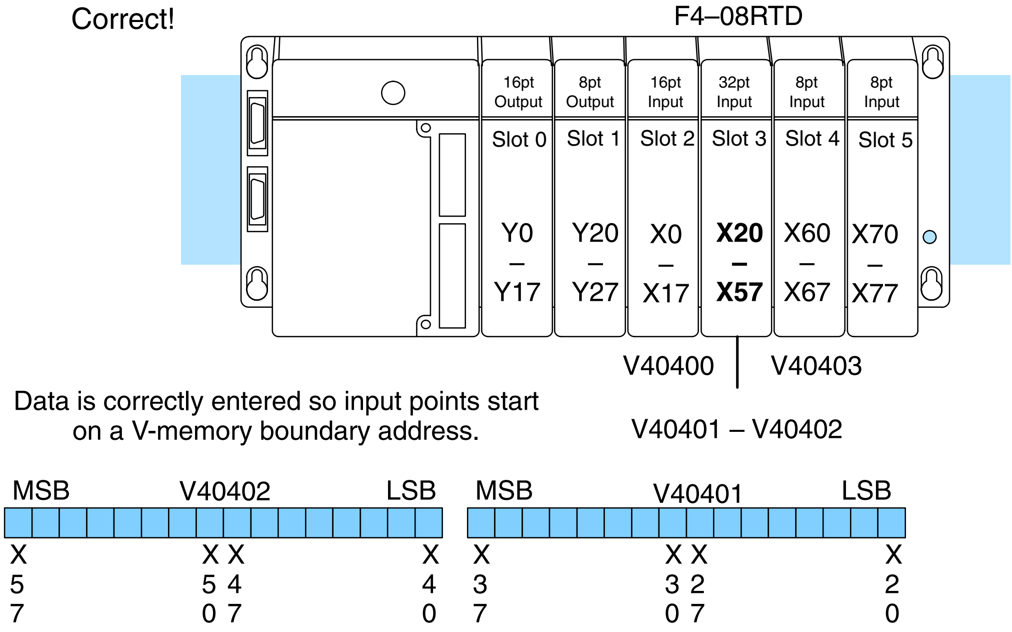
Notes:

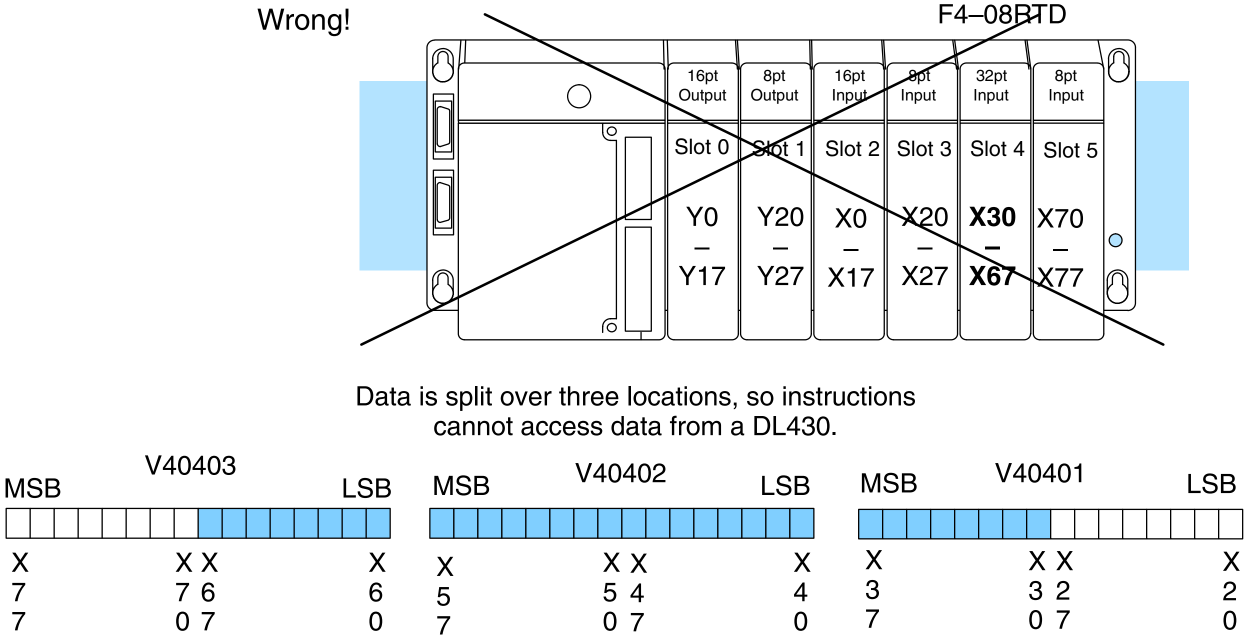
1. The three wires connecting the RTD to the module must be the same type and length. Do not use the shield or drain wire for the third connection.
2. If a RTD sensor has four wires, the extra plus (+) sense wire should be left unconnected as shown.

Module Operation

DL430 Special Requirements

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


Correct!
F4–08RTD
16pt Output
8pt Output
16pt Input
32pt Input
8pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y17
Y20 – Y27
X0 – X17
X20 – X57
X60 – X67
X70 – X77
V40400
V40403
V40401 – V40402
Data is correctly entered so input points start on a V-memory boundary address.
MSB
V40402
LSB
MSB
V40401
LSB
X57
X50
X47
X40
X37
X30
X27
X20

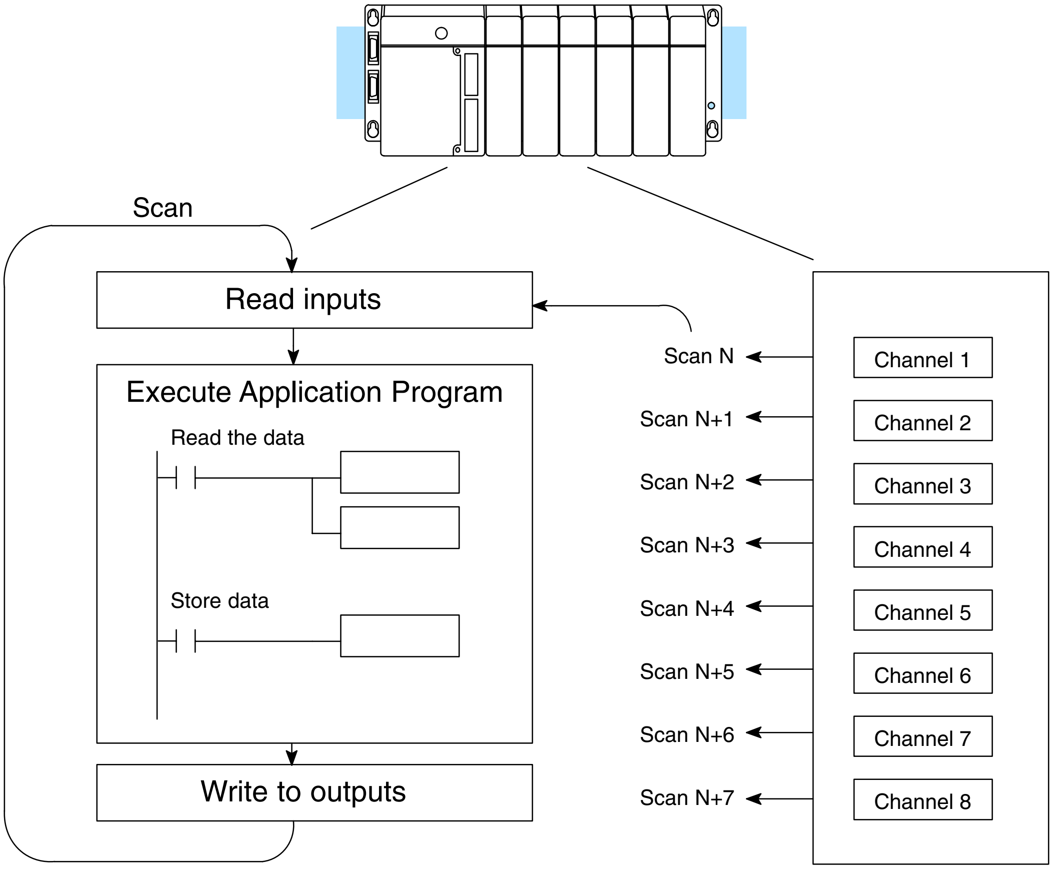

Wrong!
F4–08RTD
16pt Output
8pt Output
16pt Input
8pt Input
32pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y17
Y20 – Y27
X0 – X17
X20 – X27
X30 – X67
X70 – X77
Data is split over three locations, so instructions cannot access data from a DL430.
V40403
MSB
LSB
V40402
MSB
LSB
V40401
MSB
LSB
X77
X70
X67
X60
X57
X50
X47
X40
X37
X30
X27
X20

Channel Scanning Sequence

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

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

Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan.


Scan
Read inputs
Execute Application Program
Read the data
Store data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Channel 1
Channel 2
Channel 3
Channel 4
Channel 5
Channel 6
Channel 7
Channel 8

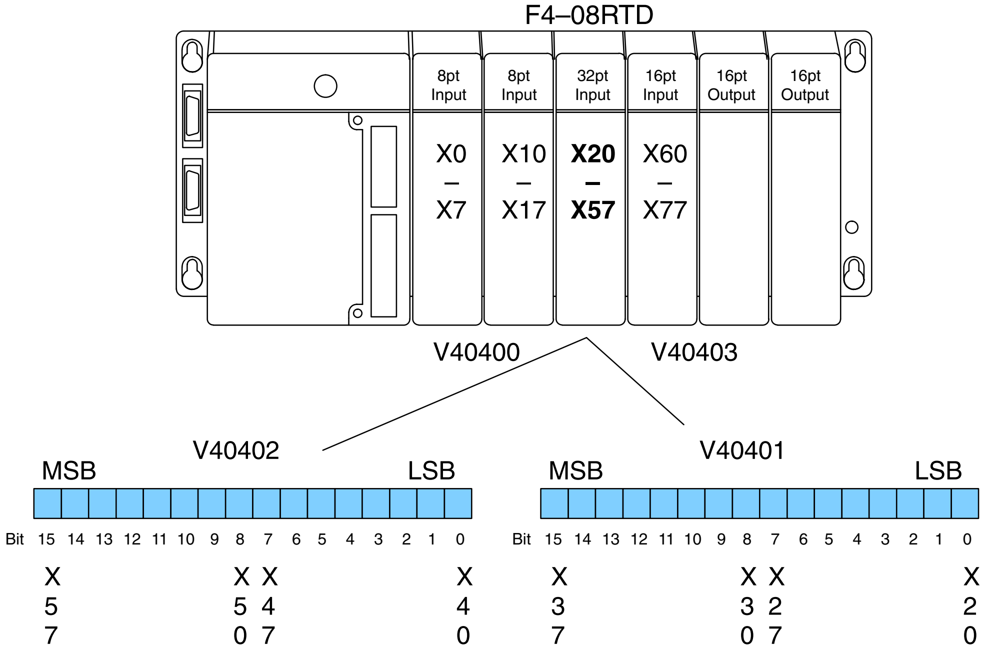
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the RTD transmitter signal and converts the signal to a 16-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete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

Identifying the Data Locations

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

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

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


F4–08RTD
8pt Input
8pt Input
32pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X57
X60 – X77
V40400
V40403
V40402
V40401
MSB
LSB
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X57
X50
X47
X40
X37
X30
X27
X20

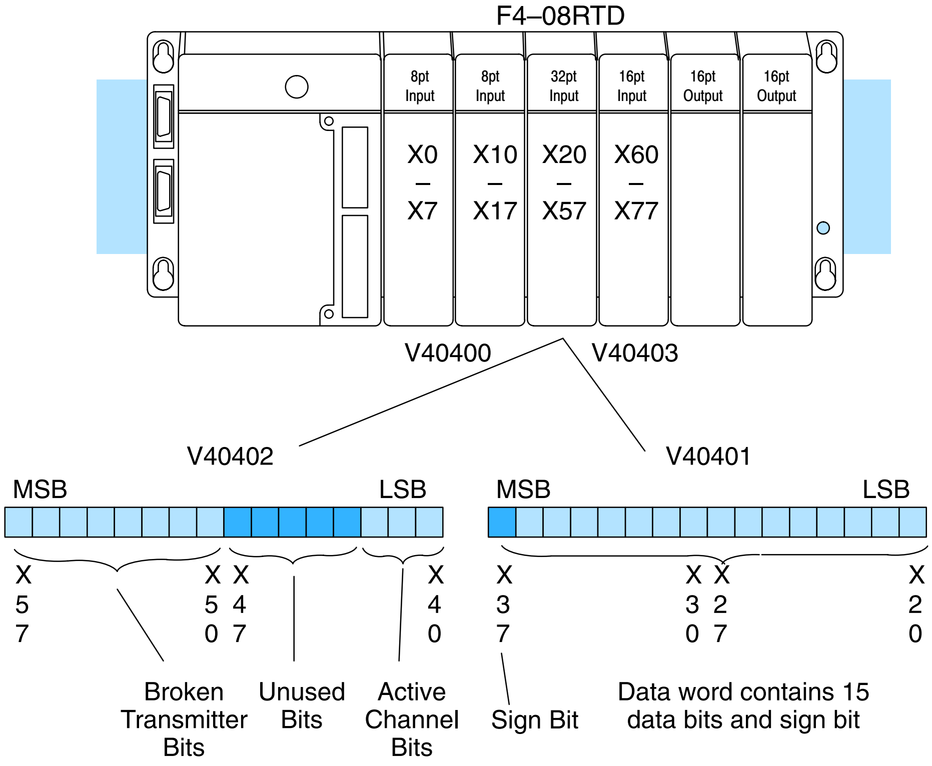
Writing the Control Program

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

Multiple Active Channels

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

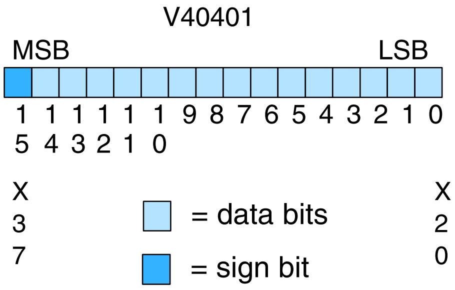
The control program must determine which channel's data is being sent from the module. If you have enabled only one channel, its data will be available on every scan. Two or more channels require demultiplexing the lower data word. Since the module communicates as X input points to the CPU, it is very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


F4–08RTD
8pt Input
8pt Input
32pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X57
X60 – X77
V40400
V40403
V40402
V40401
MSB
LSB
MSB
LSB
X57
X50
X47
X40
X37
X30
X27
X20
Broken Transmitter Bits
Unused Bits
Active Channel Bits
Sign Bit
Data word contains 15 data bits and sign bit

Analog Data and Sign Bits

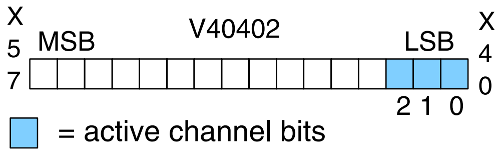
The first 16 bits represent the analog data in binary format. The MSB is the sign bit.

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


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X20
= data bits
= sign bit

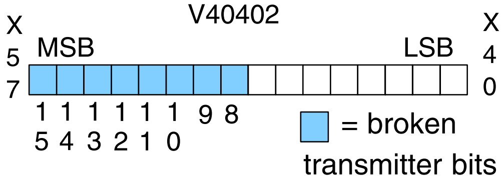
Active Channel Bits

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



Broken Transmitter Bits

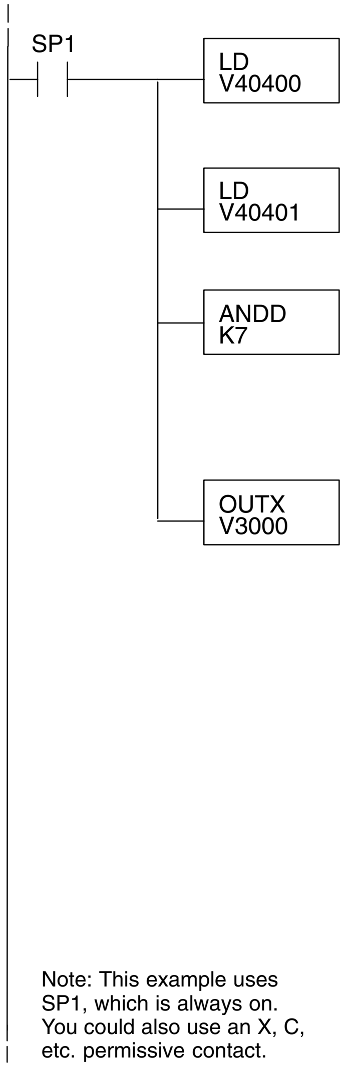
The broken transmitter bits are on when the corresponding RTD is open (00000001 = channel 1 is open, 00000010 = channel 2 is open, 11111111 = all eight channels are open, etc.).



Reading Values, DL430

✓ ✓ ✓
430 440 450

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



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

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

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

OUTX copies the 16-bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–7) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0), and if the binary value is 6, then channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

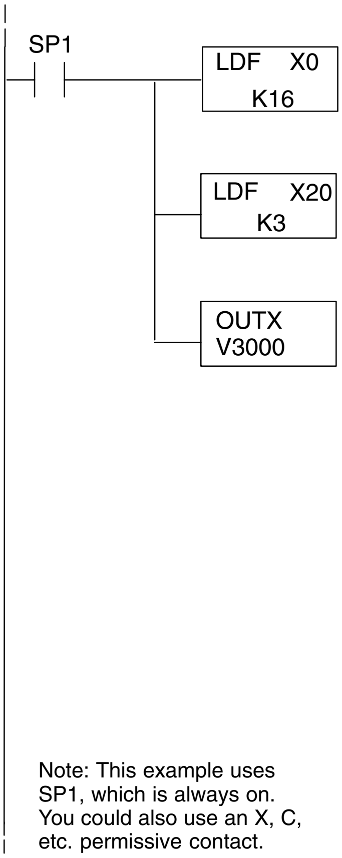
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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

Reading Values, DL440/450

☒ ☑ ☑
430 440 450

The following program example shows how to read the analog data into V-memory locations with DL440 and DL450 CPUs. Once the data is in V-memory, you can perform math on the data, compare the data against preset values, and so forth. This example will read one channel per scan, so it will take eight scans to read all eight channels. Contact SP1 is used in the example because the inputs are continually being updated. This example will not work with DL430 CPUs.



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

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

OUTX copies the 16-bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–7, which is the offset) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0), and if the binary value is 6, then channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

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

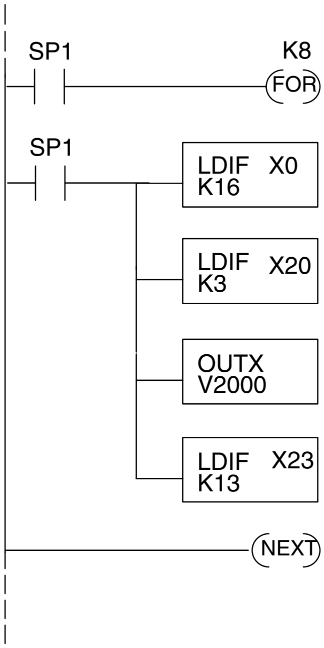
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Reading Eight Channels in One Scan, DL440/DL450

☒	✓	✓
430	440	450

The following program example shows how to read all eight channels in one scan by using a FOR/NEXT loop. This program only works with DL440 and DL450 CPUs. Before you try this method, remember that the FOR/NEXT routine shown here will add about 10–12 ms to the overall scan time. If you don't need to read the analog data on every scan, change SP1 to a permissive contact (such as an X input, CR, or stage bit) to only enable the FOR/NEXT loop when it is required.

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



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

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

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

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

Increments the temperature reading to the next channel.

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

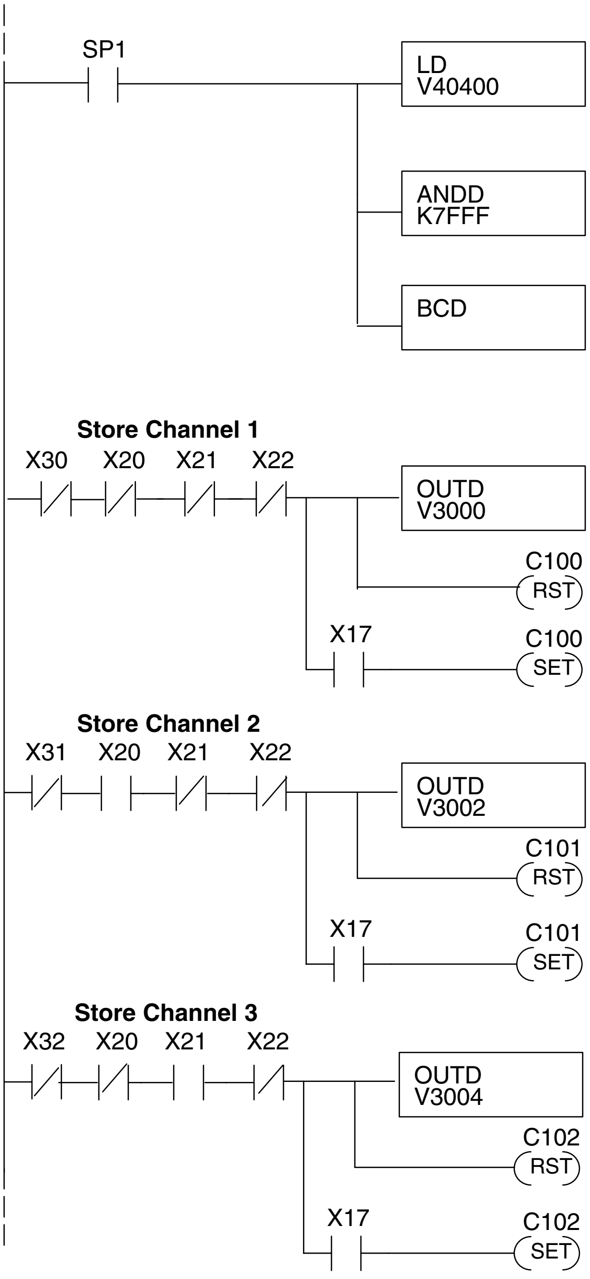
Note, this example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Using Bipolar Ranges (Magnitude Plus Sign)

☑ ☑ ☑
430 440 450

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

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



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

This instruction masks off the channel data and excludes the sign bit. Without this, the values used will not be correct, so do not forget to include it.

It is usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it. Do not use with internal PID loops because the PV requires binary data.

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

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

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

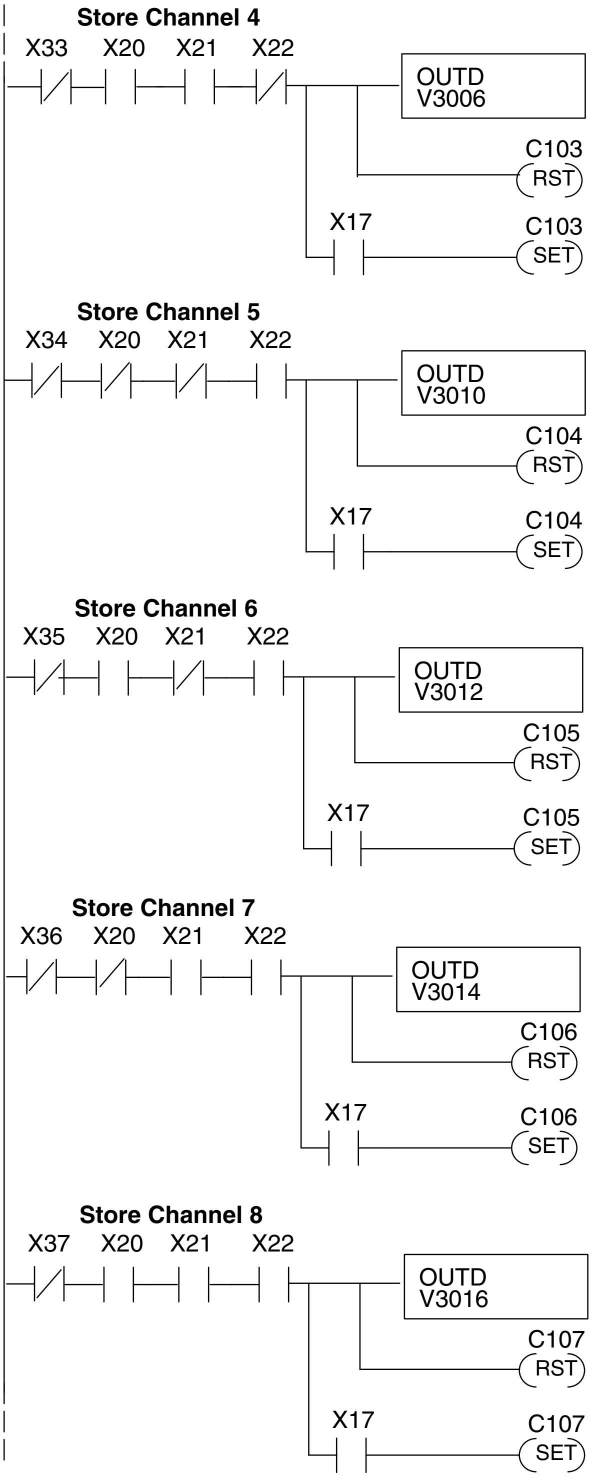
If the sign bit X17 is on, then control relay C101 is set. C101 can be used to indicate a negative channel 2 value or to call for a different message on an operator interface.

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

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

Program is continued on the next page.

Using Bipolar Ranges Example Continued


Store Channel 4
X33
X20
X21
X22
OUTD
V3006
C103
RST
X17
C103
SET
Store Channel 5
X34
X20
X21
X22
OUTD
V3010
C104
RST
X17
C104
SET
Store Channel 6
X35
X20
X21
X22
OUTD
V3012
C105
RST
X17
C105
SET
Store Channel 7
X36
X20
X21
X22
OUTD
V3014
C106
RST
X17
C106
SET
Store Channel 8
X37
X20
X21
X22
OUTD
V3016
C107
RST
X17
C107
SET

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

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

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

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

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

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

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

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

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

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

Reading the Input Data

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

10

F4-08THM 8-Channel Thermocouple Input

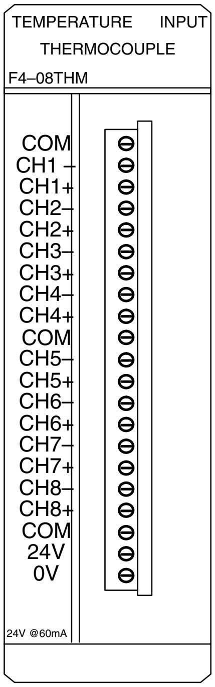
In This Chapter. . . .

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

Module Specifications

The F4-08THM 8-Channel Thermocouple Input Module provides several features and benefits.

- Eight thermocouple input channels with 16-bit voltage resolution or 0.1 °C/°F temperature resolution.
- Automatically converts type E, J, K, R, S, T, B, N, or C thermocouple signals into direct temperature readings. No extra scaling or complex conversion is required.
- Temperature data can be expressed in °F or °C.
- Module can be configured as ±5V, ±156mV, 0–5V or 0–156 mV and will convert volts and millivolt signal levels into 16-bit digital (0–65535) 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.



The following tables provide the specifications for the F4-08THM Analog Input Module. Review these specifications to make sure the module meets your application requirements.

General Specifications

Number of Channels	8, differential
Common Mode Range	±5VDC
Common Mode Rejection	90dB min. @ DC, 150dB min. @ 50/60 Hz.
Input Impedance	1MΩ
Absolute Maximum Ratings	Fault-protected inputs to ±50 VDC
Accuracy vs. Temperature	±5 ppm/°C maximum full scale calibration (including maximum offset change)
PLC Update Rate	1 channel per scan
Digital Inputs Input Points Required	16 binary data bits, 3 channel ID bits, 8 diagnostic bits 32 point (X) input module
External Power Supply	60 mA maximum, 18 to 26.4 VDC
Power Budget Requirement	110 mA maximum, 5 VDC (supplied by base)
Operating Temperature	0 to 60° C (32 to 140° F)
Storage Temperature	–20 to 70° C (–4 to 158° F)
Relative Humidity	5 to 95% (non-condensing)
Environmental air	No corrosive gases permitted
Vibration	MIL STD 810C 514.2
Shock	MIL STD 810C 516.2
Noise Immunity	NEMA ICS3–304

One count in the specification table is equal to one least significant bit of the analog data value (1 in 65535).

Thermocouple Specifications

Input Ranges	Type J –190 to 760°C –310 to 1400°F Type E –210 to 1000°C –346 to1832°F Type K –150 to 1372°C –238 to 2502°F Type R 65 to 1768°C 149 to 3214°F Type S 65 to 1768°C 149 to 3214°F Type T –230 to 400°C –382 to 752°F Type B 529 to 1820°C 984 to 3308°F Type N –70 to 1300°C –94 to 2372°F Type C 65 to 2320°C 149 to 4208°F
Display Resolution	± 0.1°C / ± 0.1°F
Cold Junction Compensation	Automatic
Warm-Up Time	30 min. typically ± 1°C repeatability
Linearity Error (End to End)	± .05°C maximum, ± .01°C typical
Maximum Inaccuracy	± 3°C (excluding thermocouple error)

Voltage Specifications

Voltage Ranges	Voltage: 0-5V, ±5V, 0-156.25mV, ± 156.25mVDC
Resolution	16 bit (1 in 65535)
Full Scale Calibration Error (Offset Error Included)	±13 counts typical, ±33 maximum
Offset Calibration Error	±1 count maximum, @ 0V input
Linearity Error (End to End)	±1 count maximum
Maximum Inaccuracy	±.02% @ 25°C (77°F)

Module Calibration

The F4-08THM module requires no calibration. The module automatically calibrates every five seconds, which removes offset and gain errors. For each thermocouple type, the temperature calculation and linearization performed by the microprocessor is accurate to within .01 °C.

Thermocouple Input Configuration Requirements

The F4-08THM module requires 32 discrete input points from the CPU. The module can be installed in any slot of a DL405 system. The limitations on the number of analog modules are:

- For local and expansion systems, the available power budget and number of 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 and I/O base for more information regarding power budget and number of local or remote I/O points.

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

Setting the Module Jumpers

Jumper Locations

Use the figure on the following page to locate the bank of ten jumpers on the PC board. Notice that the description of each jumper is just to the right of the jumpers on the PC board. To prevent losing a jumper when it is removed, store it in its original location by sliding one of its sockets over a single pin. You can select the following options by installing or removing the appropriate jumpers:

- Number of channels
- Input type
- Conversion units
- Calibrate enable

Calibrate Enable

See the following figure to locate the “Calibrate Enable” jumper. The jumper comes from the factory in the “jumper removed” setting (the jumper is installed over only one of the two pins). Installing this jumper disables the thermocouple active burn-out detection circuitry, which enables you to attach a thermocouple calibrator to the module.

To make sure that the output of the thermocouple calibrator is within the 5V common mode voltage range of the module, connect the negative side of the differential voltage input channel to the 0V terminal, then connect the thermocouple calibrator to the differential inputs (for example, Ch 3+ and Ch 3).

For the voltage input ranges, this jumper is inactive and can be installed or removed with no effect on voltage input.

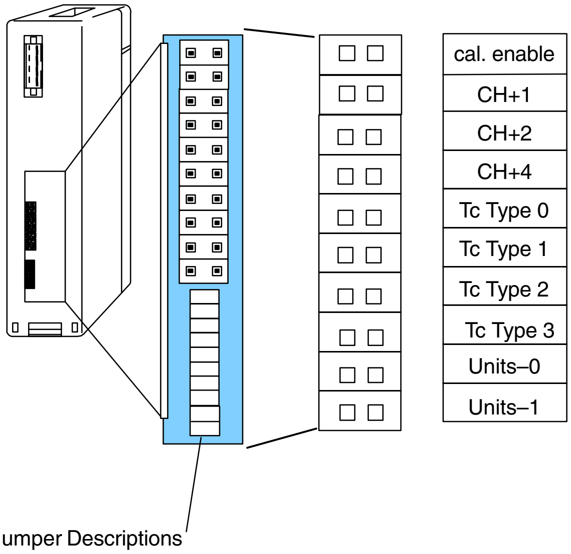
Selecting the Number of Channels

The next three jumpers labeled **CH+1**, **CH+2**, and **CH+4** determine the number of channels that will be used. The table shows how to set the jumpers for channels 1 thru 8. The module comes with three jumpers installed for eight channel operation. For example, to select channels 1 thru 3, remove the CH+1 and CH+4 jumpers and leave the CH+2 jumper installed. Any unused channels are not processed. For example, if you only select channels 1 thru 3, channels 4 through 8 will not be active.

X = jumper installed,

blank space = jumper removed

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


cal. enable
CH+1
CH+2
CH+4
Tc Type 0
Tc Type 1
Tc Type 2
Tc Type 3
Units–0
Units–1
Jumper Descriptions

Setting Input Type

The next four jumpers **(Tc Type 0, Tc Type 1, Tc Type 2, Tc Type 3)** must be set to match the type of thermocouple being used or the input voltage level. The module can be used with many types of thermocouples. Use the table to determine your settings.

The module comes from the factory with all four jumpers installed for use with a J type thermocouple. For example, to use an S type thermocouple, remove the jumper labeled Tc Type 2. All channels of the module must be the same thermocouple type or voltage range.

X = Jumper installed, and blank space = jumper removed.

Thermocouple / Voltage Inputs	Jumper			
	Tc Type 0	**Tc Type 1**	**Tc Type 2**	**Tc Type 3**
J	X	X	X	X
K		X	X	X
E	X		X	X
R			X	X
S	X	X		X
T		X		X
B	X			X
N				X
C	X	X	X	
0–5V.		X	X	
±5V.	X		X	
0–156mV.			X	
±156mV.	X	X		

Selecting the Conversion Units

Use the last two jumpers, **Units-0** and **Units-1**, to set the conversion unit used for either thermocouples or voltage inputs. The options are magnitude plus sign or 2's complement, plus Fahrenheit or Celsius for thermocouples. See the next two sections for jumper settings when using thermocouples or if using voltage inputs.

Thermocouple Conversion Units

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

For thermocouple ranges which include negative temperatures (J,E,K,T,N), the display resolution is from –3276.7 to +3276.7. For positive-only thermocouple ranges (R,S,B,C), the display resolution is 0 to 6553.5.

Negative temperatures can be represented in either 2's complement or magnitude plus sign form. If the temperature is negative, the most significant bit in the V-memory location is set (X17).

The 2's complement data format may be required to correctly display bipolar data on some operator interfaces. This data format could also be used to simplify averaging a bipolar signal. To view this data format in ***Direct***Soft, select Signed Decimal.

For unipolar thermocouple ranges (R,S,B,C), it does not matter if magnitude plus sign or 2's complement is selected.

Use the table to select settings. The module comes with both jumpers installed for magnitude plus sign conversion in Fahrenheit. For example, remove the Units-0 jumper and leave the Units-1 jumper installed for magnitude plus sign conversion in Celsius.

X = Jumper installed, and blank space = jumper removed.

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

Voltage Conversion Units

The bipolar voltage input ranges, ±5V or ±156mV (see previous page for ±5V and ±156mV settings), may be converted to a 15-bit magnitude plus sign or a 16-bit 2's complement value.

Use the table to select settings. The module comes with both jumpers installed for magnitude plus sign conversion. Remove the Units-1 jumper and leave the Units-0 jumper installed for 2's complement conversion.

X = Jumper installed, and blank space = jumper removed.

Jumper Pins	Voltage Conversion Units	
	Magnitude Plus Sign	2's Complement
Units-0	X	X
Units-1	X	

Connecting the Field Wiring

Wiring Guidelines

Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider:

- Use the shortest wiring route whenever possible.
- Use shielded wiring and ground the shield at the transmitter 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.
- Connect wiring from all unused channels to common.
- 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 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 400 mA 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: 24VDC ± 10%, Class 2, 75mA.

It is desirable in some situations to power the transmitters separately in a location remote from the PLC. This will work as long as the transmitter supply meets the voltage and current requirements and the transmitter's minus (–) side and the module supply's minus (–) side are connected together.

WARNING: If you are using the 24 VDC base power supply, 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.

The DL405 base has a switching type power supply. As a result of switching noise, you may notice some instability in the analog input data if you use the base power supply. If this is unacceptable, you should try one of the following:

1. Use a separate linear power supply.
2. Connect the 24VDC common to the frame ground, which is the screw terminal marked "G" on the base.

Unused temperature inputs should be shorted together and connected to common.

Thermocouples

Use shielded thermocouples whenever possible to minimize the presence of noise on the thermocouple wire. Ground the shield wire at one end only. For grounded thermocouples, connect the shield at the sensor end. For ungrounded thermocouples, connect the shield to the 0V (common) terminal.

Grounded Thermocouple Assembly

A grounded thermocouple provides better response time than an ungrounded thermocouple because the tip of the thermocouple junction is in direct contact with the protective case.

Ungrounded Thermocouple Assembly

An ungrounded thermocouple is electrically isolated from the protective case. If the case is electrically grounded it provides a low-impedance path for electrical noise to travel. The ungrounded thermocouple provides a more stable and accurate measurement in a noisy environment.

Exposed Grounded Thermocouple

The thermocouple does not have a protective case and is directly connected to a device with a higher potential. Grounding the thermocouple assures that the thermocouple remains within the common mode specifications. Because a thermocouple is essentially a wire, it provides a low-impedance path for electrical noise. The noise filter has a response of >100dB @ 50/60 Hz.

WARNING: A thermocouple can become shorted to a high voltage potential. Because common terminals are internally connected together, whatever voltage potential exists on one thermocouple will exist on the other channels.

Ambient Variations in Temperature

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

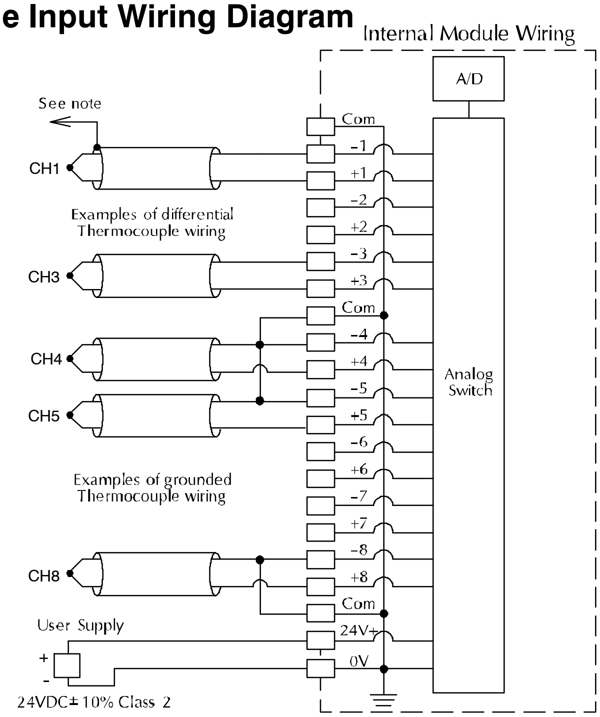
The cold junction compensation is calibrated to operate in a still-air environment. If the module is used in an application that has forced convection cooling, an error of 2–3°C may be introduced. To compensate for this you can use ladder logic to correct the values.

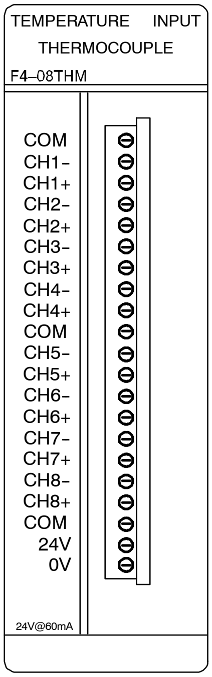
When configuring the system design it is best to locate any heat-producing devices above and away from the PLC chassis because the heat will affect the temperature readings. For example, heat introduced at one end of the terminal block can cause a channel-to-channel variation.

When exposing the F4-08THM module to abrupt ambient temperature changes it will take several minutes for the cold junction compensation and terminal block to stabilize. Errors introduced by abrupt ambient temperature changes will be less than 4°C.

Wiring Diagram Use the following diagrams to connect the field wiring.

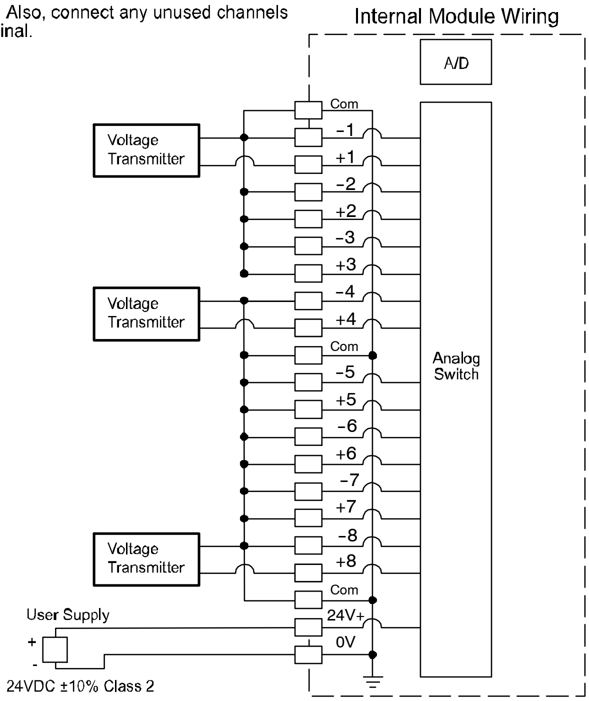
Thermocouple Input Wiring Diagram


Internal Module Wiring
A/D
See note
Com
-1
+1
-2
+2
-3
+3
Com
-4
+4
-5
+5
-6
+6
-7
+7
-8
+8
Com
24V+
0V
CH1
Examples of differential Thermocouple wiring
CH3
CH4
CH5
Examples of grounded Thermocouple wiring
CH8
Analog Switch
User Supply
+
-
24VDC± 10% Class 2


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@60mA

Voltage Input Wiring Diagram

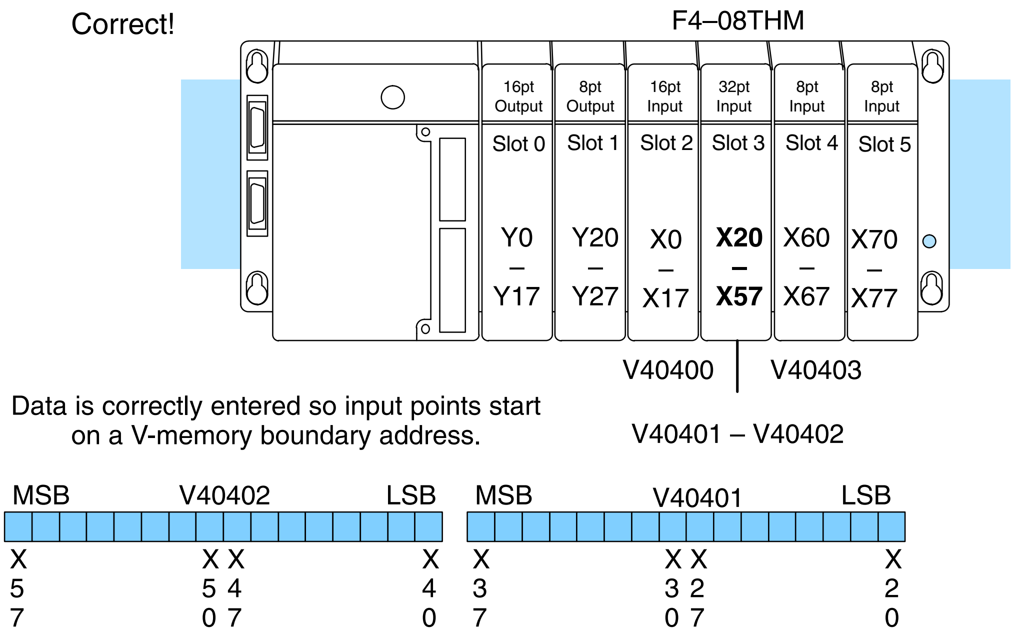
Note 3: When using 0-156mV and 5V ranges, connect CH- terminal to Com or 0V terminal to ensure common mode range acceptance. Also, connect any unused channels to Com or 0V terminal.


Internal Module Wiring
A/D
Com
-1
+1
-2
+2
-3
+3
-4
+4
Com
-5
+5
-6
+6
-7
+7
-8
+8
Com
24V+
0V
Voltage Transmitter
Voltage Transmitter
Voltage Transmitter
Analog Switch
User Supply
+
-
24VDC ±10% Class 2

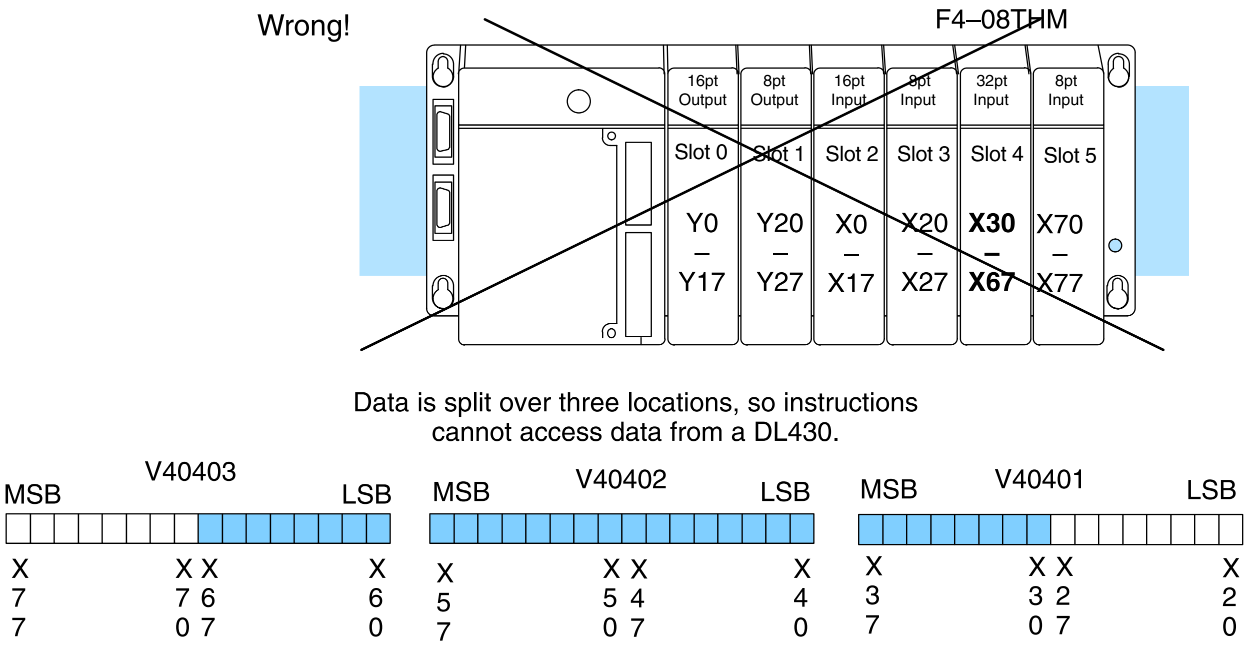
Module Operation

DL430 Special Requirements

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


Correct!
F4–08THM
16pt Output
8pt Output
16pt Input
32pt Input
8pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y17
Y20 – Y27
X0 – X17
X20 – X57
X60 – X67
X70 – X77
V40400
V40403
V40401 – V40402
MSB
V40402
LSB
MSB
V40401
LSB
X57
X50
X47
X40
X37
X30
X27
X20

Data is correctly entered so input points start on a V-memory boundary address.


Wrong!
F4–08THM
16pt Output
8pt Output
16pt Input
8pt Input
32pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y17
Y20 – Y27
X0 – X17
X20 – X27
X30 – X67
X70 – X77
MSB
V40403
LSB
MSB
V40402
LSB
MSB
V40401
LSB
X77
X70
X67
X60
X57
X50
X47
X40
X37
X30
X27
X20

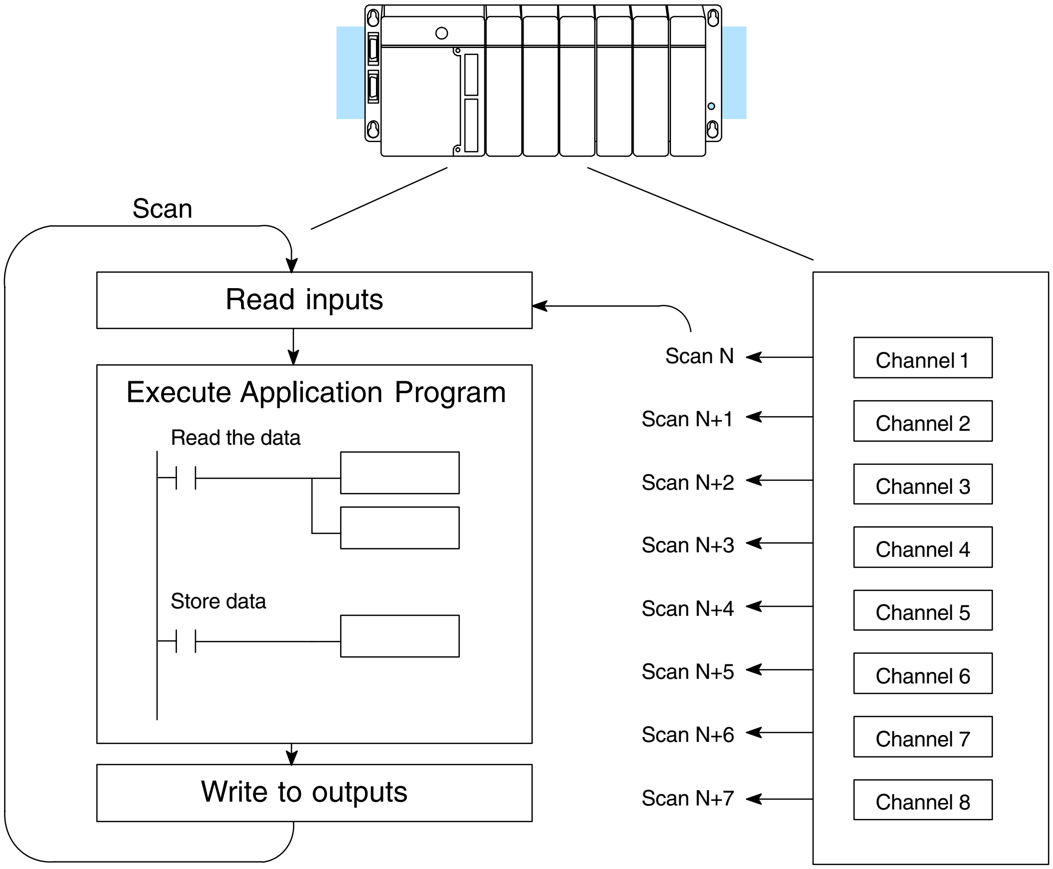
Data is split over three locations, so instructions cannot access data from a DL430.

Channel Scanning Sequence

Before you begin writing the control program, it is important to take a few minutes to understand how the module processes and represents the analog signals.

The F4–08THM 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.

Unused channels are not processed, so if you select only two channels, then each channel will be updated every other scan.


Scan
Read inputs
Execute Application Program
Read the data
Store data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Channel 1
Channel 2
Channel 3
Channel 4
Channel 5
Channel 6
Channel 7
Channel 8

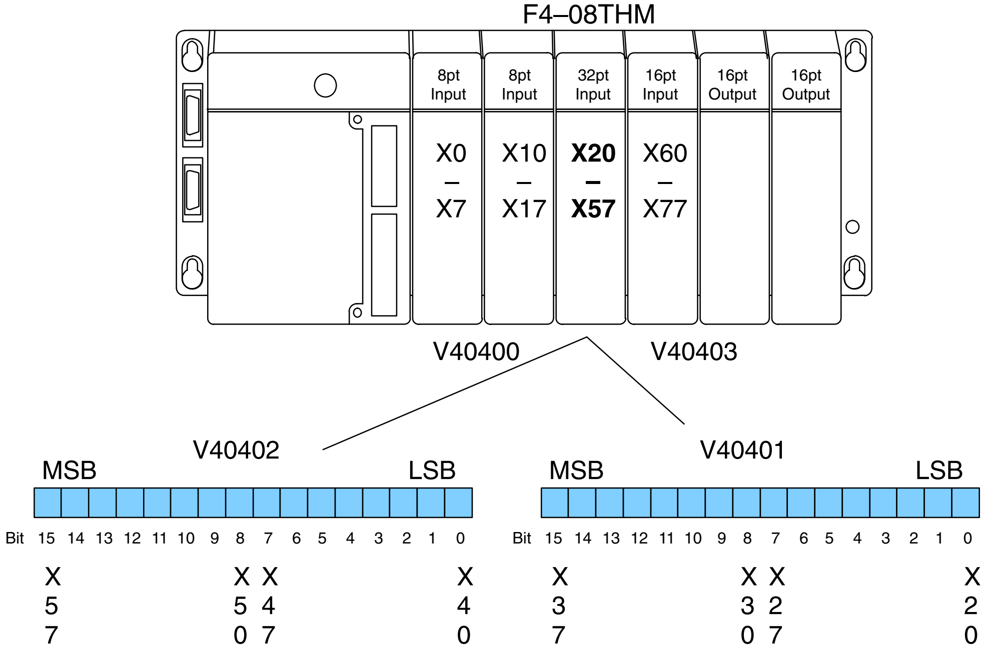
Even though the channel updates to the CPU are synchronous with the CPU scan, the module asynchronously monitors the thermocouple signal and converts the signal to a 16-bit binary representation. This enables the module to continuously provide accurate measurements without slowing down the discrete control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

Identifying the Data Locations

The F4–08THM module requires 32-point discrete input points (five bits are unused). These inputs provide:

- Three active channel bits
- A digital representation of the analog signal in 16 bits, including one sign bit.
- Individual broken transmitter detection bits for each channel.

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


F4–08THM
8pt Input
8pt Input
32pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X57
X60 – X77
V40400
V40403
V40402
V40401
MSB
LSB
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X57
X50
X47
X40
X37
X30
X27
X20

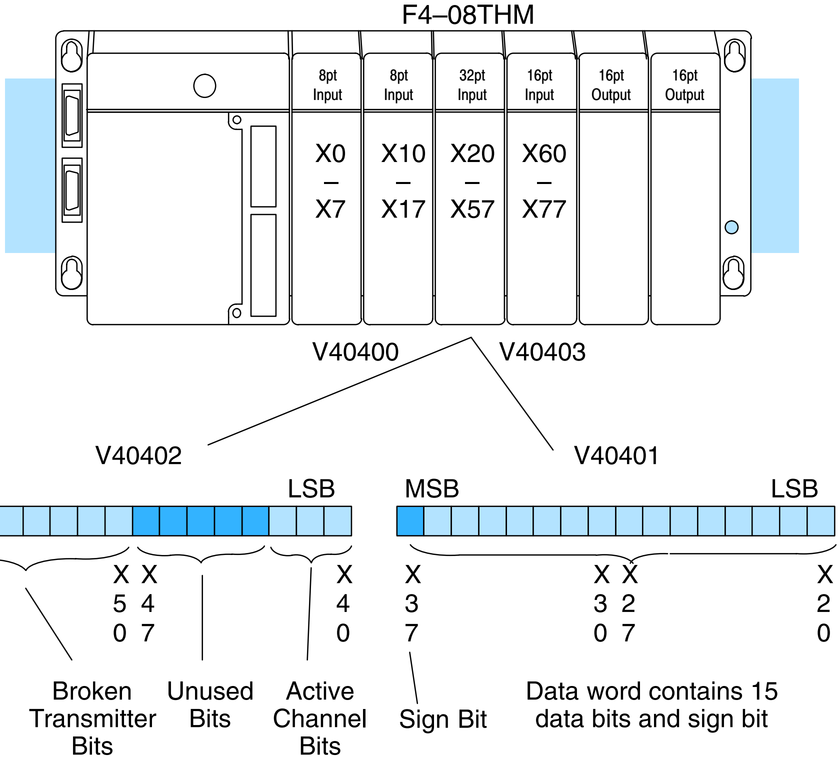
Writing the Control Program

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

Multiple Active Channels

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

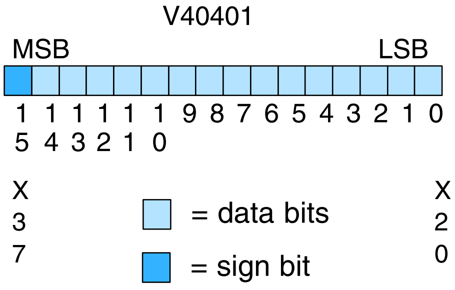
The control program must determine which channel's data is being sent from the module. If you have enabled only one channel, its data will be available on every scan. Two or more channels require demultiplexing the lower data word. Since the module communicates as X input points to the CPU, it is very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


F4–08THM
8pt Input
8pt Input
32pt Input
16pt Input
16pt Output
16pt Output
X0 – X7
X10 – X17
X20 – X57
X60 – X77
V40400
V40403
V40402
V40401
MSB
LSB
MSB
LSB
X57
X50
X47
X40
X37
X30
X27
X20
Broken Transmitter Bits
Unused Bits
Active Channel Bits
Sign Bit
Data word contains 15 data bits and sign bit

Analog Data and Sign Bits

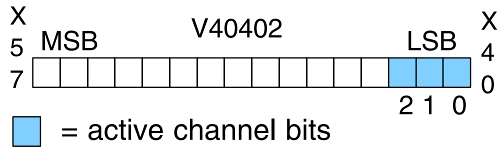
The first 16 bits represent the analog data in binary format. The MSB is the sign bit.

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


V40401
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
X37
X20
= data bits
= sign bit

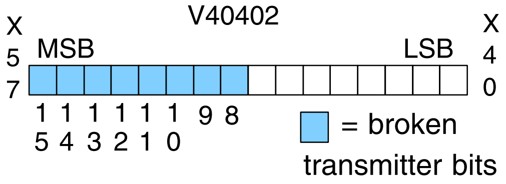
Active Channel Bits

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



Broken Transmitter Bits

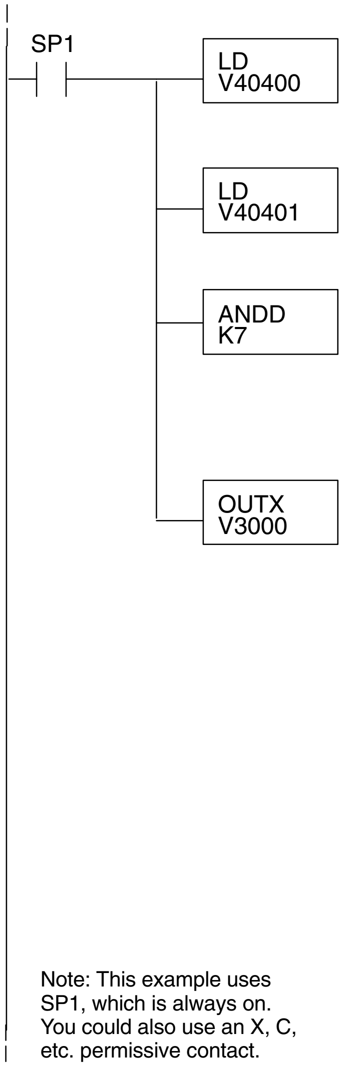
The broken transmitter bits are on when the corresponding thermocouple is open (00000001 = channel 1 is open, 00000010 = channel 2 is open, 11111111 = all eight channels are open, etc.).



Reading Values, DL430

✓ ✓ ✓
430 440 450

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



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

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

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

OUTX copies the 16-bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–7) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0), and if the binary value is 6, then channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

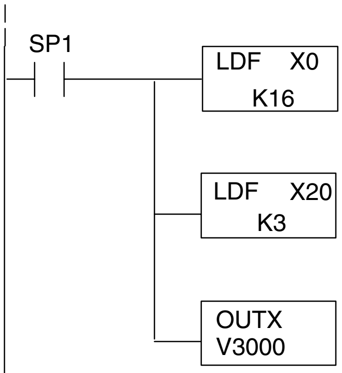
Module Reading	Acc. Bits	Offset	Data Stored in ...
Channel 1	000	0	V3000
Channel 2	001	1	V3001
Channel 3	010	2	V3002
Channel 4	011	3	V3003
Channel 5	100	4	V3004
Channel 6	101	5	V3005
Channel 7	110	6	V3006
Channel 8	111	7	V3007

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

Reading Values, DL440/450

✗	✓	✓
430	440	450

The following program example shows how to read the analog data into V-memory locations with DL440 and DL450 CPUs. Once the data is in V-memory, you can perform math on the data, compare the data against preset values, and so forth. This example will read one channel per scan, so it will take eight scans to read all eight channels. Contact SP1 is used in the example because the inputs are continually being updated. This example will not work with DL430 CPUs.



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

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

OUTX copies the 16-bit value from the first level of the accumulator stack to a source address offset by the value in the accumulator. In this case it adds the above binary value (0–7, which is the offset) to V3000. The particular channel data is then stored in its respective location: For example, if the binary value of the channel select bits is 0, then channel 1 data is stored in V-memory location V3000 (V3000 + 0), and if the binary value is 6, then channel 7 data is stored in location V3006 (V3000 + 6). See the following table.

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

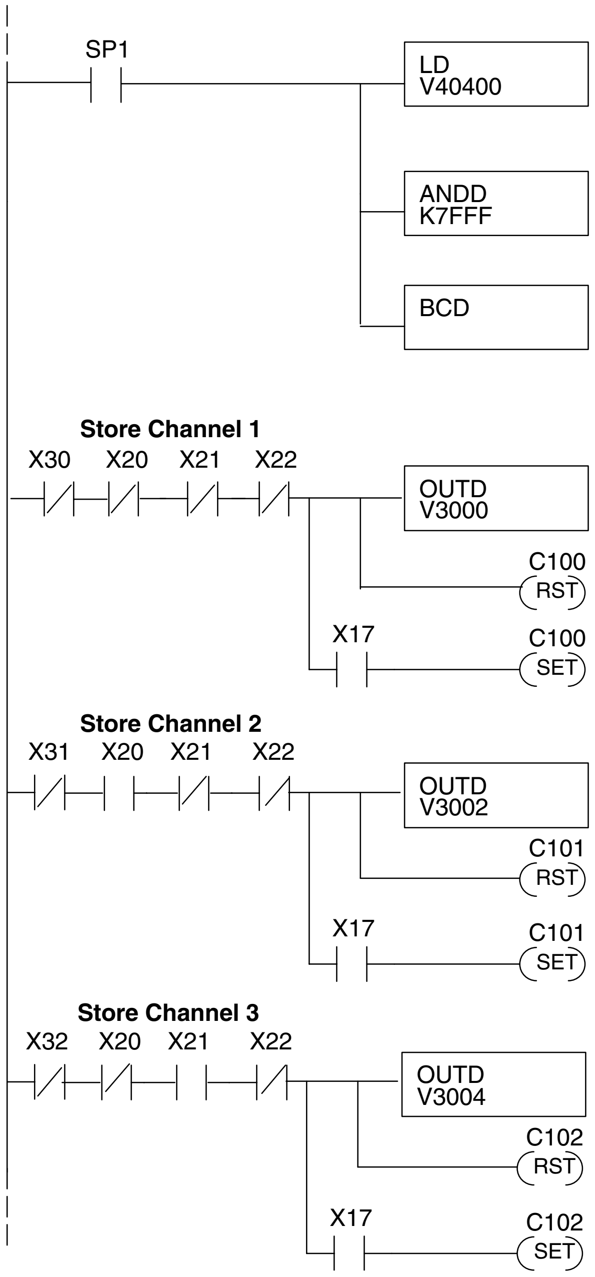
Note: This example uses SP1, which is always on. You could also use an X, C, etc. permissive contact.

Using Bipolar Ranges (Magnitude Plus Sign)

✓ ✓ ✓
430 440 450

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

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



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

This instruction masks off the channel data and excludes the sign bit. Without this, the values used will not be correct, so do not forget to include it.

It is usually easier to perform math operations in BCD, so it is best to convert the data to BCD immediately. You can leave out this instruction if your application does not require it. Do not use with internal PID loops because the PV requires binary data.

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

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

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

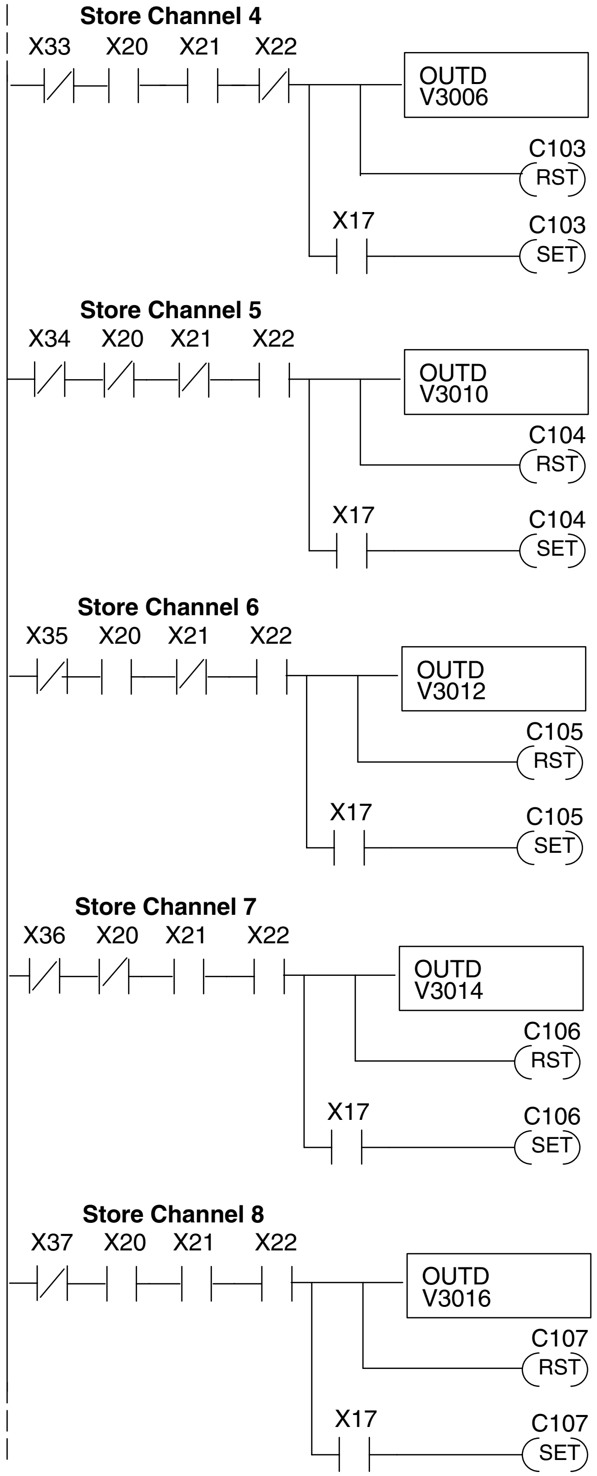
If the sign bit X17 is on, then control relay C101 is set. C101 can be used to indicate a negative channel 2 value or to call for a different message on an operator interface.

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

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

Program is continued on the next page.

Using Bipolar Ranges Example Continued


Store Channel 4
X33
X20
X21
X22
OUTD
V3006
C103
RST
X17
C103
SET
Store Channel 5
X34
X20
X21
X22
OUTD
V3010
C104
RST
X17
C104
SET
Store Channel 6
X35
X20
X21
X22
OUTD
V3012
C105
RST
X17
C105
SET
Store Channel 7
X36
X20
X21
X22
OUTD
V3014
C106
RST
X17
C106
SET
Store Channel 8
X37
X20
X21
X22
OUTD
V3016
C107
RST
X17
C107
SET

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

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

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

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

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

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

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

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

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

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

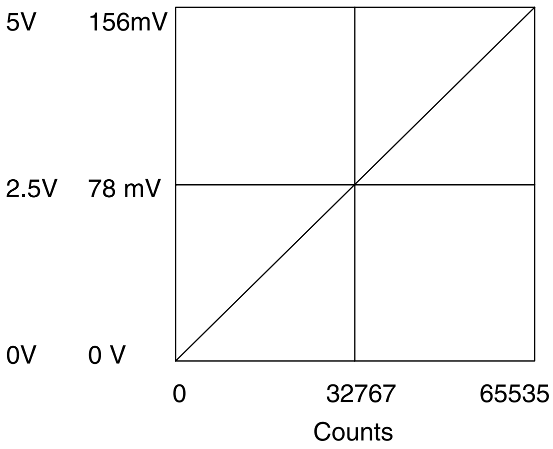
Reading the Input Data

The temperature readings have one implied decimal point. For example, a reading of 10273 is actually 1027.3 degrees.

Module Resolution 16-Bit (Unipolar Voltage Input)

☑ ☑ ☑
430 440 450

Unipolar analog signals are converted into 65536 counts ranging from 0 to 65535 (2^{16}). For example, with a 0 to 156mV signal range, 78mV would be 32767. A value of 65535 represents the upper limit of the range.



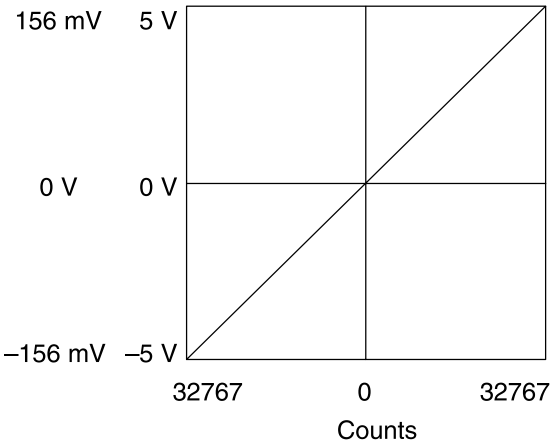
$$\text{Unipolar Resolution} = \frac{H - L}{65535}$$

H or L = high or low limit of the range

Module Resolution 15-Bit Plus Sign (Bipolar Voltage Input)

☑ ☑ ☑
430 440 450

The module has 16-bit unipolar or 15-bit + sign bipolar resolution. Bipolar analog signals are converted into 32768 counts ranging from 0 to 32767 (2^{15}). For example, with a –156mV to 156mV signal range, 156mV would be 32767. The bipolar ranges utilize a sign bit to provide 16-bit resolution. A value of 32767 can represent the upper limit of either side of the range. Use the sign bit to determine negative values.



$$\text{Bipolar Resolution} = \frac{H - L}{32767}$$

H or L = high or low limit of the range

Analog and Digital Value Conversions

✓ ✓ ✓
430 440 450

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. Remember, this module *does not* operate like other versions of analog input modules that you may be familiar with. The bipolar ranges use 0–32767 for both positive and negative voltages. The sign bit allows this and it actually provides better resolution than those modules that do not offer a sign bit. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the signal level ...
0 to 5V	$A = \frac{5D}{65535}$	$D = \frac{65535}{5}(A)$
0 to 156.25mV	$A = \frac{0.15625D}{65535}$	$D = \frac{65535}{0.15625}(A)$
±5V	$A = \frac{10D}{65535}$	$D = \frac{65535}{10}(A)$
±156.25mV	$A = \frac{0.3125D}{65535}$	$D = \frac{65535}{0.3125}(A)$

For example, if you are using the ±5V range and you have measured the signal at 2.5V, use the following formula to determine the digital value that is stored in the V-memory location that contains the data.

$$D = \frac{65535}{10}(A)$$

$$D = \frac{65535}{10}(2.5V)$$

$$D = (6553.5)\,(2.5)$$

$$D = 16383.75$$

11

D4-02DA 2-Channel Analog Output

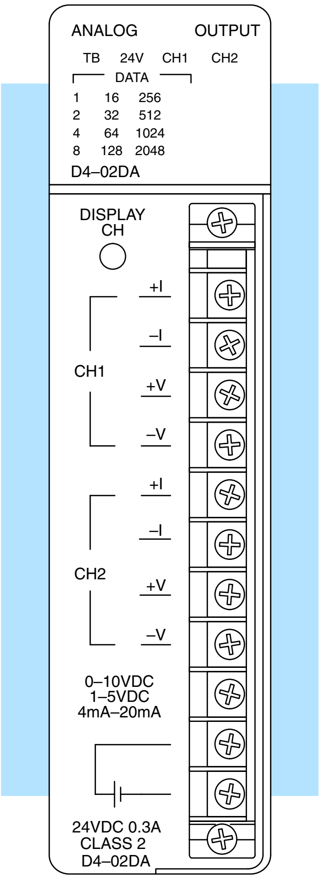
In This Chapter. . . .

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

Module Specifications

The D4–02DA 2-Channel Analog Output module provides several features and benefits.

- It provides two channels of voltage or current outputs.
- 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.
- Both analog outputs may be set in one CPU scan.


ANALOG
OUTPUT
TB
24V
CH1
CH2
DATA
1
16
256
2
32
512
4
64
1024
8
128
2048
D4–02DA
DISPLAY CH
+I
–I
CH1
+V
–V
+I
–I
CH2
+V
–V
0–10VDC
1–5VDC
4mA–20mA
24VDC 0.3A
CLASS 2
D4–02DA

Analog Output Configuration Requirements

The D4–02DA Analog Output requires 32 discrete output points from the CPU. The module can be installed in any slot of a DL405 system, including remote bases. The limitations on the number of analog modules are:

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

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

The following table provides the specifications for the D4–02DA Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

Number of Channels	2 (independent)
Output Ranges	0–10V, 1–5V, 4–20 mA
Resolution	12 bit (1 in 4096)
Output Type	Single ended
Output Impedance	0.5Ω maximum, voltage output
Output Current	5 mA maximum, voltage output
Load Impedance	550Ω max., 5.0Ω min.,current output, 2KΩ minimum, voltage output
Linearity	± 0.1% maximum
Accuracy vs. Temperature	± 70 ppm / °C maximum
Maximum Inaccuracy	± 0.2% maximum at 25° C
Conversion Method	Integration
Conversion Time	Start of scan, 30µS + one scan

General Module Specifications

PLC Update Rate	1 or 2 channels per scan
Digital Output Points Required	32 (Y) output points 12 binary data bits per channel (24 bits total with 8 unused bits)
Power Budget Requirement	250 mA (from base)
External Power Supply	24VDC, ± 10%, 300 mA, class 2
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
Insulation Resistance	10 MΩ, 500 VDC
Noise Immunity	NEMA ICS3–304

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.
- 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 D4–02DA 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. If you only have one analog module, you can use this power source instead of a separate supply. If you have more than one analog module, or you would rather use a separate supply, choose one that meets the following requirements: 24 VDC ±10%, Class 2, 300mA (per module).

Load Requirements

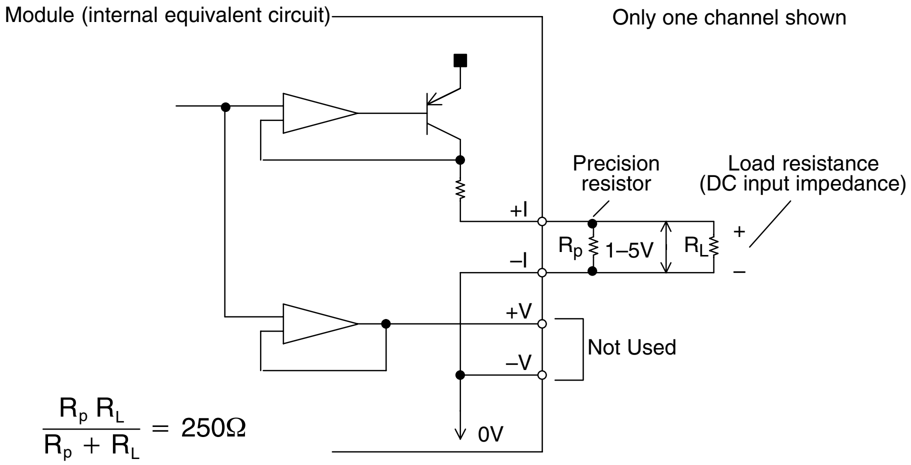
Each channel can be wired independently for voltage or current.

- Current loads must have an impedance between 5 and 500 ohms.
- Voltage loads must have an impedance greater than 2K ohms.

Using the 1–5 VDC Range

Since the module provides a 0–10V signal on the voltage outputs, you have to use the current outputs and a precision resistor to generate the 1–5V signal.

With a 250 ohm precision resistor across the current outputs, the module converts the current signals to voltage for you (4mA x 250 ohms = 1V, 20mA x 250 ohms = 5V). The transducer should be connected in parallel with the precision resistor and the parallel equivalent resistance should be approximately 250 ohms (±0.1%). Field devices that have impedances of greater than 250K ohms can be used with less than 0.1% additional inaccuracy. See the following diagram.



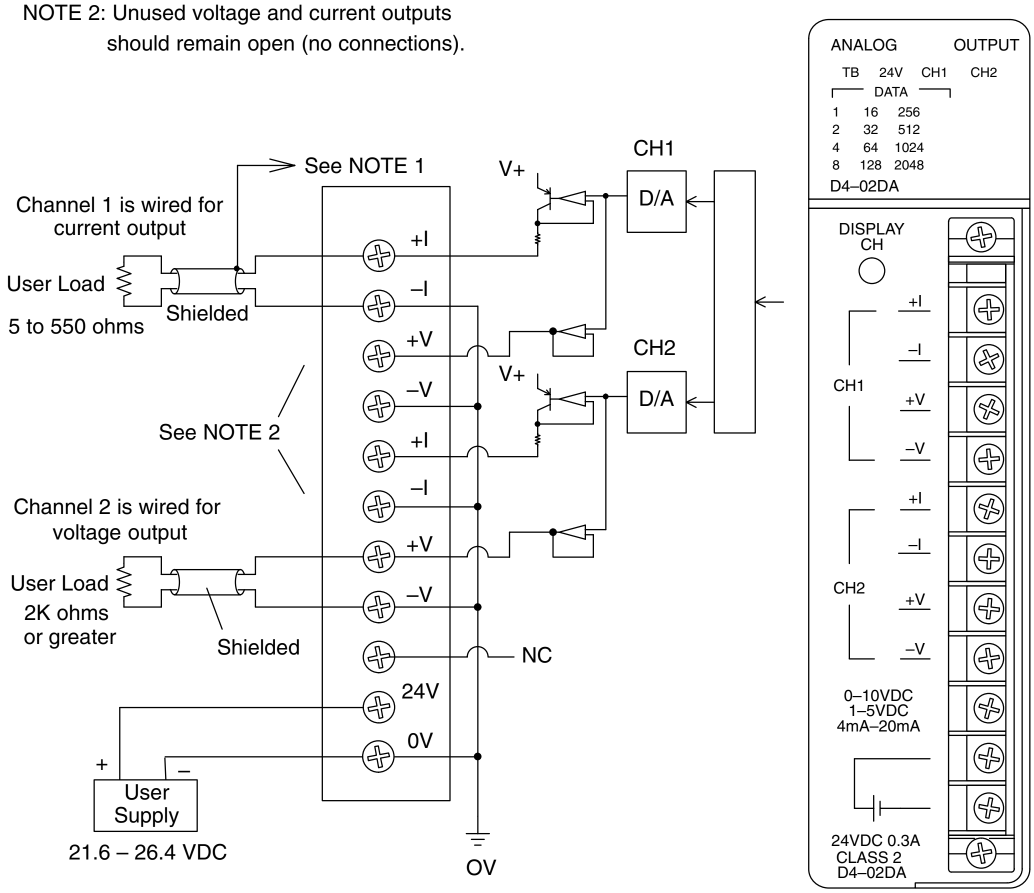
$$\frac{R_p\,R_L}{R_p + R_L} = 250\Omega$$

R_p = value of external precision resistor

R_L= load impedance

Wiring Diagram

The D4–02DA module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.



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

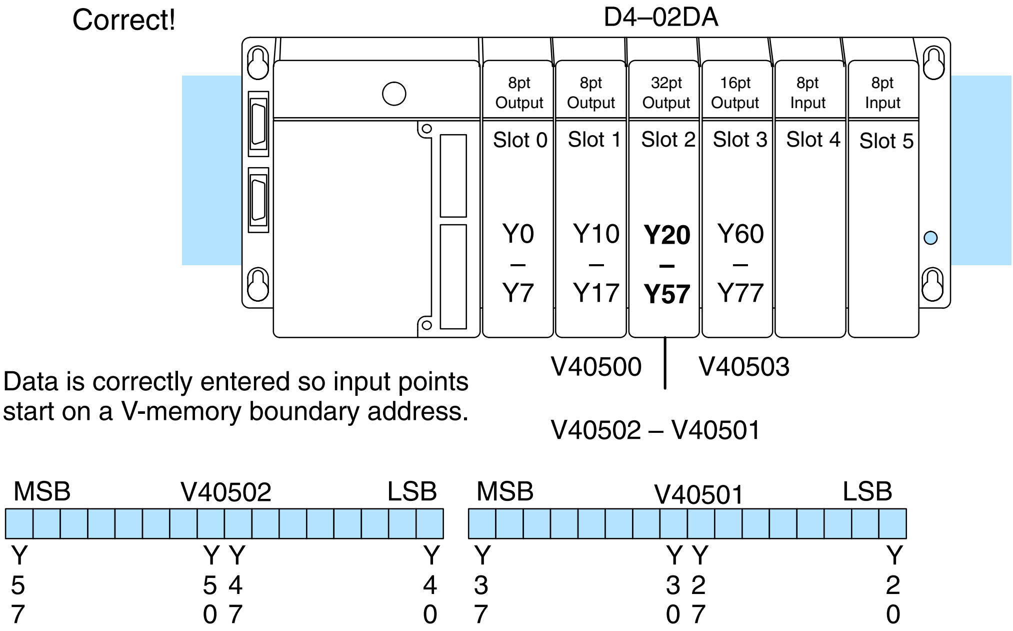
NOTE 2: Unused voltage and current outputs should remain open (no connections).

See NOTE 1
Channel 1 is wired for current output
User Load
5 to 550 ohms
Shielded
See NOTE 2
Channel 2 is wired for voltage output
User Load
2K ohms or greater
Shielded
+
–
User Supply
21.6 – 26.4 VDC
+I
–I
+V
–V
+I
–I
+V
–V
24V
0V
NC
OV
V+
V+
CH1
D/A
CH2
D/A
ANALOG
OUTPUT
TB 24V CH1 CH2
DATA
1 16 256
2 32 512
4 64 1024
8 128 2048
D4–02DA
DISPLAY CH
CH1
+I
–I
+V
–V
CH2
+I
–I
+V
–V
0–10VDC
1–5VDC
4mA–20mA
24VDC 0.3A
CLASS 2
D4–02DA

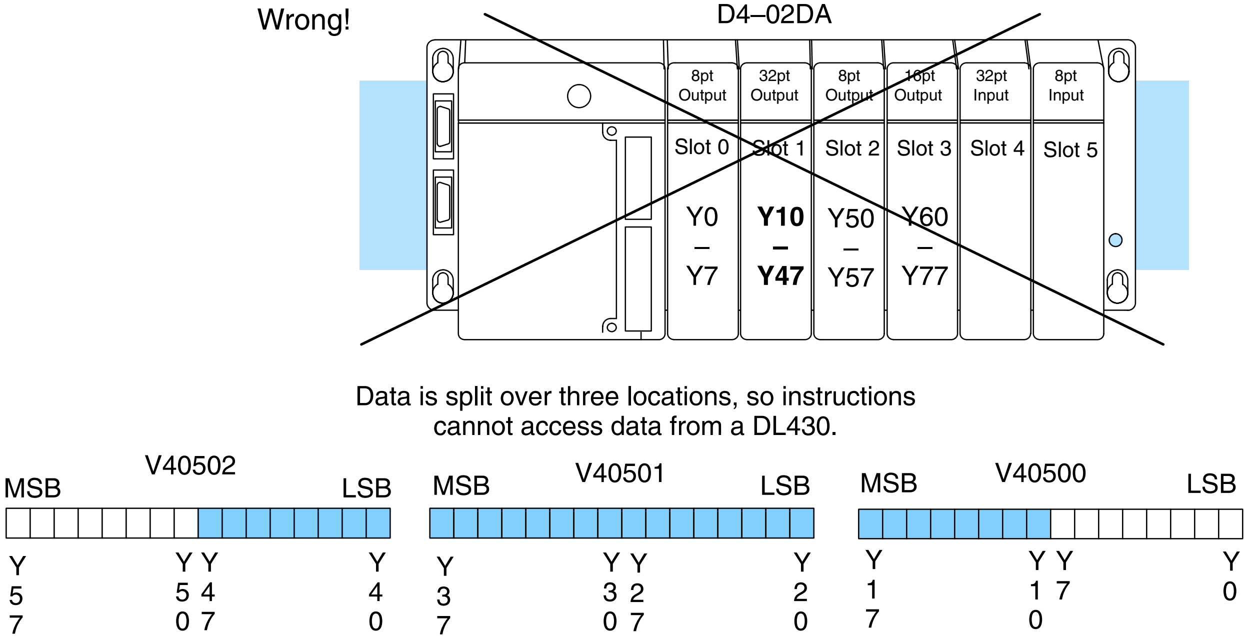
Module Operation

DL430 Special Requirements

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


Correct!
D4–02DA
8pt Output
8pt Output
32pt Output
16pt Output
8pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502 – V40501
MSB
V40502
LSB
MSB
V40501
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20

Data is correctly entered so input points start on a V-memory boundary address.

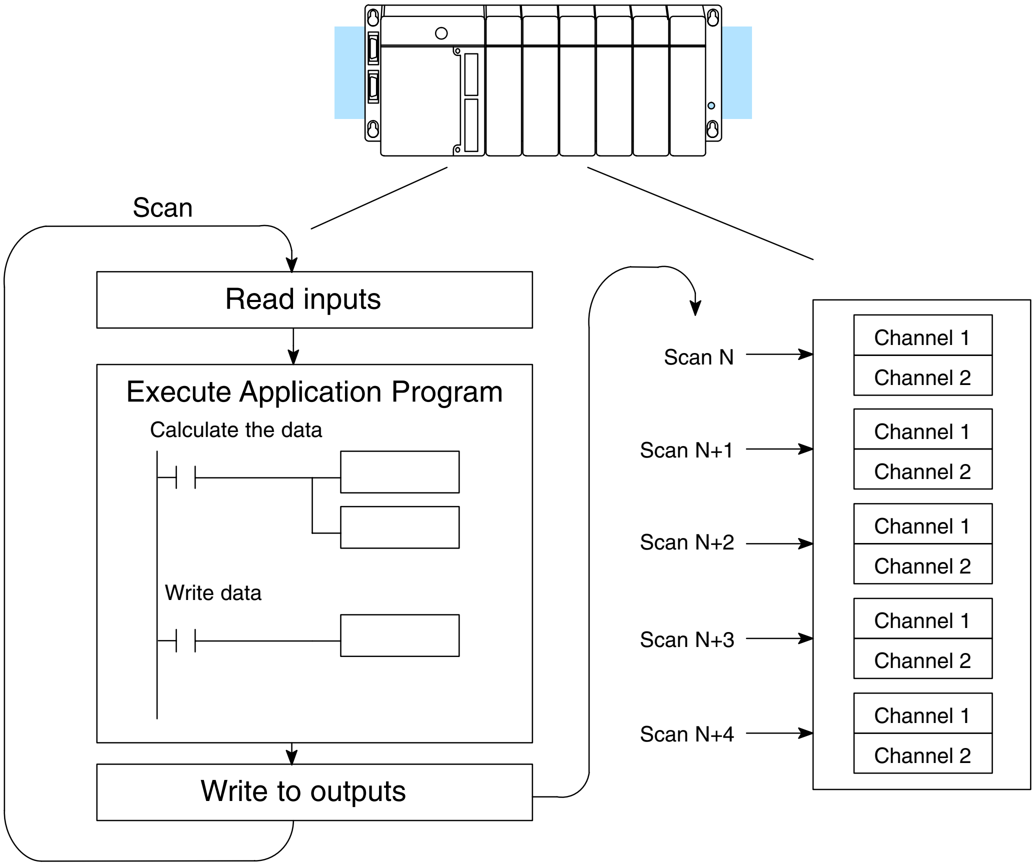

Wrong!
D4–02DA
8pt Output
32pt Output
8pt Output
16pt Output
32pt Input
8pt Input
Slot 0
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Y0 – Y7
Y10 – Y47
Y50 – Y57
Y60 – Y77
MSB
V40502
LSB
MSB
V40501
LSB
MSB
V40500
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20
Y17
Y10
Y7
Y0

Data is split over three locations, so instructions cannot access data from a DL430.

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

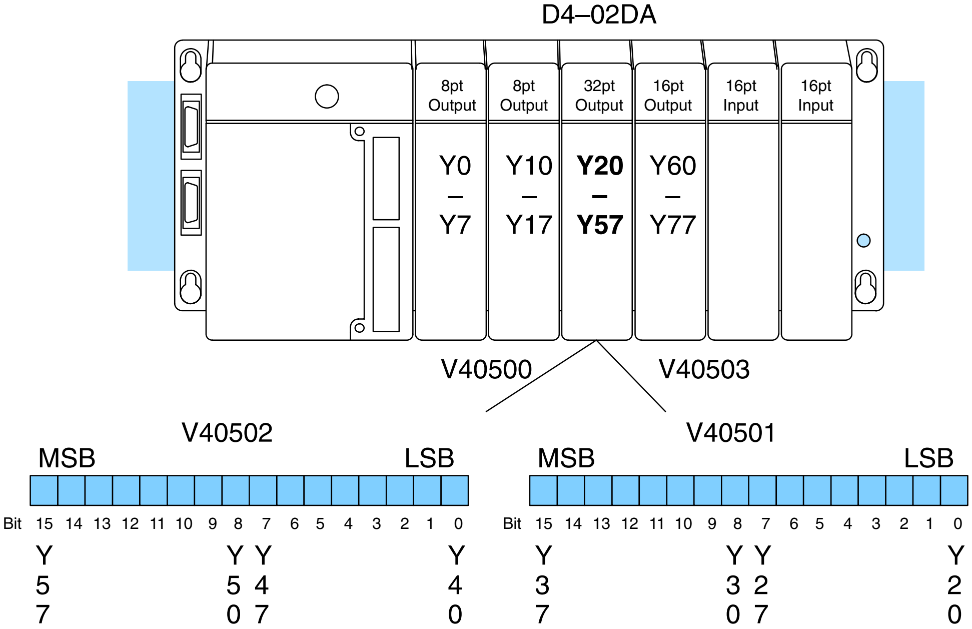
The D4–02DA module updates both channels in the same scan. The control program updates the two channels of this module independent of each other and each channel does not have to be refreshed on every scan.


Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 1
Channel 2
Channel 1
Channel 2
Channel 1
Channel 2
Channel 1
Channel 2

Output Bit Assignments

You may recall the D4–02DA module requires 32 discrete output points. These 32 points provide the digital representation of the analog signals.

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.



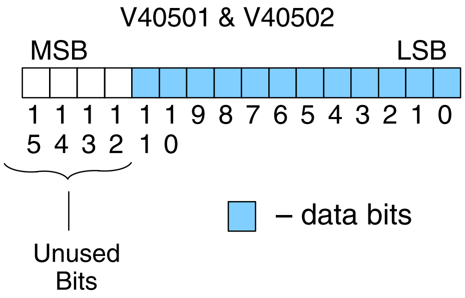
D4–02DA
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20

Within these two word locations, the individual bits represent specific information about the analog signal.

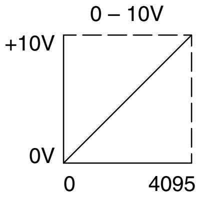
Analog Data Bits

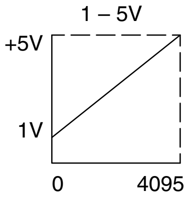
The first twelve bits of each V-memory location represent the analog data in binary format. The remaining four bits are not used and are ignored by the module.

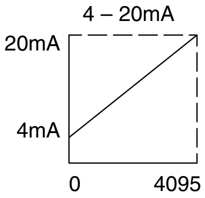
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0 – 4095 (2^{12}). For example, with a 0 to 10V scale, 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 signal change that occurs when the digital value is increased.

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

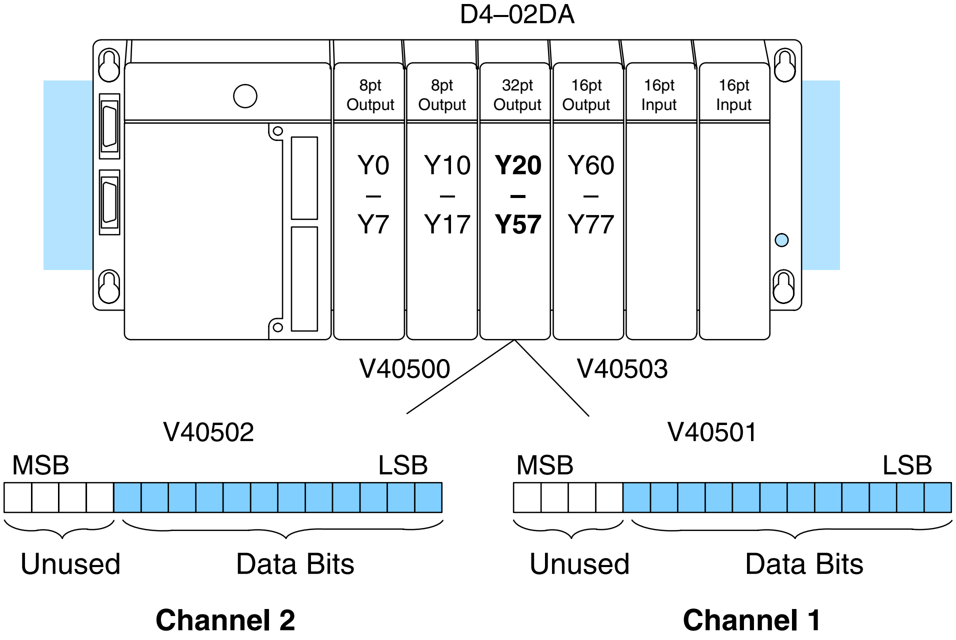
H = high limit of the signal range
L = low limit of the signal range

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

Writing the Control Program

Update Either Channel

As mentioned earlier, you can update either channel or both channels during the same scan. Since the module does not have any channel select bits, you just simply determine the location of the data word and send the data word to the output module whenever you need to update the data.


D4–02DA
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
Unused
Data Bits
Channel 2
MSB
LSB
Unused
Data Bits
Channel 1

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 all 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.

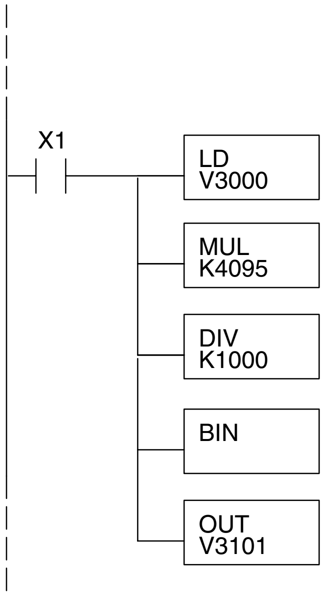
$$A = 10U \frac{4095}{10(H - L)}$$

$$A = 494 \frac{4095}{1000 - 0}$$

$$A = 2023$$

Here is 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 is 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.



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

Multiply the accumulator by 4095 (to start the conversion).

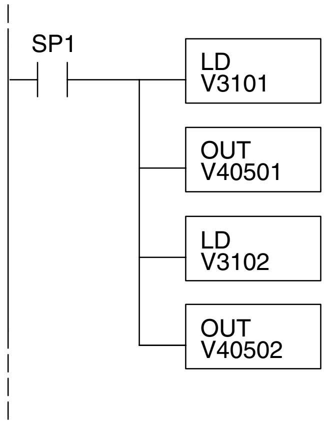
Divide the accumulator by 1000 (because we used a multiplier of 10, we have to use 1000 instead of 100).

Convert the BCD number to binary.

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

Sending the Data to the Module

The following program example shows how to send the digital values to the module.



Loads the complete data word into the accumulator. This is the digital value that will be sent to the module for channel 1.

Channel 1 is mapped to V40501, so this OUT instruction sends the channel 1 data to that address.

Loads the complete data word into the accumulator. This is the digital value that will be sent to the module for channel 2.

Channel 2 is mapped to V40502, so this OUT instruction sends the channel 2 data to that address.

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.

Range	If you know the digital value ...	If you know the analog signal level ...
0 to 10V	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}(A)$
1 to 5V	$A = \frac{4D}{4095} + 1$	$D = \frac{4095}{4}(A - 1)$
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

For example, if you are using the 0 to 10V range and you know you need a 6V signal level, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

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

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

$$D = (409.5)\ (6)$$

$$D = 2457$$

12

F4–04DA 4-Channel Analog Output

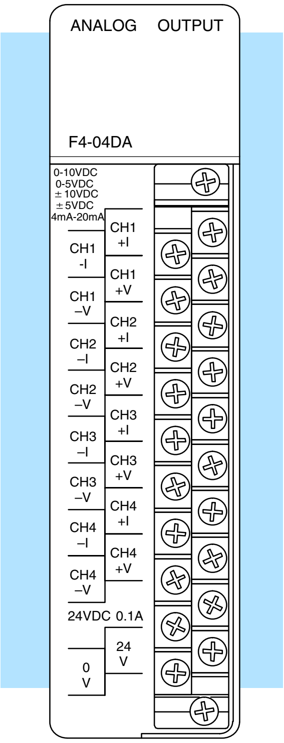
In This Chapter. . . .

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

Module Specifications

The F4–04DA 4-channel Analog Output module provides several features and benefits.

- It provides four channels of single-ended voltage or current outputs.
- 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).


ANALOG OUTPUT
F4-04DA
0-10VDC
0-5VDC
± 10VDC
± 5VDC
4mA-20mA
CH1 +I
CH1 -I
CH1 +V
CH1 –V
CH2 +I
CH2 –I
CH2 +V
CH2 –V
CH3 +I
CH3 –I
CH3 +V
CH3 –V
CH4 +I
CH4 –I
CH4 +V
CH4 –V
24VDC 0.1A
24 V
0 V

Analog Output Configuration Requirements

The F4–04DA Analog Output appears as a 16-point discrete output module. 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 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

Number of Channels	4
Output Ranges	0–5V, 0–10V, ±5V, ±10V, 4–20 mA
Resolution	12 bit (1 in 4096)
Conversion Method	Successive Approximation
Output Type	Single ended, 1 common
Output Impedance	0.2Ω typical, voltage output
Load Impedance	2KΩ minimum, voltage output 0Ω minimum, current output
Maximum Load / Voltage	680Ω/18V, 1KΩ/24V, 1.5KΩ/36V, current output
Voltage Output Current	5 mA sink or source
Short-Circuit Current	15 mA typical, voltage output
Linearity Error	±1 count (±0.025%) maximum
Gain Calibration Error	±8 counts maximum, voltage output –8 to +11 counts maximum, current output
Offset Calibration Error	±2 counts maximum, voltage output –5 to +9 counts maximum, current output
Conversion Time	5 µs maximum, settling time 0.3 ms maximum, digital out to analog out

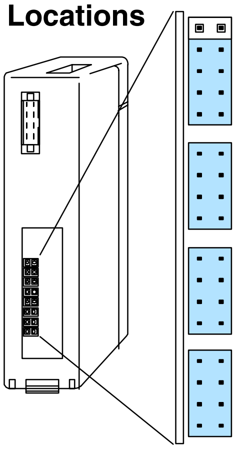
General Module Specifications

Digital Output Points Required	16 point (Y) outputs, 12 bits binary data, 4 channel select bits
Power Budget Requirement	120 mA @ 5 VDC (from base)
External Power Supply	24 VDC, 100 mA, class 2 ±10% (add 20 mA for each current loop used)
Accuracy vs. Temperature	±50 ppm / °C maximum full scale ±25 ppm / °C maximum offset
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 4096).

Setting the Module Jumpers

Jumper Locations

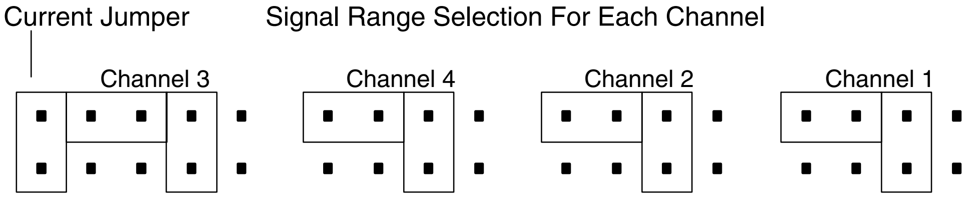


If you examine the rear of the module, you will notice several jumpers. These jumpers are used to select the signal range for each channel. There is also a current jumper. Some of the ranges can be selected with the current jumper installed or removed. To prevent losing jumpers when they are removed, a good place to store them is to reinstall each removed jumper over a single pin.

The signal range choices include five commonly encountered signal ranges: ±5 VDC, ±10 VDC, 4 – 20 mA, 0 – 5 VDC, and 0 – 10 VDC. The jumper settings for these signal ranges are shown in the table below.

In addition, the module supports some additional ranges that are not encountered very often. Use the last table in this section for those jumper settings.

The module is set at the factory for a 4–20 mA signal on all four channels. If this is acceptable you do not have to change any of the jumpers. The following diagram shows how the jumpers are set from the factory.



Selecting Common Output Signal Ranges

If you are using any 4–20 mA signals, you must have the current jumper installed. You can still select some of the other voltage ranges (except for the ±10 VDC range, which requires that the current jumper be removed). The following table shows the jumper selections for the commonly used ranges.

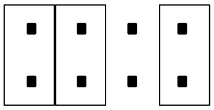
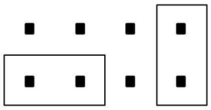
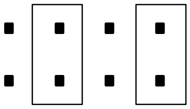
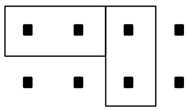
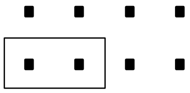
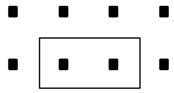
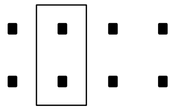
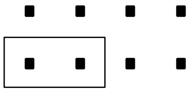
Bipolar Signal Range	Current Jumper Installed	Current Jumper Removed
±5 VDC		
±10 VDC	Not supported with current jumper installed	

Table is continued on the next page.

Commonly Used Signal Ranges (continued)

Unipolar Signal Range	Current Jumper Installed	Current Jumper Removed
4 to 20 mA		Not supported with current jumper removed
0 to +5 VDC		
0 to +10 VDC		

Selecting Special Signal Ranges

The F4–04DA module supports additional ranges that are not encountered very often. The following table shows the jumper selections for these ranges.

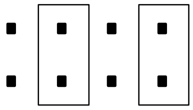
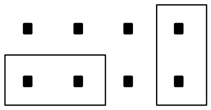
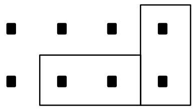
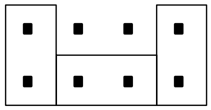
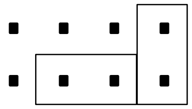
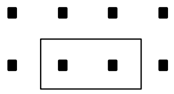
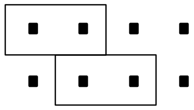
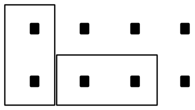
Signal Range	Current Jumper Installed	Current Jumper Removed
–10 to 0 VDC		Not supported with current jumper removed
–5 to 0 VDC		Not supported with current jumper removed
–2.5 to 0 VDC		Not supported with current jumper removed
± 1.25 VDC		Not supported with current jumper removed

Table is continued on the next page.

Special Signal Ranges (continued)

Signal Range	Current Jumper Installed	Current Jumper Removed
±2.5 VDC	Not supported with current jumper installed	
0 to +2.5 VDC		Not supported with current jumper removed
+1.25 to +6.25 VDC	Not supported with current jumper installed	
+2.5 to +7.5 VDC	Not supported with current jumper installed	

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.
- 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 F4–04DA 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: 21.6–26.4 VDC, Class 2, 500mA current. If you are using current loops, add 20 mA per current loop.

Load Requirements

Each channel can be wired independently for voltage or current. However, you cannot use both ±10 VDC and 4–20 mA signals on the same module.

- Current loads must have an impedance between 0 and 1000 ohms.
- Voltage loads must have an impedance greater than 2K ohms.

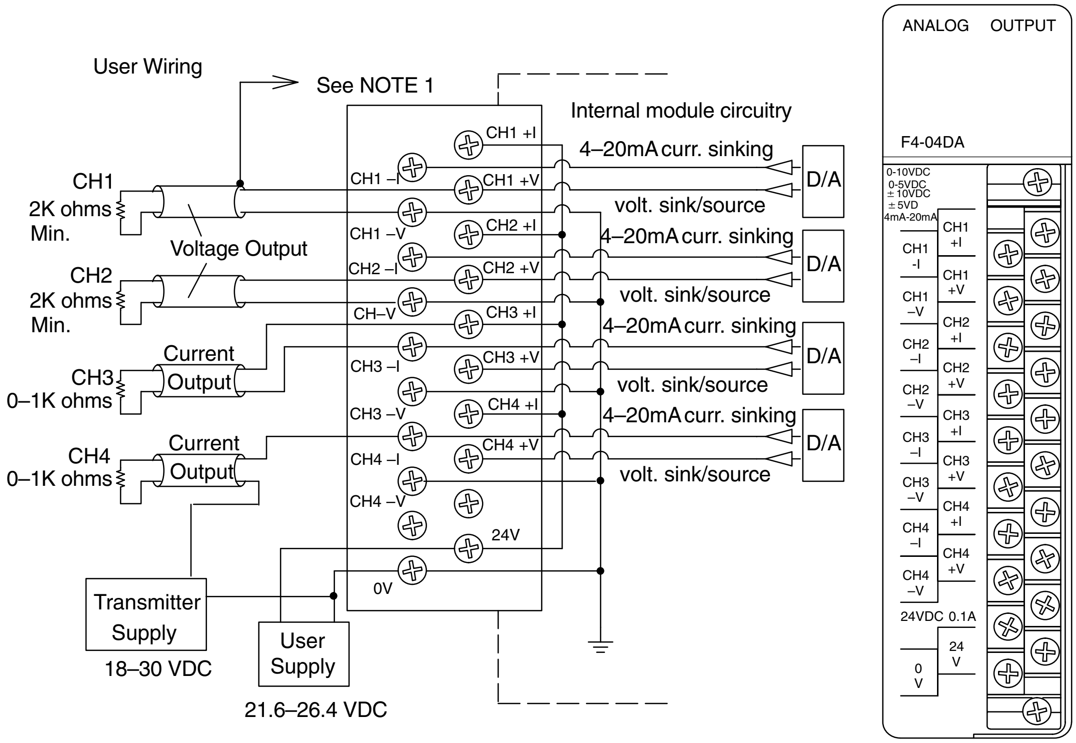
Removable Connector

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

Wiring Diagram

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

NOTE 2: Unused voltage and current outputs should remain open (no connections).

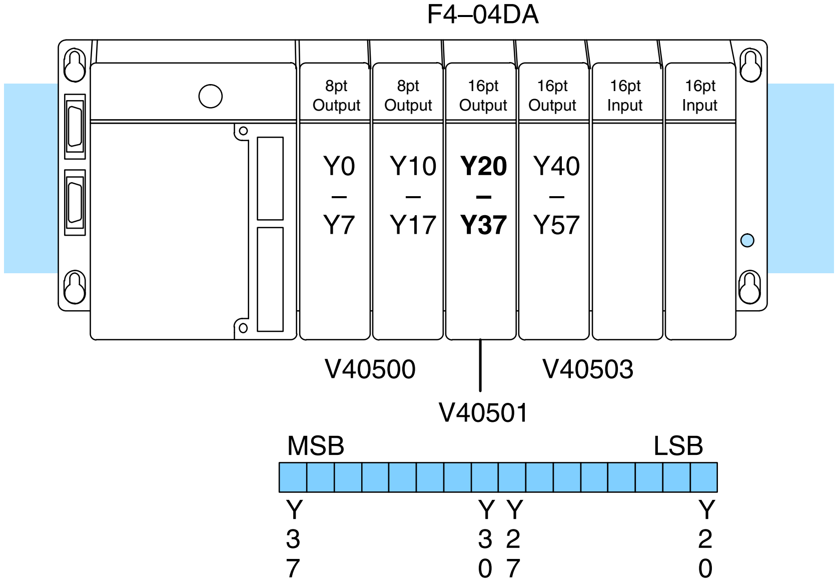

User Wiring
See NOTE 1
Internal module circuitry
CH1
2K ohms
Min.
Voltage Output
CH2
2K ohms
Min.
CH3
0–1K ohms
Current
Output
CH4
0–1K ohms
Current
Output
CH1 +I
CH1 –I
CH1 +V
CH1 –V
CH2 +I
CH2 –I
CH2 +V
CH–V
CH3 +I
CH3 –I
CH3 +V
CH3 –V
CH4 +I
CH4 –I
CH4 +V
CH4 –V
24V
0V
4–20mA curr. sinking
volt. sink/source
4–20mA curr. sinking
volt. sink/source
4–20mA curr. sinking
volt. sink/source
4–20mA curr. sinking
volt. sink/source
D/A
D/A
D/A
D/A
Transmitter
Supply
18–30 VDC
User
Supply
21.6–26.4 VDC
ANALOG OUTPUT
F4-04DA
0-10VDC
0-5VDC
± 10VDC
± 5VD
4mA-20mA
CH1 +I
CH1 -I
CH1 +V
CH1 –V
CH2 +I
CH2 –I
CH2 +V
CH2 –V
CH3 +I
CH3 –I
CH3 +V
CH3 –V
CH4 +I
CH4 –I
CH4 +V
CH4 –V
24VDC 0.1A
24 V
0 V

Module Operation

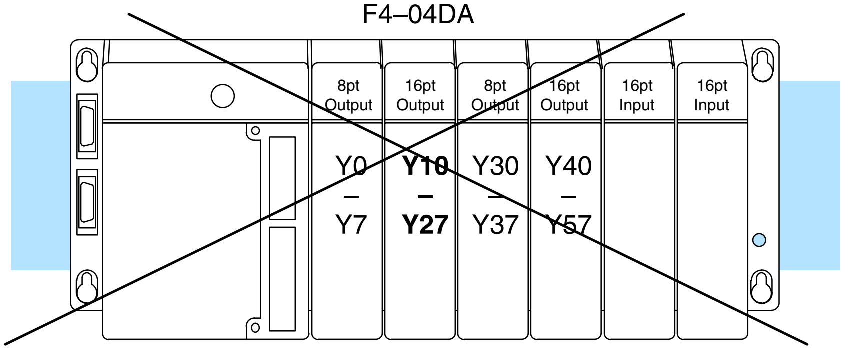
DL430 Special Requirements

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.

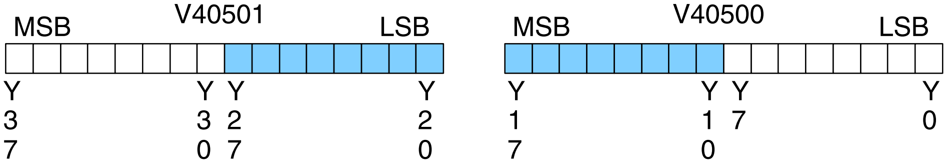
Correct!


F4–04DA
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40501
V40503
MSB
LSB
Y37
Y30
Y27
Y20

Wrong!


F4–04DA
8pt Output
16pt Output
8pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y27
Y30 – Y37
Y40 – Y57

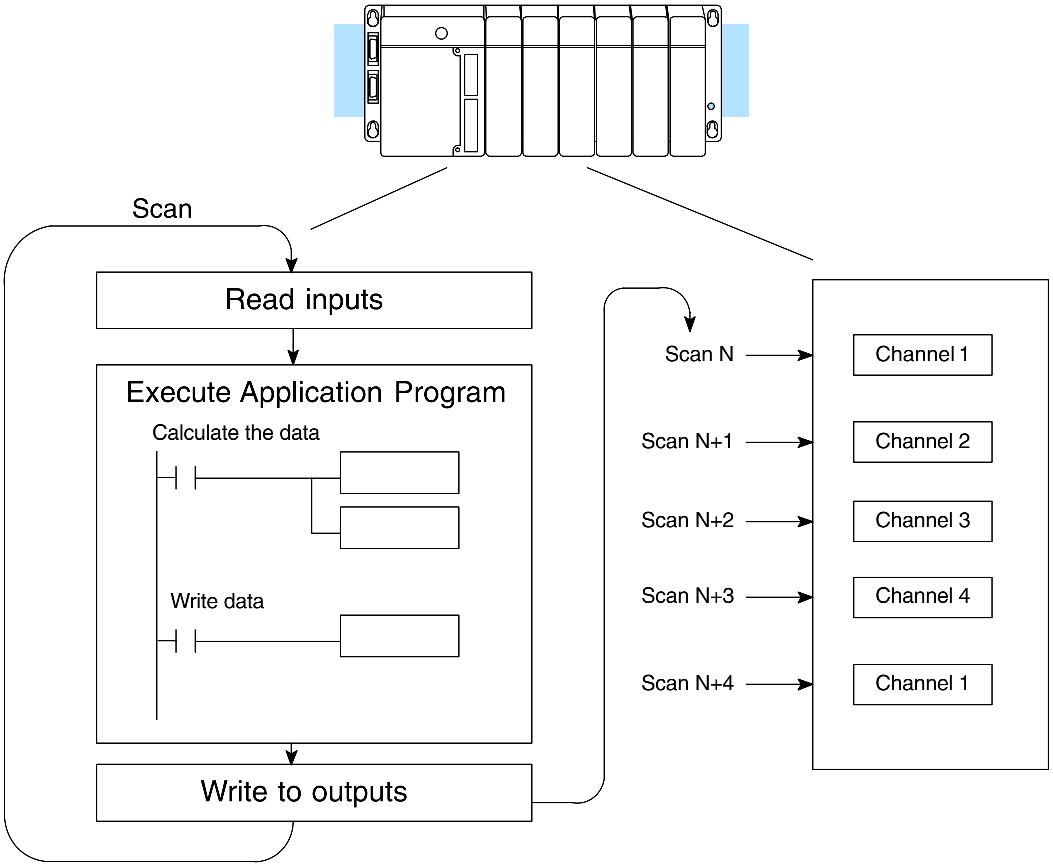
Data is split over two locations, so instructions cannot access data from a DL430.


V40501
MSB
LSB
Y37
Y30
Y27
Y20
V40500
MSB
LSB
Y17
Y10
Y7
Y0

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 module allows you to update the channels in any order. Your control program determines which channel gets updated on any given scan by *turning off* a bit that is associated with each channel. With a DL440 or DL450 CPU, you can use immediate instructions to update all four channels in the same scan (we will show you how to do this later).

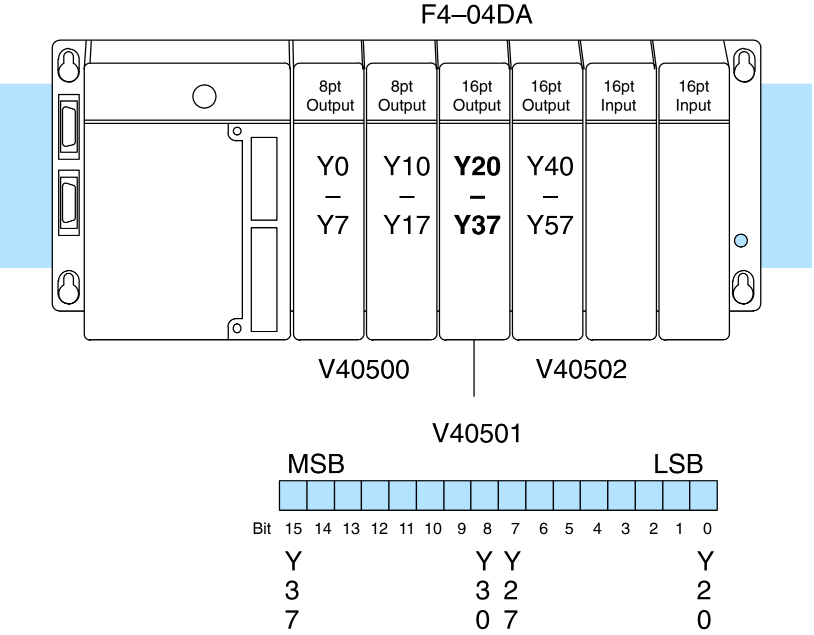

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 3
Channel 4
Channel 1

Output Bit Assignments

You may recall the F4–04DA 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.

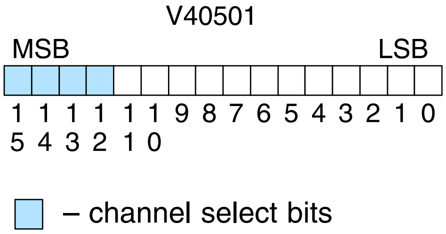

F4–04DA
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40502
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y37
Y30
Y27
Y20

Within this V-memory location the individual bits represent specific information about the analog signal.

Channel Select Bits

The last four bits (outputs) select the channel that will be updated with the data. The bits are assigned as follows.

Bit	Y point	Channel
12	Y34	1
13	Y35	2
14	Y36	3
15	Y37	4

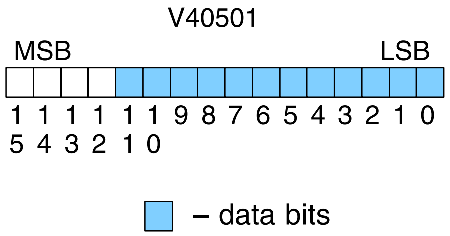

V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– channel select bits

When the output is off, the data currently in the accumulator is sent to the corresponding channel. If the bit is on, the corresponding channel holds the last value that was received from the CPU.

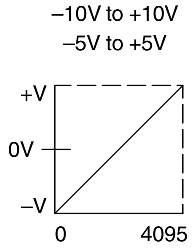
Analog Data Bits

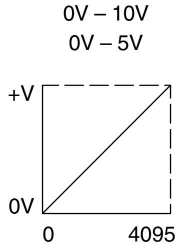
The first twelve bits of the V-memory location represent the analog data in binary format.

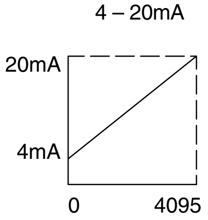
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



Since the module has 12-bit resolution, the analog signal is made of 4096 “pieces” ranging from 0 – 4095 (2^{12}). For example, with a 0 to 10V scale, 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 signal change that occurs when the digital value is increased.

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

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

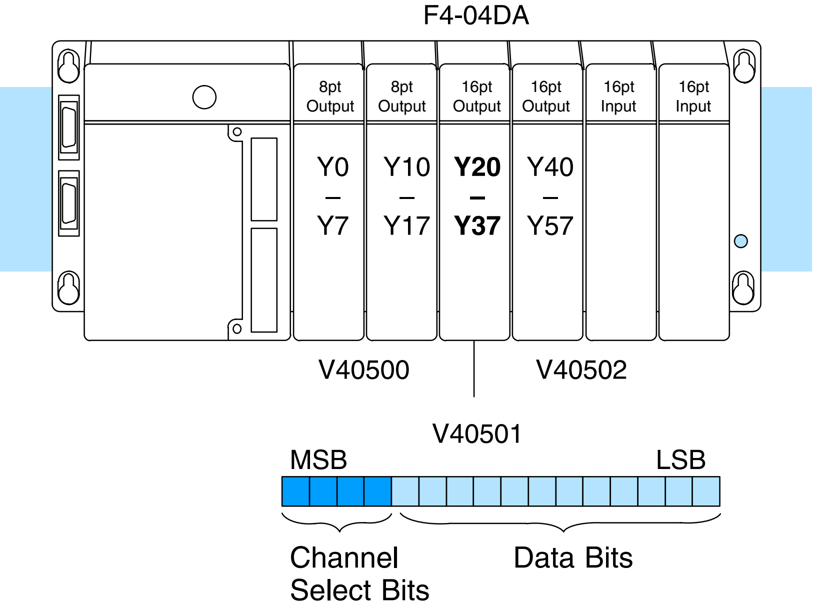
Signal Range	Span (H – L)	Divide By	Smallest Change
± 10V	20V	4095	4.88 mV
± 5V	10V	4095	2.44 mV
0 to 5V	5V	4095	1.22 mV
0 to 10V	10V	4095	2.44 mV
4 to 20mA	16mA	4095	3.91 µA

Now that you understand how the analog signal is represented by the digital value, you’re ready to write the control program.

Writing the Control Program

Update Any Channel

As mentioned earlier, you can update any channel or any channels during the same scan. 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).


F4-04DA
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
Channel Select Bits
Data Bits

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 almost all 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.

$$A = 10U \frac{4095}{10(H - L)}$$

$$A = 494 \frac{4095}{1000 - 0}$$

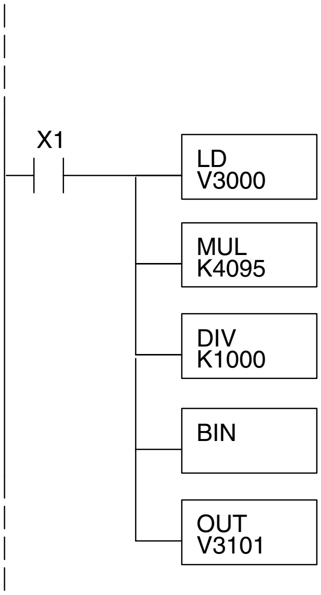
$$A = 2023$$

Engineering Unit Conversion

✓ 430 ✓ 440 ✓ 450

Here is 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 all four channels if you are using different data for each channel.

NOTE: The DL405 offers various instructions that allow you to perform math operations using binary, BCD, etc. It is 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary.

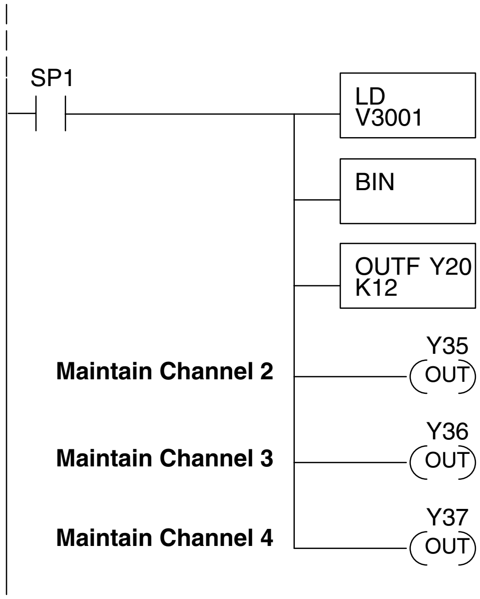
Store the result in V3101. This is the digital value, in binary form, that should be sent to the module.

Sending Data to One Channel

The following programs show you how to update a single channel. Notice the DL440 and DL450 CPUs require slightly different programs than the DL430 CPU. 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 with DL440 and DL450 CPUs. This example assumes you already have the data loaded in V3001.

☒	✓	✓
430	440	450

DL440/450 Example



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).

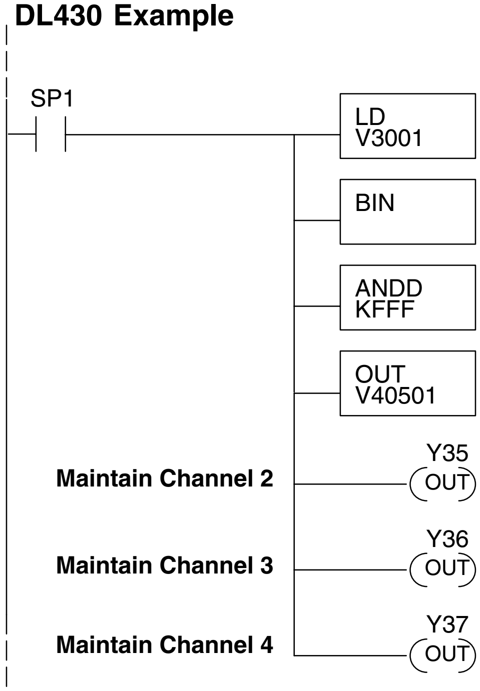
The OUTF sends the 12 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.

When the output selection bit is on, the module maintains the channel at its current value. If the output selection bit is off, the corresponding channel is updated with the data from the accumulator. This example assumes that Y34 is off and not used elsewhere in the program. By turning Y35 – Y37 on, only channel 1 gets updated. See the table below.

Output	Channel
Y34	Ch. 1
Y35	Ch. 2
Y36	Ch. 3
Y37	Ch. 4

✓	✓	✓
430	440	450

DL430 Example



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 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.

When the output selection bit is on, the module maintains the channel at its current value. If the output selection bit is off, the corresponding channel is updated with the data from the accumulator. This example assumes that Y34 is off and not used elsewhere in the program. By turning Y35 – Y37 on, only channel 1 gets updated. See the table below.

Output	Channel
Y34	Ch. 1
Y35	Ch. 2
Y36	Ch. 3
Y37	Ch. 4

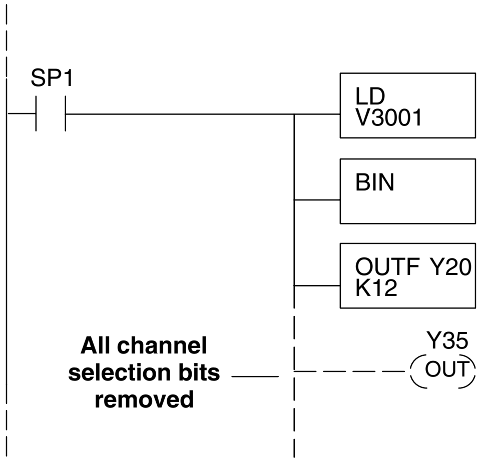
Sending the Same Data to All Channels

☒ ☑ ☑
430 440 450

The following programs show you how to update all channels with the same data. The primary difference from the previous example programs is that you do not have to include the channel selection bits. By leaving the selection bits off, the module uses the same data to update all the channels.

Notice the DL440 and DL450 CPUs require slightly different programs than the DL430 CPU. 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 with DL440 and DL450 CPUs. This example assumes you already have the data loaded in V3001.

DL440/450 Example



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.

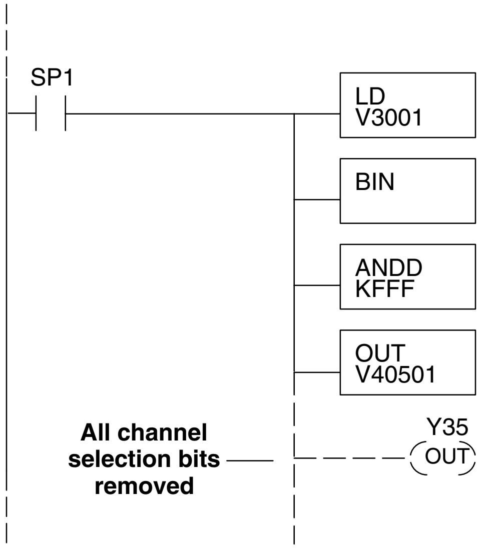
The BIN instruction converts the accumulator data to binary (you must omit this step if you've already converted the data elsewhere).

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.

Y34 – Y37 must be off.

☑ ☑ ☑
430 440 450

DL430 Example



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 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.

Y34 – Y37 must be off.

Sequencing the Channel Updates

The next four 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 most 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.

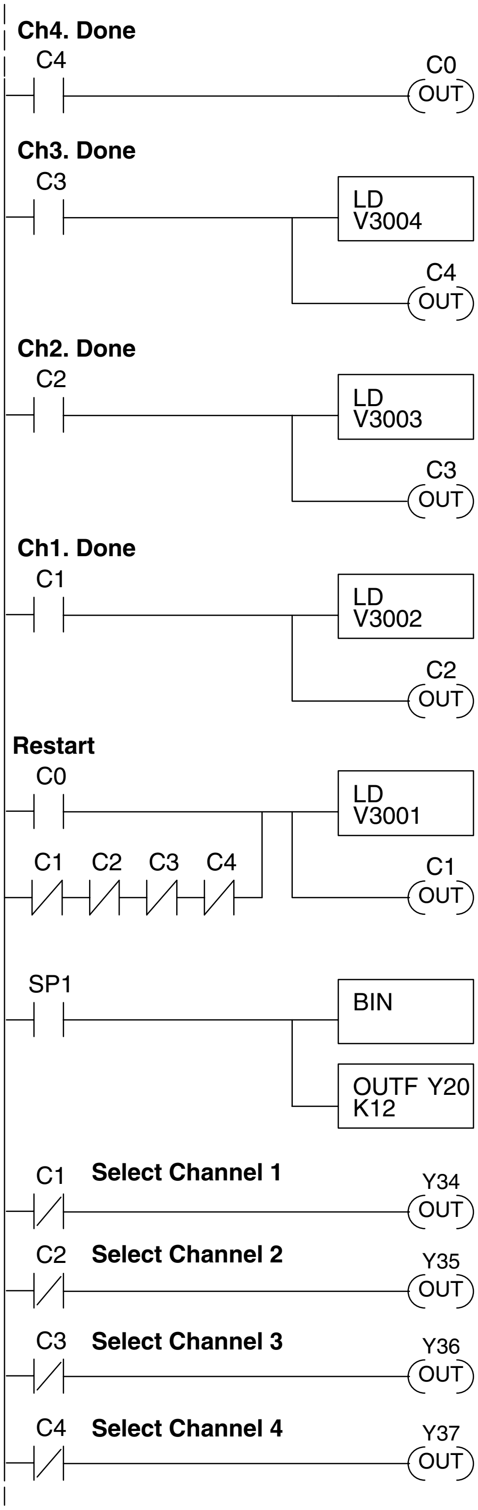
The next two examples, 3 and 4, are slightly more complex. However, they do not depend on the use of control relays to provide channel sequencing. Instead, they use function boxes to increment a channel pointer value in V-memory. Then, other instructions perform bit manipulations to properly position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

In the last example, we show you how to update all four channels in the same scan with DL440 and DL450 CPUs. However, this increases 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
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
LD
V3001
C1 C2 C3 C4
C1
OUT
SP1
BIN
OUTF Y20
K12
C1 Select Channel 1
Y34
OUT
C2 Select Channel 2
Y35
OUT
C3 Select Channel 3
Y36
OUT
C4 Select Channel 4
Y37
OUT

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

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 below).

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).

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).

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.

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.

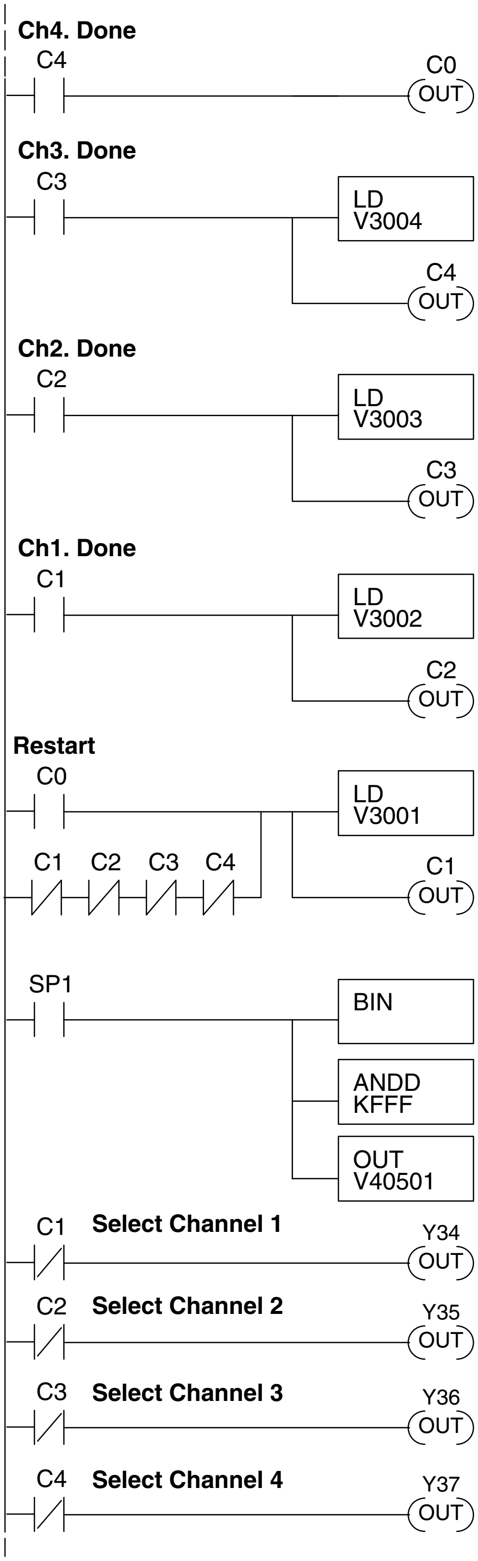
When the output selection bit is on (Y34–Y37 in this case), the module maintains the channel at its current value. If the output selection bit is off, the corresponding channel is updated with the data from the accumulator.

Output	Channel
Y34	Ch. 1
Y35	Ch. 2
Y36	Ch. 3
Y37	Ch. 4

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
C0
OUT
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
LD
V3001
C1
C2
C3
C4
C1
OUT
SP1
BIN
ANDD
KFFF
OUT
V40501
C1
Select Channel 1
Y34
OUT
C2
Select Channel 2
Y35
OUT
C3
Select Channel 3
Y36
OUT
C4
Select Channel 4
Y37
OUT

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

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 below).

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).

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).

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.

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.

When the output selection bit is on (Y34 – Y37 in this case), the module maintains that channel at its current value. If the output selection bit is off, the corresponding channel is updated with the data from the accumulator.

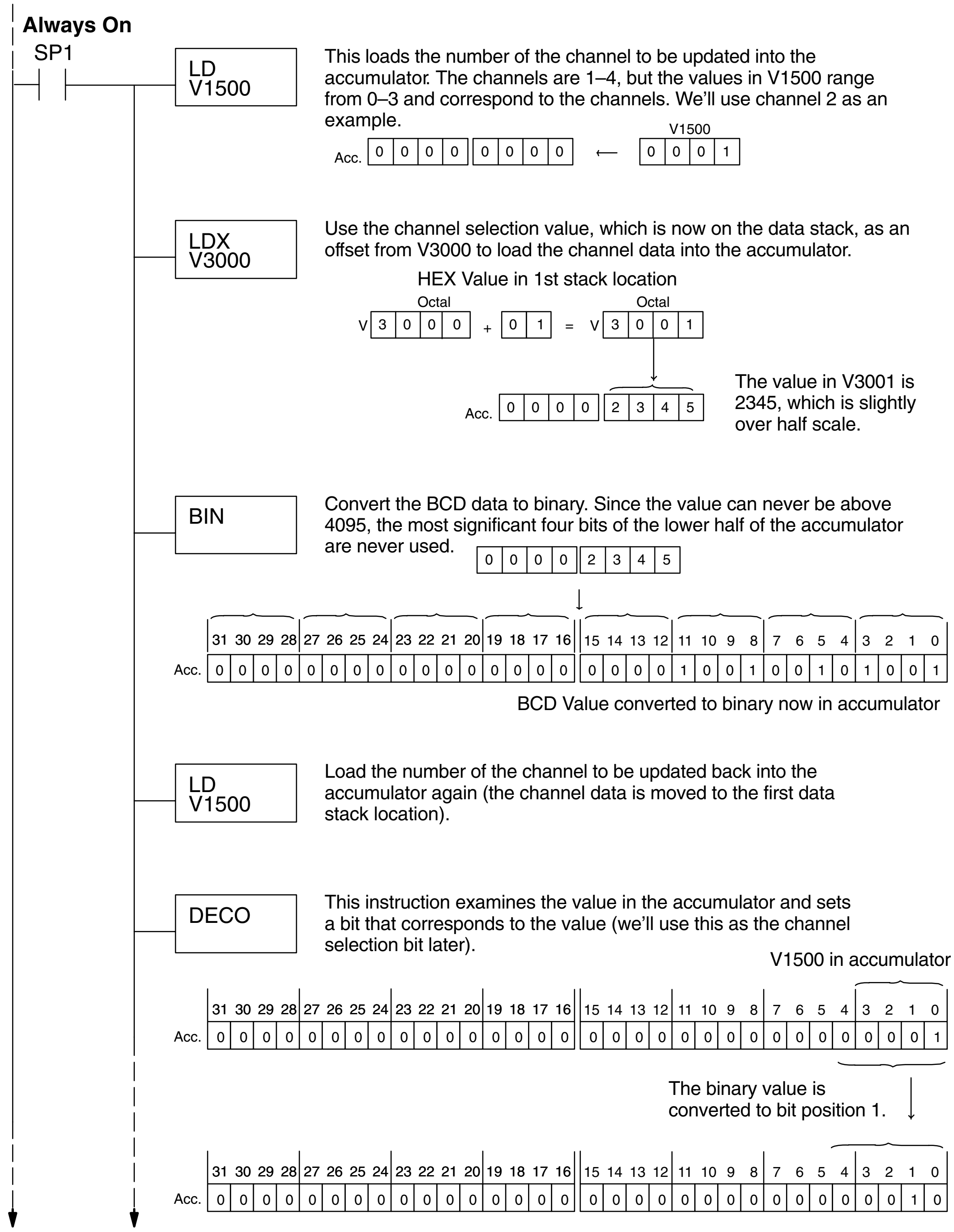
Output	Channel
Y34	Ch. 1
Y35	Ch. 2
Y36	Ch. 3
Y37	Ch. 4

Sequencing Example 3, 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 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
Acc.
0 0 0 0 0 0 0 0
V1500
0 0 0 1
LDX
V3000
HEX Value in 1st stack location
Octal
V 3 0 0 0 + 0 1 = V 3 0 0 1
Octal
Acc.
0 0 0 0 2 3 4 5
BIN
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
LD
V1500
DECO
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
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 1 0

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.

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.

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

Convert the BCD data to binary. Since the value can never be above 4095, the most significant four bits of the lower half of the accumulator are never used.

BCD Value converted to binary now in accumulator

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).

This instruction examines the value in the accumulator and sets a bit that corresponds to the value (we'll use this as the channel selection bit later).

The binary value is converted to bit position 1.

rogram is continued on the next page.

Example 3 Continued

INV

This inverts the accumlator bits. Remember, this module activates the channels when they are off, so the channel selection bit needs to be a zero. That's why we use the invert instruction.

	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.	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1

**SHFL
K12**

This instruction moves the channel selection bit 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.	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	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 lower 16 bits of the accumulator to 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.	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	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	1	1	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

**OUTF Y20
K16**

Sends the first 16 bits stored in the accumulator to the analog module. The most significant four bits of the analog word contain the channel selection bits. The remaining 12 bits contain the analog data.

**INCB
V1500**

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

V1500 K3
┤ = ├

**LD
K0**

When channel 4 has been updated, then reset the channel selection memory location to 0 (remember, 0 is for channel 1).

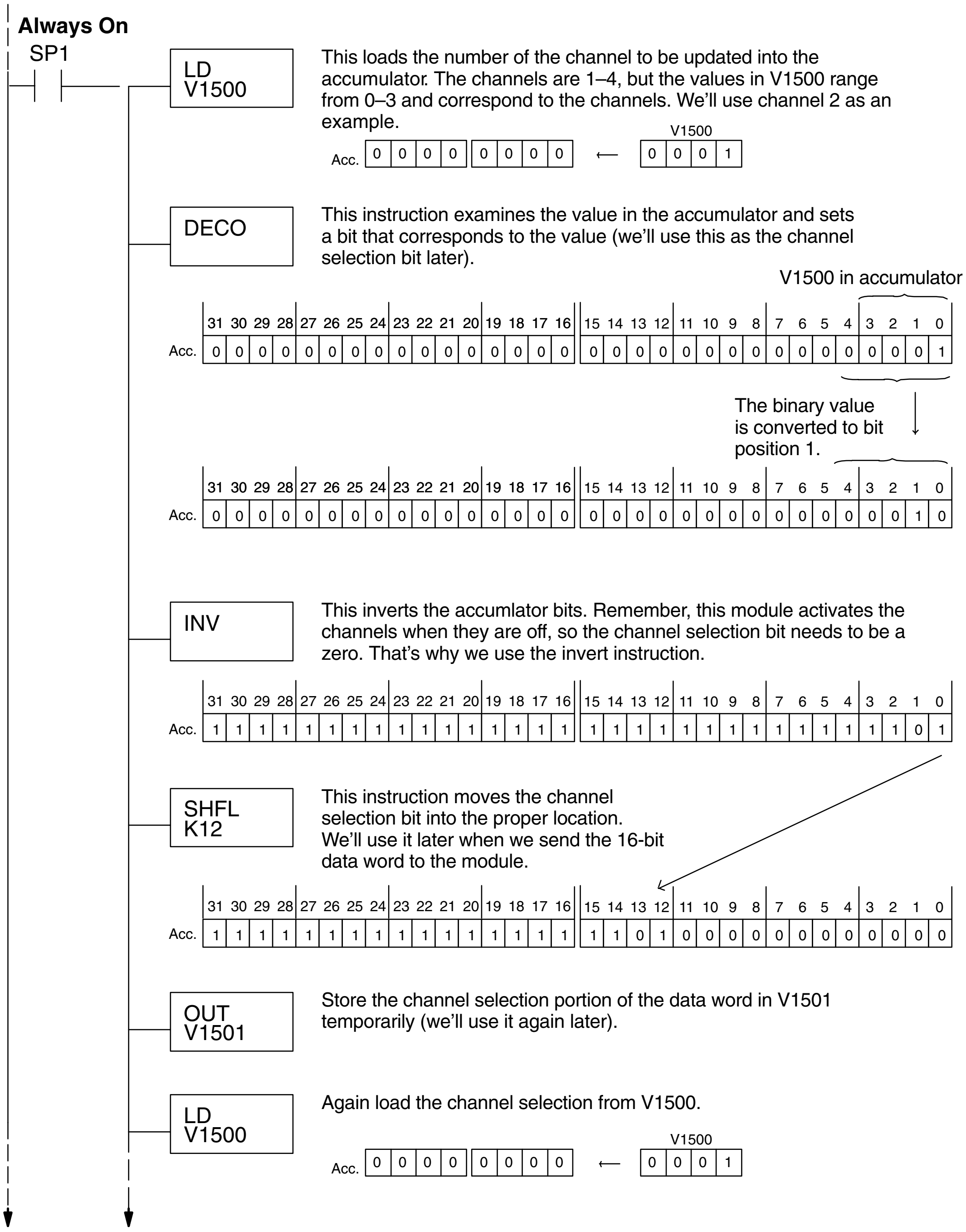
**OUT
V1500**

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



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

This instruction examines the value in the accumulator and sets a bit that corresponds to the value (we'll use this as the channel selection bit later).

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

The binary value is converted to bit position 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	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

This inverts the accumlator bits. Remember, this module activates the channels when they are off, so the channel selection bit needs to be a zero. That's why we use the invert instruction.

	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.	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1

This instruction moves the channel selection bit 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.	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

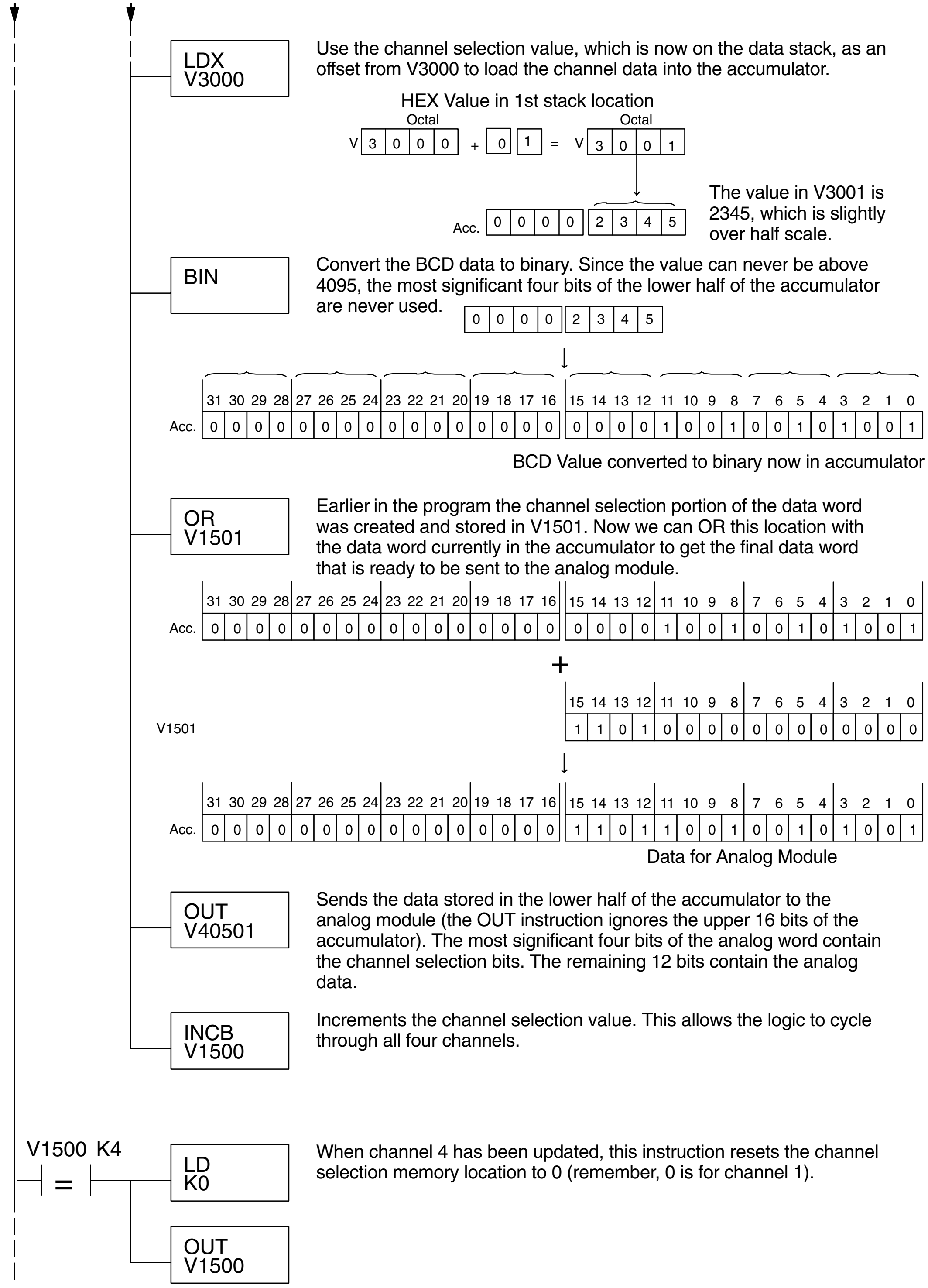
Store the channel selection portion of the data word in V1501 temporarily (we'll use it again later).

Again load the channel selection from V1500.

Acc. 0 0 0 0 0 0 0 0 ⟵ V1500 0 0 0 1

Program is continued on the next page.

Example 4 Continued



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, the most significant four bits of the lower half of the accumulator are never 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

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

Data for Analog Module

OUT V40501: Sends 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: Increments the channel selection value. This allows the logic to cycle through all four channels.

V1500 = K4, LD K0, OUT V1500: When channel 4 has been updated, this instruction resets the channel selection memory location to 0 (remember, 0 is for channel 1).

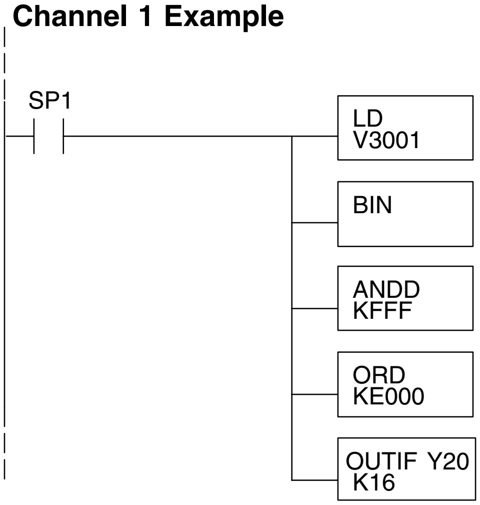
Updating All Channels in a Single Scan, DL440/450

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 that it slows 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



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

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.

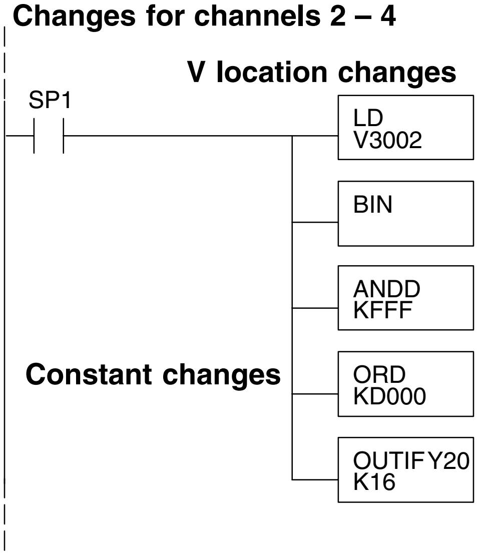
The ORD instruction (with K E000) has the effect of turning on Y35–Y37 and turning off Y34. By leaving Y34 off, channel 1 is updated with the data.

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



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

Location	Channel
V3001	1
V3002	2
V3003	3
V3004	4

The ORD instruction has the effect of selecting the appropriate channel to be updated. The following constants are used.

Constant	Channel
K E000	1
K D000	2
K B000	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 helpful during machine startup or troubleshooting. The following table provides formulas to make this conversion easier.

Range	If you know the digital value ...	If you know the analog signal level ...
0 to 5V	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}(A)$
0 to 10V	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}(A)$
±5V	$A = \frac{10D}{4095} - 5$	$D = \frac{4095}{10}(A + 5)$
±10V	$A = \frac{20D}{4095} - 10$	$D = \frac{4095}{20}(A + 10)$
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

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

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

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

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

$$D = 3276$$

Now you have all the necessary information to get your analog module installed and operating correctly.

13

F4-04DA-1 4-Channel Analog Current Output

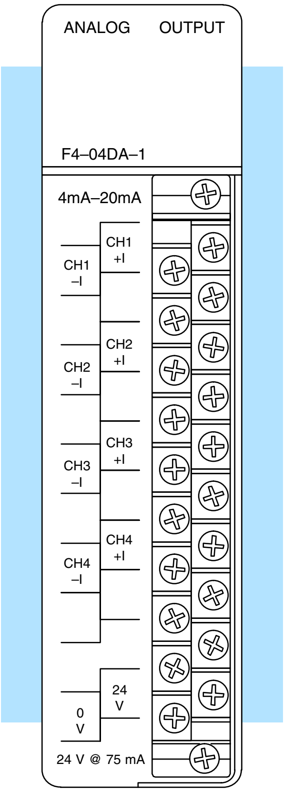
In This Chapter. . . .

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

Module Specifications

The F4–04DA–1 Analog Current Output Module provides several features and benefits.

- It is a direct replacement for the popular F4–04DA module in applications set for 4–20 mA output range.
- It provides four channels of 4–20 mA single ended current outputs.
- 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).



The following tables provide the specifications for the F4–04DA–1 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

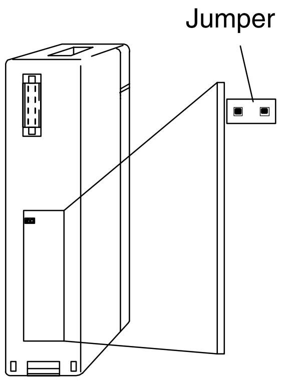
Number of Channels	4, single ended (one common)
Output Range	4–20 mA
Resolution	12 bit (1 in 4095)
Output Type	Outputs sink 4-20 mA from external supply
External Load Resistance	0Ω minimum
Maximum Loop Supply	30 VDC
Peak Output Voltage	40 VDC (clamped, transient suppressed)
Maximum Load / Power Supply	620Ω/18V, 910Ω/24V, 1200Ω/30V
Linearity Error (best fit)	±1 count (±0.025%) maximum
Gain Calibration Error	±5 counts maximum
Offset Calibration Error	±3 counts maximum
Maximum Inaccuracy	±0.1% @ 25°C (77°F) ±0.3% @ 0 to 60°C (32 to140°F)
Conversion Time	100 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 and 4 active channel bits
Power Budget Requirement	70 mA @ 5 VDC (from base)
External Power Supply	21.6–26.4 VDC, 75 mA, class 2 (add 20 mA for each current loop used)
Accuracy vs. Temperature	±57 ppm / °C full scale calibration range (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 Jumper

Before installing and wiring the module, you'll need to decide the proper jumper setting for your application. The module has one jumper, located in the open cutout at the rear of the housing. When the jumper is installed (which is the factory default setting), the module operates in Standard Mode. In this mode, the channel select bits are binary encoded, and you have access to the Output Enable control bit. We recommend this setting for new applications, since it offers more overall features and easier CPU programming.



Installed = Standard Mode

Removed = F4–04DA Compatibility Mode

When the jumper is removed, the module operates in the F4–04DA Compatibility Mode. If you have an existing F4–04DA application that uses 4–20 mA outputs only, choosing this mode will allow existing ladder logic to work with this module. In the F4–04DA Compatibility Mode, each channel has an individual channel select output bit (the Output Enable control bit is not accessible).

For either mode, the module requires 16 (Y) output points. Choose the mode of operation that best fits your application.

NOTE: If you have selected the F4–04DA Compatibility Mode (jumper removed), refer to the chapter on the F4–04DA for output bit assignments and ladder logic examples. **The remainder of the information in this chapter applies only to the F4–04DA–1 operating in the Standard Mode (with jumper installed).**

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–1 requires at least one field-side supply. You may use the same or separate power sources for the module supply and loop supply. The module requires 21.6 to 26.4 VDC, Class 2, at 75 mA current. The four current loops require 18 to 30 VDC, at 20 mA each.

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. You may use one of these instead of a separate supply if you have only a couple of analog modules. The current required is 75 mA (module), plus 80 mA (four current loops) for a total of 155 mA.

In some situations it's desirable to power the loops separately due to power budget or due to their remote location from the PLC. This will work, as long as the loop supply meets the voltage and current requirements, and its minus (–) side and the module supply's (–) side are connected together.

WARNING: If you are using the 24 VDC base power supply, 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.

Load Requirements

Each channel in use must have a load impedance less than 620 ohms at18V, 910 ohms at 24V, or 1200 ohms at 30V. Unused channels must be left disconnected.

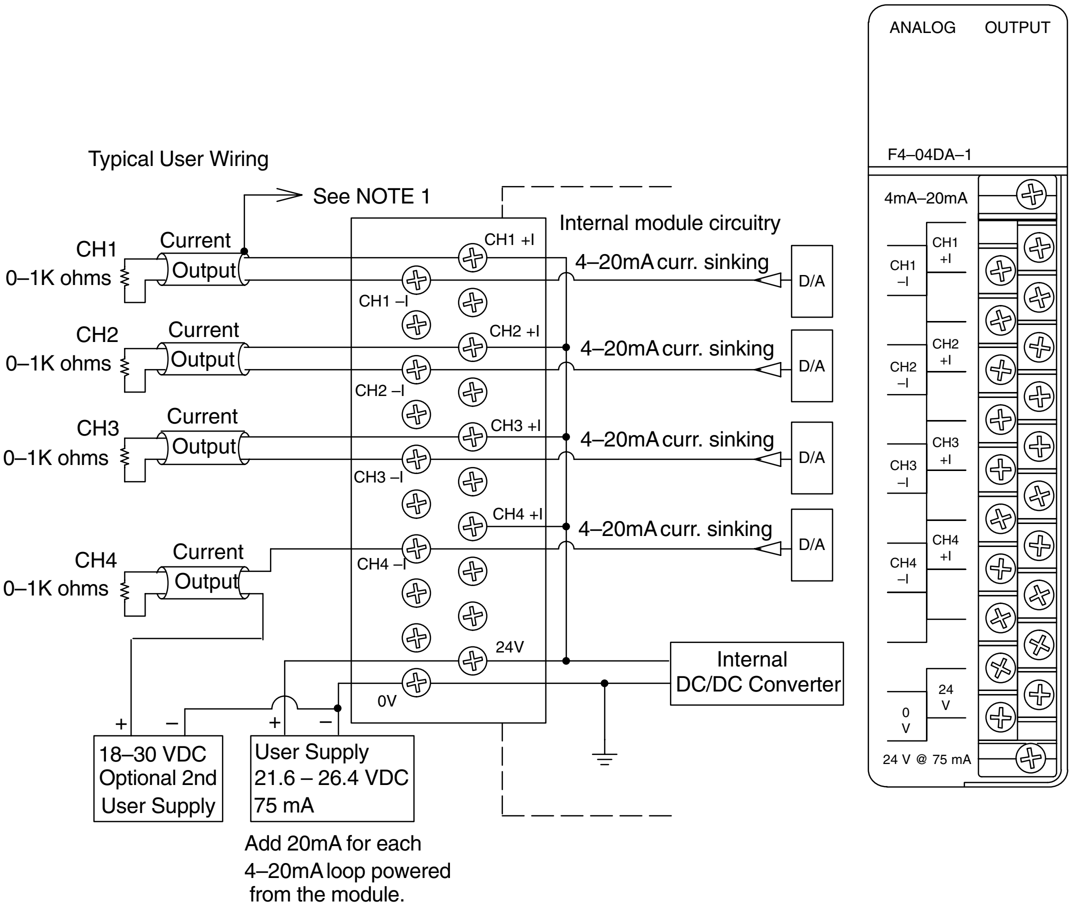
Removable Connector

The F4–04DA–1 module has a removable connector to make wiring easier. Simply loosen the retaining screws and gently pull the connector from the module. Use the following diagram to connect the field wiring. The diagram shows separate module and loop supplies for channel 4. If you only want to use one field-side supply, just combine the supplies' positive (+) terminals into one node, and remove the loop supply.

Wiring Diagram

NOTE 1: Shields should be connected to the 0V terminal of the module terminal block.

NOTE 2: Unused current outputs should remain open (no connections).

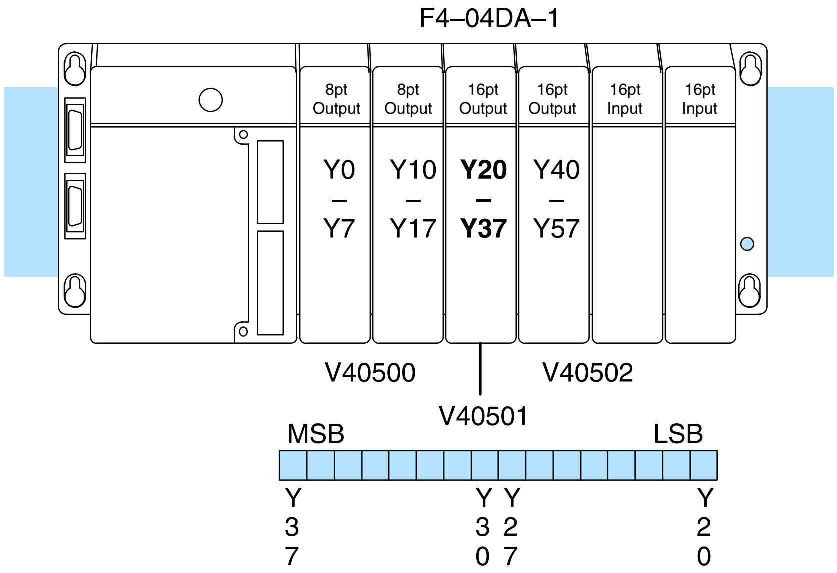

Typical User Wiring
See NOTE 1
Internal module circuitry
CH1 0–1K ohms
Current Output
CH1 +I
CH1 –I
4–20mA curr. sinking
D/A
CH2 0–1K ohms
CH2 +I
CH2 –I
CH3 0–1K ohms
CH3 +I
CH3 –I
CH4 0–1K ohms
CH4 +I
CH4 –I
24V
0V
Internal DC/DC Converter
+ –
18–30 VDC Optional 2nd User Supply
User Supply 21.6 – 26.4 VDC 75 mA
Add 20mA for each 4–20mA loop powered from the module.
ANALOG OUTPUT
F4–04DA–1
4mA–20mA
0 V
24 V
24 V @ 75 mA

Module Operation

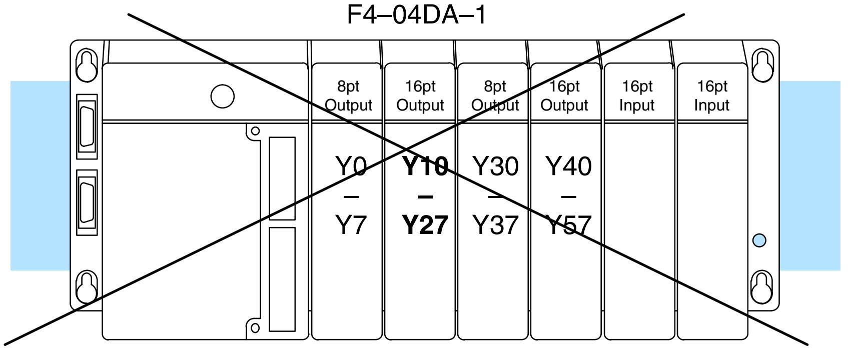
DL430 Special Requirements

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.

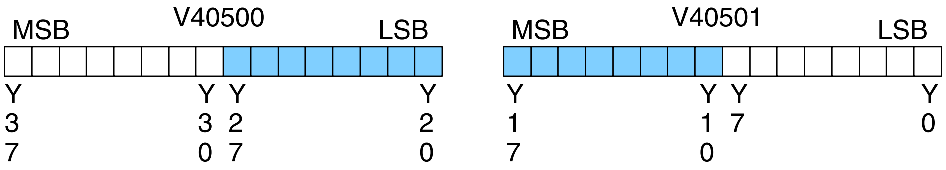
Correct!


F4–04DA–1
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40501
V40502
MSB
LSB
Y37
Y30
Y27
Y20

Wrong!


F4–04DA–1
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.


V40500
MSB
LSB
Y37
Y30
Y27
Y20
V40501
MSB
LSB
Y17
Y10
Y7
Y0

Analog Output Configuration Requirements

The D4–04DA–1 Analog Output module 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:

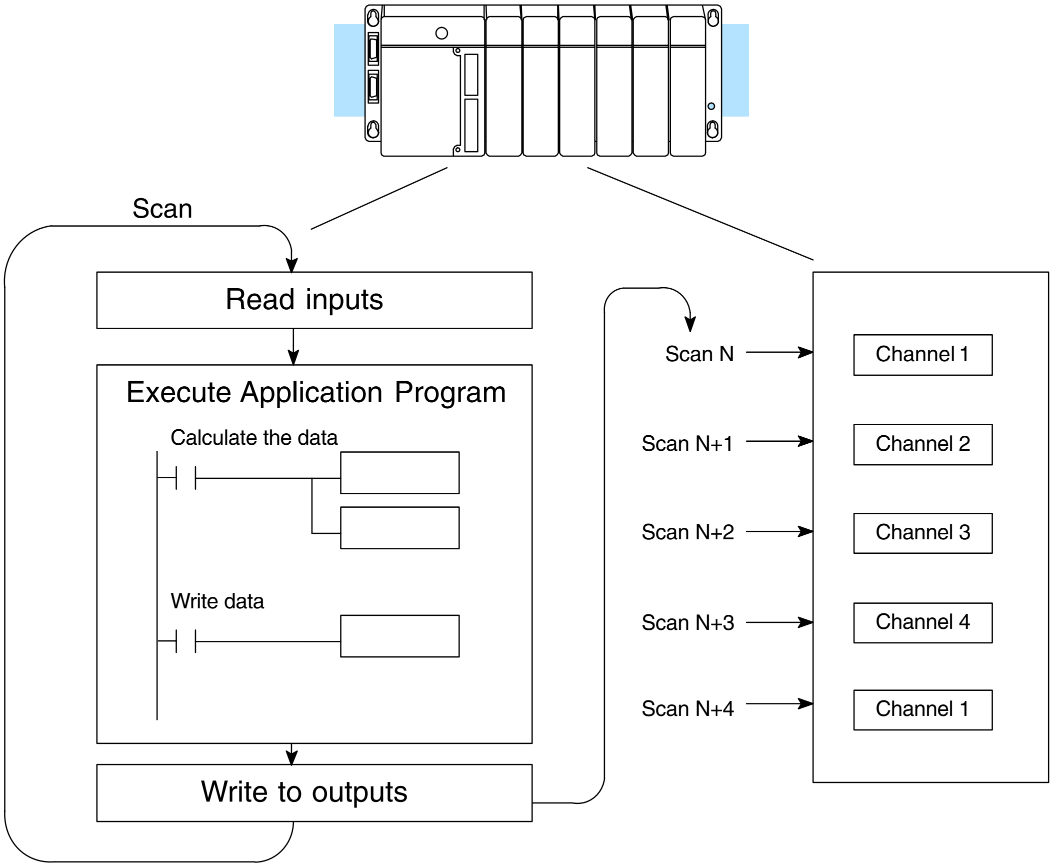
- For local and expansion systems, the available power budget and discrete I/O points.
- For remote I/O systems, the available power budget and number of remote I/O points.

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.

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

The F4–04DA–1 module allows you to update the channels in any order. Your control program determines which channel gets updated on any given scan. The exact method depends on the operating mode you selected when setting the jumper. 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).

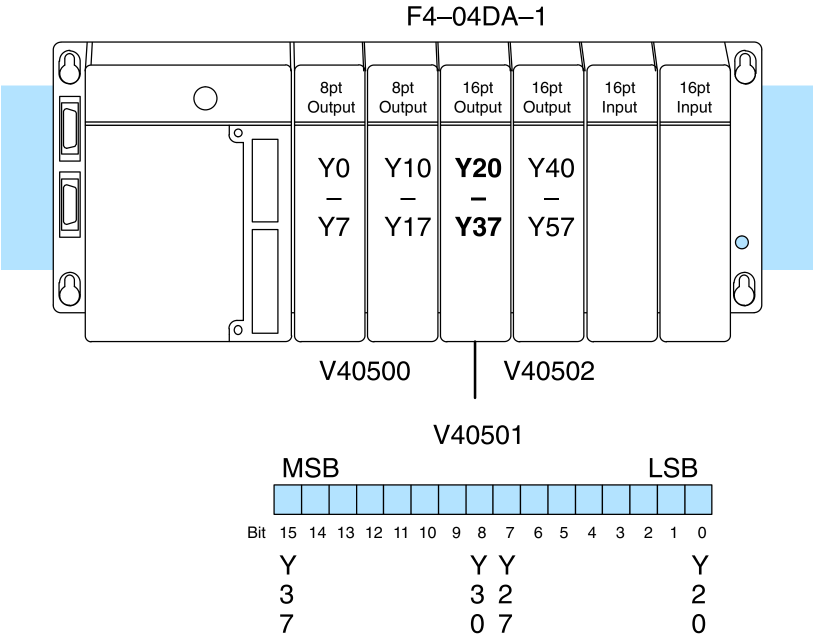

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 3
Channel 4
Channel 1

Output Bit Assignments

You may recall the F4–04DA–1 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.

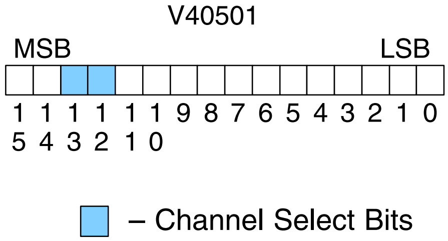
Since all output points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4–04DA–1
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
Y37
Y30
Y27
Y20

Within this V-memory location the individual bits represent specific information about the channel selected and the analog signal.

Channel Select Bits

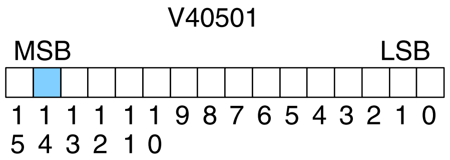
Output bits 12 and 13 are the channel select outputs. They are binary encoded to select the channel that will be updated with the data. The bits are assigned as follows.


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– Channel Select Bits

Bit 13	Bit 12	Channel
Off	Off	1
Off	On	2
On	Off	3
On	On	4

Output Enable Bit

Output bit 14 is the Output Enable control bit for all four channels. When it is off, all channel output currents decrease to their lowest level, which is 4 mA for connected loads. Disabling the outputs also clears the module's output data registers for each channel. To resume analog output levels, first the Output Enable control bit must turn on. Then, the CPU must write new data to each channel to restore the output current for that channel.



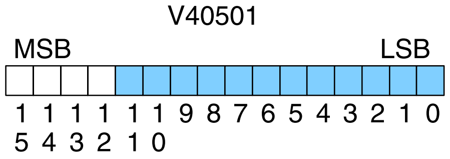
🟦 – Output Enable Bit

OFF = Disable (and clear)
ON = Enable

Analog Data Bits

The first twelve bits of the V-memory location represent the analog data in binary format. Each bit has a binary weight according to the following table.

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

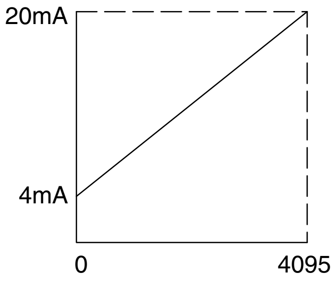


🟦 – data bits

The remaining bit (bit 15) is not used and is ignored by the module.

Module Resolution

Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0 – 4095 (2^{12}). For the 4 to 20 mA scale, sending a 0 produces a 4 mA signal, and 4095 gives a 20 mA signal. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The graph to the right shows the linear relationship between the data value and output signal level.



Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal change that occurs when the digital value increases by 1 LSB.

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

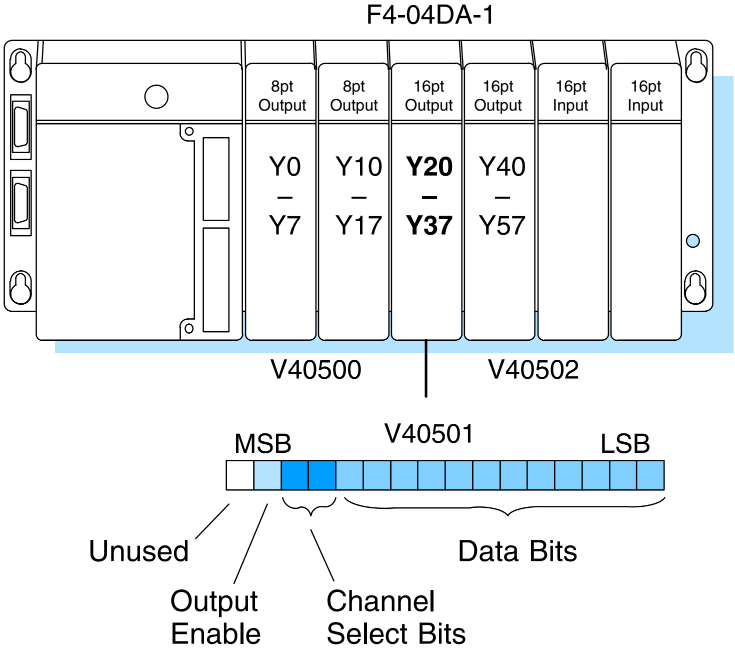
H = high limit of the signal range
L = low limit of the signal range

Signal Range	Span (H – L)	Divide By	Smallest Change
4 to 20mA	16mA	4095	3.91 µA

Writing the Control Program

Update Any Channel

As mentioned earlier, you can update any channel each 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 select outputs to determine which channel gets updated (more on this later).


F4-04DA-1
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
Unused
Output Enable
Channel Select Bits
Data Bits

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 almost all 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.

$$A = 10U \frac{4095}{10(H - L)}$$

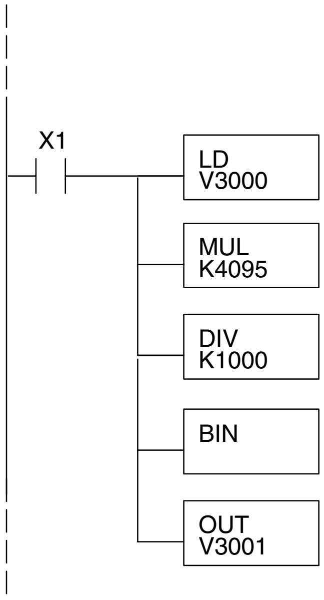
$$A = 494 \frac{4095}{1000 - 0}$$

$$A = 2023$$

✓	✓	✓
430	440	450

Here is 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 all four channels if you are using different data for each channel.

NOTE: The DL405 offers various instructions that allow you to perform math operations using binary, BCD, etc. It is 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary.

Store the result in V3001. 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 available for the DL450 CPU.

V-Memory Register Addresses for 16-Point Output (Y) Locations

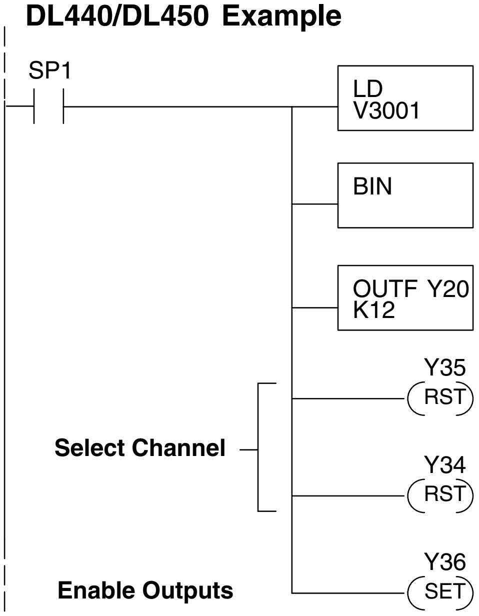
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

Sending Data to One Channel

The following programs show you how to update a single channel. Notice the DL430 CPU requires 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 with DL440 and DL450 CPUs. This example assumes you already have the data loaded in V3001.

430	440	450
☒	✓	✓

DL440/DL450 Example



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.

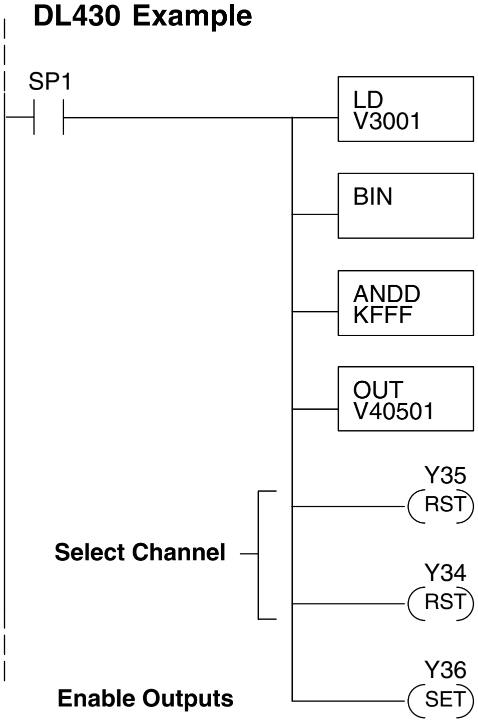
Turn Y35 off and Y34 off to update Channel 1.

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

Turn on Y36 to enable all four output channels.

430	440	450
✓	✓	✓

DL430 Example



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 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.

Turn Y35 off and Y34 off to update Channel 1.

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

Turn on Y36 to enable all four output channels.

Sequencing the Channel Updates

The next four 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.

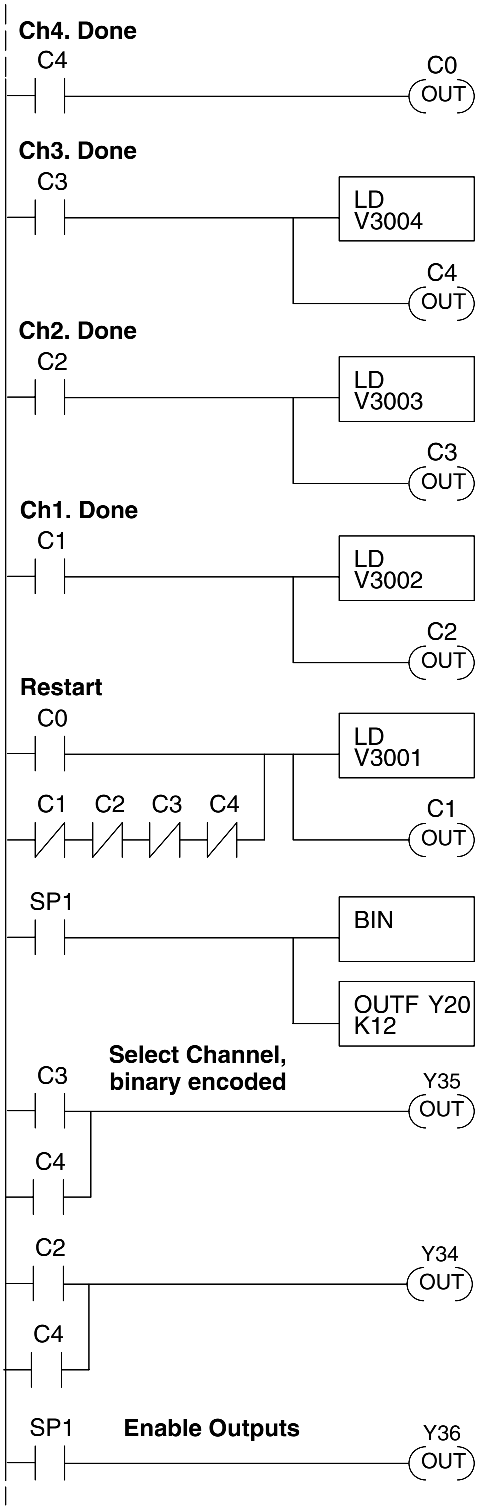
The next two examples, 3 and 4, are slightly more complex. However, they do not depend on the use of control relays to provide channel sequencing. Instead, they use function boxes to increment a channel pointer value in V-memory. Then, other instructions perform bit manipulations to position the channel 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.

In the last example, we show 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
Ch3. Done
C3
LD V3004
C4
OUT
Ch2. Done
C2
LD V3003
C3
OUT
Ch1. Done
C1
LD V3002
C2
OUT
Restart
C0
LD V3001
C1 C2 C3 C4
C1
OUT
SP1
BIN
OUTF Y20 K12
C3
Select Channel, binary encoded
Y35
OUT
C4
C2
Y34
OUT
C4
SP1
Enable Outputs
Y36
OUT

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

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 below).

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).

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).

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.

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.

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

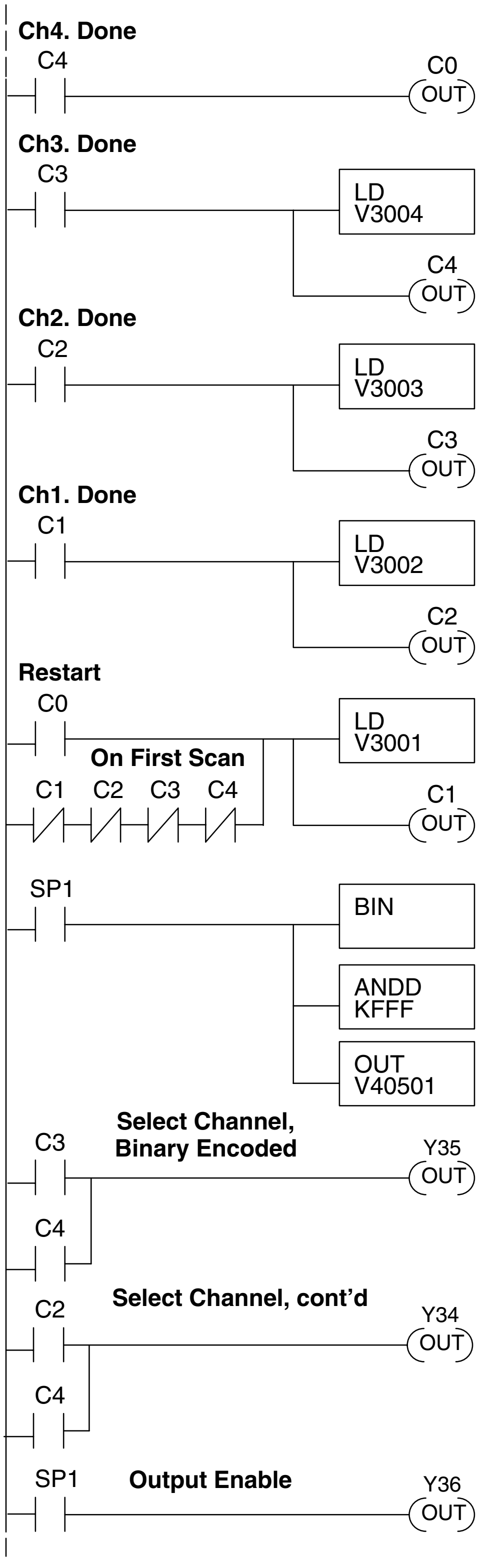
CR(on)	Y35	Y34	Channel
C1	Off	Off	Ch. 1
C2	Off	On	Ch. 2
C3	On	Off	Ch. 3
C4	On	On	Ch. 4

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
C0
OUT
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
On First Scan
C1 C2 C3 C4
LD
V3001
C1
OUT
SP1
BIN
ANDD
KFFF
OUT
V40501
Select Channel, Binary Encoded
C3
C4
Y35
OUT
Select Channel, cont'd
C2
C4
Y34
OUT
SP1
Output Enable
Y36
OUT

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

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 below).

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).

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).

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.

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.

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

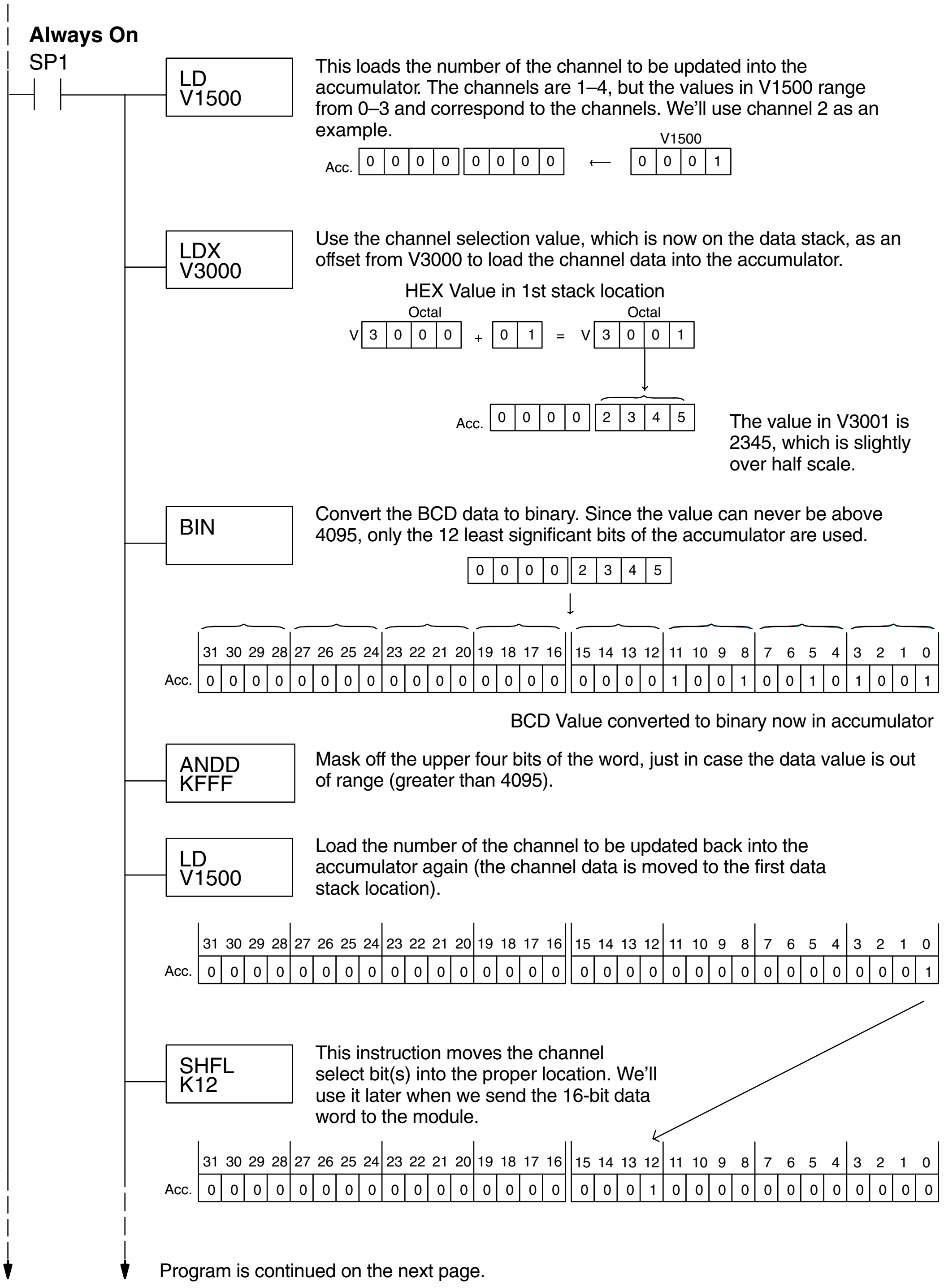
Enables all four output channels. SP1 is always on.

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



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

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.

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

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

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).

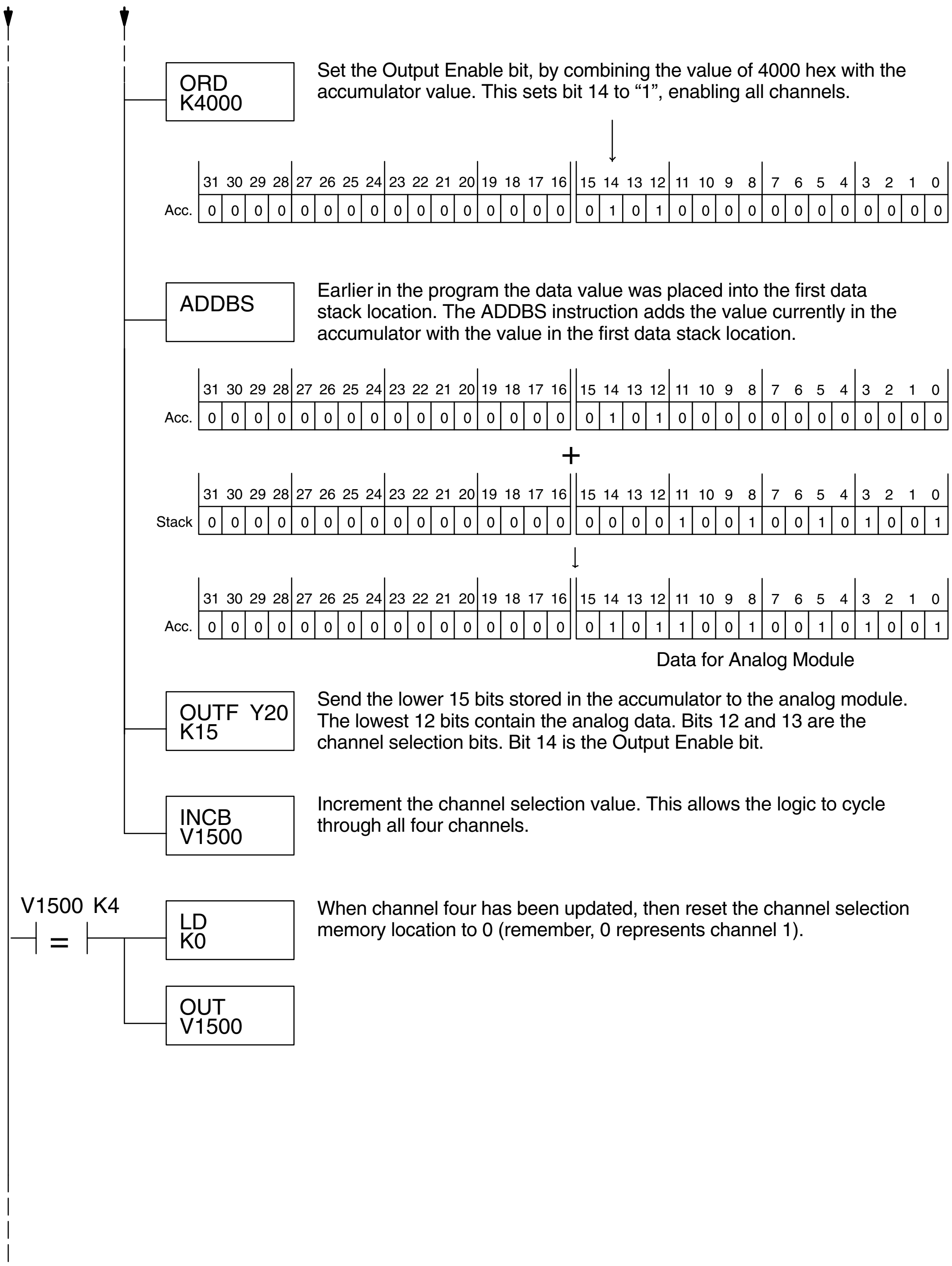
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

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



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

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

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.

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

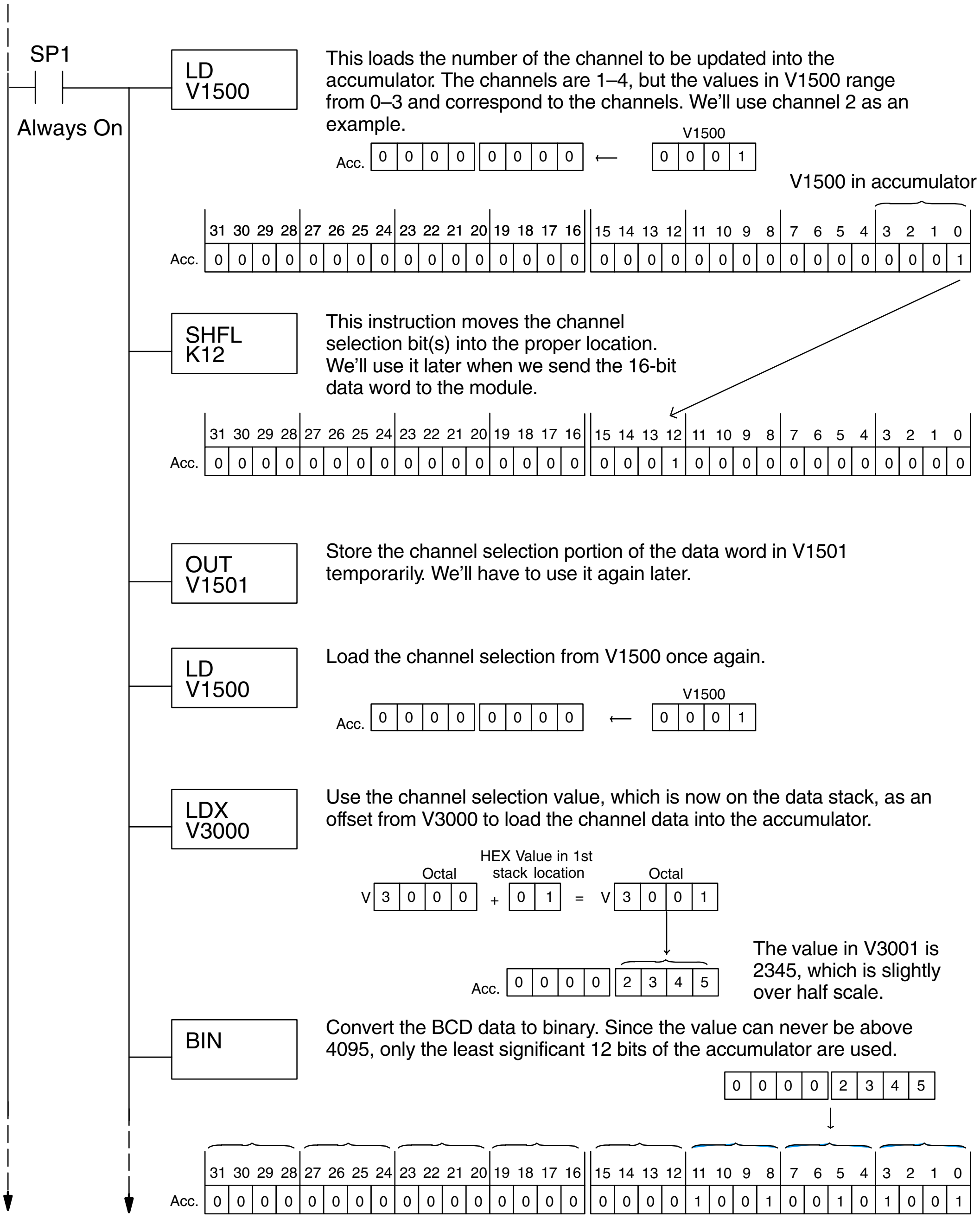
When channel four has been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1).

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
Acc. 0 0 0 0 0 0 0 0 ⟵ V1500 0 0 0 1
V1500 in accumulator
Acc. 31–16: 0000 0000 0000 0000; 15–0: 0000 0000 0000 0001
SHFL K12
Acc. 31–16: 0000 0000 0000 0000; 15–0: 0001 0000 0000 0000
OUT V1501
LD V1500
Acc. 0 0 0 0 0 0 0 0 ⟵ V1500 0 0 0 1
LDX V3000
Octal V 3 0 0 0 + HEX Value in 1st stack location 0 1 = Octal V 3 0 0 1
Acc. 0 0 0 0 2 3 4 5
BIN
0 0 0 0 2 3 4 5
Acc. 31–16: 0000 0000 0000 0000; 15–0: 0000 1001 0010 1001

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.

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.

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

Load the channel selection from V1500 once again.

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.

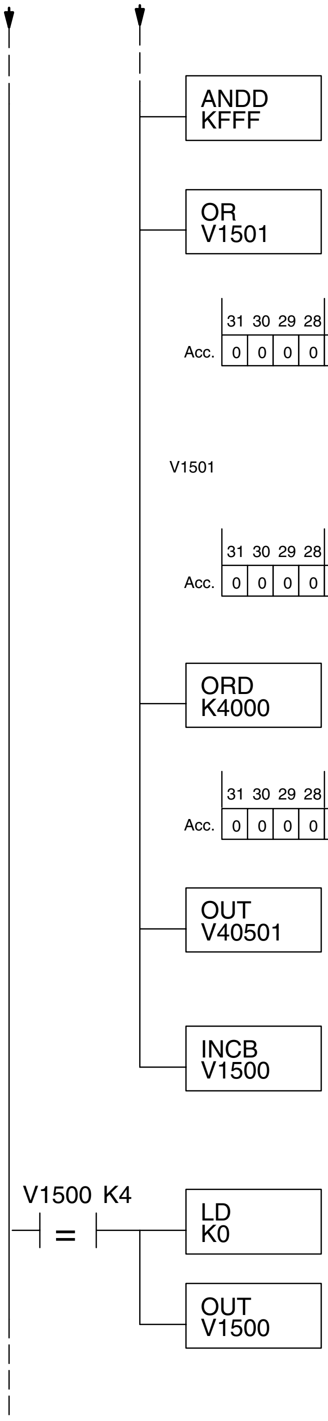
The value in V3001 is 2345, which is slightly over half scale.

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.

BCD Value converted to binary now in accumulator.

Program is continued on the next page.

Example 4 Continued



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

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

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

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.

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

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

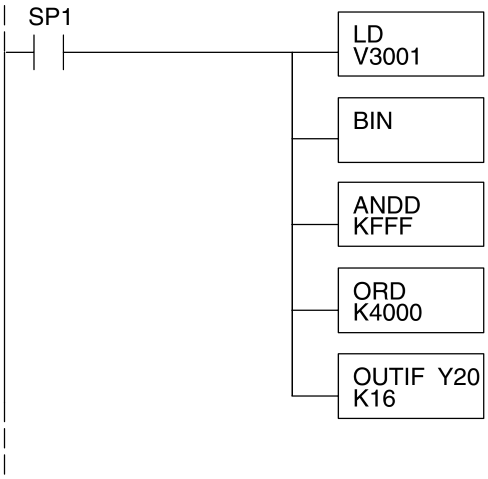
Updating all Channels in a Single Scan

✗	✓	✓
430	440	450

By using the Immediate instructions found in the DL440 and DL450 CPUs (not DL430s), you can easily update all four channels in a single scan. Before choosing this method, remember that it slows 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.

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



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

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.

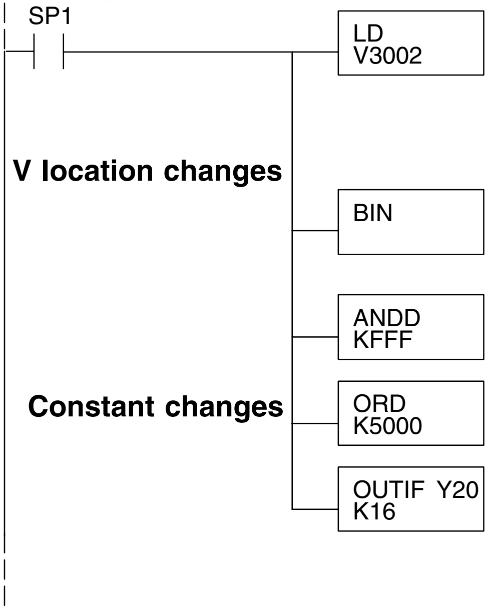
The ORD instruction (with K4000) sets the Output Enable bit. Outputs Y34 and Y35 are left off to select channel 1 for updating with the data.

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



The LD instruction loads the data for channel 2 into the accumulator.

Location	Channel
V3001	1
V3002	2
V3003	3
V3004	4

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

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

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.

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.

Range	If you know the digital value ...	If you know the analog signal level ...
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

For example, if you need a 9mA signal level, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

$$D = \frac{4095}{16}(A-4)$$

$$D = \frac{4095}{16}(9mA-4)$$

$$D = (255.94)\ (5)$$

$$D = 1280$$

14

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

In This Chapter. . . .

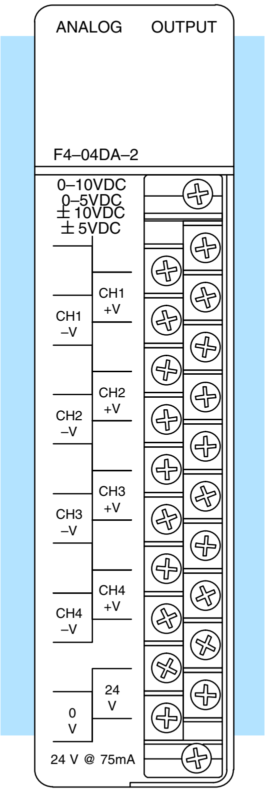
— Module Specifications
— Setting the Module Jumpers
— Connecting the Field Wiring
— Module Operation
— Writing the Control Program
— Configuration Cross-Reference, F4–04DA to F4–04DA–2

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

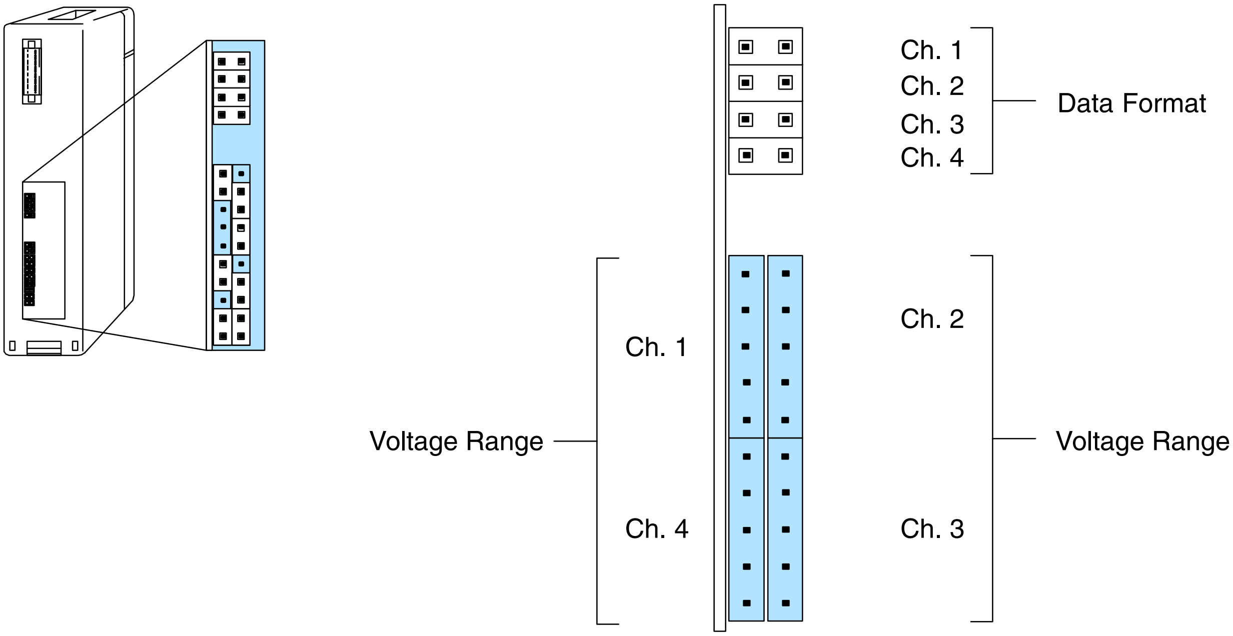
General Module Specifications

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.



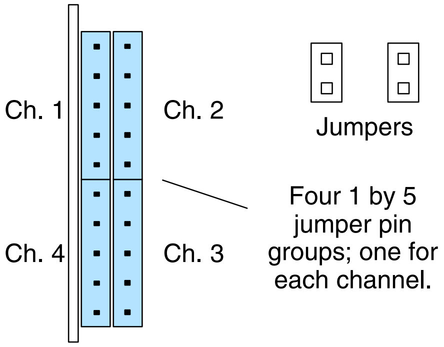
NOTE: If you are replacing an F4–04DA module with this one in an existing application, skip to the special section at the end of this chapter “Configuration Cross-Reference F4–04DA to F4–04DA–2”.

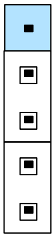
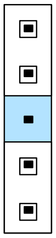
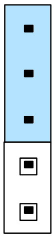
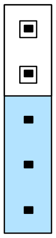
Factory Default Settings

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.

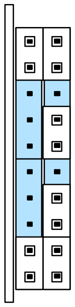
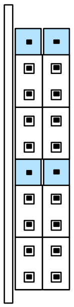
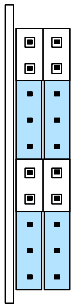
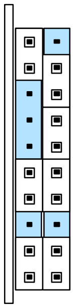
Voltage Range Selection

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.

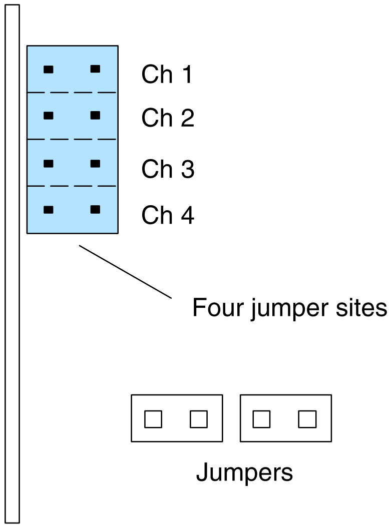
Example 1	Jumper Settings	Example 2	Jumper Settings
Ch. 1 ±10 VDC Ch. 2 ±5 VDC Ch. 3 0 to 5 VDC Ch. 4 0 to 10VDC		Ch. 1 0 to 5 VDC Ch. 2 (same) Ch. 3 (same) Ch. 4 (same)	
Example 3	**Jumper Settings**	**Example 4**	**Jumper Settings**
Ch. 1 ±10 VDC Ch. 2 (same) Ch. 3 (same) Ch. 4 (same)		Ch. 1 ±10 VDC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

Data Format Selection

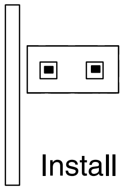
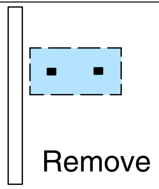
The upper jumper block consists of four pairs of jumper posts. Functionally, the jumper posts are separated into four horizontal pairs, as shown to the right. Each jumper site corresponds to a particular channel as labeled.

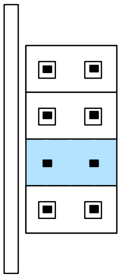
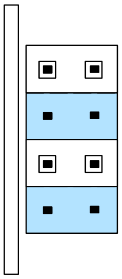
Installing a jumper sets the corresponding channel for 0 to 4095 data format conversion range. This is typically used with unipolar voltage range selections. The sign bit ***is not*** active in this data format.

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.



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.

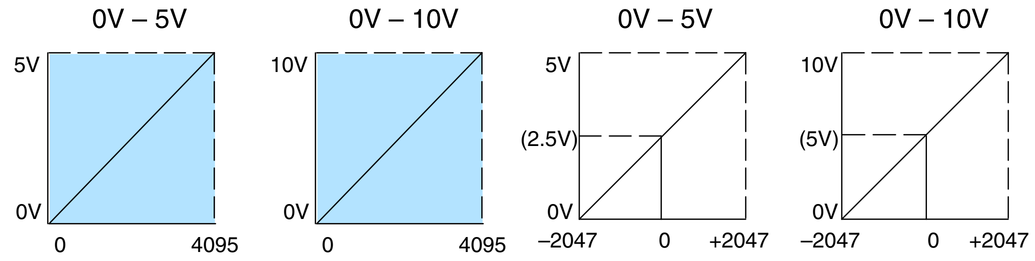
Output Data Format	Jumper Setting	Data Format	Jumper Setting
0 to 4095	 Install	–2047 to +2047	 Remove

Example 1	Jumper Settings	Example 2	Jumper Settings
Ch 1 0 to 4095 Ch 2 0 to 4095 Ch 3 –2047 to +2047 Ch 4 0 to 4095		Ch.= 1 0 to 4095 Ch 2 –2047 to +2047 Ch 3 0 to 4095 Ch 4 –2047 to +2047	

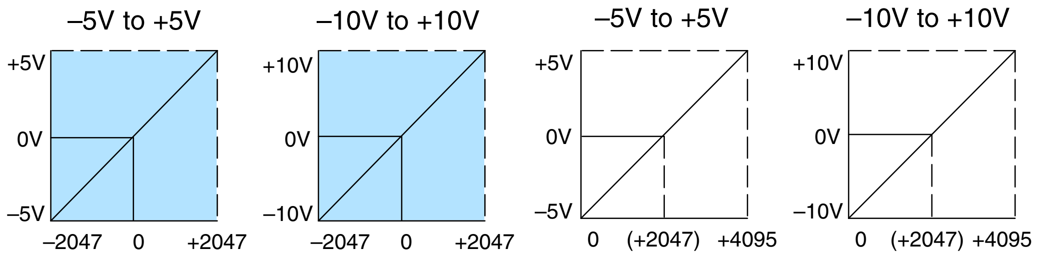
NOTE: The Data Format jumpers can be configured in a way (not shown above) for F4–04DA compatibility. If you choose to set the jumpers for F4–04DA compatibility, note that the ladder program examples in this chapter do not apply to that mode. Refer to “Configuration Cross-Reference, F4–04DA to F4–04DA–2” 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.

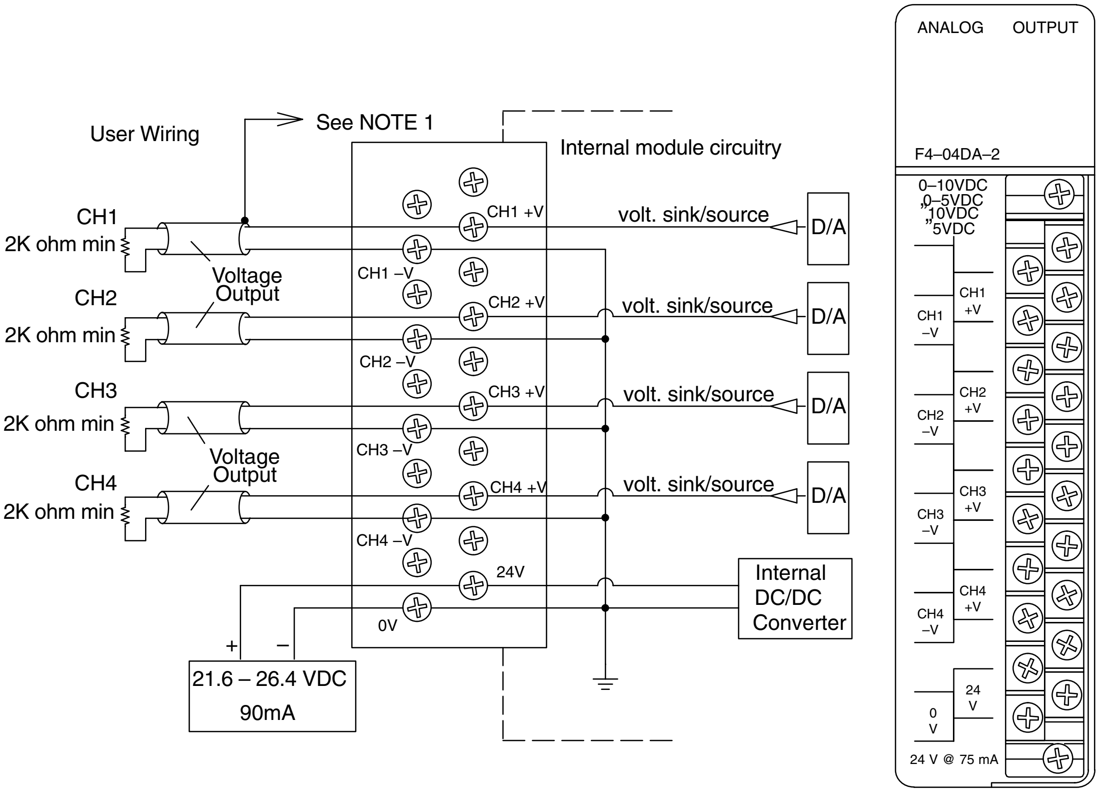
Load Requirements Each channel can be connected to a load with an impedance of 2k ohms or greater. Unused outputs should be left unconnected.

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).

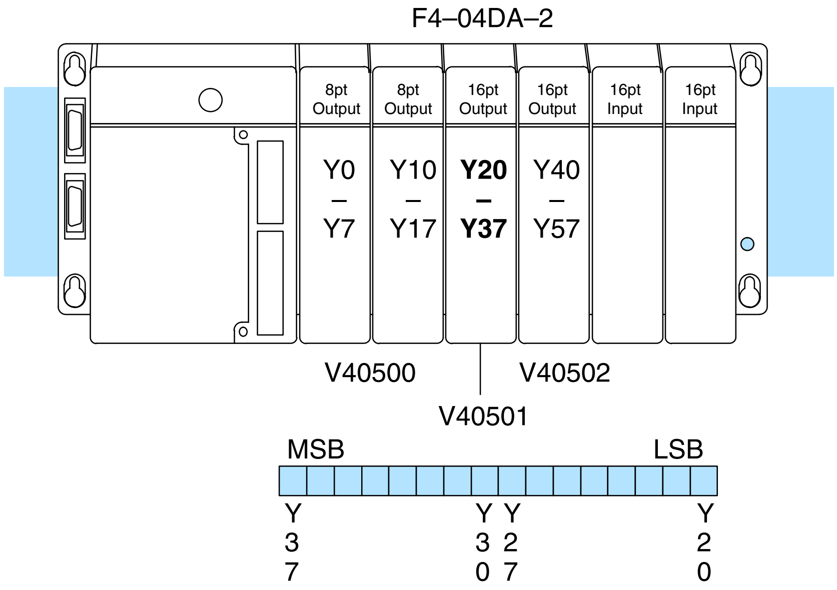

User Wiring
See NOTE 1
Internal module circuitry
CH1
2K ohm min
CH2
2K ohm min
CH3
2K ohm min
CH4
2K ohm min
Voltage Output
Voltage Output
CH1 +V
CH1 –V
CH2 +V
CH2 –V
CH3 +V
CH3 –V
CH4 +V
CH4 –V
24V
0V
volt. sink/source
volt. sink/source
volt. sink/source
volt. sink/source
D/A
D/A
D/A
D/A
Internal DC/DC Converter
+
–
21.6 – 26.4 VDC
90mA
ANALOG OUTPUT
F4–04DA–2
0–10VDC
0–5VDC
„10VDC
„5VDC
CH1 +V
CH1 –V
CH2 +V
CH2 –V
CH3 +V
CH3 –V
CH4 +V
CH4 –V
24 V
0 V
24 V @ 75 mA

Module Operation

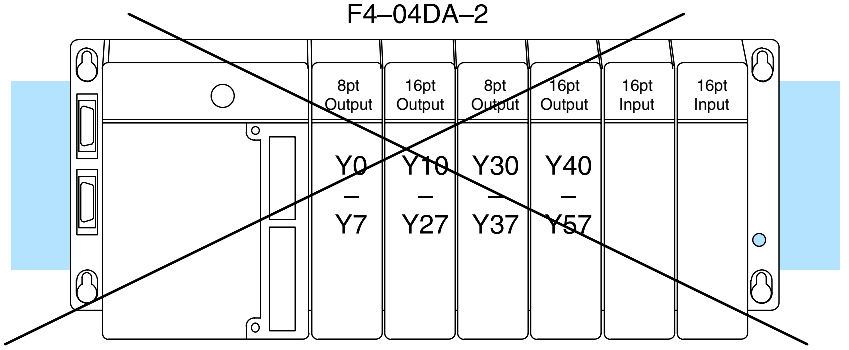
DL430 Special Requirements

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.

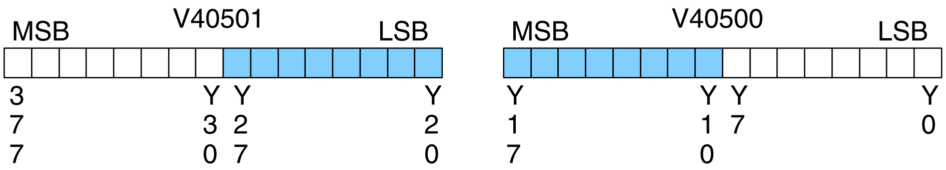
Correct!


F4–04DA–2
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40501
V40502
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

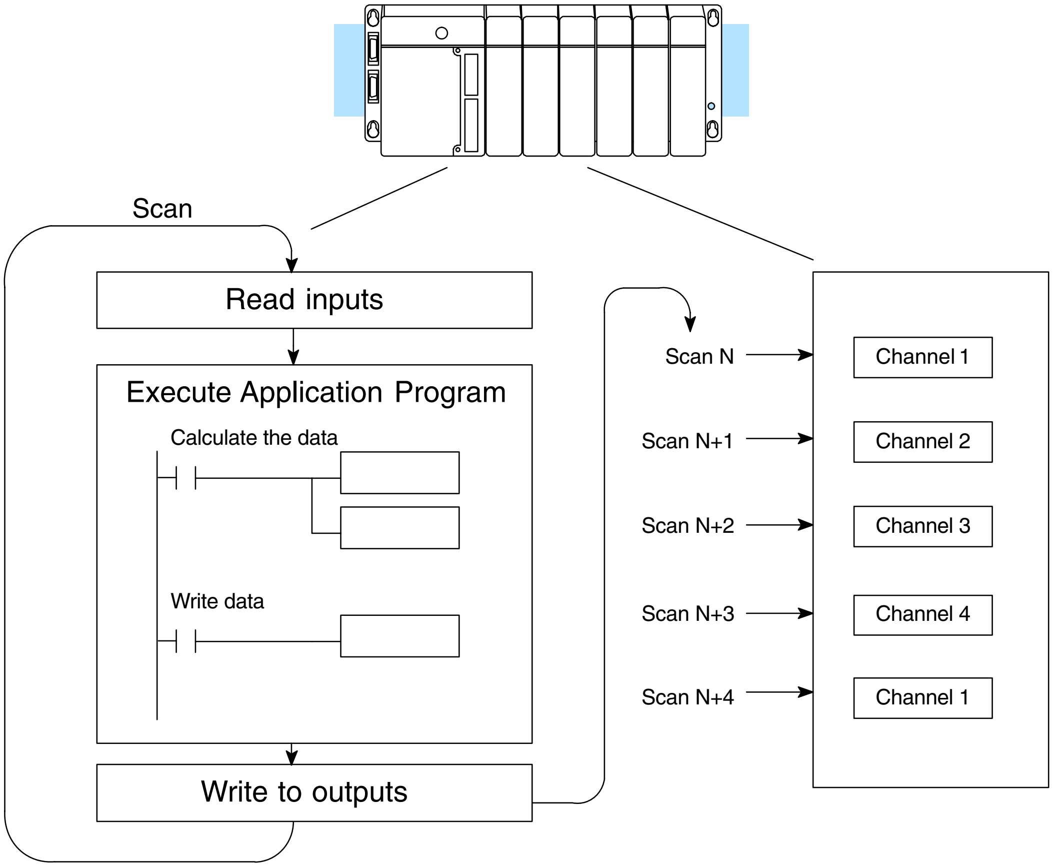
Data is split over two locations, so instructions cannot access data from a DL430.


MSB
V40501
LSB
3 7 7
Y 3 0
Y 2 7
Y 2 0
MSB
V40500
LSB
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).

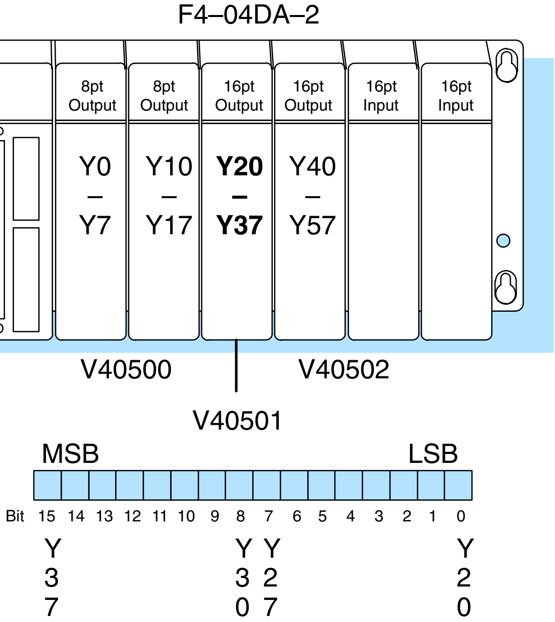

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 3
Channel 4
Channel 1

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.

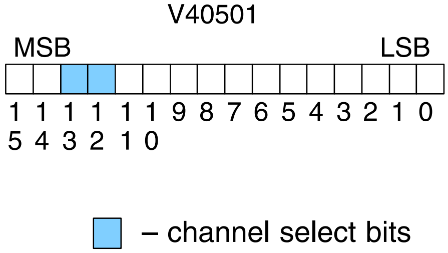

F4–04DA–2
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40501
V40502
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y37
Y30
Y27
Y20

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

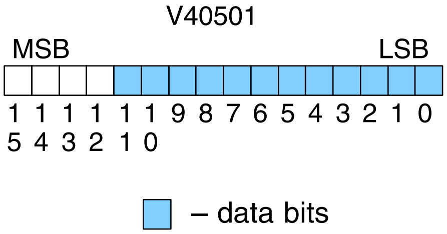

V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– channel select bits

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.

Analog Data Bits

The first twelve bits of the V-memory location represent the analog data in binary format.

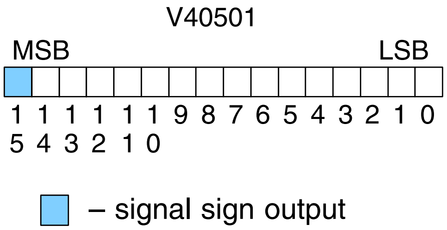
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



Signal Sign Bit

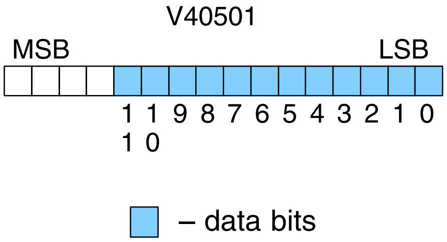
The most significant bit (bit 15) is the sign bit, when the data format for any channel is selected to be –2047 to +2047. By controlling this output (Y37 in our example), you can easily select positive or negative values.

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



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”.



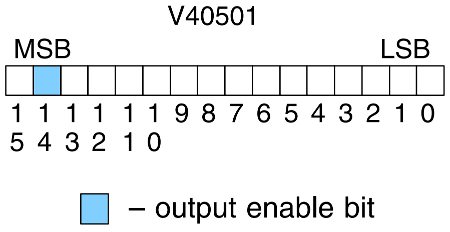
Bit 11 must be “0” for output format of –2047 to +2047.

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.

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

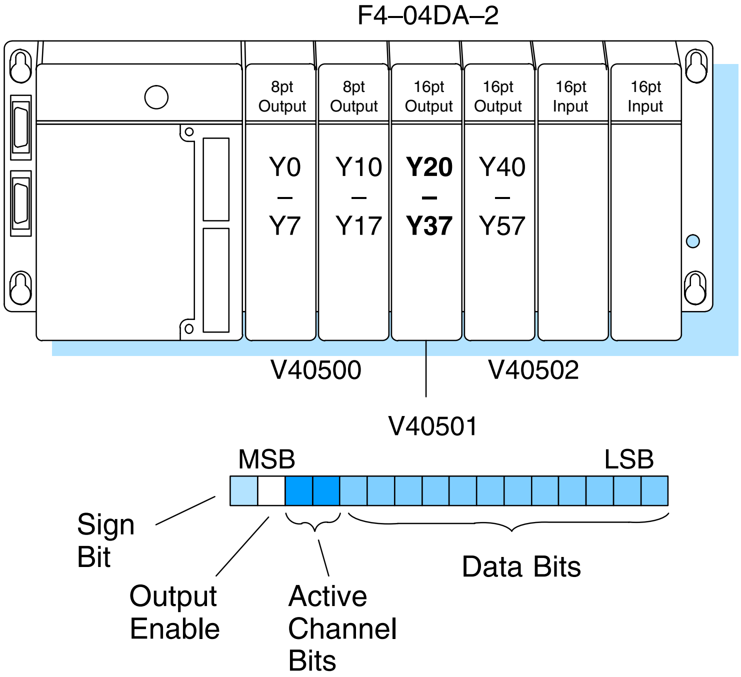
L = low limit of the signal range

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).


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
Sign Bit
Output Enable
Active Channel Bits
Data Bits

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

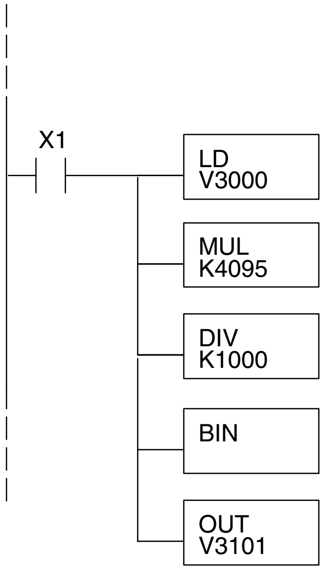
Consider the following example which controls pressure from 0.0 to 99.9 PSI. By using the formula, you can easily determine the digital value that should be sent to the module. The example shows the conversion required to yield 49.4 PSI. Notice the formula uses a multiplier of 10. This is because the 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary.

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

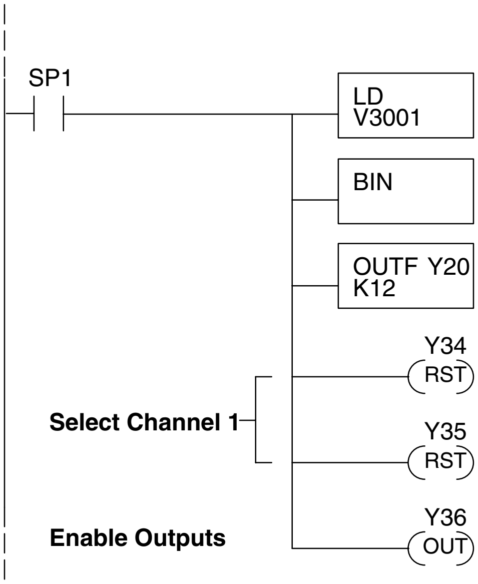
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

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



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.

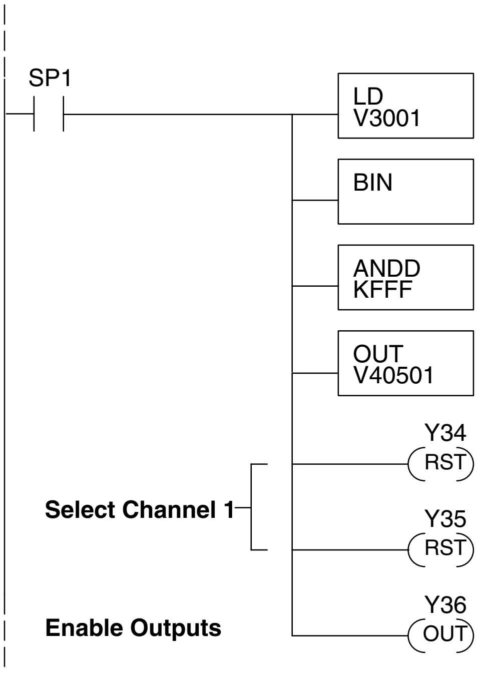
Select channel 1 for updating.

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



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.

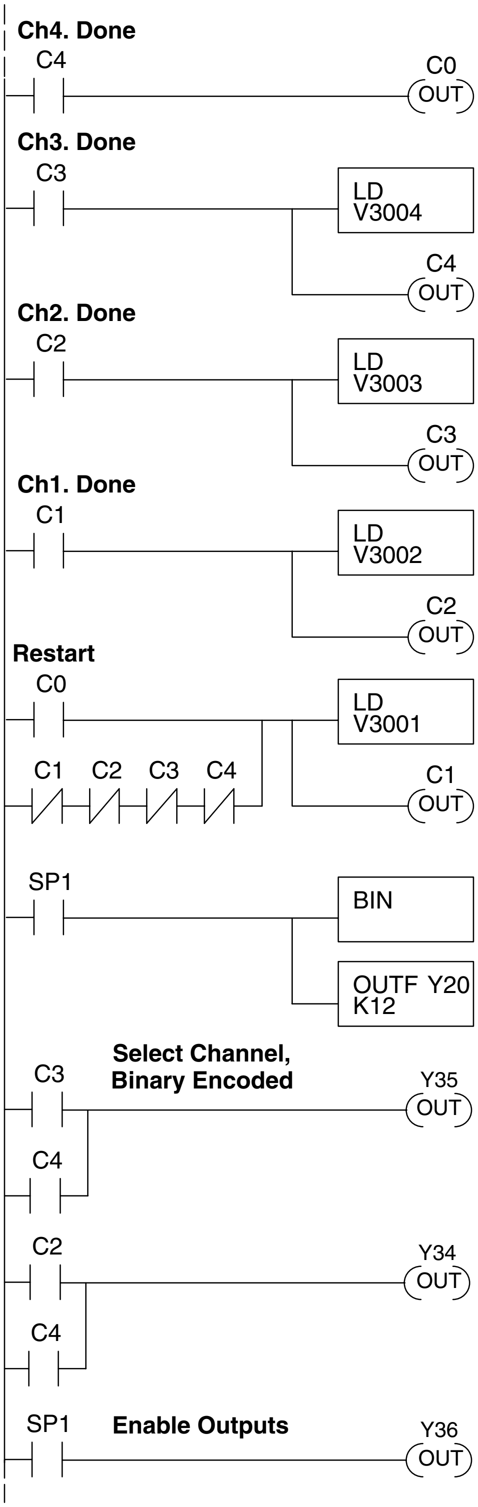
Example 5 is similar to example 2, but is modified for use with bipolar outputs. This example works with DL430, DL440 and DL450 CPUs.

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
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
LD
V3001
C1 C2 C3 C4
C1
OUT
SP1
BIN
OUTF Y20
K12
Select Channel,
Binary Encoded
C3
Y35
OUT
C4
C2
Y34
OUT
C4
SP1
Enable Outputs
Y36
OUT

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

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).

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).

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).

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.

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.

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

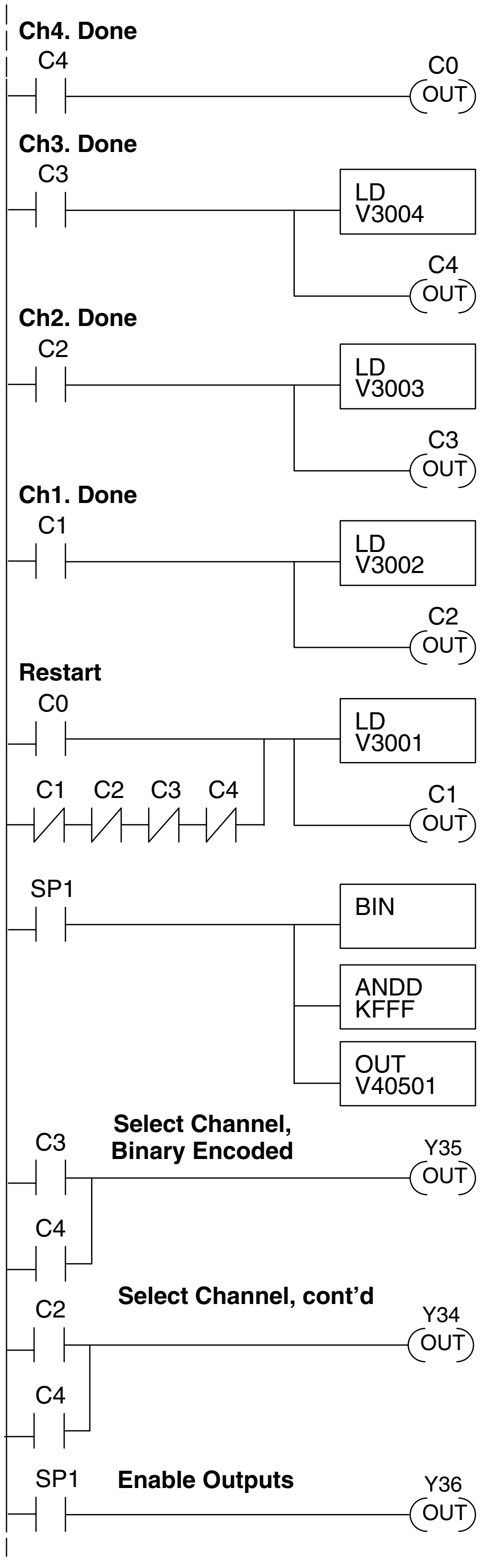
CR(on)	Y35	Y34	Channel
C1	Off	Off	Ch. 1
C2	Off	On	Ch. 2
C3	On	Off	Ch. 3
C4	On	On	Ch. 4

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
C0
OUT
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
LD
V3001
C1 C2 C3 C4
C1
OUT
SP1
BIN
ANDD
KFFF
OUT
V40501
Select Channel, Binary Encoded
C3
Y35
OUT
C4
Select Channel, cont'd
C2
Y34
OUT
C4
SP1
Enable Outputs
Y36
OUT

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

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).

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).

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).

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.

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.

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

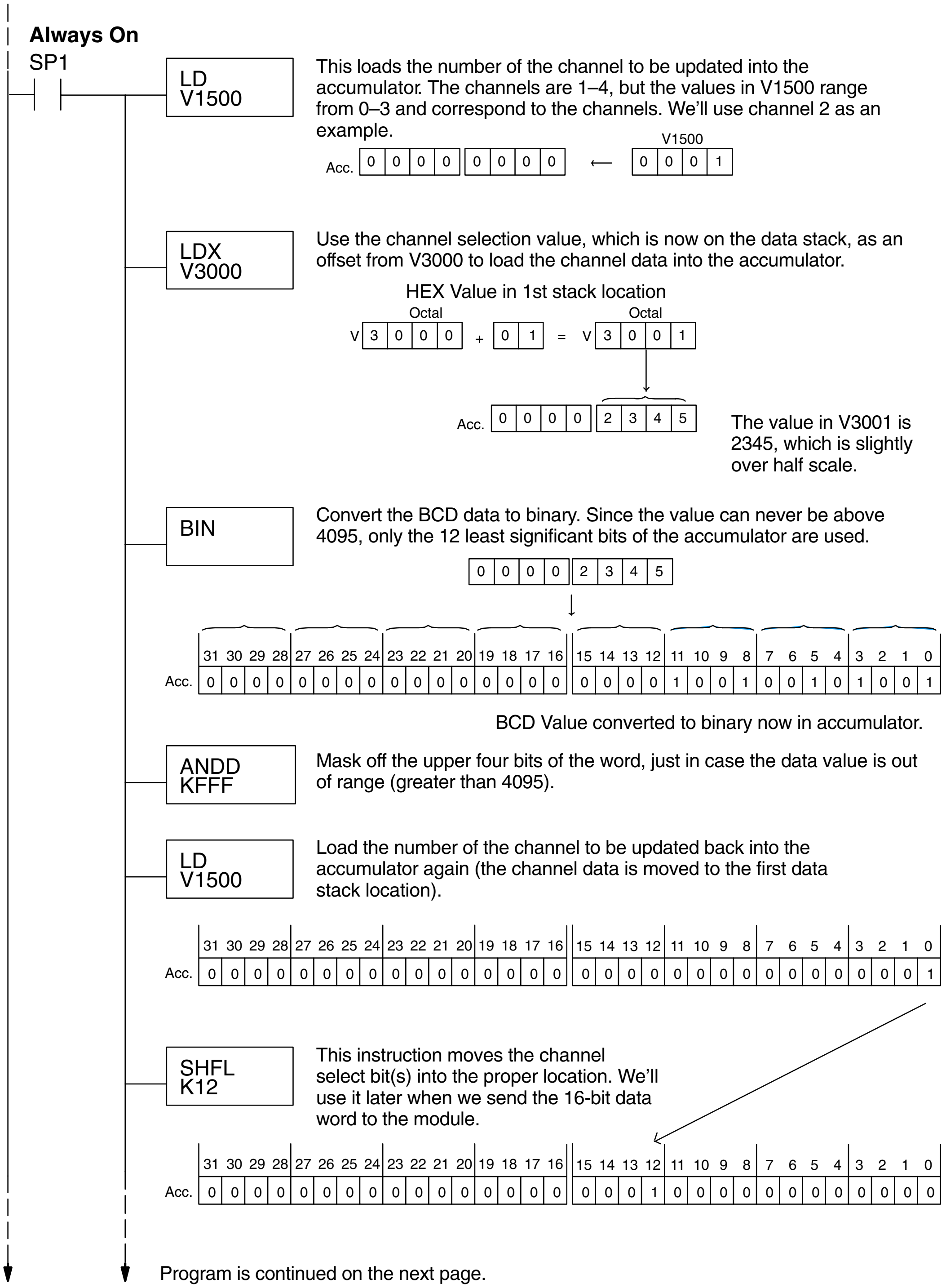
Enables all four output channels. SP1 is always on.

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



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.

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.

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

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.

	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.

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

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).

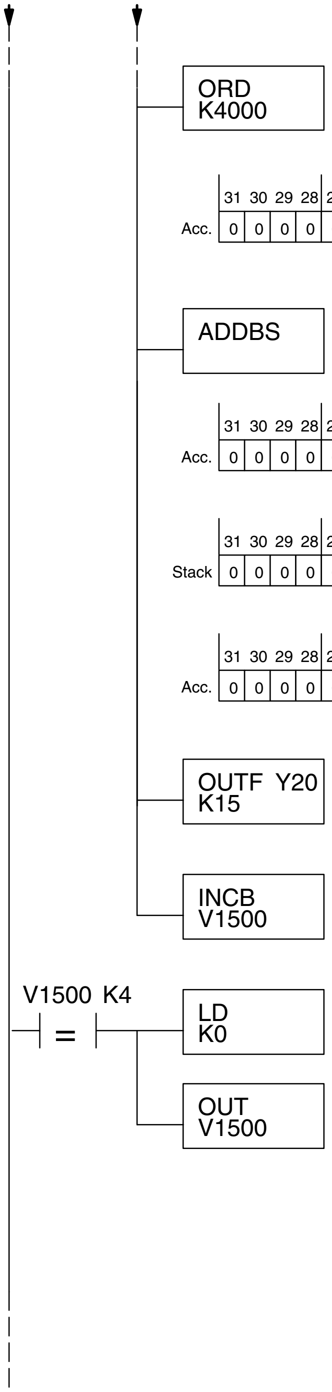
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

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



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

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

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.

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

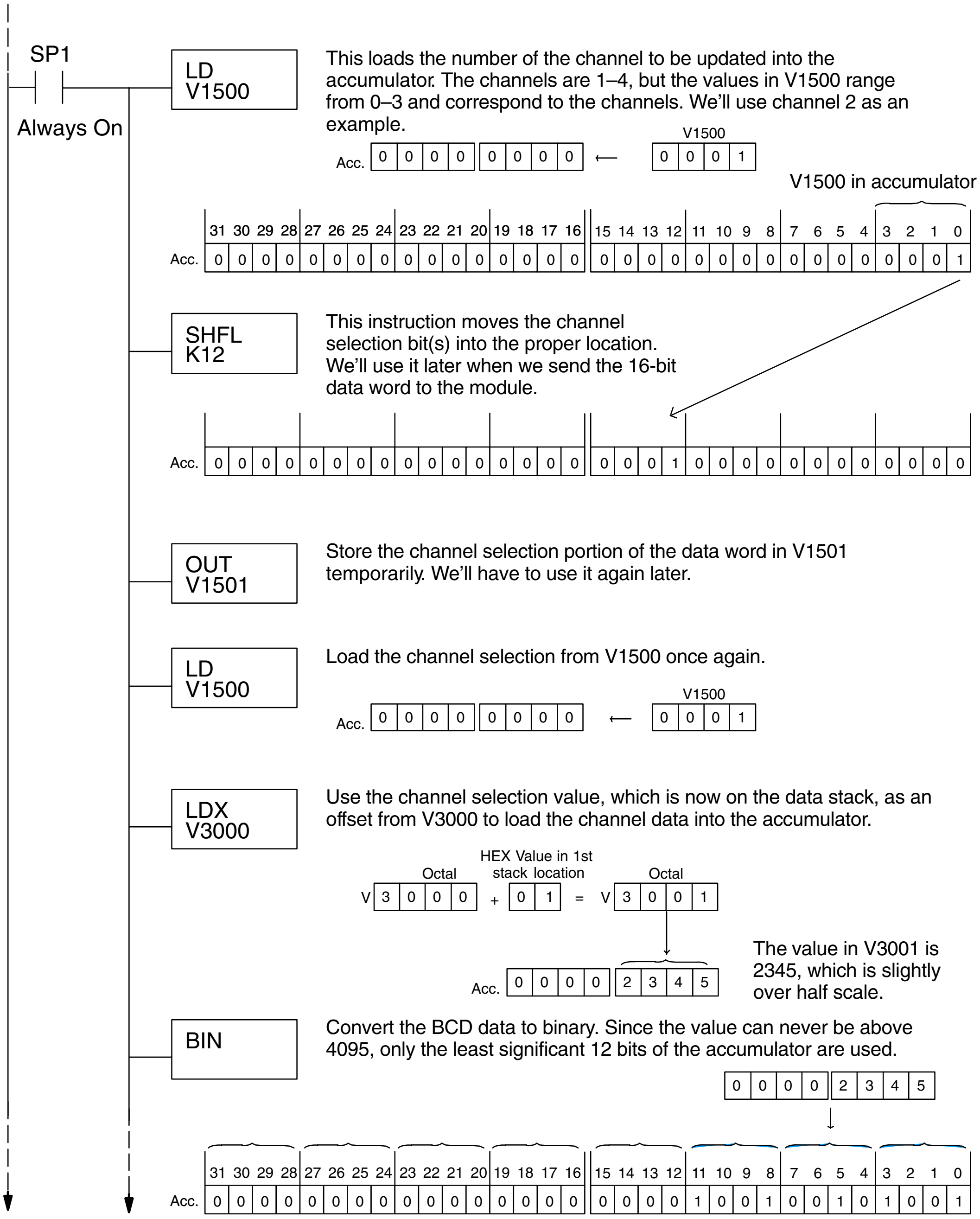
When channel four has been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1).

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
Acc.
V1500
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
SHFL
K12
OUT
V1501
LD
V1500
LDX
V3000
Octal
HEX Value in 1st stack location
Octal
V 3 0 0 0 + 0 1 = V 3 0 0 1
Acc. 0 0 0 0 2 3 4 5
BIN

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.

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.

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

Load the channel selection from V1500 once again.

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.

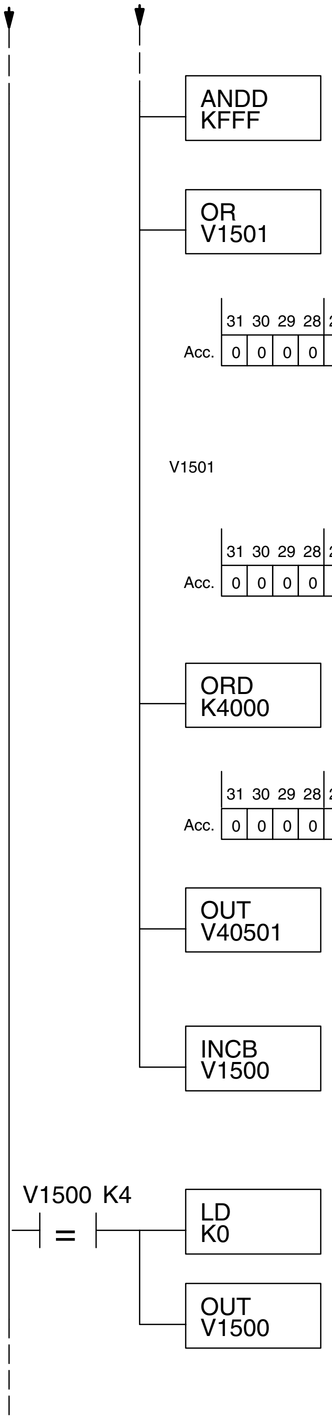
The value in V3001 is 2345, which is slightly over half scale.

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.

Program is continued on the next page.

BCD Value converted to binary now in accumulator.

Example 4 Continued



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

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

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

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.

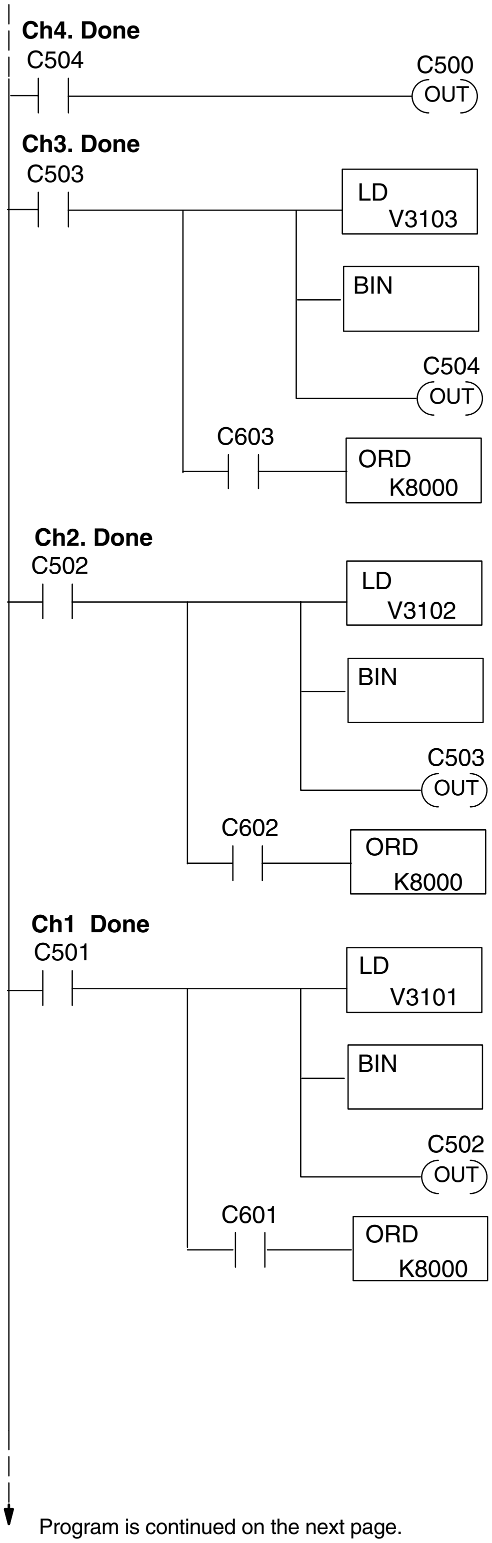
Increment the channel selection value. This allows the logic to cycle through all four channels.

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

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
Ch3. Done
C503
LD
V3103
BIN
C504
OUT
C603
ORD
K8000
Ch2. Done
C502
LD
V3102
BIN
C503
OUT
C602
ORD
K8000
Ch1 Done
C501
LD
V3101
BIN
C502
OUT
C601
ORD
K8000

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

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.

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).

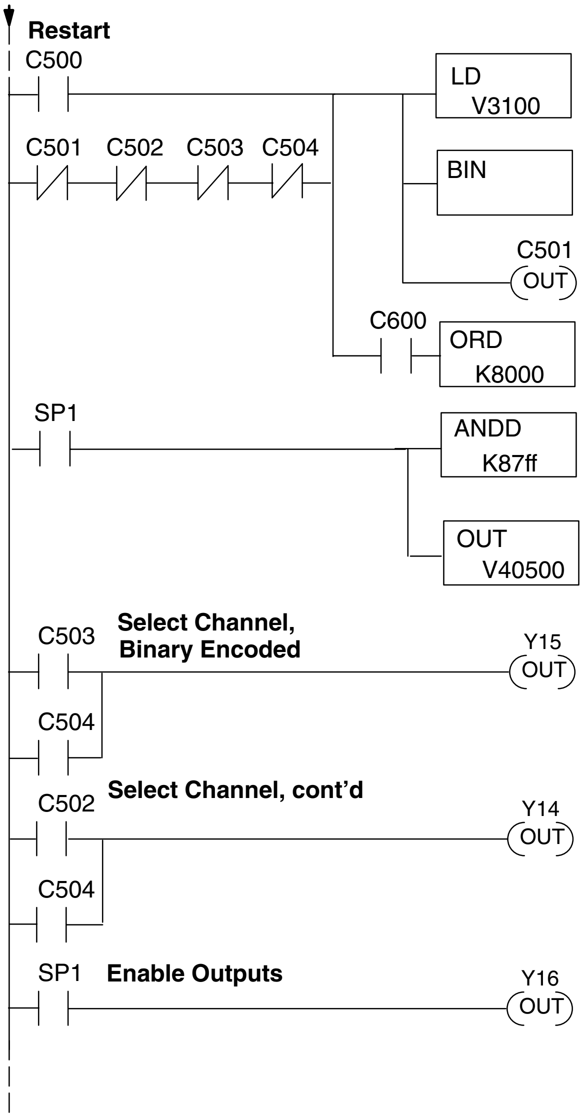
When channel 3's data is negative (indicated by C602), set the sign bit.

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
C600
ORD
K8000
SP1
ANDD
K87ff
OUT
V40500
C503
Select Channel,
Binary Encoded
Y15
OUT
C504
C502
Select Channel, cont'd
Y14
OUT
C504
SP1
Enable Outputs
Y16
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.

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

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

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.

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

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

Enables all four output channels. SP1 is always on.

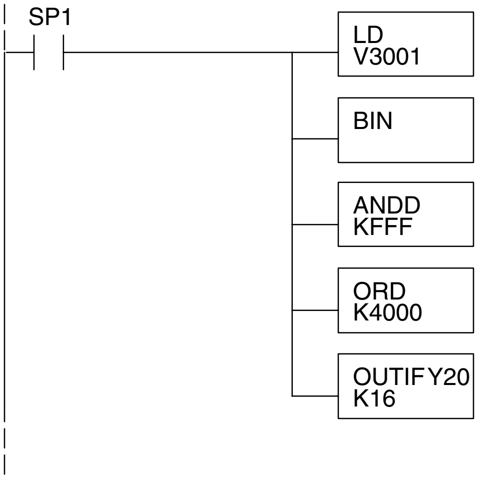
Updating all Channels in a Single Scan, DL440/450

☒ ✓ ✓
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



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

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.

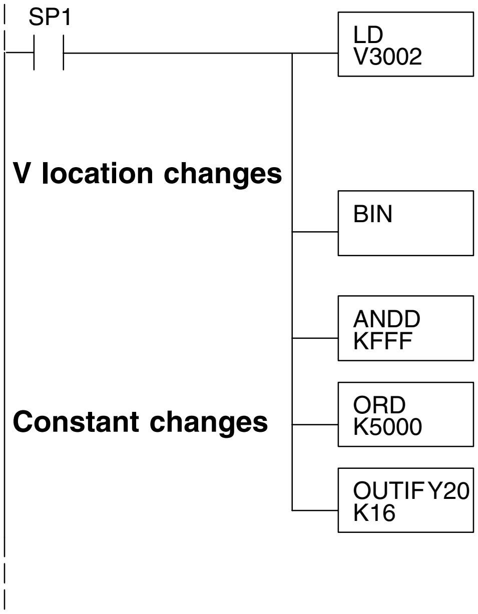
The ORD instruction (with K4000) sets the Output Enable bit. Outputs Y34 and Y35 are left off to select channel 1 for updating with the data.

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



The LD instruction loads the data for channel 2 into the accumulator.

Location	Channel
V3001	1
V3002	2
V3003	3
V3004	4

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

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

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.

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	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}(A)$
	–2047 to +2047	$A = \frac{5(D + 2047)}{4095}$	$D = \frac{4095(A)}{5} - 2047$
0 – 10V	0 to 4095	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}(A)$
	–2047 to +2047	$A = \frac{10(D + 2047)}{4095}$	$D = \frac{4095(A)}{10} - 2047$
± 5V	0 to 4095	$A = \frac{10D}{4095} - 5$	$D = \frac{4095}{10}(A + 5)$
	–2047 to +2047	$A = \frac{5(D)}{2047}$	$D = \frac{2047(A)}{5}$
±10V	0 to 4095	$A = \frac{20D}{4095} - 10$	$D = \frac{4095}{20}(A + 10)$
	–2047 to +2047	$A = \frac{10(D)}{2047}$	$D = \frac{2047(A)}{10}$

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.

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

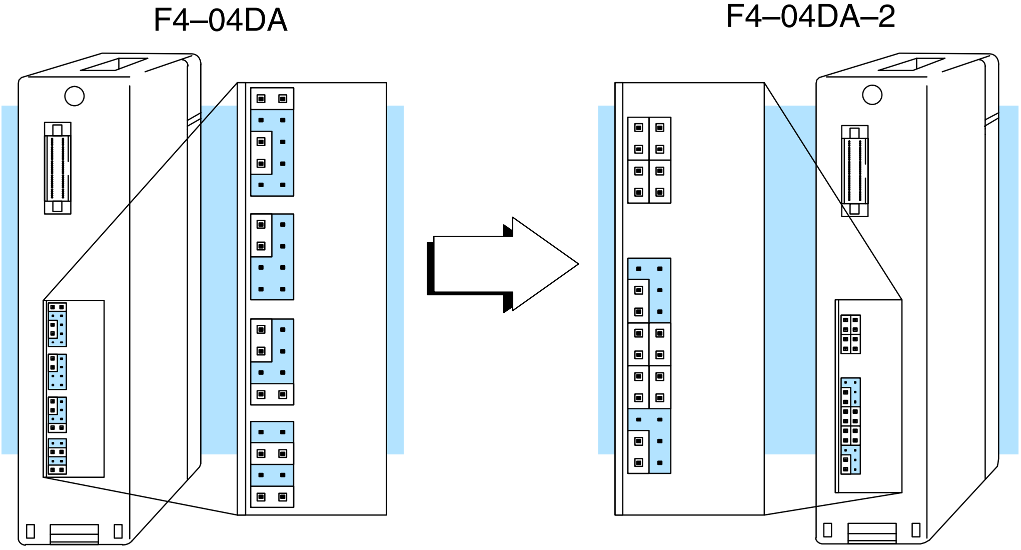
$$D = \frac{-6141}{10}$$

$$D = -614$$

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.

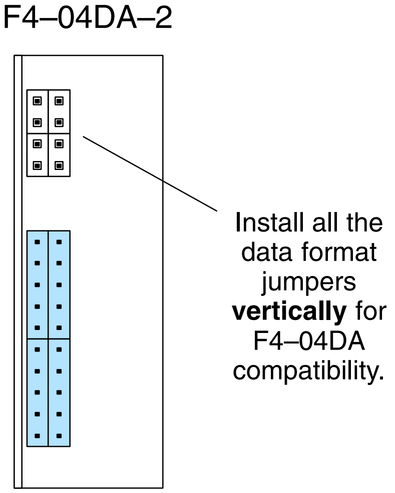


In the following procedure, we will examine the present jumper settings on the F4–04DA. Just follow the steps to translate that configuration into its equivalent on the F4–04DA–2 module.

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.

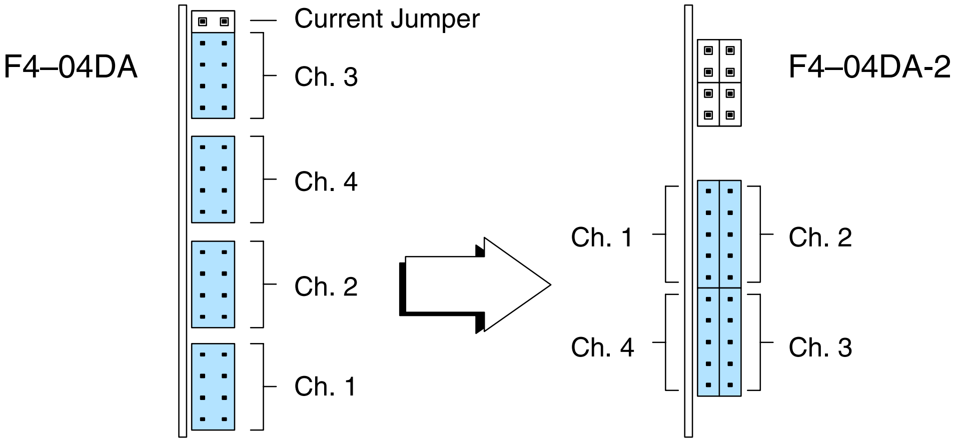
You may recall that the F4–04DA has only one data format (0 to 4095) for all channels. This data format is in effect now in the F4–04DA–2's compatibility mode.



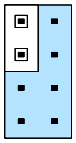
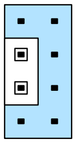
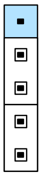
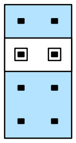
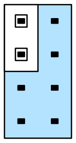
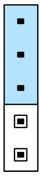
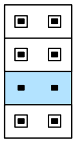
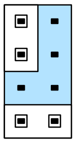
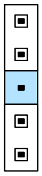
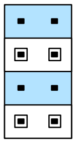
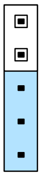
Install all the data format jumpers **vertically** for F4–04DA compatibility.

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		

15

F4–08DA–1 8-Channel Analog Current Output

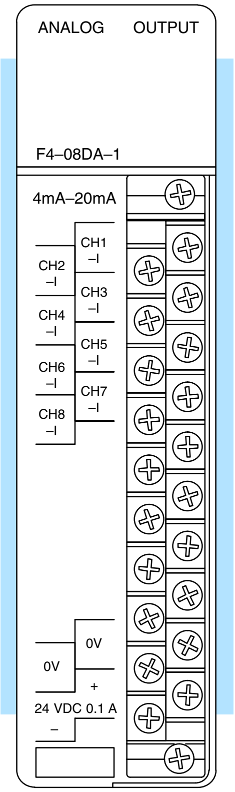
In This Chapter. . . .

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

Module Specifications

The F4–08DA–1 Analog Current Output Module provides several features and benefits.

- It provides eight channels of 4–20 mA single ended current outputs.
- 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.
- From one to eight analog outputs may be updated in one CPU scan (DL440 and DL450 CPUs only).
- No jumper settings are required.



Analog Output Configuration Requirements

The F4–08DA–1 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 tables provide the specifications for the F4–08DA–1 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

Number of Channels	8, single ended (one common)
Output Range	4–20 mA current
Resolution	12 bit (1 in 4095)
Output Type	Outputs sink 4–20 mA from external supply
Peak Output Voltage	40 VDC (no transient voltage suppression)
External Load Resistance	0–480Ω at 18V, 220–740Ω at 24V, 1550–1760Ω at 48V
Maximum Loop Supply	48 VDC (with load resistance in proper range)
Crosstalk	–70 dB, ±1 count maximum
Linearity Error (end-to-end) and Relative Accuracy	±1 count maximum
Full Scale Calibration Error (offset error included)	±8 counts maximum (20mA at 25°C)
Offset Calibration Error	±3 counts maximum (4mA at 25°C)
Maximum Inaccuracy	±0.2% at 25°C (77°F) ±0.4% at 0 to 60°C (32 to140°F)
Conversion Time	400 μs maximum, for full scale change 2.25 to 4.5 mS for digital output to analog out

General Module Specifications

Digital Output Points Required	16 point (Y) outputs, 12 bits binary data, 3 bits channel select, 1 bit output enable
Power Budget Requirement	90 mA at 5 VDC (supplied by base power supply)
External Power Supply	21.6-26.4 VDC, 100 mA, class 2 (add 20 mA for each current loop used)
Accuracy vs. Temperature	±57 ppm / °C full scale calibration range (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

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.
- 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 F4–08DA–1 requires at least one field-side supply. You may use the same or separate power sources for the module supply and loop supply. The module requires 21.6 – 26.4 VDC, Class 2, 100mA current. The eight current loops require 18 to 48 VDC, at 20 mA each.

The DL430/440/450 CPUs, D4–RS Remote I/O Controller, and D–EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. You may use one of these instead of a separate supply if there is only a couple of analog modules. The current required is 100 mA (module), plus 160 mA (eight current loops) for a total of 260 mA.

In some situations it is desirable to power the loops separately due to power budget or due to their remote location from the PLC. This will work, as long as the loop supply meets the voltage and current requirements, and its minus (–) side and the module supply's minus (–) side connect together.

Load Requirements

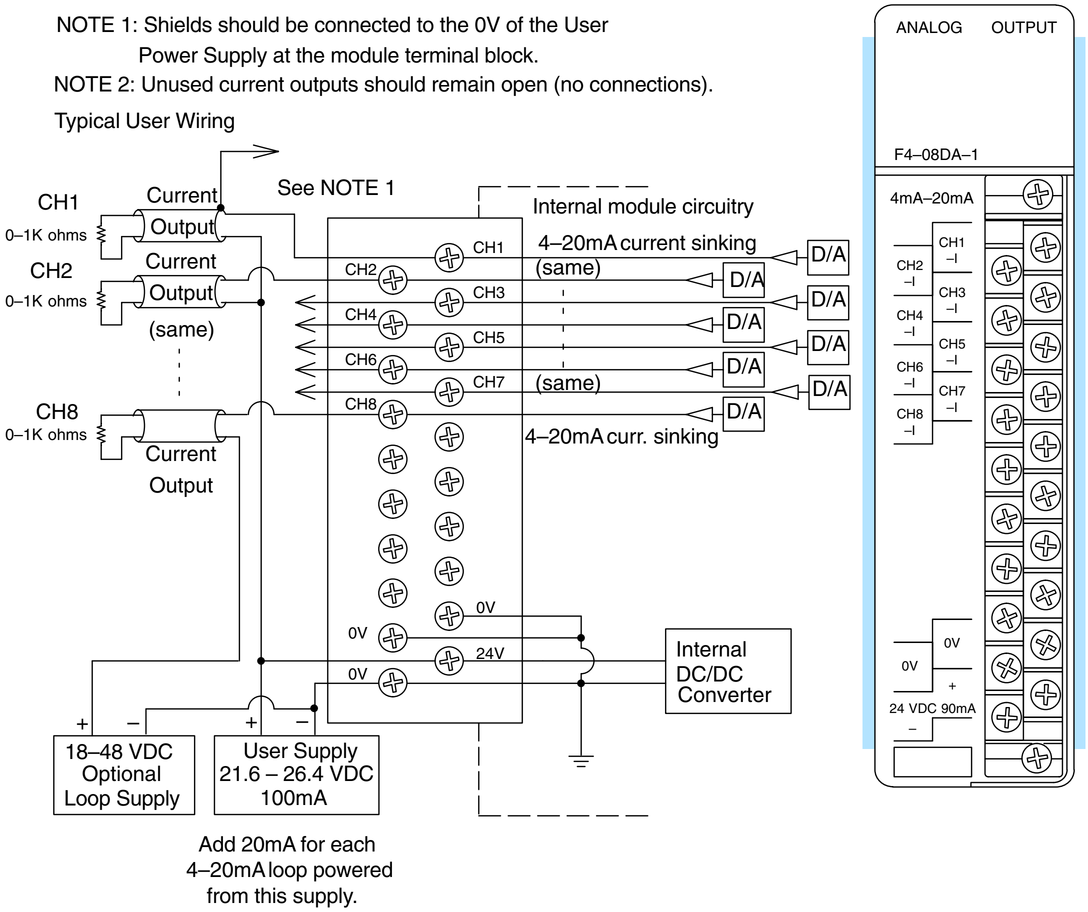
Each channel in use must have a load impedance of 0–480Ω at 18V, 220–740Ω @ 24V, or 1550–1760Ω at 48V. Unused channels must be left disconnected.

WARNING: If you are using the 24 VDC base power supply, 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.

Removable Connector

The F4–08DA–1 module has a removable connector to make wiring easier. Simply loosen the retaining screws and gently pull the connector from the module. The diagram shows separate module and loop supplies for channel 8. If you desire to use only one field-side supply, just combine the supplies' positive terminals into one node, and remove the loop supply.

Wiring Diagram



NOTE 1: Shields should be connected to the 0V of the User Power Supply at the module terminal block.

NOTE 2: Unused current outputs should remain open (no connections).

Typical User Wiring

Add 20mA for each 4–20mA loop powered from this supply.

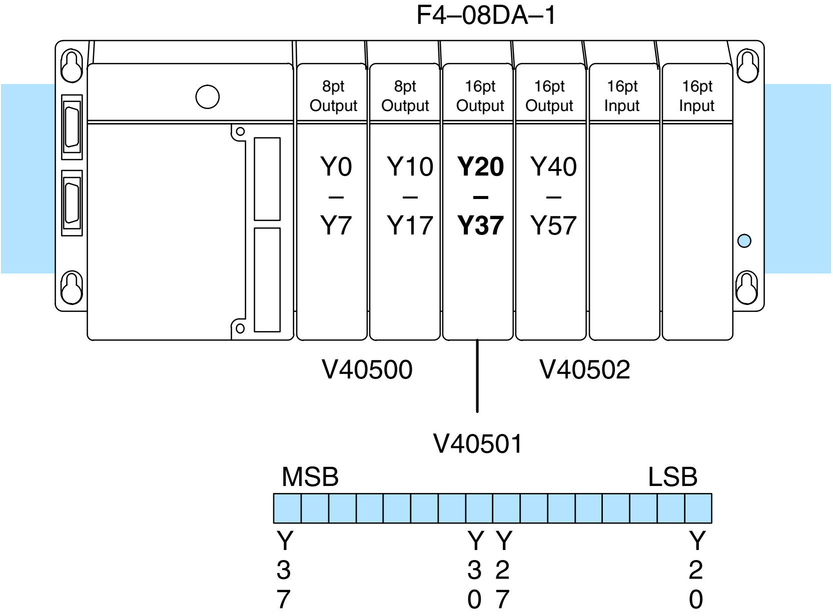
See NOTE 1
CH1
0–1K ohms
Current Output
CH2
0–1K ohms
Current Output
(same)
CH8
0–1K ohms
Current Output
Internal module circuitry
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
4–20mA current sinking
(same)
(same)
4–20mA curr. sinking
D/A
0V
0V
24V
0V
Internal DC/DC Converter
+
–
18–48 VDC Optional Loop Supply
+
–
User Supply 21.6 – 26.4 VDC 100mA
ANALOG OUTPUT
F4–08DA–1
4mA–20mA
CH1 –I
CH2 –I
CH3 –I
CH4 –I
CH5 –I
CH6 –I
CH7 –I
CH8 –I
0V
0V
+
24 VDC 90mA
–

Module Operation

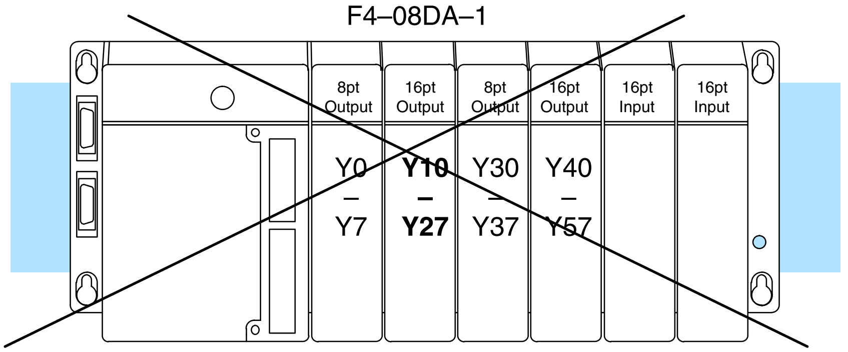
DL430 Special Requirements

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'll 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.

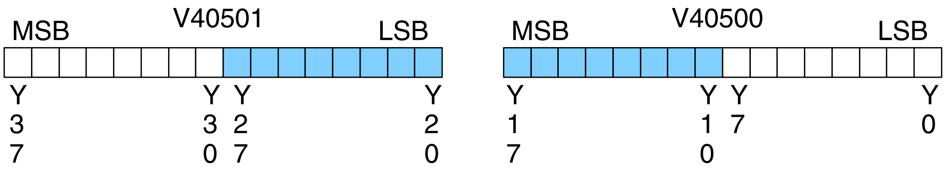
Correct!


F4–08DA–1
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
Y37
Y30
Y27
Y20

Wrong!


F4–08DA–1
8pt Output
16pt Output
8pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y27
Y30 – Y37
Y40 – Y57

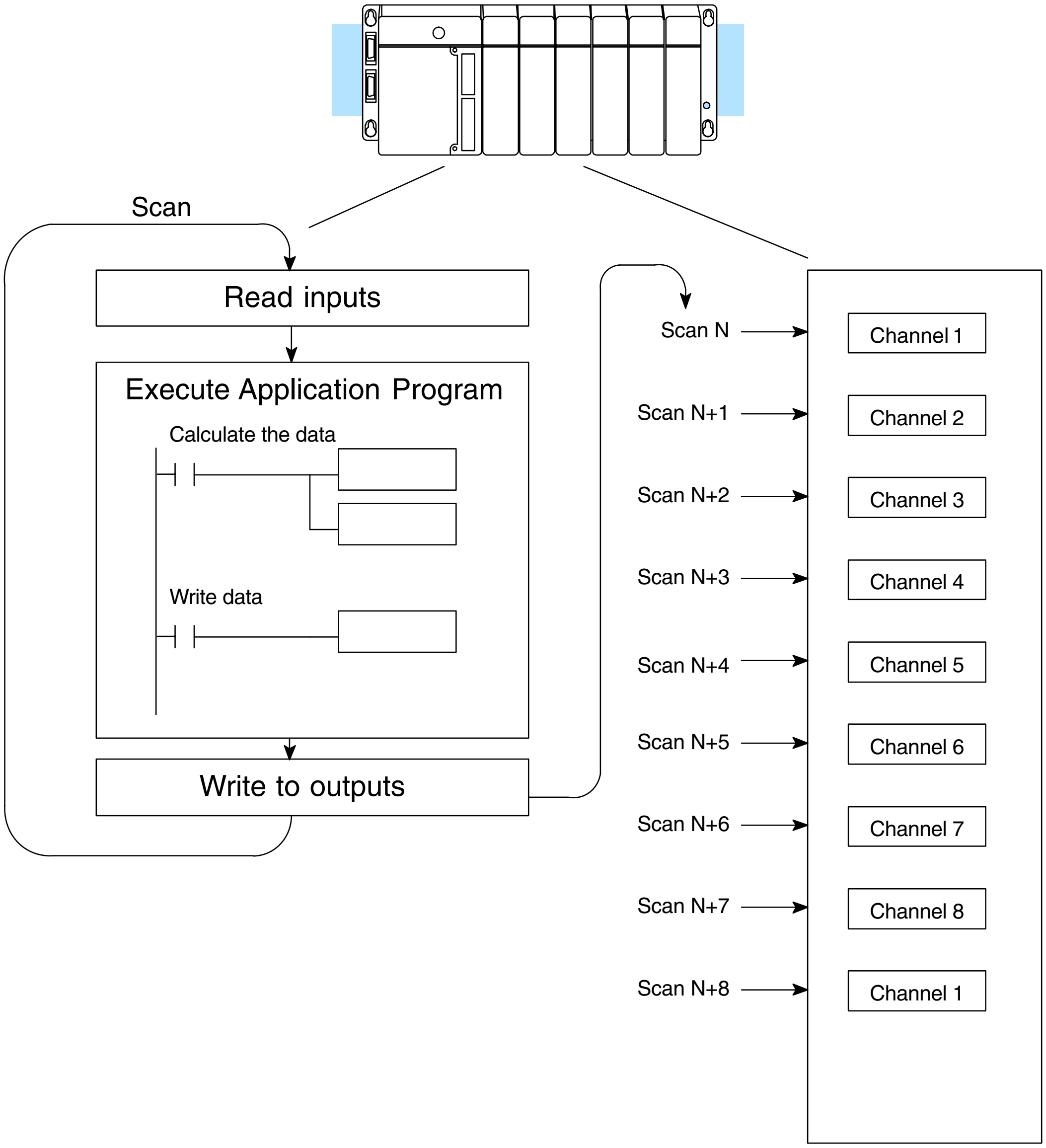
Data is split over two locations, so instructions cannot access data from a DL430.


MSB
V40501
LSB
Y37
Y30
Y27
Y20
MSB
V40500
LSB
Y17
Y10
Y7
Y0

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 F4–08DA–1 module allows you to update the channels in any order. Your control program determines which channels get updated on any given scan. A typical ladder program will update one channel per CPU scan. So, all eight channels can be updated every eight scans. With a DL440 or DL450 CPU, you can use Immediate instructions to update all eight channels in the same scan (we'll show you how to do this later).

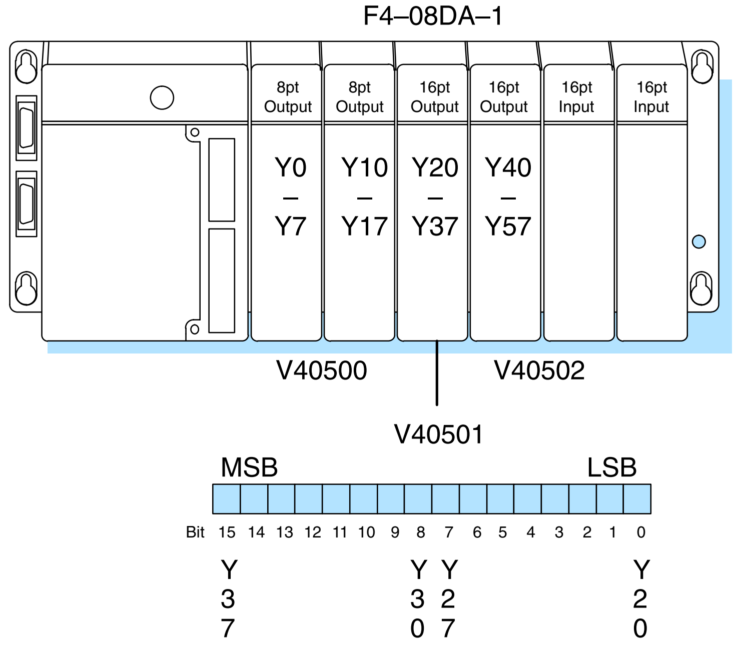

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Scan N+8
Channel 1
Channel 2
Channel 3
Channel 4
Channel 5
Channel 6
Channel 7
Channel 8
Channel 1

Output Bit Assignments

You may recall the F4–08DA–1 module requires 16 discrete output points from the CPU. These points provide:

- The digital representation of one analog signal per scan.
- Identification of the channel that is to receive the data.
- Output enable control for all channels.

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.

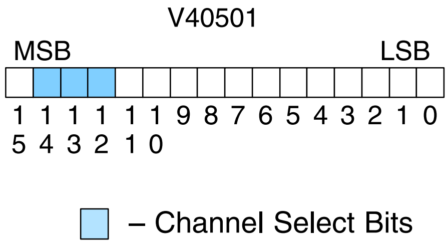

F4–08DA–1
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40502
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y37
Y30
Y27
Y20

Within this V-memory location the individual bits represent specific information about the channel selected and the analog signal.

Channel Select Bits

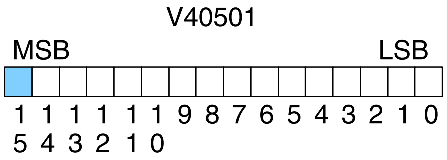
Output bits 12, 13, and 14 of the data word are the channel select outputs. They are binary encoded to select the channel from 1 to 8 that will be updated with the data.

Bit 14	Bit 13	Bit 12	Channel
Off	Off	Off	1
Off	Off	On	2
Off	On	Off	3
Off	On	On	4
On	Off	Off	5
On	Off	On	6
On	On	Off	7
On	On	On	8


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– Channel Select Bits

Output Enable Bit

Output bit 15 is the Output Enable control bit for all eight channels. When the bit is off, all eight channel output currents decrease to their lowest level, which is 4 mA for connected loads. Disabling the outputs also clears all eight output data registers. To resume analog output levels, first the Output Enable control bit must turn on. Then, the CPU must write new data to each channel to restore the output current for that channel.



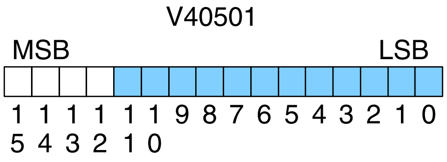
🟦 – Output Enable Bit

OFF = Disable (and clear)
ON = Enable

Analog Data Bits

The first twelve bits of the data word represent the analog data in binary format.

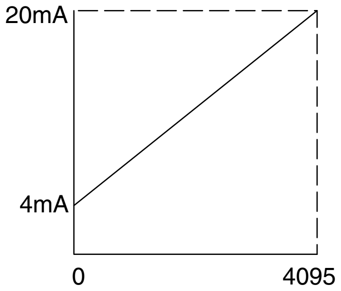
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



🟦 – data bits

Module Resolution

Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0–4095 (2^{12}). For the 4 to 20 mA scale, sending a 0 produces a 4 mA signal, and 4095 gives a 20 mA signal. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The graph to the right shows the linear relationship between the data value and output signal level.



Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal change that occurs when the digital value is increased by 1 LSB.

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

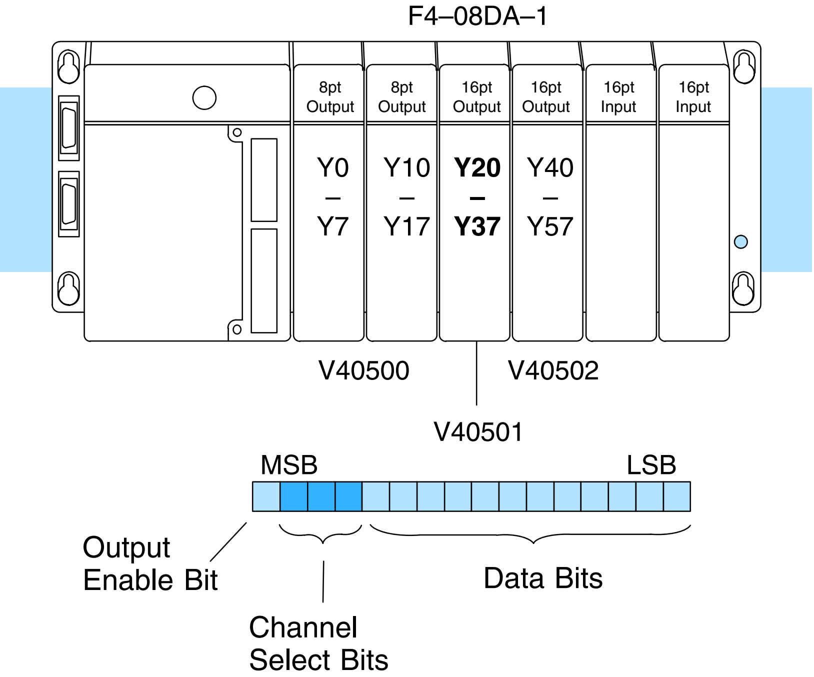
H = high limit of the signal range
L = low limit of the signal range

Signal Range	Span (H – L)	Divide By	Smallest Change
4 to 20mA	16mA	4095	3.91 µA

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 select outputs to determine which channel gets updated (more on this later).


F4–08DA–1
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
Output Enable Bit
Channel Select Bits
Data Bits

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.

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.

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

A = analog value (0 – 4095)

U = engineering units

H = high limit of the engineering unit range

L = low limit of the engineering unit range

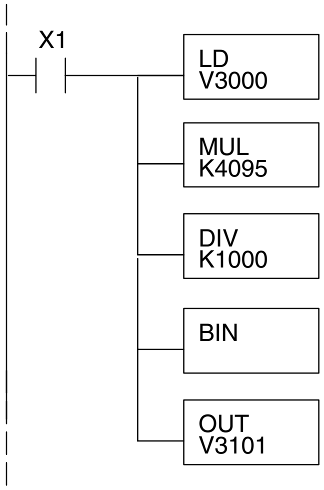
$$A = 10U \frac{4095}{10(H - L)}$$

$$A = 494 \frac{4095}{1000 - 0}$$

$$A = 2023$$

Here is 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 all eight channels if you are 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary.

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 slot of your analog module. See Appendix A for additional addresses available in the DL450 CPU.

V-Memory Register Addresses for 16-Point Output (Y) Locations

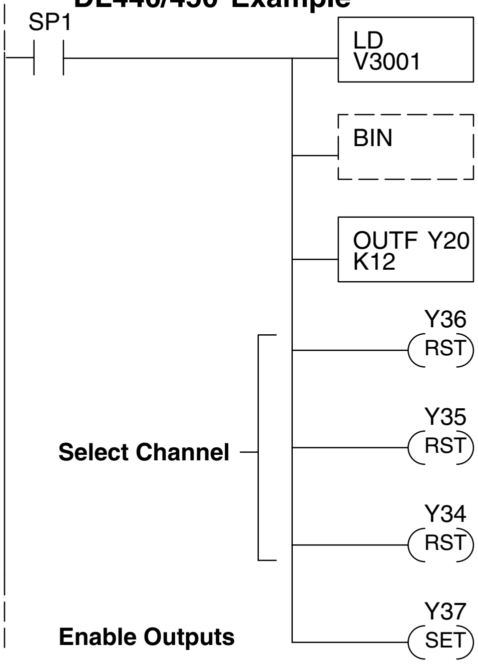
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

Sending Data to One Channel

The following programs show you how to update a single channel. Notice the DL430 CPU requires a slightly different program than the DL440/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. These examples assume you already have the data loaded in V3001.

✗ ✓ ✓
430 440 450

DL440/450 Example



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.

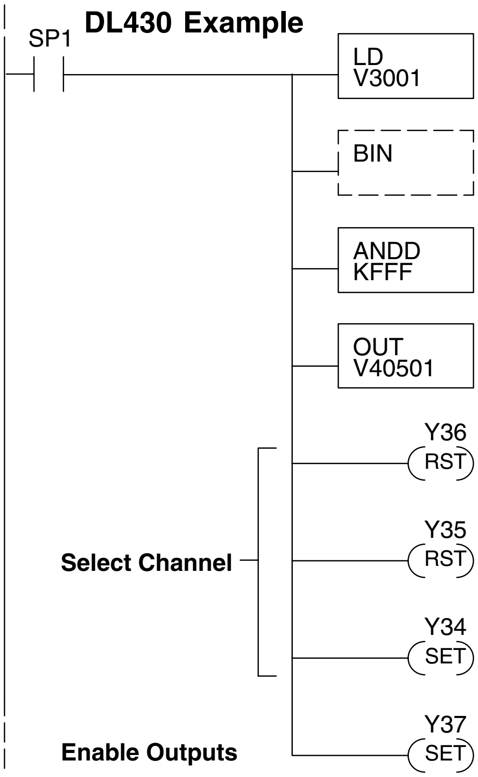
Turn Y36, Y35, and Y34 off to update Channel 1.

Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels.

✓ ✓ ✓
430 440 450

DL430 Example



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 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.

Turn Y36, Y35, and Y34 off to update Channel 1.

Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels.

Sequencing the Channel Updates

The next four 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 eight channels over eight 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 C8 as index numbers corresponding to the channel updated on any particular scan. At the end of each scan, only one control relay C1 through C8 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 2 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 4 with DL430 CPUs. Either example will work with DL440 or DL450 CPUs.

In the fifth example, we show you how you can update all eight 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 eight channels on every scan.

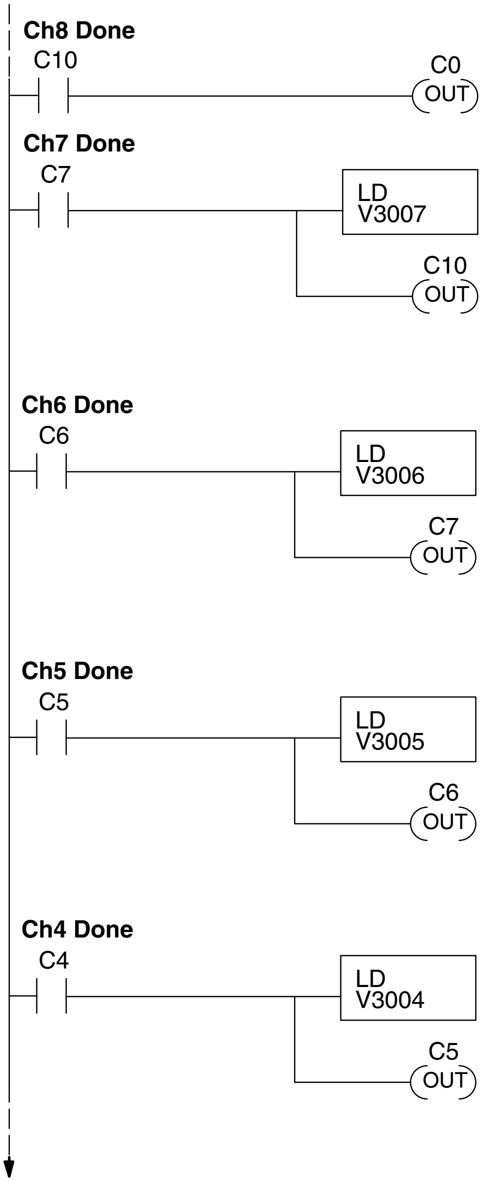
In the last example, we show you how you can update a single channel during the scan with DL440 and DL450 CPUs using the Immediate instructions.

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 have already loaded the data according to the following table. It is important to use the rungs in the order shown for the program to work. This example will not work with DL430 CPUs.

V-Memory Locations for Output Data in Examples 1 and 2								
Channel Number	1	2	3	4	5	6	7	8
V–Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007



When channel 8 is updated, C0 restarts the update sequence.

When channel 7 has been updated, this rung loads the data for channel 8 into the accumulator.

Turning on C10 triggers the channel update (see the channel select rungs).

When channel 6 has been updated, this rung loads the data for channel 7 into the accumulator.

Turning on C7 triggers the channel update (see the channel select rungs).

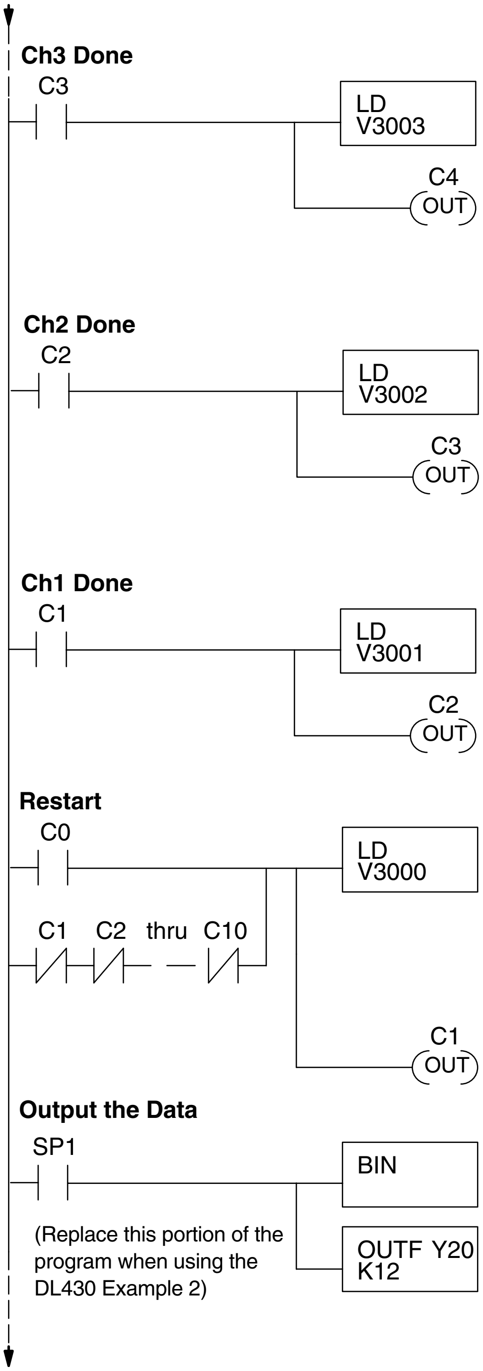
When channel 5 has been updated, this rung loads the data for channel 6 into the accumulator.

Turning on C6 triggers the channel update (see the channel select rungs).

When channel 4 has been updated, this rung loads the data for channel 5 into the accumulator.

Turning on C5 triggers the channel update (see the channel select rungs).

Example 1 Continued



(Replace this portion of the program when using the DL430 Example 2)

When channel 3 has been updated, this rung loads the data for channel 4 into the accumulator.

Turning on C4 triggers the channel update (see the channel select rungs).

When channel 2 has been updated, this rung loads the data for channel 3 into the accumulator.

Turning on C3 triggers the channel update (see the channel select rungs).

When channel 1 has been updated, this rung loads the data for channel 2 into the accumulator.

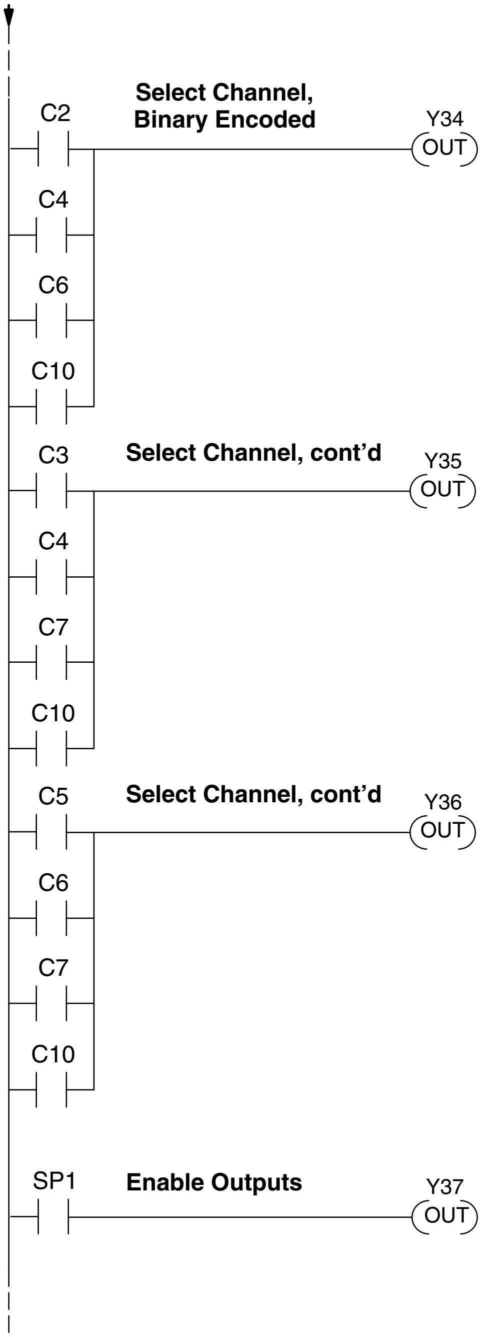
Turning on C2 triggers the channel update (see the channel select rungs).

This rung loads the data for channel 1 into the accumulator. C0 restarts the sequence after channel 8 is done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C1 through C10 are off (all eight contacts not shown here due to space constraints). In this case, we also start the sequence with channel 1.

Turning on C1 triggers the channel update (see the channel select rungs).

This rung converts the accumulator data for all channels (one per scan) 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.

Example 1 Continued


C2
Select Channel, Binary Encoded
Y34
OUT
C4
C6
C10
C3
Select Channel, cont'd
Y35
OUT
C4
C7
C10
C5
Select Channel, cont'd
Y36
OUT
C6
C7
C10
SP1
Enable Outputs
Y37
OUT

Set Y36, Y35, and Y34 to the binary code which selects the output channel 1 through 8, based on the control relay status.

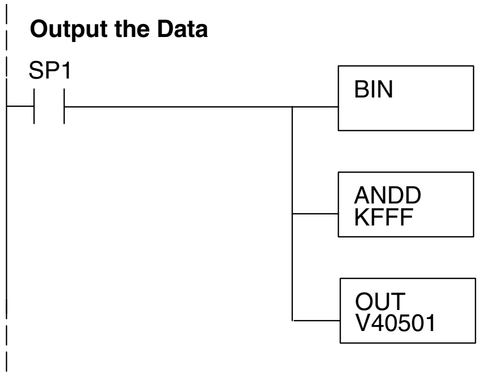
CR(on)	Y36	Y35	Y34	Channels
C1	Off	Off	Off	Ch. 1
C2	Off	Off	On	Ch. 2
C3	Off	On	Off	Ch. 3
C4	Off	On	On	Ch. 4
C5	On	Off	Off	Ch. 5
C6	On	Off	On	Ch. 6
C7	On	On	Off	Ch. 7
C10	On	On	On	Ch. 8

Enable all 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 or the output enable bits are not accidentally changed by the data in the accumulator. Replace the “Output the Data” rung in the middle of Example 1 with the new rung below. Be sure to retain the original order of the rungs shown in Example 1 for the program to work. This example will also work with DL440 and DL450 CPUs.



This rung converts the accumulator data for channels 1 through 8 (one per scan) 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 for the first bank of 8 channels, but the actual value depends on the location of the module in your application.

Sequencing Example 3, DL440/450

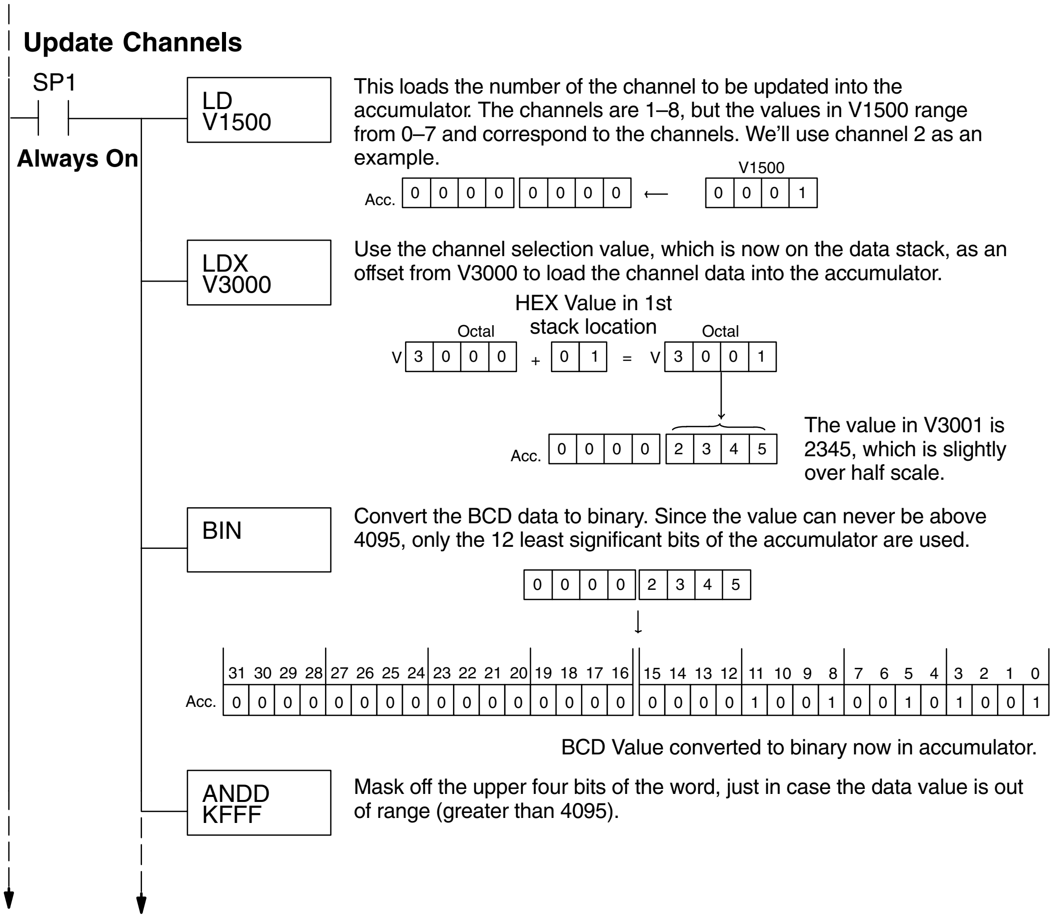
☒ ☑ ☑
430 440 450

The following program example shows how to send digital values to the module when you have more than one channel. This example works only for DL440 and DL450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 3								
Channel Number	1	2	3	4	5	6	7	8
V-Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007

The channel index is stored in V1500. It varies from 0 to 7, pointing to channels as shown: 0 – Ch. 1, 1 – Ch. 2, and 7 – Ch. 8. This example assumes V1500 is initialized to “0” earlier in the program.

This example program updates one channel during each scan. The program comments for this portion also shows the accumulator status at each step. The last portion of the program increments the channel index number and resets it after eight scans.


Update Channels
SP1
Always On
LD
V1500
Acc. 0 0 0 0 0 0 0 0
V1500
0 0 0 1
LDX
V3000
HEX Value in 1st
stack location
Octal
V 3 0 0 0 + 0 1 = V 3 0 0 1
Acc. 0 0 0 0 2 3 4 5
BIN
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
ANDD
KFFF

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

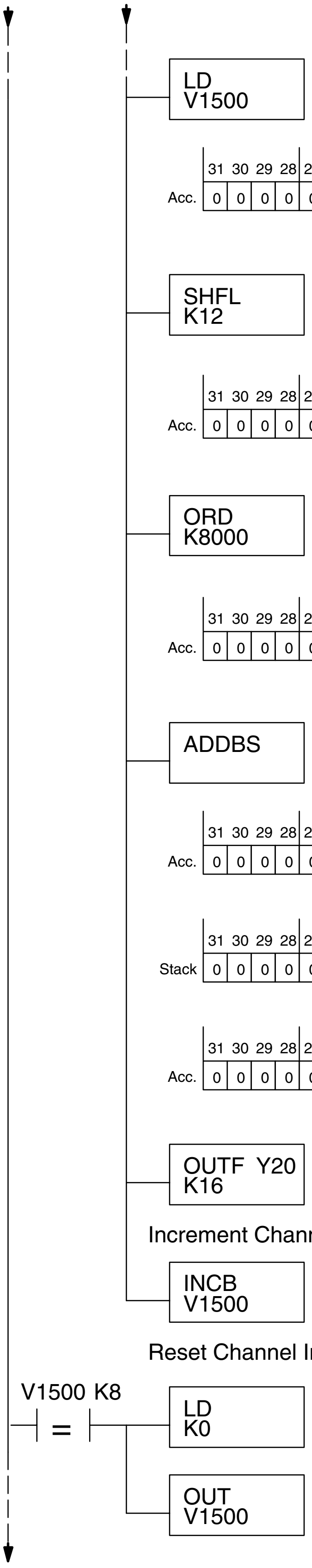
The value in V3001 is 2345, which is slightly over half scale.

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.

BCD Value converted to binary now in accumulator.

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

Example 3 Continued



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

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

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 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	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

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	1	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
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	1	0	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

Send the lower 16 bits stored in the accumulator to the analog module. The lowest 12 bits contain the analog data. Bits 12, 13, and 14 are the channel selection bits. Bit 15 is the Output Enable bit.

Increment Channel Index

Increment the channel index value. This allows the logic to cycle through all eight channels.

Reset Channel Index

When channel 8 has been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1).

Sequencing Example 4, DL430

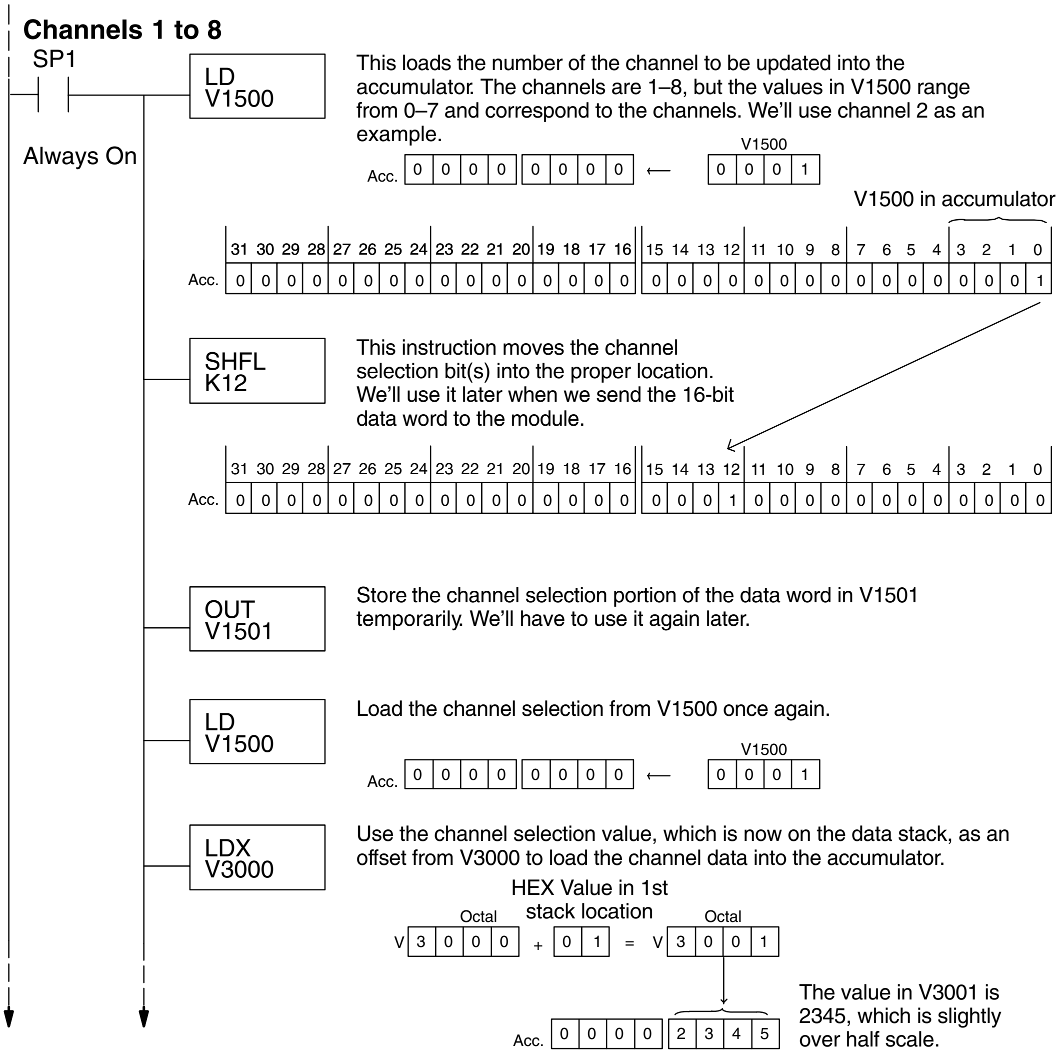
✓ ✓ ✓
430 440 450

The following program example shows how to send digital values to the module when you have more than one channel. This example works for DL430, DL440 or DL450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 4								
Channel Number	1	2	3	4	5	6	7	8
V–Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007

The channel index is stored in V1500. It varies from 0 to 7, pointing to channels as shown: 0 – Ch. 1, 1 – Ch. 2, and 7 – Ch. 8. This example assumes V1500 is initialized to “0” earlier in the program.

The first portion of the program updates one channel during each scan. The program comments show the accumulator status at each step. The last portion of the program increments the channel index number and resets it after eight scans.


Channels 1 to 8
SP1
Always On
LD V1500
SHFL K12
OUT V1501
LD V1500
LDX V3000
Acc.
V1500
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
HEX Value in 1st stack location
Octal
V 3 0 0 0 + 0 1 = V 3 0 0 1
Acc. 0 0 0 0 2 3 4 5

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

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

Load the channel selection from V1500 once again.

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.

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

Example 4 Continued

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.

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 send 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
K8000

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 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	1	0	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.

Increment Channel Index

INCB
V1500

Increment the channel index value. This allows the logic to cycle through all eight channels.

Reset Channel Index

V1500K8
⊣ = ⊢

LD
K0

When channel 8 has been updated, then reset the channel index memory location to 0 (remember, 0 represents channel 1).

OUT
V1500

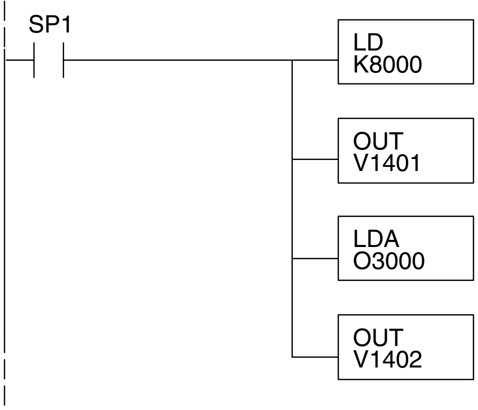
Updating all Channels in a Single Scan, DL440/450

☒	✓	✓
430	440	450

By using the Immediate instructions found in the DL440 and DL450 CPUs, you can easily update all eight channels in a single scan. Before choosing this method, remember it slows CPU scan time (approximately 12 mS). 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 V3000 to V3007 for channels 1 to 7 respectively.

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

Initialize the Immediate Analog Output Pointers



The LD instruction loads the data into the accumulator. The constant K8000 represents channel 1 selected, Output Enable bit is on.

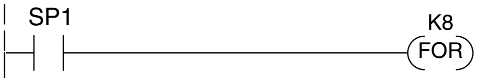
We store the channel index pointer in V1401. The program increments this in each pass, resetting it after eight scans.

The Load Address instruction takes the octal 3000 and converts it to hex, then puts it in the accumulator. V3000 is the location which contains the data for channel 1.

V1402 contains the pointer for channel 1 to 8 data.

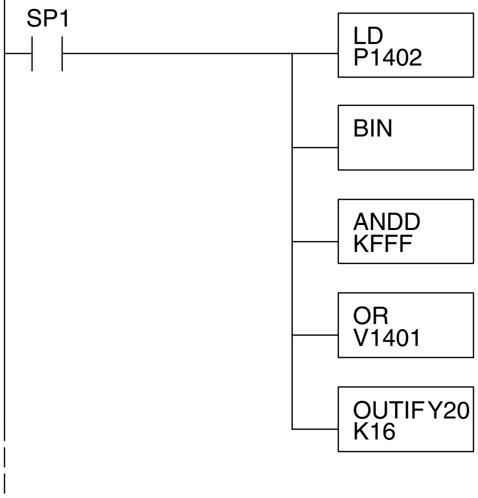
The following FOR–NEXT loop updates all eight channels in a single scan.

FOR–NEXT Loop



The following FOR–NEXT loop updates channels 1 through 8. The valid range for the constant K is 1 to 8 for the module. For example, a value of 1 will update channel 1.

Update Channels



Load the analog output value into the accumulator. V1300 = Ch. 1, V3001 = Ch. 2, ... V3007 = Ch. 8. The number at V1402 is a pointer to the address of the value.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y36, Y35, and Y34 with the data in the accumulator.

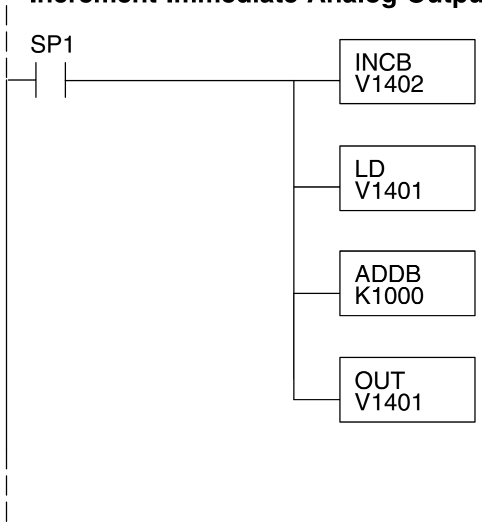
Immediately write the lower 16 bits of the accumulator to the module's lower data word. This updates channels 1 through 8 during the FOR–NEXT loop.

Updating all Channels in a Single Scan, Continued

☒ ✓ ✓
430 440 450

Now we increment the Immediate Analog Output Pointers for the channel, before the next pass through the FOR–NEXT loop.

Increment Immediate Analog Output Pointers



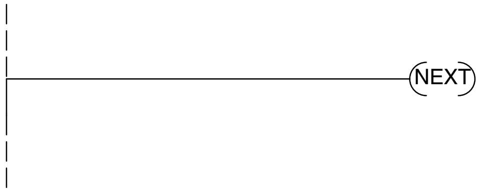
Increment the analog output data pointer.

Load the curent channel index count into the accumulator.

Add 1000 hex to the channel index value, since the channel select field is in the most significant four bits.

Save the incremented channel index value for the next pass through the FOR–NEXT loop.

Here is the end of the FOR–NEXT loop.

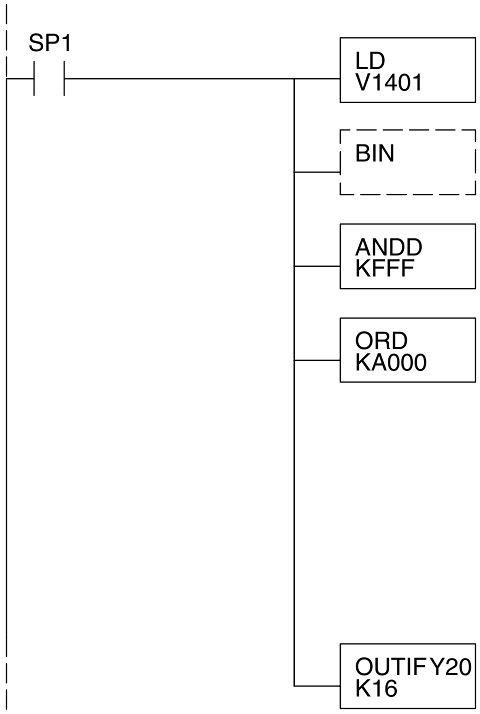


Execute the loop above the number of times specified in the FOR instruction.

Updating a Single Channel During a Scan, DL440/450

☒ ✓ ✓
430 440 450

You can also update just a single channel during a ladder logic scan by using the Immediate instructions found in the DL440 and DL450 CPUs. By removing the FOR–NEXT rungs and a couple of other rungs, we create the example below. This example assumes the data is already loaded in V1401.



Load the analog output value for the channel from V1401 into the accumulator.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits (channel index) with the desired channel with the data in the accumulator. We chose channel 3 here.

Index	Channel
8000	Ch. 1
9000	Ch. 2
A000	Ch. 3
B000	Ch. 4
C000	Ch. 5
D000	Ch. 6
E000	Ch. 7
F000	Ch. 8

Immediately write the lower 16 bits of the accumulator to the module's lower data word. This updates the selected channel.

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.

Range	If you know the digital value ...	If you know the analog signal level ...
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

For example, if you need an 9mA signal level, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

$$D = \frac{4095}{16}(A-4)$$

$$D = \frac{4095}{16}(9\text{mA}-4)$$

$$D = (255.94)\ (5)$$

$$D = 1280$$

16

F4–16DA–1 16-Channel Analog Current Output

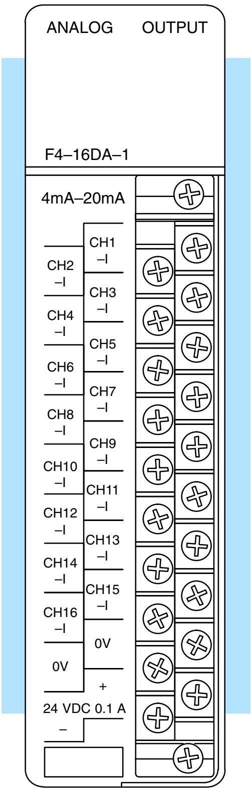
In This Chapter. . . .

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

Module Specifications

The F4–16DA–1 Analog Current Output module provides several features and benefits.

- It provides sixteen channels of 4–20 mA single ended current outputs.
- 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.
- From two to sixteen analog outputs may be updated in one CPU scan (DL440 and DL450 CPUs only).
- No jumper settings are required.


ANALOG OUTPUT
F4–16DA–1
4mA–20mA
CH1 –I
CH2 –I
CH3 –I
CH4 –I
CH5 –I
CH6 –I
CH7 –I
CH8 –I
CH9 –I
CH10 –I
CH11 –I
CH12 –I
CH13 –I
CH14 –I
CH15 –I
CH16 –I
0V
0V
+
24 VDC 0.1 A
–

Analog Output Configuration Requirements

The F4–16DA–1 Analog Output module requires 32 discrete output 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 output points.
- For remote I/O systems, the available power budget and remote I/O points.

The following tables provide the specifications for the F4–16DA–1 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

Number of Channels	16, single ended (one common)
Output Range	4–20 mA current
Resolution	12 bit (1 in 4095)
Output Type	Outputs sink 4–20 mA from external supply
Peak Output Voltage	40 VDC (no transient voltage suppression)
External Load Resistance	0–480Ω @ 18V, 220–740Ω @ 24V, 1550–1760Ω @ 48V
Maximum Loop Supply	48 VDC (with load resistance in proper range)
Crosstalk	–70 dB, ±1 count maximum
Linearity Error (end-to-end) and Relative Accuracy	±1 count maximum (20mA at 25°C)
Full Scale Calibration Error (offset error included)	±8 counts maximum (20mA at 25°C)
Offset Calibration Error	±3 counts maximum (4mA at 25°C)
Maximum Inaccuracy	±0.2% @ 25°C (77°F) ±0.4% @ 0 to 60°C (32 to 140°F)
Conversion Time	400 μs maximum, for full scale change 4.5 to 9 mS for digital output to analog out

General Module Specifications

Digital Output Points Required	32 point (Y) outputs, 2 sets each of 12 bits binary data, 3 bits channel select, 1 bit output enable
Power Budget Requirement	90 mA @ 5 VDC (supplied by base)
External Power Supply	21.6 – 26.4 VDC, 100 mA, class 2 (add 20 mA for each current loop used)
Accuracy vs. Temperature	±57 ppm / °C full scale calibration range (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

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.
- 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 F4–16DA–1 requires at least one field-side supply. You may use the same or separate power sources for the module supply and loop supply. The module requires 21.6 to 26.4 VDC, Class 2, at 100mA current. The sixteen current loops require 18 to 48 VDC, at 20 mA each.

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. You may use one of these instead of a separate supply if there is only one F4–16DA–1 module with 15 channels or fewer in use. The current required is 100 mA (module), plus 320 mA (sixteen current loops) for a total of 420 mA.

In some situations it's desirable to power the loops separately due to power budget or due to their remote location from the PLC. This will work, as long as the loop supply meets the voltage and current requirements, and its minus (–) side and the module supply's minus (–) side connect together.

Load Requirements

Each channel in use must have a load impedance within a range determined by the loop supply voltage: 0–480Ω @ 18V, 220–740Ω @ 24V,1550-1760Ω @ 48V. Unused channels must be left disconnected.

WARNING: If you are using the 24 VDC base power supply, 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.

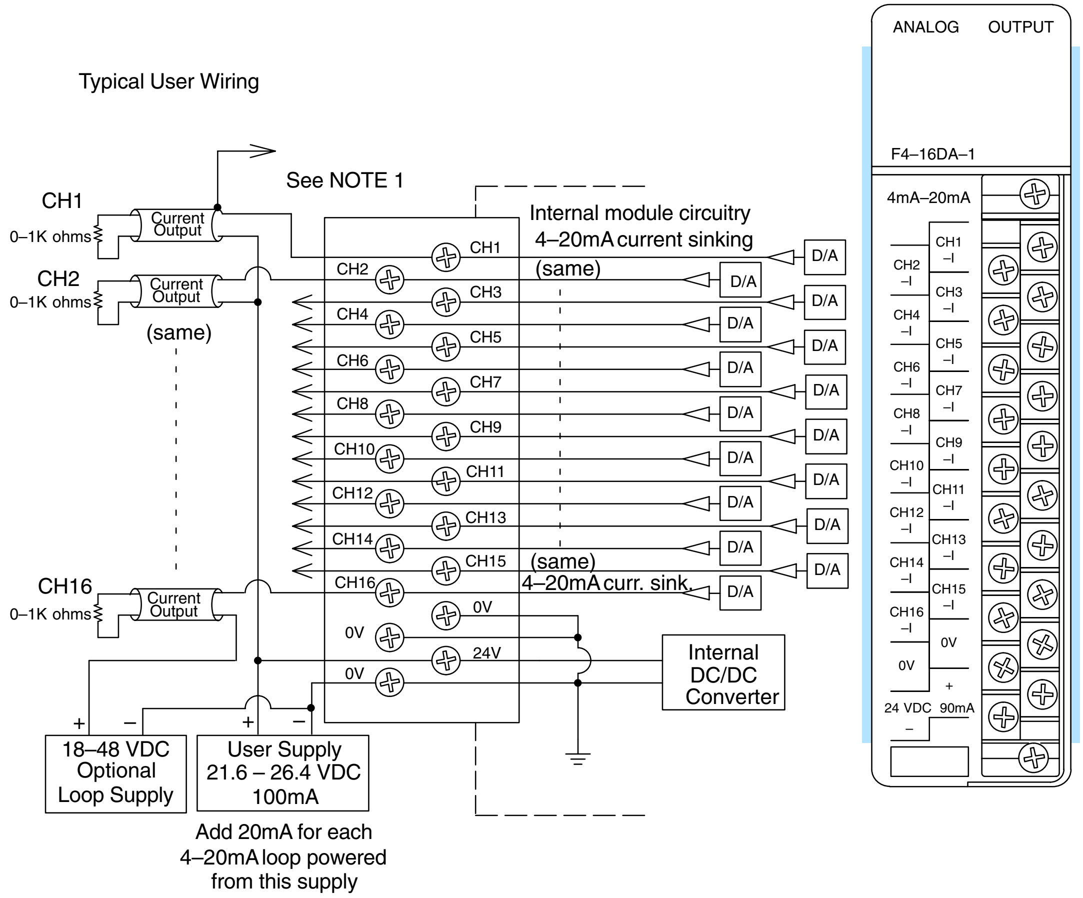
Removable Connector

The F4–16DA–1 module has a removable connector to make wiring easier. Simply loosen the retaining screws and gently pull the connector from the module. The diagram shows separate module and loop supplies for channel 16. If you desire to use only one field-side supply, just combine the supplies' positive terminals into one node, and remove the loop supply.

Wiring Diagram

NOTE 1: Shields should be connected to the 0V of the User Power Supply at the module terminal block.

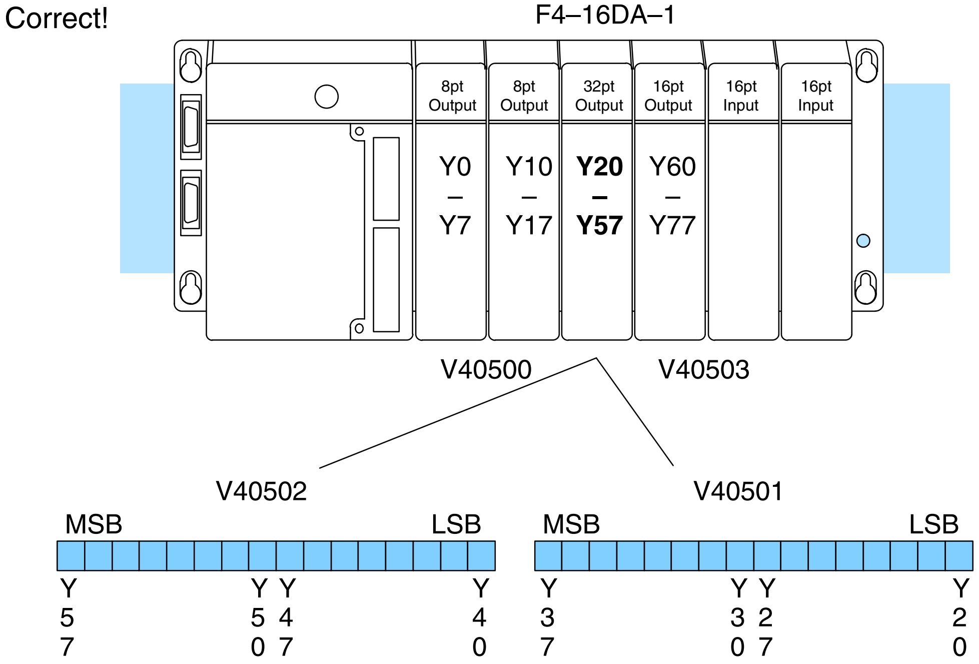
NOTE 2: Unused current outputs should remain open (no connections).

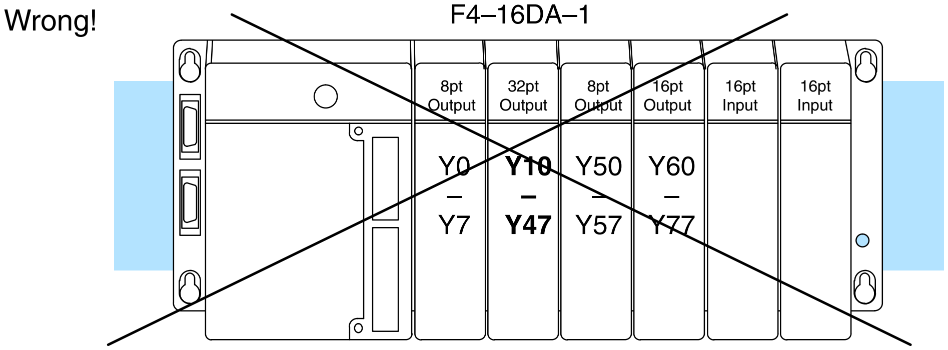

Typical User Wiring
See NOTE 1
CH1
0–1K ohms
Current Output
CH2
0–1K ohms
Current Output
(same)
CH16
0–1K ohms
Current Output
Internal module circuitry
4–20mA current sinking
(same)
(same)
4–20mA curr. sink.
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
0V
0V
24V
0V
D/A
Internal DC/DC Converter
+
–
18–48 VDC Optional Loop Supply
+
–
User Supply 21.6 – 26.4 VDC 100mA
Add 20mA for each 4–20mA loop powered from this supply
ANALOG OUTPUT
F4–16DA–1
4mA–20mA
CH1 –I
CH2 –I
CH3 –I
CH4 –I
CH5 –I
CH6 –I
CH7 –I
CH8 –I
CH9 –I
CH10 –I
CH11 –I
CH12 –I
CH13 –I
CH14 –I
CH15 –I
CH16 –I
0V
0V
+
24 VDC 90mA
–

Module Operation

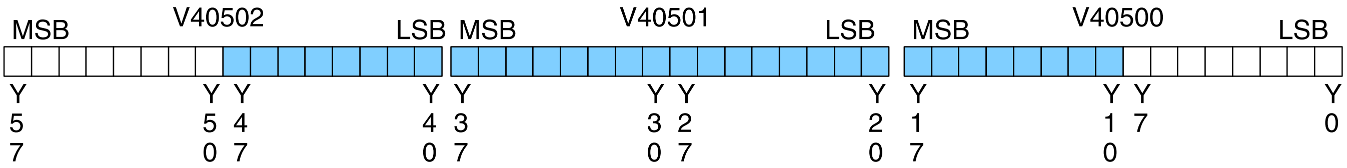
DL430 Special Requirements

Even though the module can be placed in any slot, it is important to examine the configuration if you're using a DL430 CPU. As you'll 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 can't access the data.


Correct!
F4–16DA–1
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
MSB
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20


Wrong!
F4–16DA–1
8pt Output
32pt Output
8pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y47
Y50 – Y57
Y60 – Y77

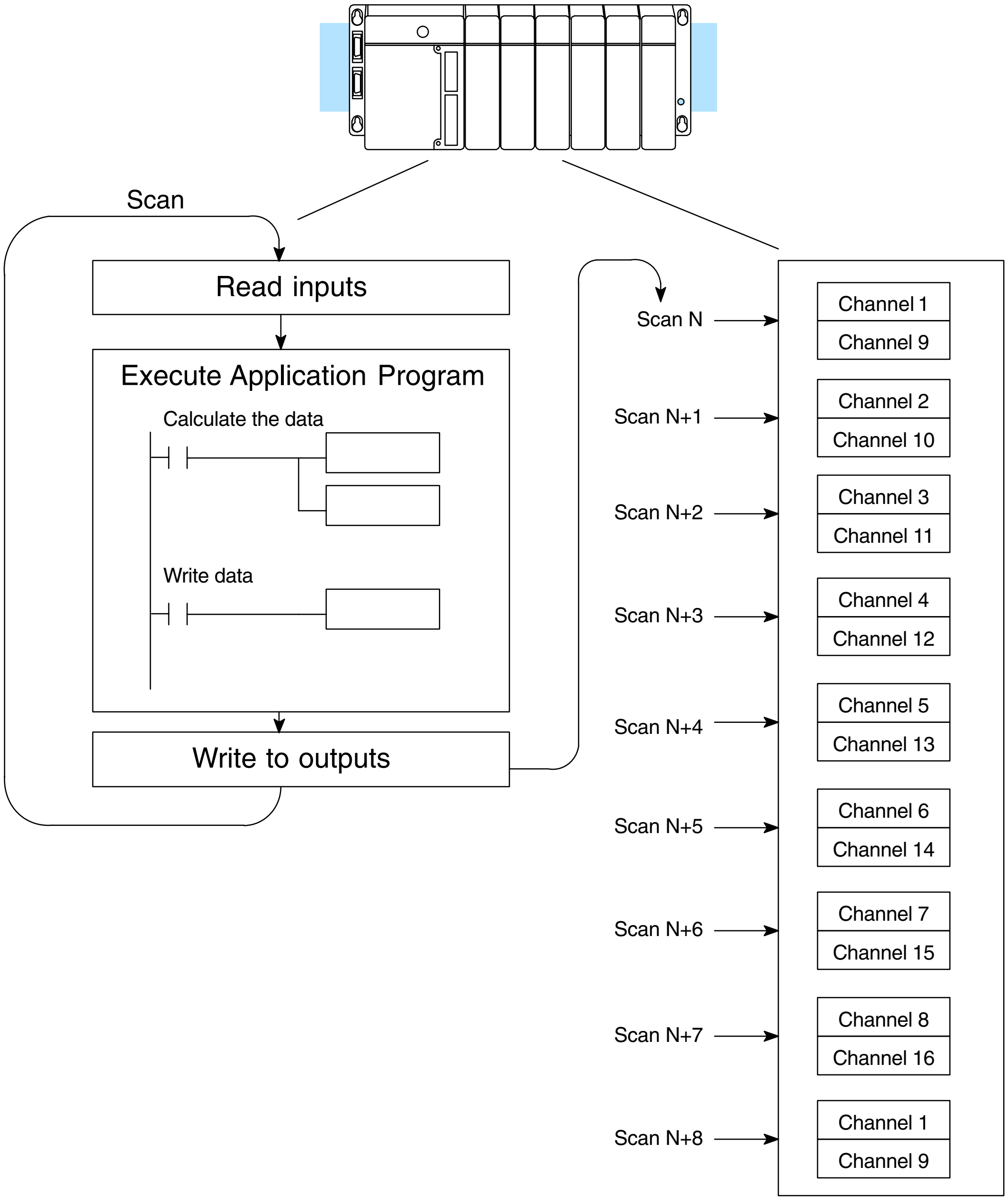
Data is split over three locations, so instructions cannot access data from a DL430


MSB
V40502
LSB
MSB
V40501
LSB
MSB
V40500
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20
Y17
Y10
Y7
Y0

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 F4–16DA–1 module allows you to update the channels in any order. Your control program determines which channels get updated on any given scan. The channels are organized as two banks of outputs, eight channels each. A typical ladder program will update one channel in each bank, or two channels per CPU scan. So, all sixteen channels can be updated every eight scans. With a DL440 or DL450 CPU, you can use Immediate instructions to update all sixteen channels in the same scan (we'll show you how to do this later).



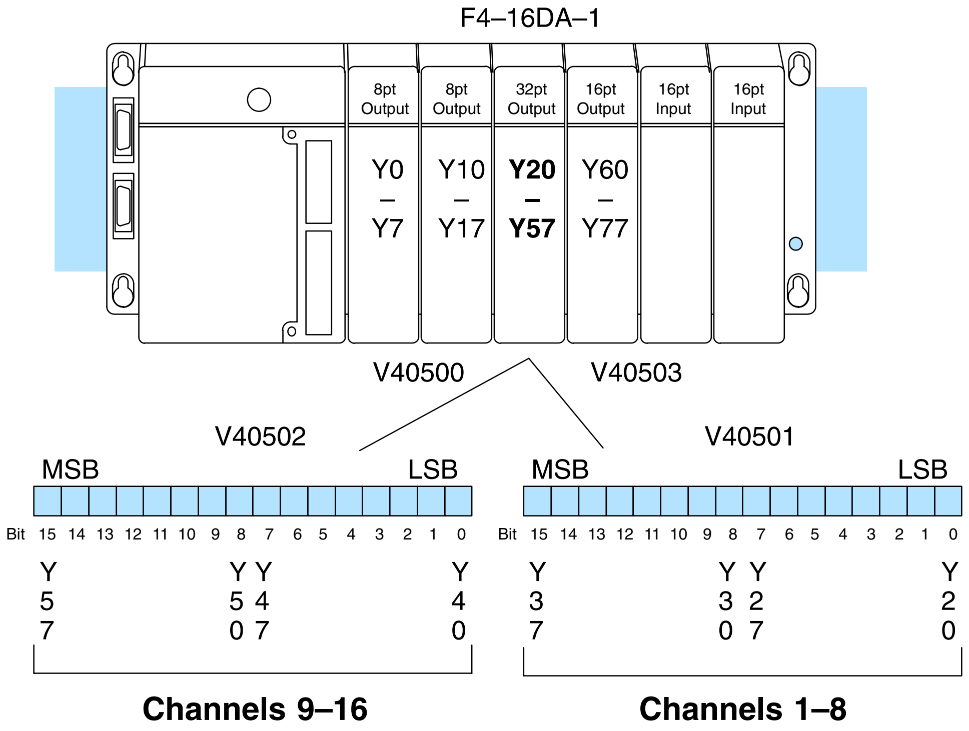
Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Scan N+8
Channel 1
Channel 9
Channel 2
Channel 10
Channel 3
Channel 11
Channel 4
Channel 12
Channel 5
Channel 13
Channel 6
Channel 14
Channel 7
Channel 15
Channel 8
Channel 16
Channel 1
Channel 9

Output Bit Assignments

You may recall the F4–16DA–1 module requires 32 discrete output points from the CPU. These points provide:

- The digital representation of two analog signals per scan.
- Identification of the two channels that are to receive the data.
- Output enable control for all channels.

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

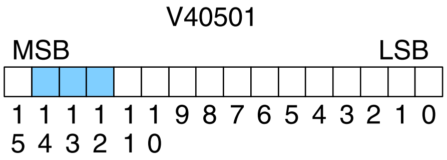

F4–16DA–1
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50
Y47
Y40
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y37
Y30
Y27
Y20
Channels 9–16
Channels 1–8

Within these V-memory locations the individual bits represent specific information about the channels selected and the analog signals.

Channel Select Bits

Output bits 12, 13, and 14 of both data words are the channel select outputs. They are binary encoded to select the channel that will be updated with the data. The bits of the lower data word select channels 1 to 8 as follows.

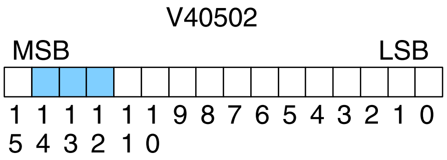
Bit 14	Bit 13	Bit 12	Channel
Off	Off	Off	1
Off	Off	On	2
Off	On	Off	3
Off	On	On	4
On	Off	Off	5
On	Off	On	6
On	On	Off	7
On	On	On	8


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

🟦 – Channel Select Bits, channels 1 through 8

The bits of the upper data word select channels 9 to 16 as follows.

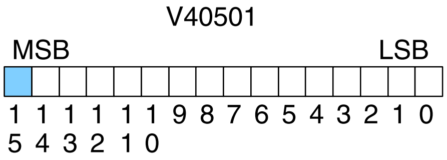
Bit 14	Bit 13	Bit 12	Channel
Off	Off	Off	9
Off	Off	On	10
Off	On	Off	11
Off	On	On	12
On	Off	Off	13
On	Off	On	14
On	On	Off	15
On	On	On	16


V40502
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

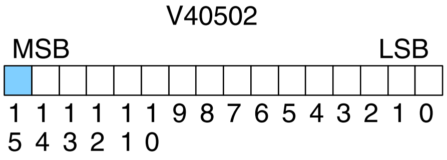
🟦 – Channel Select Bits, channels 9 through 16

Output Enable Bits

Output bit 15 is the Output Enable control bit in each data word for all eight channels of each bank. When either bit is off, the corresponding group of eight channel output currents decrease to their lowest level, which is 4 mA for connected loads. Disabling the outputs also clears all eight output data registers for each bank. To resume analog output levels, first the appropriate Output Enable control bit must turn on. Then, the CPU must write new data to each channel to restore the output current for that channel.


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

🟦 – Output Enable Bit, channels 1 through 8


V40502
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

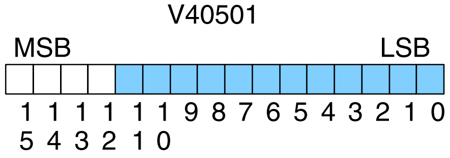
🟦 – Output Enable Bit, channels 9 through 16

OFF = Disable (and clear)
ON = Enable

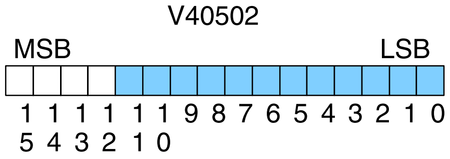
Analog Data Bits

The first twelve bits of both V-memory locations 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



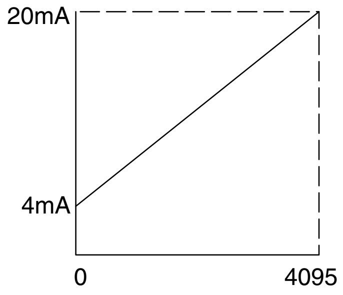
🟦 – data bits,
channels 1 through 8



🟦 – data bits,
channels 9 through 16

Module Resolution

Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0 – 4095 (2^{12}). For the 4 to 20 mA scale, sending a 0 produces a 4 mA signal, and 4095 gives a 20 mA signal. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The graph to the right shows the linear relationship between the data value and output signal level.



Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal change that occurs when the digital value is increased by 1 LSB.

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

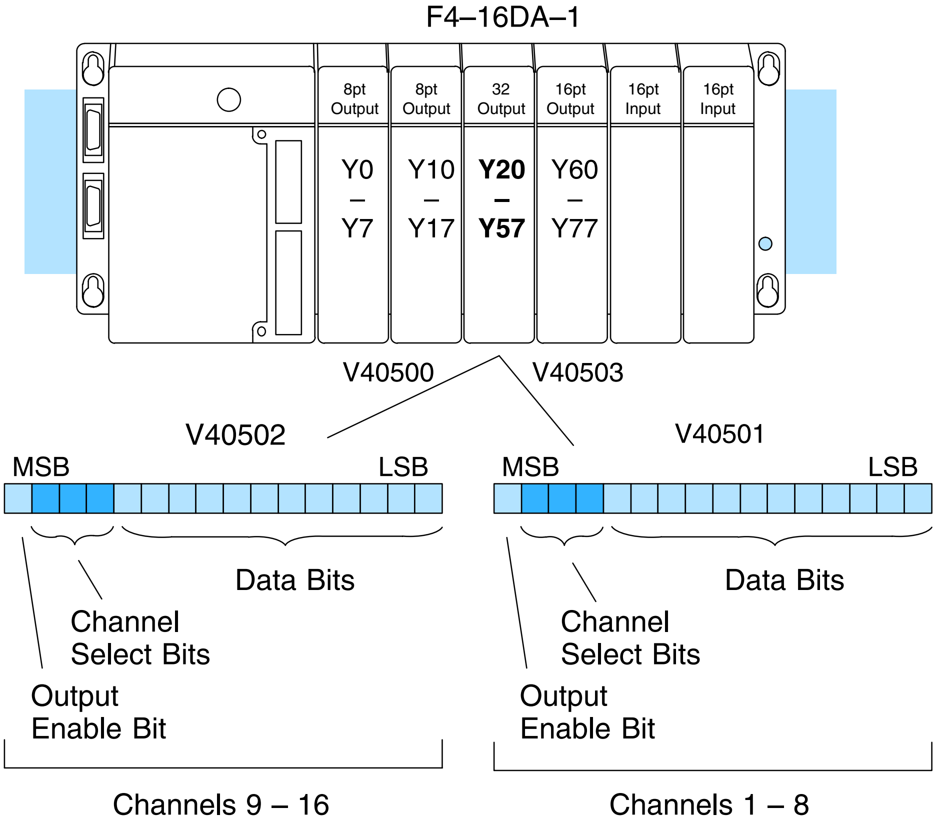
H = high limit of the signal range
L = low limit of the signal range

Signal Range	Span (H – L)	Divide By	Smallest Change
4 to 20mA	16mA	4095	3.91 µA

Writing the Control Program

Update Any Channel

As mentioned earlier, you can update two channels 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 select outputs to determine which channels get updated (more on this later).


F4–16DA–1
8pt Output
8pt Output
32 Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
MSB
LSB
Data Bits
Channel Select Bits
Output Enable Bit
Data Bits
Channel Select Bits
Output Enable Bit
Channels 9 – 16
Channels 1 – 8

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.

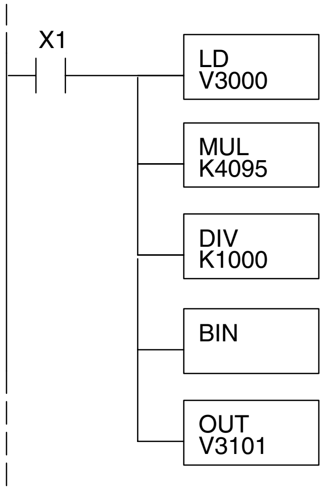
$$A = 10U \frac{4095}{10(H - L)}$$

$$A = 494 \frac{4095}{1000 - 0}$$

$$A = 2023$$

Here is 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 all sixteen 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary (you must omit this step if you have converted the data elsewhere).

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

V-Memory Registers

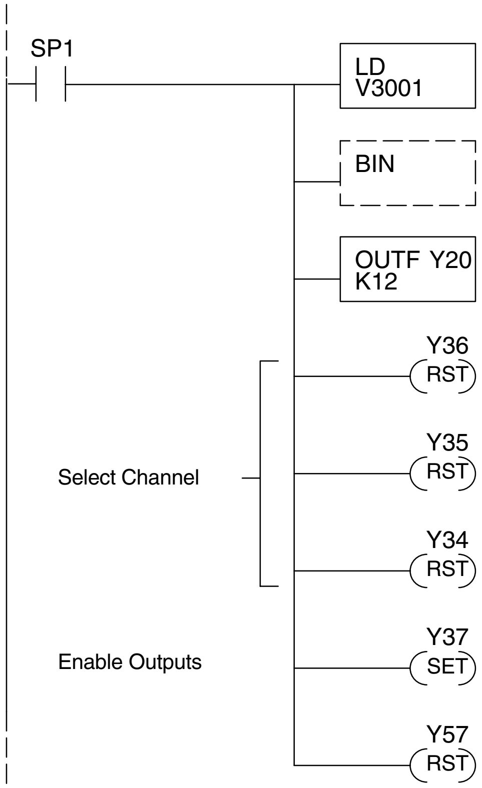
The ladder program examples that follow occasionally use certain V-memory register addresses in the CPU that correspond to 16-bit Y output modules. Use the table below to find the V-memory address for the particular slot of your analog module. See Appendix A for additional addresses for DL450 CPUs.

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

Sending Data to One Channel, DL440/450

☒ ☑ ☑
430 440 450

The following two programs show you how to update a single channel. Notice the DL440/DL450 and DL430 CPUs use slightly different programs, although the DL430 example will also work with 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. This example assumes you already have the data loaded in V3001.



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.

Turn Y36, Y35, and Y34 off to update Channel 1.

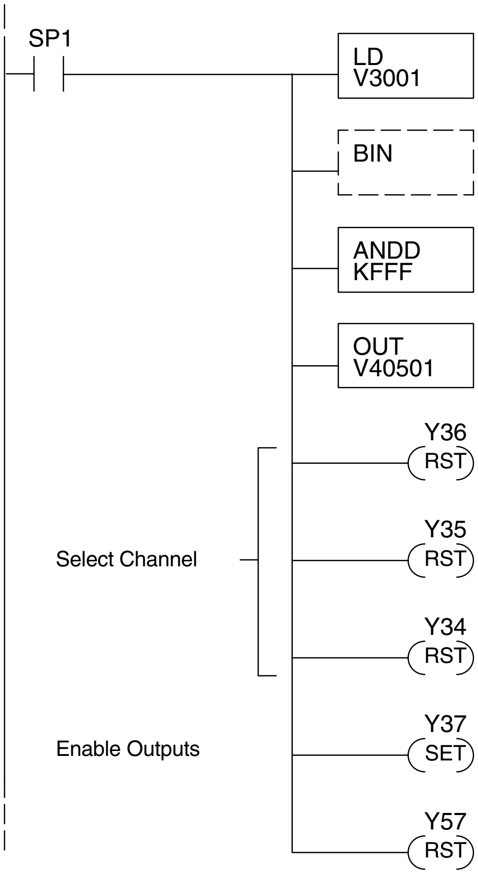
Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels of the first bank.

Turn off Y57 to disable the second bank of channels, 9 through 16.

Sending Data to One Channel, DL430

✓	✓	✓
430	440	450



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 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.

Turn Y36, Y35, and Y34 off to update Channel 1.

Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels of the first bank.

Turn off Y57 to disable the second bank of channels, 9 through 16.

Sequencing the Channel Updates

The next four example programs show you how to send digital values to the module when you have more than one channel. These examples automatically update all sixteen channels over eight scans. Channel updates occur in pairs, such as channels 1 and 9, channels 2 and 10, and so on. This method uses the same binary code for the channel select bits of both groups of outputs, simplifying the programs. You may recall we stated earlier you could update all sixteen channels in the same scan. That's true, you can do that if you have a DL440/450 CPU; however, this can increase the scan time and you may not always need to update all sixteen channels on every scan. We show you how to do this later in this chapter.

The first two sequencing examples are fairly simple and will work in most situations. We recommend these for new users. They use control relays C1 through C8 as index numbers corresponding to the pair of channels updated on any particular scan. At the end of each scan, only one control relay C1 through C8 is on. On each subsequent scan, the next control relay energizes. The channel sequencing automatically begins with channels 1 and 9 on the first scan, or after any disruption in the logic.

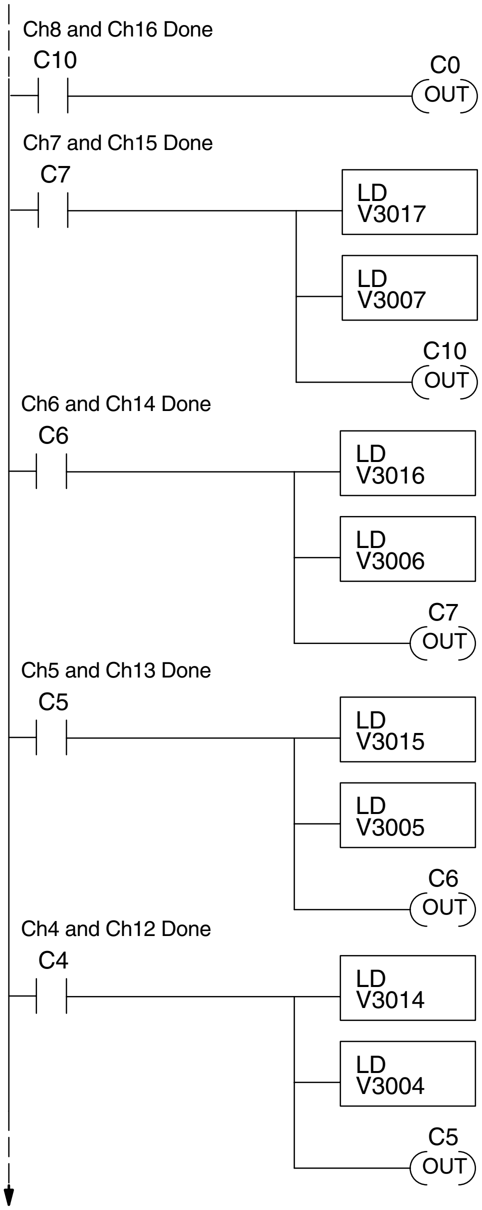
The last two examples, 3 and 4, are slightly more complex. However, they do not depend on the use of control relays to provide channel sequencing. Instead, they use function boxes to increment a channel pointer value in V-memory. Then, other instructions perform bit manipulations to position the channel select bits properly in the output words to the module.

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 have already loaded the data according to the following table. It is important to use the rungs in the order shown for the program to work.

V-Memory Locations for Output Data in Examples 1 and 2								
Channel Number	1	2	3	4	5	6	7	8
V-Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007
Channel Number	9	10	11	12	13	14	15	16
V-Memory Storage	3010	3011	3012	3013	3014	3015	3016	3017



When channels 8 and 16 are updated, C0 restarts the update sequence.

When channels 7 and 15 have been updated, this rung loads the data for channel 16 into the accumulator. The second LD instruction places the data for channel 8 in the accumulator and pushes channel 16’s data onto the stack.

Turning on C10 triggers the channel update (see the channel select rungs).

When channels 6 and 14 have been updated, this rung loads the data for channel 15 into the accumulator. The second LD instruction places the data for channel 7 in the accumulator and pushes channel 15’s data onto the stack.

Turning on C7 triggers the channel update (see the channel select rungs).

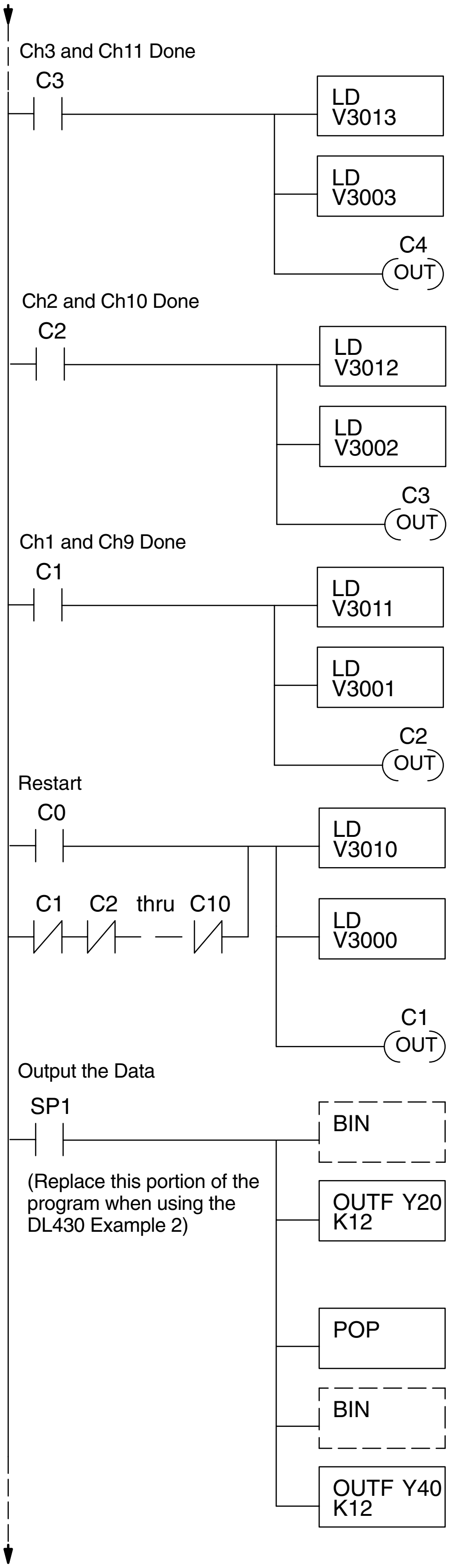
When channels 5 and 13 have been updated, this rung loads the data for channel 14 into the accumulator. The second LD instruction places the data for channel 6 in the accumulator and pushes channel 14’s data onto the stack.

Turning on C6 triggers the channel update (see the channel select rungs).

When channels 4 and 12 have been updated, this rung loads the data for channel 13 into the accumulator. The second LD instruction places the data for channel 5 in the accumulator and pushes channel 13’s data onto the stack.

Turning on C5 triggers the channel update (see the channel select rungs).

Example 1 Continued


Ch3 and Ch11 Done
C3
LD V3013
LD V3003
C4
OUT
Ch2 and Ch10 Done
C2
LD V3012
LD V3002
C3
OUT
Ch1 and Ch9 Done
C1
LD V3011
LD V3001
C2
OUT
Restart
C0
C1 C2 thru C10
LD V3010
LD V3000
C1
OUT
Output the Data
SP1
(Replace this portion of the program when using the DL430 Example 2)
BIN
OUTF Y20 K12
POP
BIN
OUTF Y40 K12

When channels 3 and 11 have been updated, this rung loads the data for channel 12 into the accumulator. The second LD instruction places the data for channel 4 in the accumulator and pushes channel 12's data onto the stack.

Turning on C4 triggers the channel update (see the channel select rungs).

When channels 2 and 10 have been updated, this rung loads the data for channel 11 into the accumulator. The second LD instruction places the data for channel 3 in the accumulator and pushes channel 11's data onto the stack.

Turning on C3 triggers the channel update (see the channel select rungs).

When channels 1 and 9 have been updated, this rung loads the data for channel 10 into the accumulator. The second LD instruction places the data for channel 2 in the accumulator and pushes channel 10's data onto the stack.

Turning on C2 triggers the channel update (see the channel select rungs).

This rung loads the data for channel 9 into the accumulator. The second LD instruction places the data for channel 1 in the accumulator and pushes channel 9's data onto the stack. C0 restarts the sequence after channels 8 and 16 are done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C1 through C10 are off (all eight contacts are not shown here due to space constraints). In this case, we also start the sequence with channels 1 and 9.

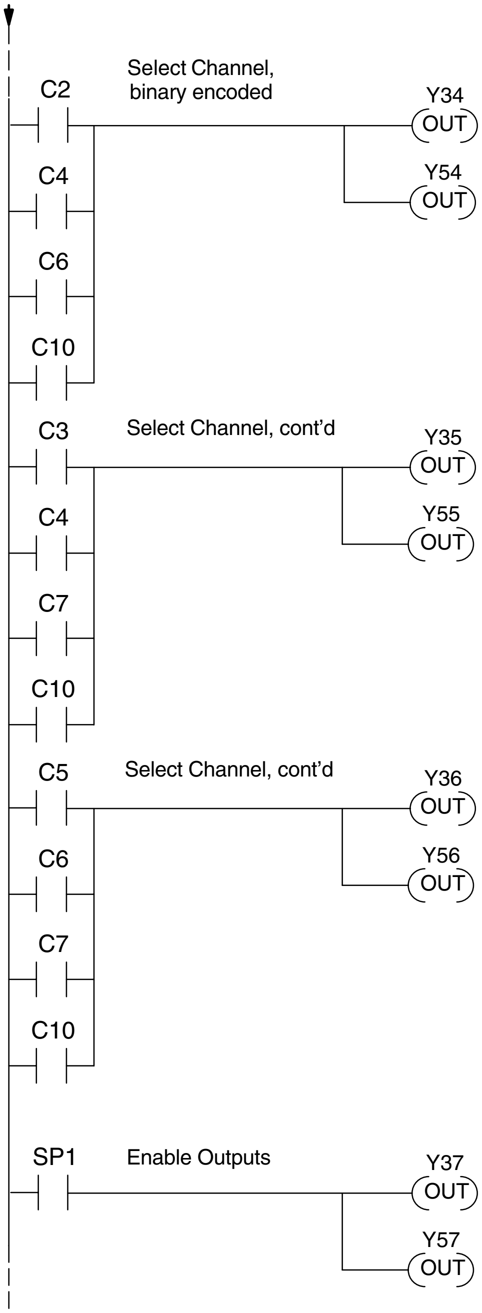
Turning on C1 triggers the channel update (see the channel select rungs).

This rung converts the accumulator data for channels 1 through 8 (one per scan) 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 for the first bank of eight channels. Our example starts with Y20, but the actual value depends on the location of the module in your application.

We move the data for channels 9 through 16 (one per scan) from the stack to the accumulator.

This rung converts the accumulator data for channels 9 through 16 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 for the second bank of eight channels. In our example, it is Y40, but the actual value depends on the location of the module in your application.

Example 1 Continued


C2
C4
C6
C10
Select Channel, binary encoded
Y34
OUT
Y54
OUT
C3
C4
C7
C10
Select Channel, cont'd
Y35
OUT
Y55
OUT
C5
C6
C7
C10
Select Channel, cont'd
Y36
OUT
Y56
OUT
SP1
Enable Outputs
Y37
OUT
Y57
OUT

Set Y36, Y35, and Y34 to the binary code which selects the output channel 1 through 8, based on the control relay status. We use the same binary code to select channels 9 through 16.

CR(on)	Y36,Y56	Y35,Y55	Y34,Y54	Channels
C1	Off	Off	Off	Ch. 1 and 9
C2	Off	Off	On	Ch. 2 and 10
C3	Off	On	Off	Ch. 3 and 11
C4	Off	On	On	Ch. 4 and 12
C5	On	Off	Off	Ch. 5 and 13
C6	On	Off	On	Ch. 6 and 14
C7	On	On	Off	Ch. 7 and 15
C10	On	On	On	Ch. 8 and 16

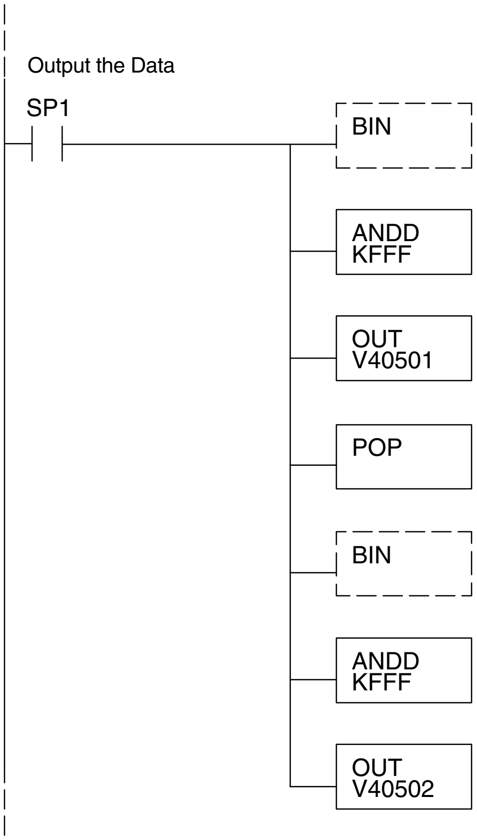
Enable channels 1 through 8. SP1 is always on.

Enable channels 9 through 16.

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 or the output enable bits are not accidentally changed by the data in the accumulator. Replace the “Output the Data” rung in the middle of Example 1 with the new rung below. Be sure to retain the original order of the rungs shown in Example 1 for the program to work. This will also work with DL440 and DL450 CPUs.



This rung converts the accumulator data for channels 1 through 8 (one per scan) 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 for the first bank of 8 channels, but the actual value depends on the location of the module in your application.

We move the data for channels 9 through 16 (one per scan) from the stack to the accumulator.

This rung converts the accumulator data for channels 9 through 16 (one per scan) 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 uses V40502 for the second bank of 8 channels, but the actual value depends on the location of the module in your application.

Sequencing Example 3, DL440/450

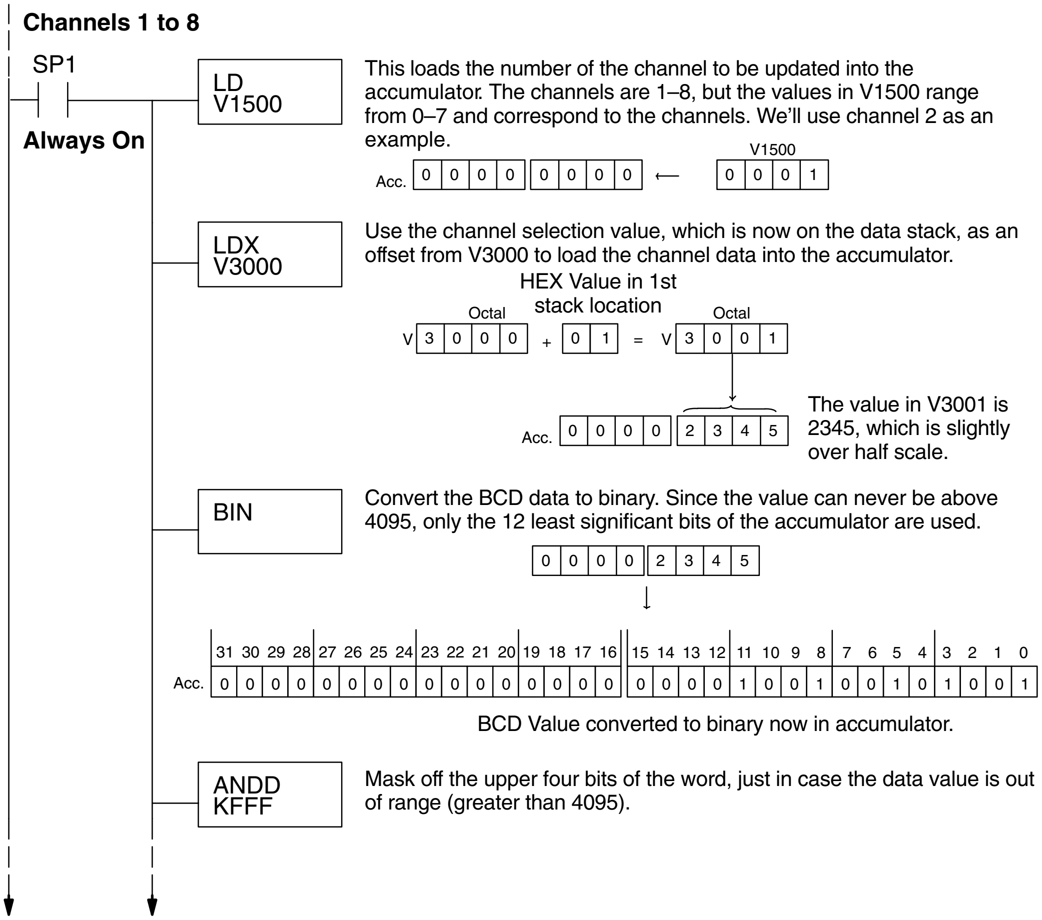
☒ ✓ ✓
430 440 450

The following program example shows how to send digital values to the module when you have more than one channel. This example works only for DL440 and DL450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 3								
Channel Number	1	2	3	4	5	6	7	8
V-Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007
Channel Number	9	10	11	12	13	14	15	16
V-Memory Storage	3010	3011	3012	3013	3014	3015	3016	3017

The channel index is stored in V1500. It varies from 0 to 7, pointing to pairs of channels as shown: 0 = ch. 1 and 9, 1 = ch. 2 and 10, ... and 7 = ch. 8 and16. This example assumes V1500 is initialized to “0” earlier in the program.

The first portion of the program updates one channel from the first bank of eight channels (1 to 8) during each scan. The program comments for this portion also shows the accumulator status at each step. The second part of the program updates one channel from the second bank of eight channels (9 to 16) during each scan (accumulator status not shown). So, there are two channels which are updated on each scan. The last portion of the program increments the channel index number and resets it after eight scans.


Channels 1 to 8
SP1
Always On
LD V1500
LDX V3000
BIN
ANDD KFFF
V1500
Acc.
Octal
HEX Value in 1st stack location
V 3 0 0 0 + 0 1 = V 3 0 0 1
Acc. 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

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

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

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.

BCD Value converted to binary now in accumulator.

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

Example 3 Continued

**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

**ORD
K8000**

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to "1", enabling channels 1 through 8.

	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	1	0	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	1	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
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

↓

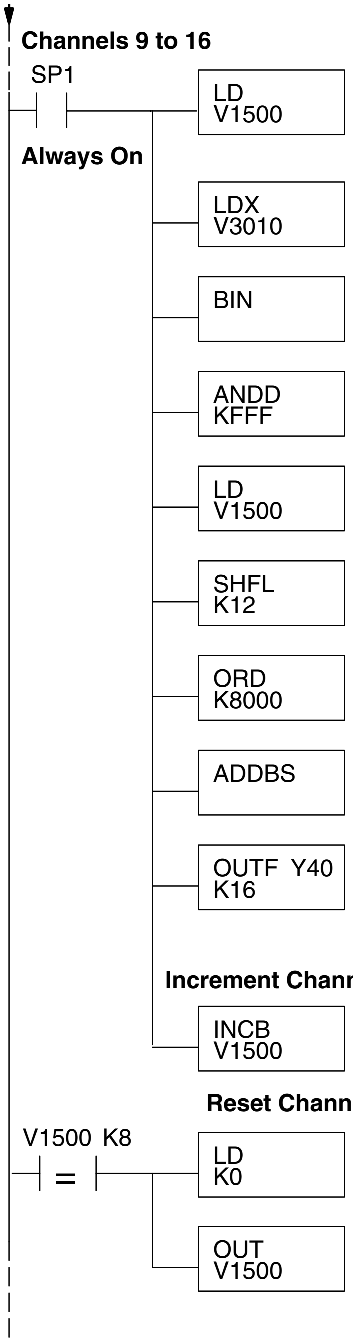
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

**OUTF Y20
K16**

Send the lower 16 bits stored in the accumulator to the analog module.. The lowest 12 bits contain the analog data. Bits 12, 13, and 14 are the channel selection bits. Bit 15 is the Output Enable bit. This example uses Y20, but the actual value depends on the location of the modules in your application.

Example 3 Continued



This loads the number of the channel to be updated into the accumulator. The channels are 9–16, but the values in V1500 range from 0–7 and correspond to the channels. We'll use channel 10 as an example.

Use the channel selection value, which is now on the data stack, as an offset from V3010 to load the channel data into the accumulator.

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.

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

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).

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.

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to "1", enabling channels 9 through 16.

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.

Send the lower 16 bits stored in the accumulator to the analog module. The lowest 12 bits contain the analog data. Bits 12, 13, and 14 are the channel selection bits. Bit 15 is the Output Enable bit. This example uses Y20, but the actual value depends on the location of the modules in your application.

Increment the channel index value. This allows the logic to cycle through all sixteen channels (8 pairs).

When channels 8 and 16 have been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1 and channel 9).

Sequencing Example 4, DL430

✓ ✓ ✓
430 440 450

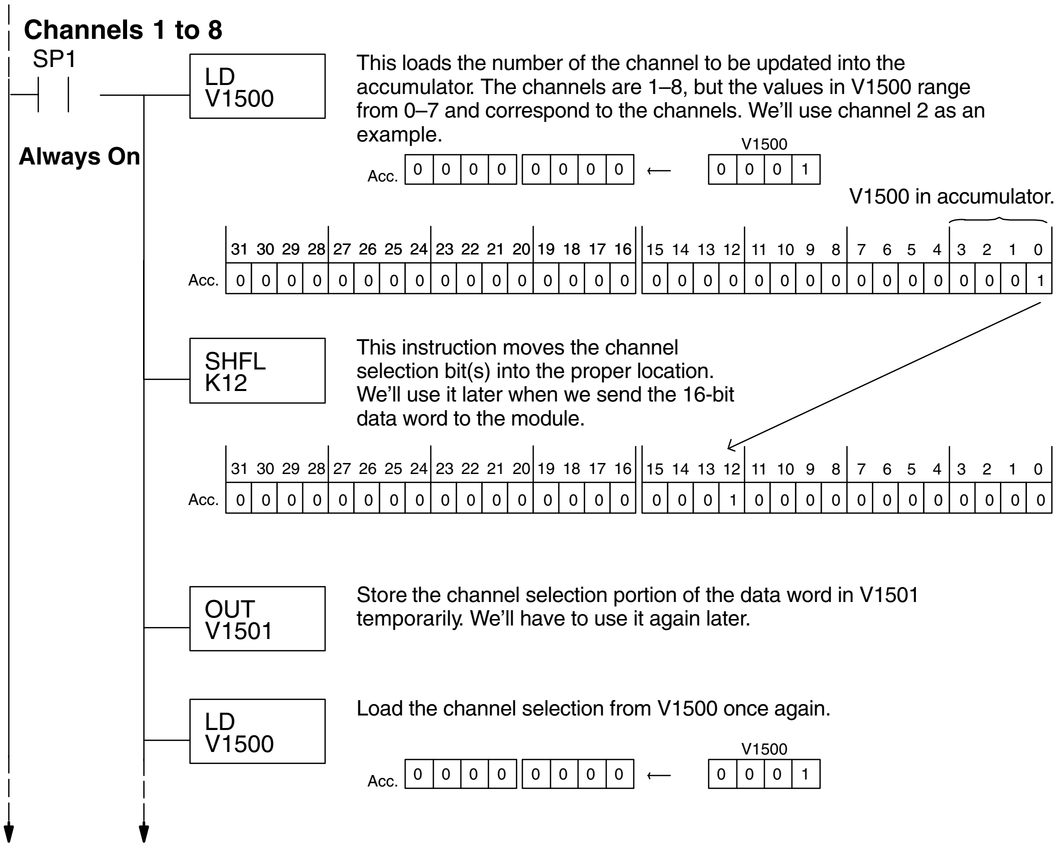
The following program example shows how to send digital values to the module when you have more than one channel. This example also works for DL440 and DL450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 4

Channel Number	1	2	3	4	5	6	7	8
V–Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007
Channel Number	9	10	11	12	13	14	15	16
V–Memory Storage	3010	3011	3012	3013	3014	3015	3016	3017

The channel index is stored in V1500. It varies from 0 to 7, pointing to pairs of channels as shown: 0 = ch. 1 and 9, 1 = ch. 2 and 10, ... and 7 = ch. 8 and16. This example assumes V1500 is initialized to “0” earlier in the program.

The first portion of the program updates one channel from the first bank of eight channels (1 to 8) during each scan. The program comments for this portion also show the accumulator status at each step. The second part of the program updates one channel from the second bank of eight channels (9 to 16) during each scan (accumulator status not shown, for brevity). So, there are two channels which are updated on each scan. The last portion of the program increments the channel index number and resets it after eight scans.


Channels 1 to 8
SP1
Always On
LD
V1500
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
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
OUT
V1501
LD
V1500
Acc. 0 0 0 0 0 0 0 0 ⟵ V1500 0 0 0 1

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

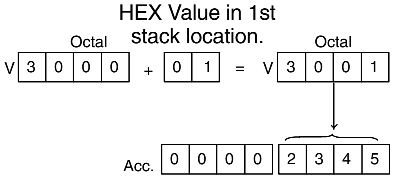
Store the channel selection portion of the data word in V1501 temporarily. We’ll have to use it again later.

Load the channel selection from V1500 once again.

Example 4 Continued

**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.



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.

**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 send 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
K8000**

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to “1”, enabling channels 1 through 8.

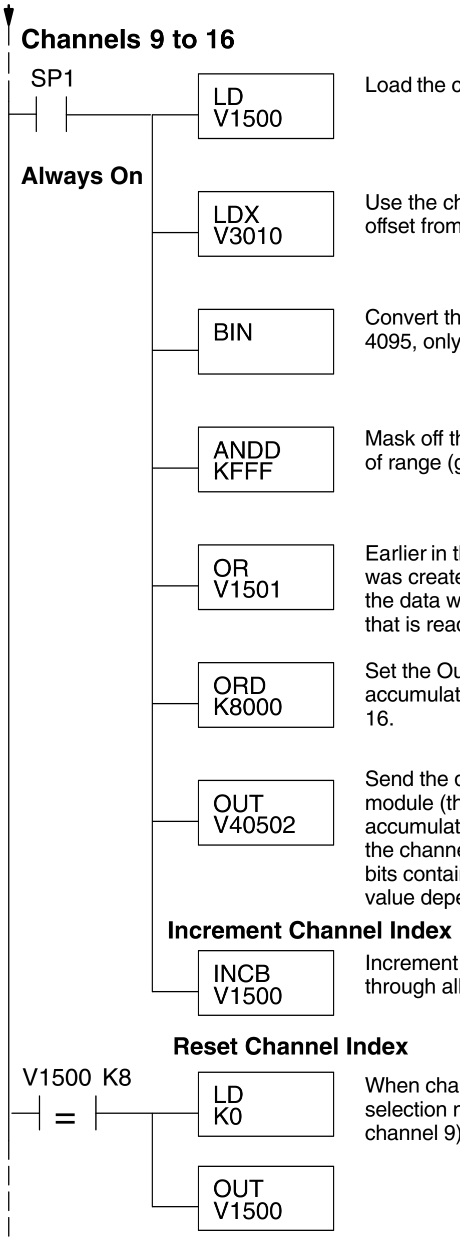
↓

	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	1	0	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. This example uses V40501, but the actual value depends on the location of the module in your application.

Example 4 Continued



Load the channel selection from V1500 once again.

Use the channel selection value, which is now on the data stack, as an offset from V3010 to load the channel data into the accumulator.

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.

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

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 send to the analog module.

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to “1”, enabling channels 9 through 16.

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 and the output enable bit. The remaining 12 bits contain the analog data. This example uses V40502, but the actual value depends on the location of the module in your application.

Increment the channel index value. This allows the logic to cycle through all sixteen channels (8 pairs).

When channels 8 and 16 have been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1 and channel 9).

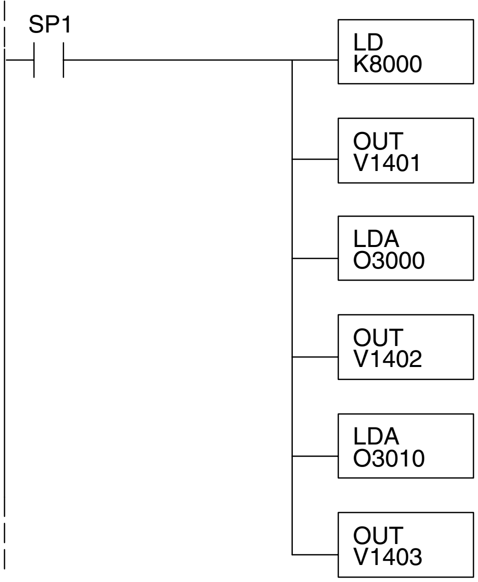
Updating All Channels in a Single Scan, DL440/450

☒	✓	✓
430	440	450

By using the Immediate instructions found in the DL440 and DL450 CPUs, you can easily update all sixteen channels in a single scan. This method slows 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 V3000 to V3007 for channels 1 to 7 respectively, and data loaded in V3010 to V3017 for channels 9 to 16 respectively.

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

Initialize the Immediate Analog Output Pointers



The LD instruction loads the data into the accumulator. The constant K8000 represents channel 1 and 9 selected, Output Enable bit is on.

We store the channel index pointer in V1401. The program increments this in each pass, resetting it after eight scans.

The Load Address instruction takes the octal 1401 and converts it to hex, then puts it in the accumulator. V3000 is the location wich contains the data for channel 1.

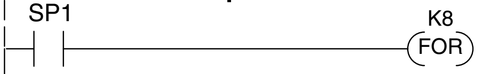
V1402 contains the pointer for channel 1 to 8 data.

The Load Address instruction takes the octal 3010 and converts it to hex, then puts it in the accumulator. V3010 is the location which contains the data for channel 9.

V1403 contains the pointer for channel 9 to 16 data.

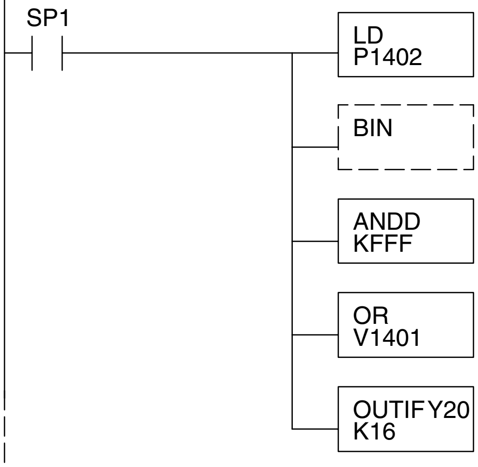
The following FOR–NEXT loop updates all 16 channels in a single scan. The first portion of the loop updates channels 1 through 8.

FOR-NEXT Loop



The following FOR–NEXT loop updates channels 1 through 16. The valid range for the constant K is 2 to 8 for the module. For example, a value of 2 will update channels 1 and 9, then 2 and 10.

Update Channels 1 - 8



Load the analog output value into the accumulator. V3000 = Ch. 1, V3001 = Ch. 2, ... V3007 = Ch. 8. The number at V1402 is a pointer to the address of the value.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

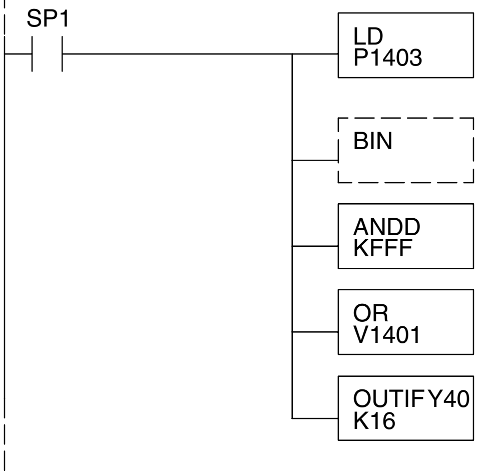
Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y36, Y35, and Y34 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module’s lower data word. This updates channels 1 through 8 during the FOR–NEXT loop. This example uses Y20, but the actual value depends on the location of the modules in your application.

The second portion of the FOR–NEXT loop updates channels 9 through 16.

Update Channels 9 - 16



Load the analog output value into the accumulator. V3010 = Ch. 9, V3011 = Ch. 10, ... V3017 = Ch. 16. The number at V1403 is a pointer to the address of the value.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

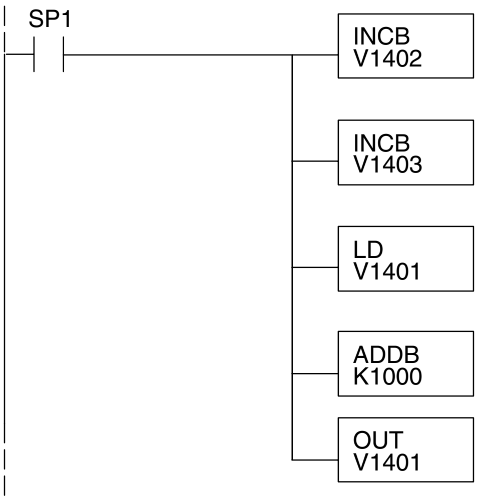
Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y56, Y55, and Y54 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module's upper data word. This updates channels 9 through 16 during the FOR–NEXT loop. This example uses Y40, but the actual value depends on the location of the modules in your application.

Now we increment the Immediate Analog Output Pointers for the next pair of channels, before the next pass through the FOR-NEXT loop.

Increment Immediate Analog Output Pointers



Increment the analog output data pointer for channels 1 through 8.

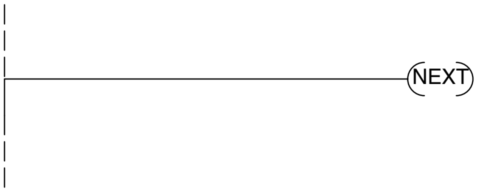
Increment the analog output data pointer for channels 9 through 16.

Load the current channel index count into the accumulator.

Add 1000 hex to the channel index value, since the channel select field is in the most significant four bits.

Save the incremented channel index value for the next pass through the FOR–NEXT loop.

Here is the end of the FOR-NEXT loop.



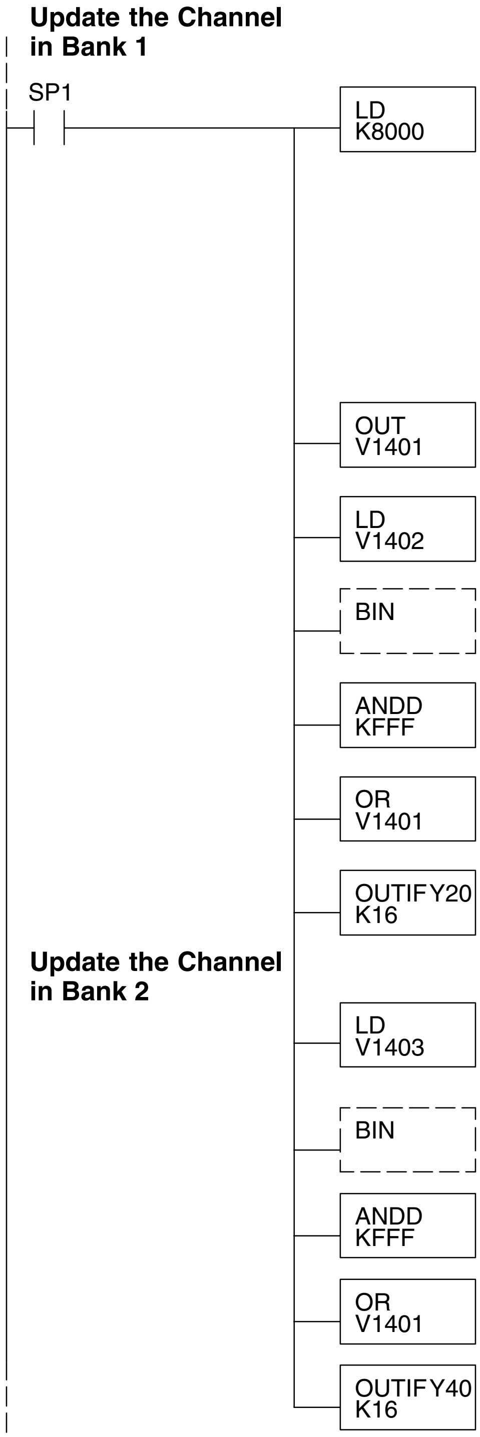
Execute the loop above the number of times specified in the FOR instruction.

The program above can be modified to immediately output any number of channel pairs by changing the constant in the FOR instruction.

Updating a Pair of Channels in a Single Scan, DL440/450

☒	☑	☑
430	440	450

You can also update just a pair of channels in a single scan by using the Immediate instructions found in the DL440 and DL450 CPUs. By removing the FOR–NEXT rungs and couple of other rungs, we create the example below. This example assumes you will use one channel from the first eight, and a corresponding channel from the second bank of eight. Also, the data must already be loaded in V1401 and V1402 respectively.



Load a constant representing the channel pair to be updated into the accumulator.

Index	Channels
8000	Ch. 1 and 9
9000	Ch. 2 and 10
A000	Ch. 3 and 11
B000	Ch. 4 and 12
C000	Ch. 5 and 13
D000	Ch. 6 and 14
E000	Ch. 7 and 15
F000	Ch. 8 and 16

We store the channel index in V1401.

Load the analog output value for the channel in bank 1 from V1402 into the accumulator.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y36, Y35, and Y34 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module’s lower data word. This updates a channel in the first bank of eight. This example uses Y20, but the actual value depends on the location of the modules in your application.

Load the analog output value for the channel in bank 2 from V1403 into the accumulator.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y56, Y55, and Y54 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module’s upper data word. This updates a channel in the second bank of eight. This example uses Y40, but the actual value depends on the location of the modules in your application.

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.

Range	If you know the digital value ...	If you know the analog signal level ...
4 to 20mA	$A = \frac{16D}{4095} + 4$	$D = \frac{4095}{16}(A - 4)$

For example, if you need a 9mA signal level, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

$$D = \frac{4095}{16}(A–4)$$

$$D = \frac{4095}{16}(9mA–4)$$

$$D = (255.94)\ (5)$$

$$D = 1280$$

17

F4–04DAS–1 4-Channel Isolated 4–20mA Output

In This Chapter. . . .

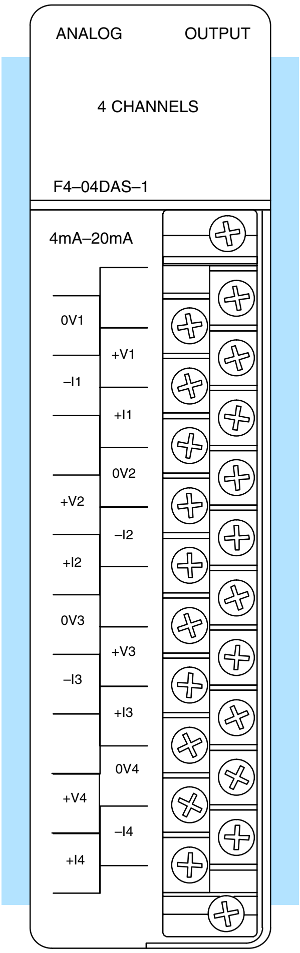
— Module Specifications
— Connecting the Field Wiring
— Module Operation
— Writing the Control Program

Module Specifications

The F4–04DAS–1 4-channel Isolated Analog Output module provides several features and benefits.

- Each analog output is isolated from the other outputs.
- 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).
- Provides four channels of isolated current outputs if used with independent loop power supplies.

Firmware Requirements:
When using this module with an H4–EBC, the H4–EBC must have firmware version 2.1.46 or later.



Analog Output Configuration Requirements

The F4–04DAS–1 Analog Output requires 32 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 F4–04DAS–1 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

Number of Channels	4, isolated current sourcing
Output Ranges	4–20mA
Resolution	16 bit (1 in 65536)
Isolation Voltage	± 750V continuous, channel to channel, channel to logic
Load Impedance	0–525Ω
Loop Supply	18–32VDC
Linearity Error (end to end)	± 10 counts (± 0.015%) of full scale
Offset Calibration Error	± 13 counts (± 0.02%)
Full Scale Calibration Error	± 8 counts maximum (offset error included)
Maximum Inaccuracy	± 0.07% at 25°C (77°F) ± 0.18% at 0 to 60°C (32 to 140°F)
Conversion Settling Time	3 ms to 0.1% of full scale

General Module Specifications

Digital Output Output Points Required	16 data bits, 2 channel ID, 1 output enable 32(Y) output points
Power Budget Requirement	100mA @ 5 VDC (from base)
External Power Supply	50mA per channel, class 2
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

Connecting the Field Wiring

Wiring Guidelines

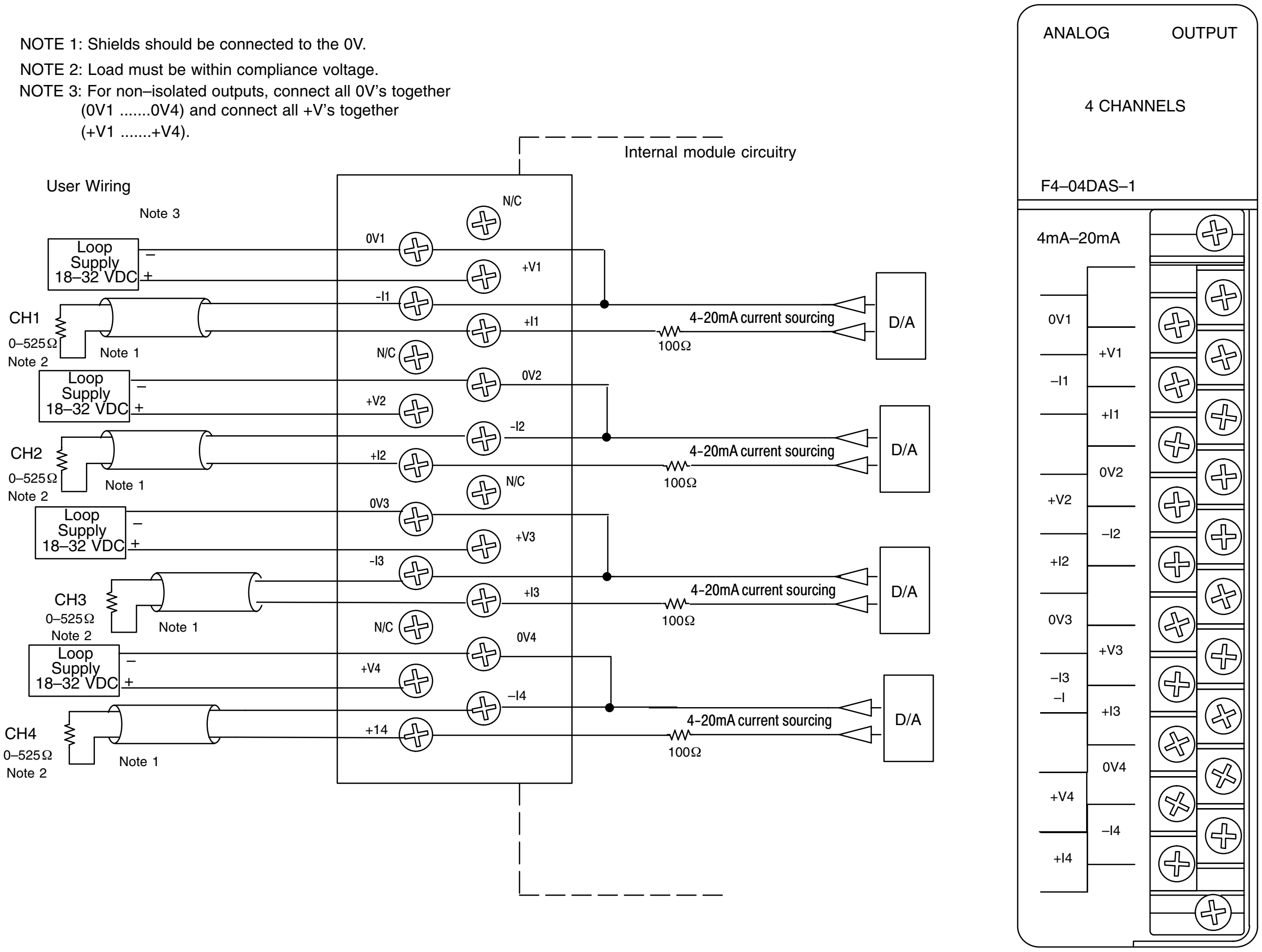
Your company may have guidelines for wiring and cable installation. If so, you should check those before you begin the installation. Here are some general things to consider.

- 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.
- 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.

Removable Connector

The F4–04DAS–1 module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

Wiring Diagram



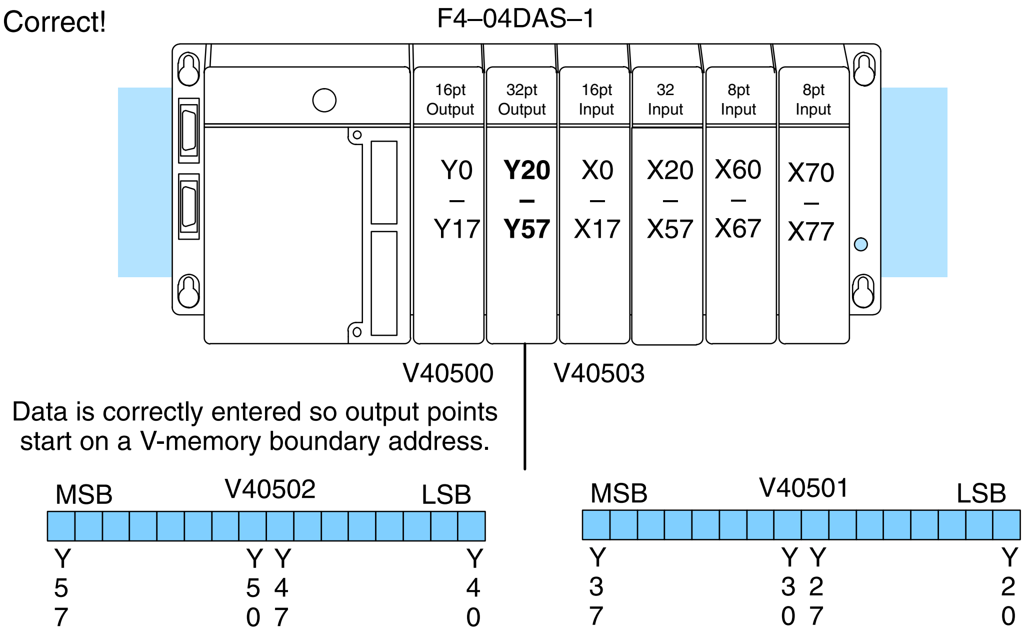
NOTE 1: Shields should be connected to the 0V.
NOTE 2: Load must be within compliance voltage.
NOTE 3: For non–isolated outputs, connect all 0V's together (0V1 ....0V4) and connect all +V's together (+V1 ....+V4).

Internal module circuitry
User Wiring
Note 3
Loop Supply 18–32 VDC
–
+
CH1
0–525Ω
Note 2
Note 1
CH2
CH3
CH4
N/C
0V1
+V1
–I1
+I1
N/C
0V2
+V2
–I2
+I2
N/C
0V3
+V3
–I3
+I3
N/C
0V4
+V4
–I4
+14
4–20mA current sourcing
100Ω
D/A
ANALOG
OUTPUT
4 CHANNELS
F4–04DAS–1
4mA–20mA
–I3
–I

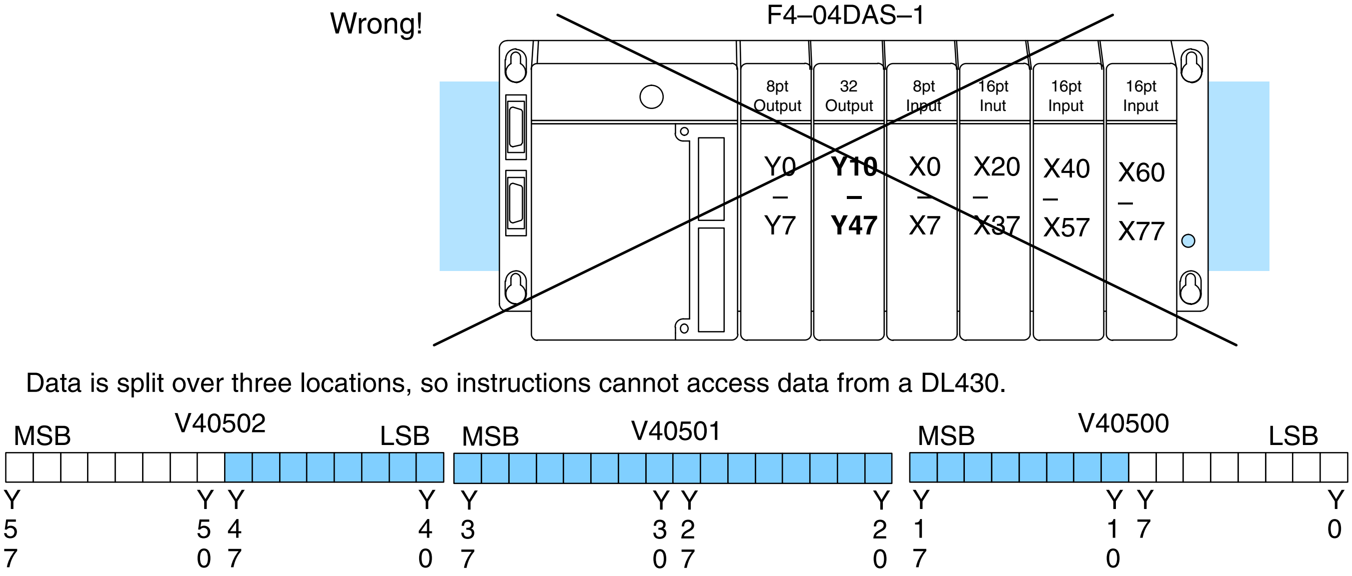
Module Operation

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!
F4–04DAS–1
16pt Output
32pt Output
16pt Input
32 Input
8pt Input
8pt Input
Y0 – Y17
Y20 – Y57
X0 – X17
X20 – X57
X60 – X67
X70 – X77
V40500
V40503
MSB
V40502
LSB
MSB
V40501
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20

Data is correctly entered so output points start on a V-memory boundary address.

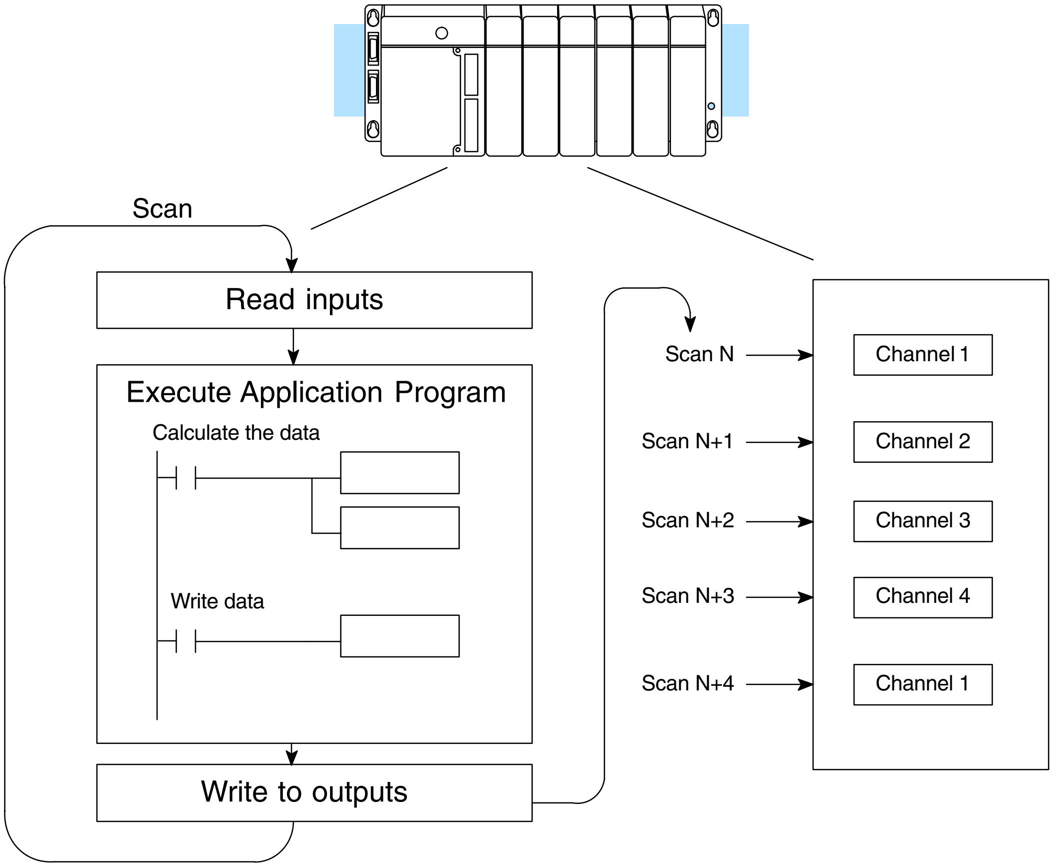

Wrong!
F4–04DAS–1
8pt Output
32 Output
8pt Input
16pt Inut
16pt Input
16pt Input
Y0 – Y7
Y10 – Y47
X0 – X7
X20 – X37
X40 – X57
X60 – X77
MSB
V40502
LSB
MSB
V40501
MSB
V40500
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20
Y17
Y10
Y7
Y0

Data is split over three locations, so instructions cannot access data from a DL430.

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 F4–04DAS–1 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 will show you how to do this later).

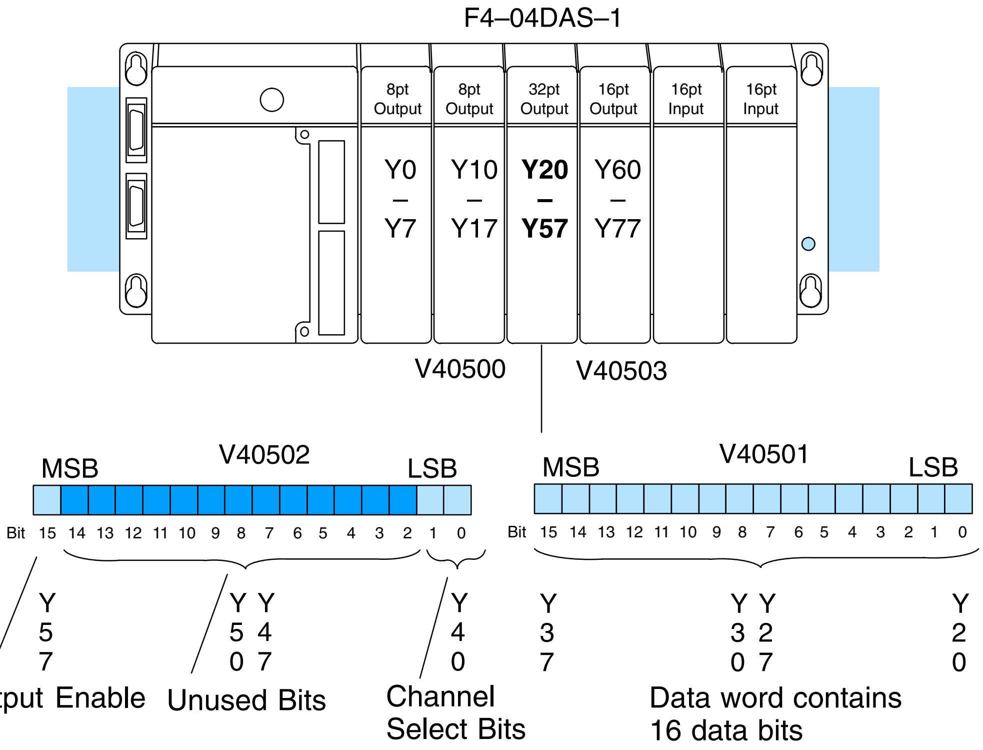

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 3
Channel 4
Channel 1

Output Bit Assignments

You may recall the F4–04DAS–1 module requires 32 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.

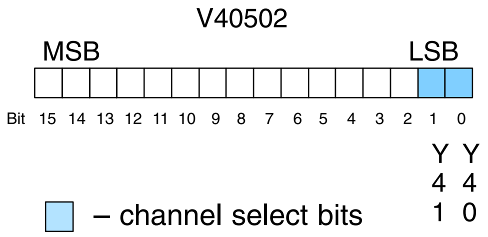
Since all output points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4–04DAS–1
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
MSB
LSB
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50 Y47
Y40
Y37
Y30 Y27
Y20
Output Enable Bit
Unused Bits
Channel Select Bits
Data word contains 16 data bits

Within this V-memory location the individual bits represent specific information about the analog signal.

Channel Select Bits

Bits 16 and 17 are binary encoded to select the channel that will be updated with the data. The bits are assigned as follows.

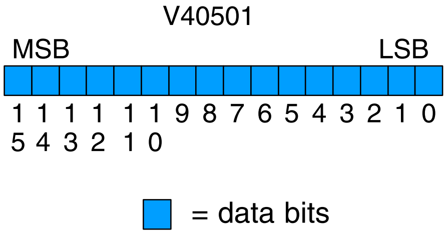

V40502
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y41 Y40
– channel select bits

Y41	Y40	Channel Number
0	0	1
0	1	2
1	0	3
1	1	4

Analog Data Bits

The first sixteen bits of the V-memory location represent the analog data in binary format.

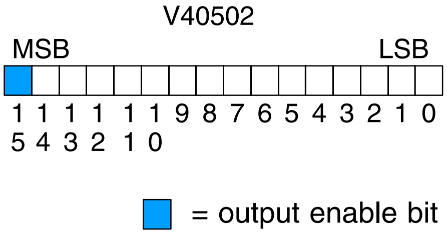
Bit	Value	Bit	Value
0	1	8	256
1	2	9	512
2	4	10	1024
3	8	11	2048
4	16	12	4096
5	32	13	8192
6	64	14	16384
7	128	15	32768



Output Enable Bit

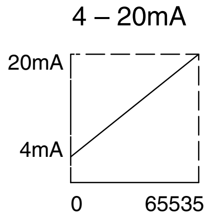
The most significant bit of the second word (or MSW) is the Output Enable Bit. Turning it on enables all four channels to be updated. Turning it off causes all output signal levels to go to 4mA and clears the module's internal data registers for all channels.

After an off-to-on transition of this bit, each output stays at 4mA until the channel and the CPU writes a non-zero value to it.



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 4mA signal and 65535 to get a 20mA 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.



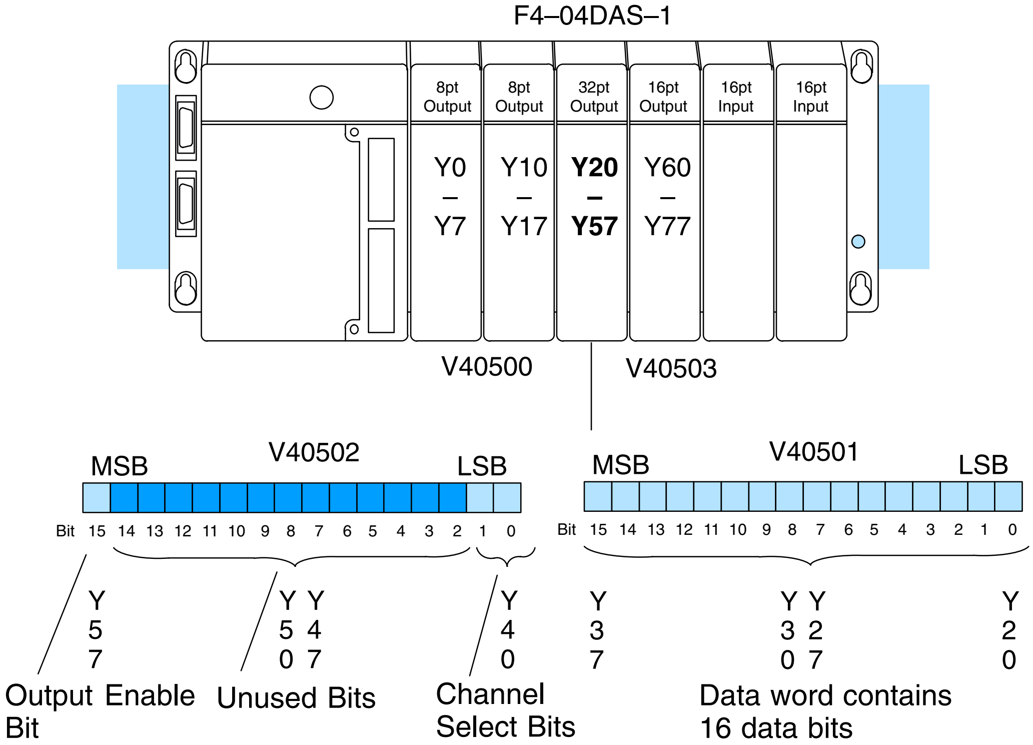
$$\text{Resolution} = \frac{H - L}{65535}$$

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

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).


F4–04DAS–1
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
MSB
LSB
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50 Y47
Y40
Y37
Y30 Y27
Y20
Output Enable Bit
Unused Bits
Channel Select Bits
Data word contains 16 data bits

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 almost all 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.

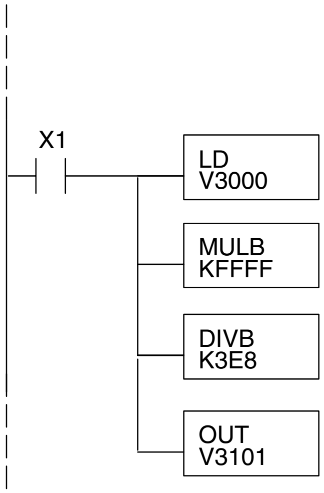
$$A = 10U \frac{65535}{10(H - L)} \qquad A = 494 \frac{65535}{1000 - 0} \qquad A = 32374$$

Engineering Unit Conversion

✓ ✓ ✓
430 440 450

Here is 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 all four channels if you are using different data for each channel.

NOTE: The DL405 offers various instructions that allow you to perform math operations using binary, BCD, etc. When using this module, it is usually easier to perform any math calculations in binary because of the large numbers involved.



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

Multiply the accumulator by 65535 (to start the conversion).

Divide the accumulator by 1000 (3E8 hex, because we used a multiplier of 10, we have to use 1000 instead of 100).

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.

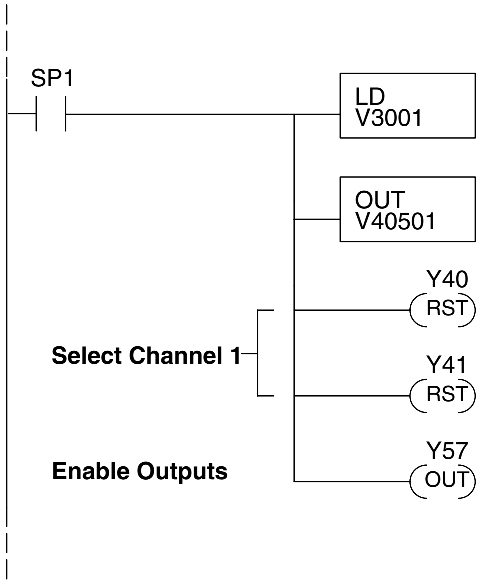
V-Memory Register Addresses for 16-Point Output (Y) Locations										
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

Sending Data to One Channel

✓ ✓ ✓
430 440 450

The following programs show you how to update a single channel. Notice that the BCD method uses a slightly different program than the binary method . Both examples assume you already have the data loaded in V3001.

Binary Example


SP1
LD
V3001
OUT
V40501
Y40
RST
Select Channel 1
Y41
RST
Y57
OUT
Enable Outputs

Data is in a range of 0–FFFF (hex).

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.

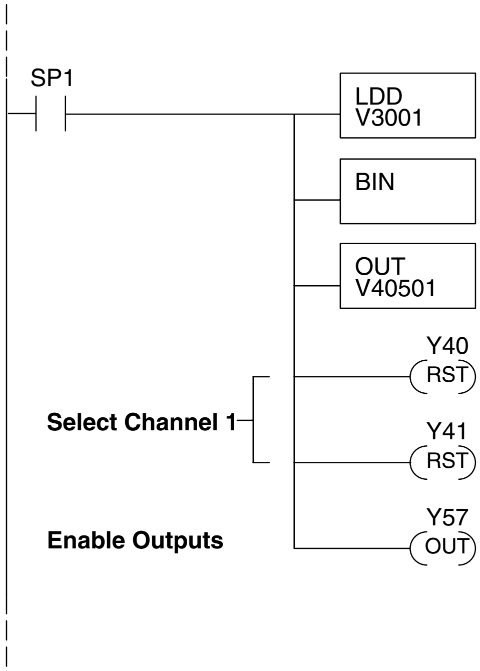
The OUT sends the 16 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.

Select channel 1 for updating.

Y41	Y40	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.

BCD Example


SP1
LDD
V3001
BIN
OUT
V40501
Y40
RST
Select Channel 1
Y41
RST
Y57
OUT
Enable Outputs

Data is in a range of 0–65535 (2 words).

The LDD 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.

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 1 for updating.

Y41	Y40	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 three example programs show you how to send digital values to the module when you have more than one channel. The first two examples will automatically update all four channels over four scans, while the last example updates all four channels in one scan.

The first sequencing example is fairly simple and will work in almost all situations. We recommend it for new users. It uses 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.

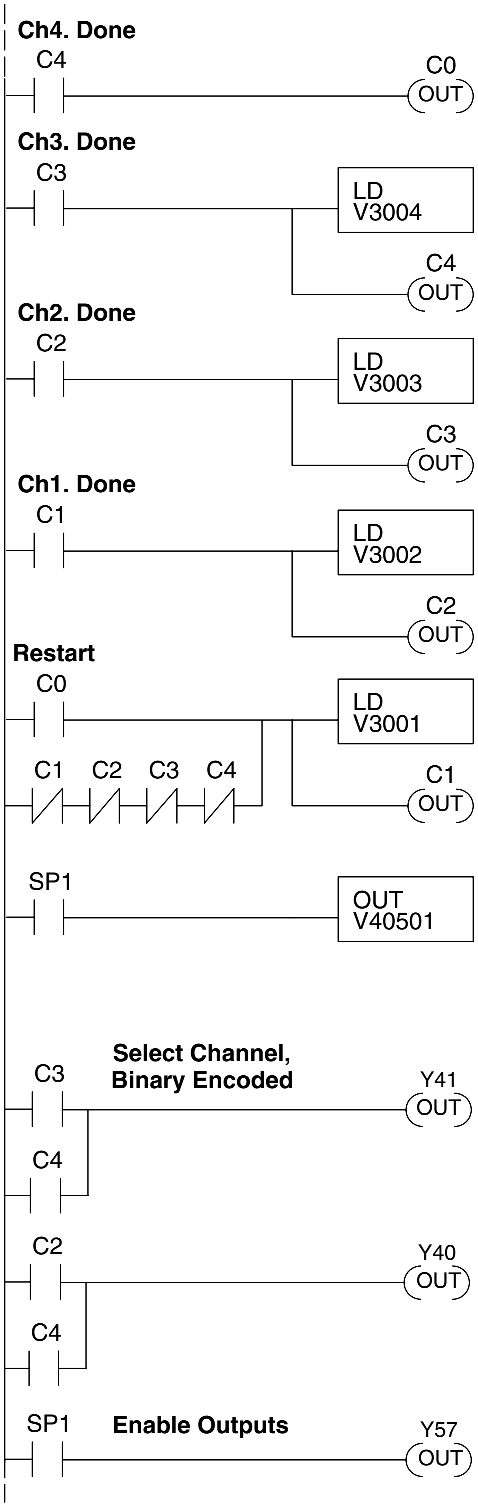
The second example is slightly more complex. However, it does not depend on the use of control relays to provide channel sequencing. Instead, it uses function boxes to increment a channel pointer value in V-memory. Then, other instructions perform bit manipulations to position the channel select bits properly in the output word to the module.

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, DL430/440/450

✓	✓	✓
430	440	450

This 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 binary format in V3001, V3002, V3003, and V3004 for channels 1 – 4 respectively (note that these locations are in a range of 0–FFFF hex). It is important to use the rungs in the order shown for the program to work.


Ch4. Done
C4
C0
OUT
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
LD
V3001
C1 C2 C3 C4
C1
OUT
SP1
OUT
V40501
Select Channel,
Binary Encoded
C3
Y41
OUT
C4
C2
Y40
OUT
C4
SP1
Enable Outputs
Y57
OUT

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

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).

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).

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).

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.

This rung 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.

Set Y41 and Y40 to select the output channel, based on the control relay status.

CR(on)	Y41	Y40	Channel
C1	Off	Off	Ch. 1
C2	Off	On	Ch. 2
C3	On	Off	Ch. 3
C4	On	On	Ch. 4

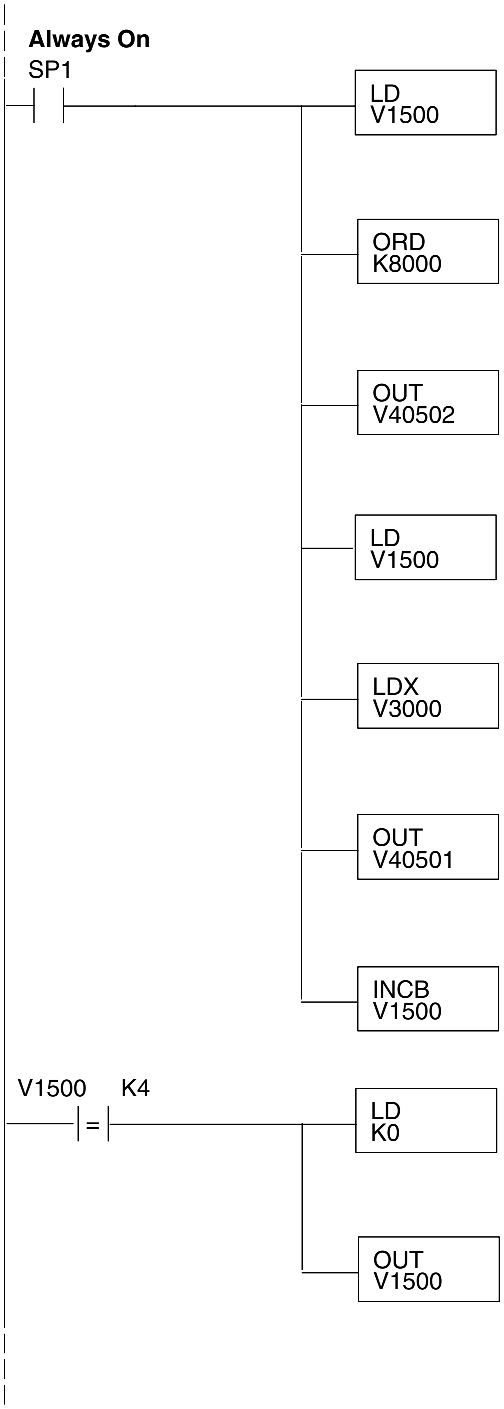
Enables all four output channels. SP1 is always on.

Sequencing Example 2, DL430/440/450

✓ ✓ ✓
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 have the data in binary format and are using the following data locations.

- V3000 – channel 1 data V3001 – channel 2 data
- V3002 – channel 3 data V3003 – channel 4 data
- V1500 – channel to update: 0 = ch. 1, 1 = ch. 2, 2 = ch. 3, 3 = ch. 4



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.

Logically ORs the value in the accumulator with the constant 8000, which sets the Output Enable Bit.

The result is stored in this location.

Again load the channel selection from V1500 back into the accumulator.

Use the channel selection value as an offset from V3000 to load the channel data into the accumulator.

Sends 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).

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

When channel 4 has been updated, this instruction resets the channel selection memory location to 0 (0 is for channel 1).

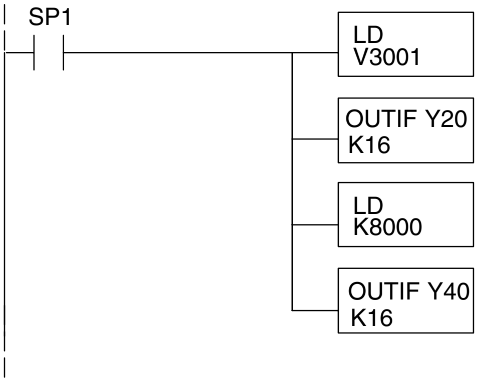
Updating all Channels in a Single Scan, DL440/450

☒	✓	✓
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 are using binary format and 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



The LD instruction loads the data into the accumulator. Specifiying V3001 selects channel 1.

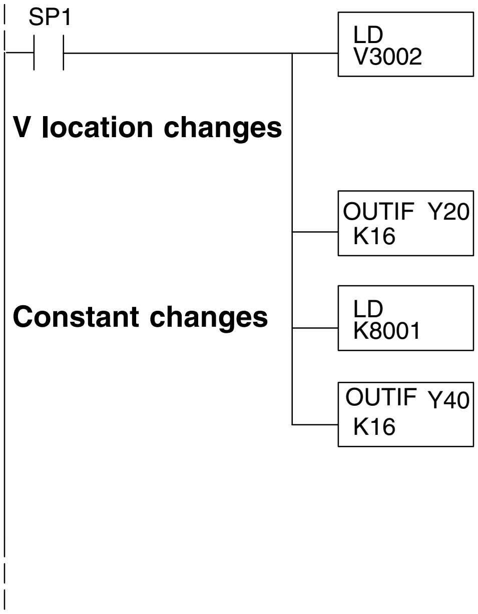
The OUTIF instruction sends 16 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.

Loads the constant 8000 into the accumulator.

The OUTIF instruction sends 16 bits to the data word. Our example starts with Y40, but the actual value depends on the location of the module in your application.

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 second LD instruction. The constant loaded with the second LD instruction is different for each channel. The following example shows where these differences occur.

Changes for channels 2 – 4



The LD instruction loads the data into the accumulator. Specifying V3002 selects channel 2. Here are the locations for each of the four channels.

Location	Channel
V3001	1
V3002	2
V3003	3
V3004	4

The OUTIF instruction 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.

Loads the constant 8001 into the accumulator.

The OUTIF instruction sends 16 bits to the data word. Our example starts with Y40, but the actual value depends on the location of the module in your application. The following constants are used.

Constant	Channel
K 8000	1
K 8001	2
K 8002	3
K 8003	4

Analog and Digital Value Conversions

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

Current Range	Output Format	If you know the digital value ...	If you know the analog signal level ...
4 – 20mA	0 to 65535	$A = \frac{16D}{65535} + 4$	$D = \frac{65535}{16}(A - 4)$

18

F4–04DAS–2 4-Channel Isolated 0–5V, 0–10V Output

In This Chapter. . . .

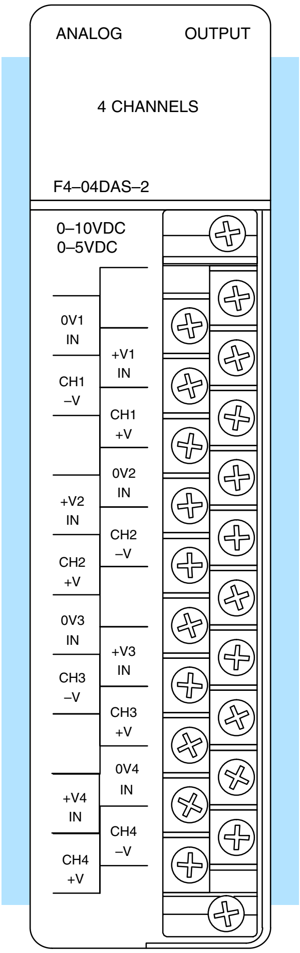
— Module Specifications
— Setting the Module Jumpers
— Connecting the Field Wiring
— Module Operation
— Writing the Control Program

Module Specifications

The F4–04DAS–2 4-channel Isolated Analog Output module provides several features and benefits.

- Each analog output is isolated from the other outputs.
- 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 (D4–440 and D4–450 CPUs only).
- Provides four channels of isolated voltage outputs if used with independent loop power supplies.

Firmware Requirements:
When using this module with an H4–EBC, the H4–EBC must have firmware version 2.1.46 or later.



Analog Output Configuration Requirements

The F4–04DAS–2 Analog Output requires 32 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 F4–04DAS–2 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

Number of Channels	4, Isolated
Output Ranges	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
Offset Calibration Error	± 13 counts (± 0.02%)
Full Scale Calibration Error	± 32 counts (0.05%)
Maximum Inaccuracy	± 0.07% at 25° C (77° F) ± 0.18% at 0 to 60° C (32 to 140° F)
Conversion Settling Time	3 ms to 0.1% of full scale

General Module Specifications

Digital Output Output Points Required	16 data bits, 2 channel ID, 1 output enable 32(Y) output points
Power Budget Requirement	60mA @ 5 VDC (from base)
External Power Supply	60mA per channel, 21.6VDC–26.4VDC, class 2
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

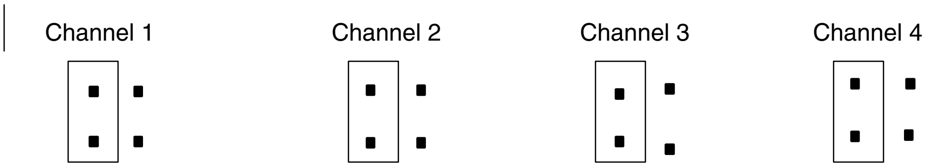
Setting the Module Jumpers

If you examine the rear of the module, you will notice several jumpers. These jumpers are used to select the signal range for each channel.

The signal range choices are 0 – 5 V and 0 – 10 V. The jumper settings for these signal ranges are shown in the table below.

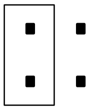
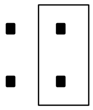
The module is set at the factory for a 0–5V signal on all four channels. If this is acceptable you do not have to change any of the jumpers. The following diagram shows how the jumpers are set from the factory.

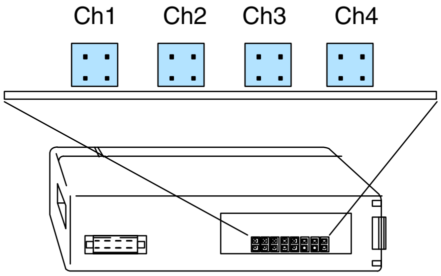
Signal Range Selection For Each Channel



Output Range Selection

Use the following table to select the output voltage range for each channel.

Channel Signal Range	Jumper Setting
0–5 VDC Place Jumper on **LEFT**	
0–10 VDC Place Jumper on **RIGHT**	



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.
- 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.

Removable Connector

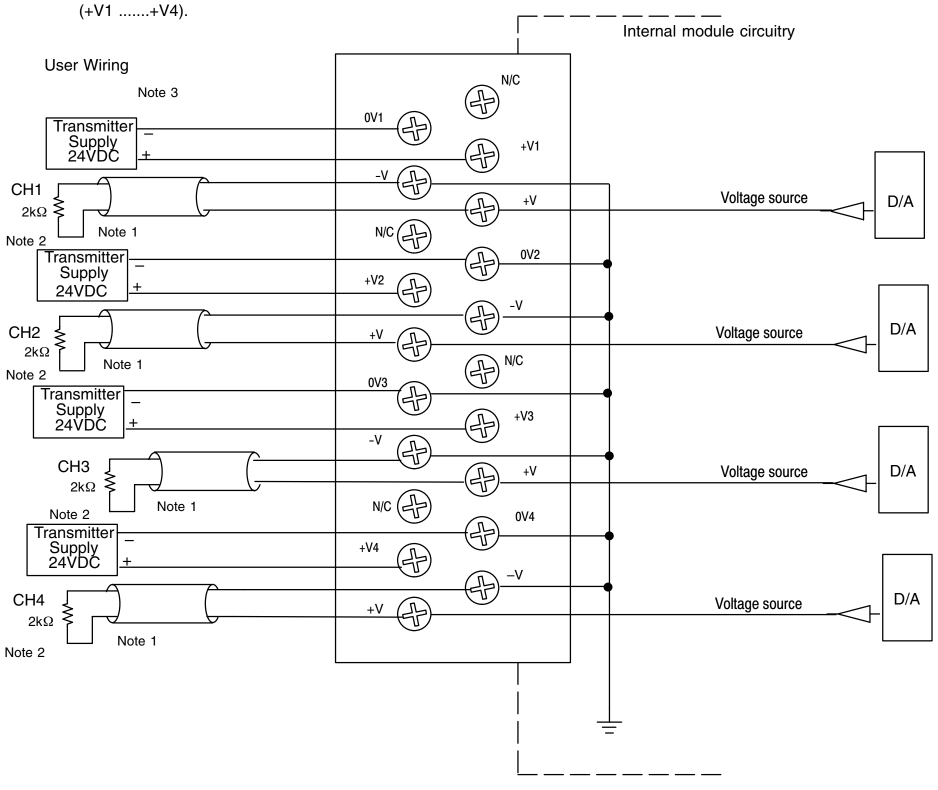
The F4–04DAS–2 module has a removable connector to make wiring easier. Simply remove the retaining screws and gently pull the connector from the module.

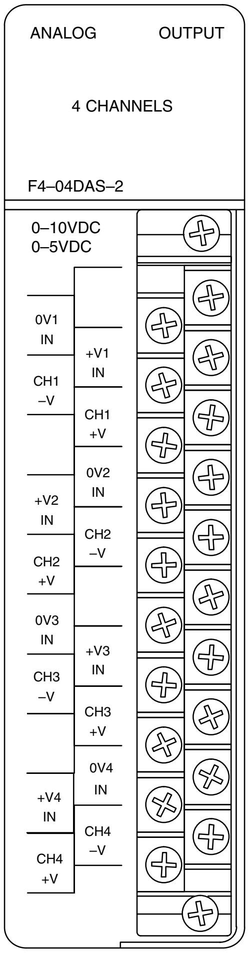
Wiring Diagram

NOTE 1: Shields should be connected to the 0V.

NOTE 2: Load must be within compliance voltage.

NOTE 3: For non–isolated outputs, connect all 0V's together
(0V1 ....0V4) and connect all +V's together
(+V1 ....+V4).


Internal module circuitry
User Wiring
Note 3
Transmitter Supply 24VDC
–
+
CH1
2kΩ
Note 1
Note 2
CH2
CH3
CH4
N/C
0V1
+V1
–V
+V
N/C
0V2
+V2
–V
+V
N/C
0V3
+V3
–V
+V
N/C
0V4
+V4
–V
+V
Voltage source
D/A

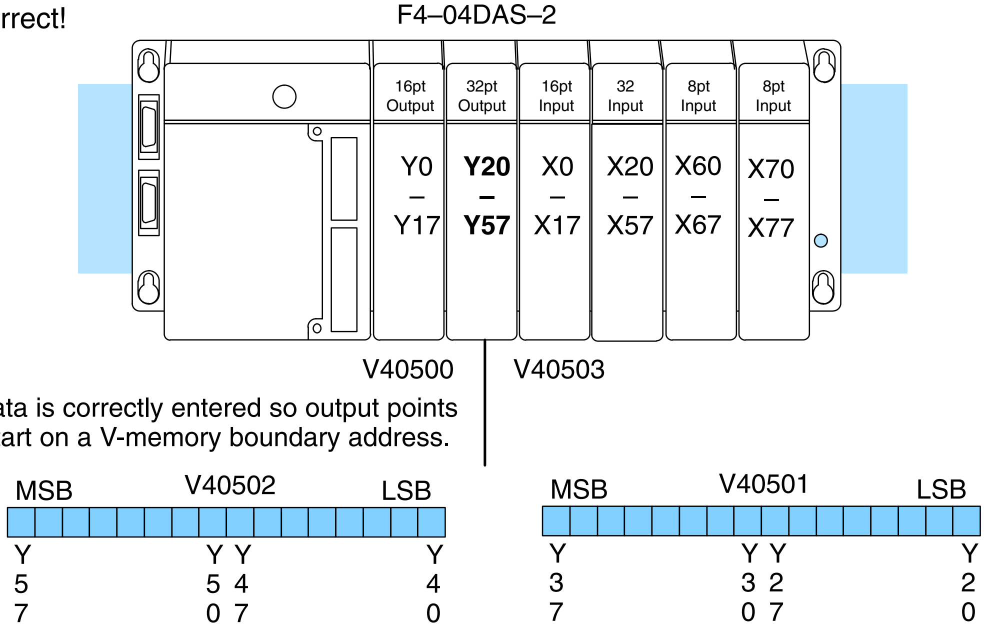

ANALOG OUTPUT
4 CHANNELS
F4–04DAS–2
0–10VDC
0–5VDC
0V1 IN
+V1 IN
CH1 –V
CH1 +V
0V2 IN
+V2 IN
CH2 –V
CH2 +V
0V3 IN
+V3 IN
CH3 –V
CH3 +V
0V4 IN
+V4 IN
CH4 –V
CH4 +V

Module Operation

D4–430 Special Requirements

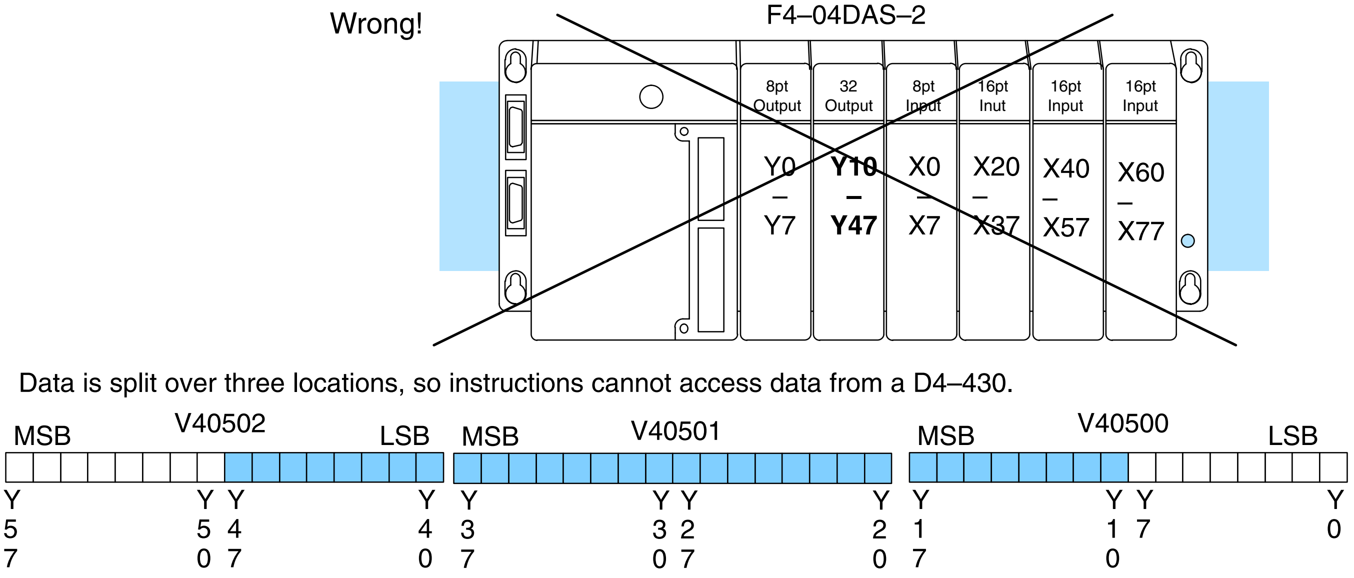
Even though the module can be placed in any slot, it is important to examine the configuration if you are using a D4–430 CPU. As you will see in the section on writing the program, you use V-memory locations to send the analog data. As shown in the following diagram, if you place the module so the output points do not start on a V-memory boundary, the instructions cannot access the data.

Correct!


F4–04DAS–2
16pt Output
32pt Output
16pt Input
32 Input
8pt Input
8pt Input
Y0 – Y17
Y20 – Y57
X0 – X17
X20 – X57
X60 – X67
X70 – X77
V40500
V40503
MSB
V40502
LSB
MSB
V40501
LSB
Y57
Y50 Y47
Y40
Y37
Y30 Y27
Y20

Data is correctly entered so output points start on a V-memory boundary address.

Wrong!

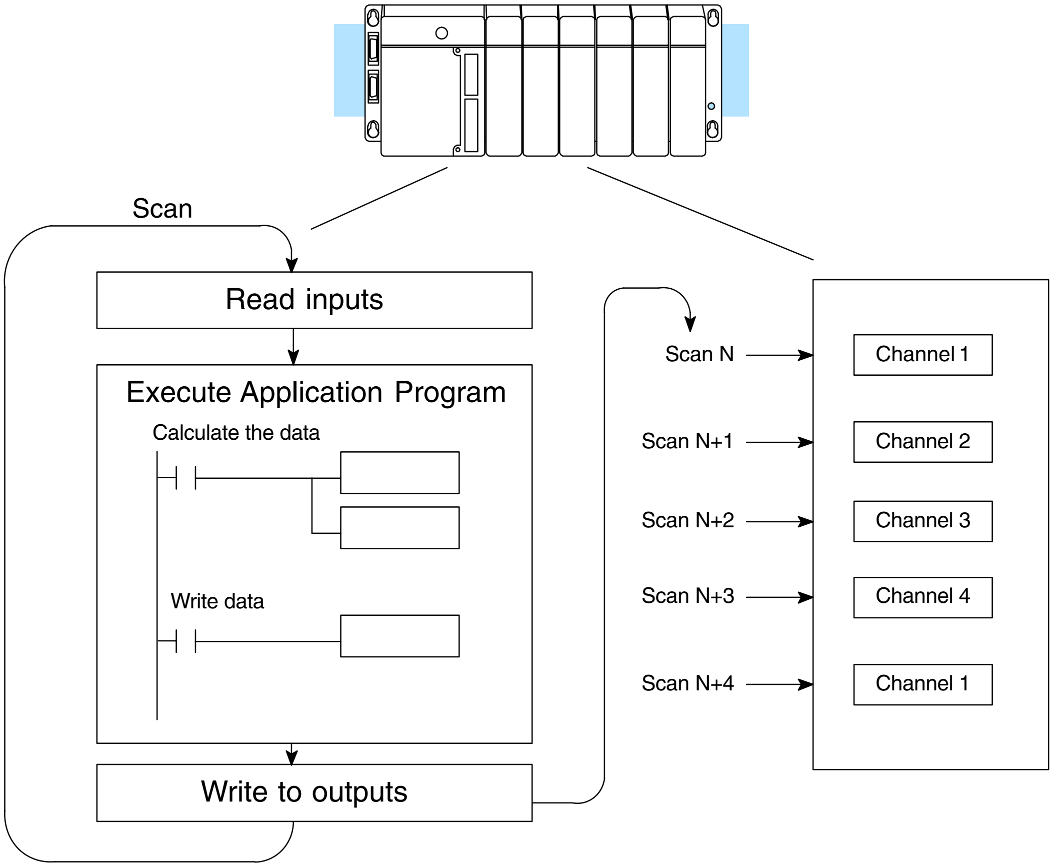

F4–04DAS–2
8pt Output
32 Output
8pt Input
16pt Inut
16pt Input
16pt Input
Y0 – Y7
Y10 – Y47
X0 – X7
X20 – X37
X40 – X57
X60 – X77
MSB
V40502
LSB
MSB
V40501
MSB
V40500
LSB
Y57
Y50 Y47
Y40
Y37
Y30 Y27
Y20
Y17
Y10 Y7
Y0

Data is split over three locations, so instructions cannot access data from a D4–430.

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 F4–04DAS–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 D4–440 or D4–450 CPU, you can use immediate instructions to update all four channels in the same scan (we will show you how to do this later).

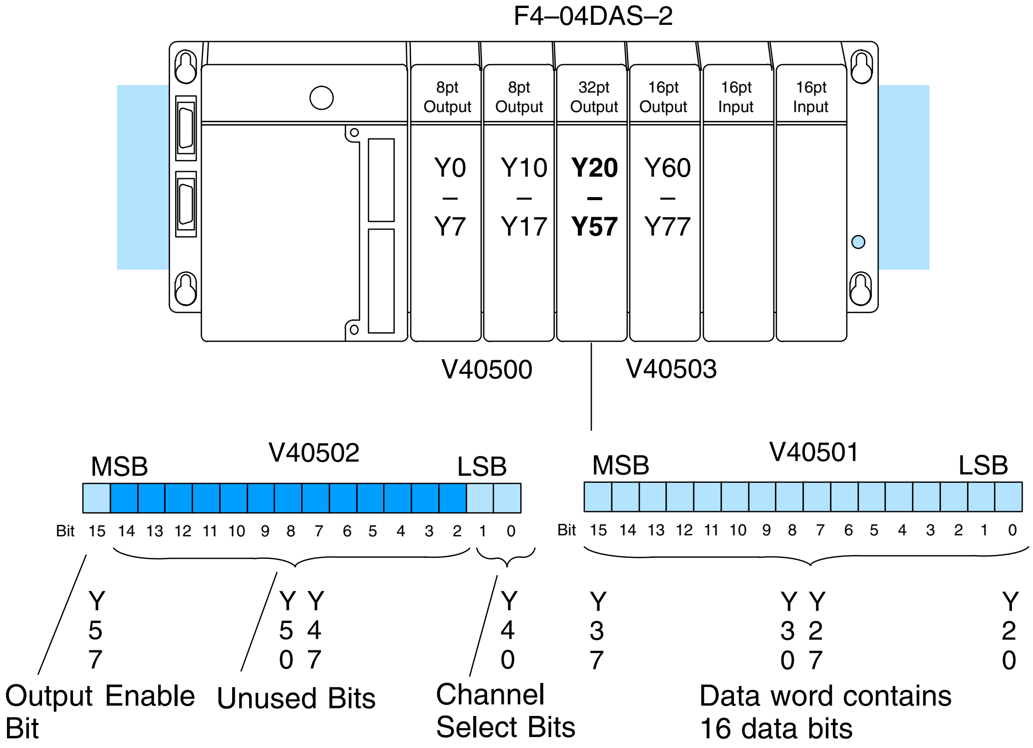

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Channel 1
Channel 2
Channel 3
Channel 4
Channel 1

Output Bit Assignments

You may recall the F4–04DAS–2 module requires 32 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.

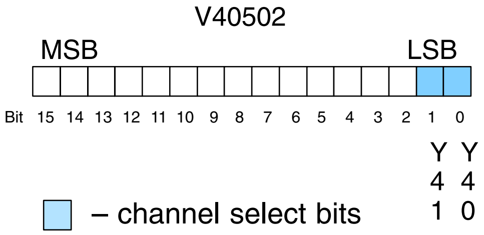
Since all output points are automatically mapped into V-memory, it is very easy to determine the location of the data word that will be assigned to the module.


F4–04DAS–2
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
MSB
LSB
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50 Y47
Y40
Y37
Y30 Y27
Y20
Output Enable Bit
Unused Bits
Channel Select Bits
Data word contains 16 data bits

Within this V-memory location the individual bits represent specific information about the analog signal.

Channel Select Bits

Bits 0 and 1 of the upper V-memory word are binary encoded to select the channel that will be updated with the data. The bits are assigned as follows.

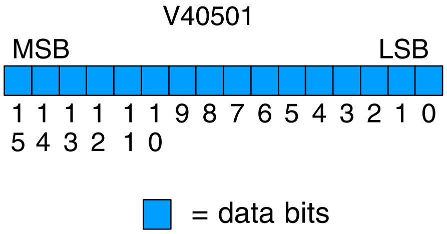

V40502
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y41 Y40
– channel select bits

Y41	Y40	Channel Number
0	0	1
0	1	2
1	0	3
1	1	4

Analog Data Bits

The first sixteen bit V-memory location represents the analog data in binary format.

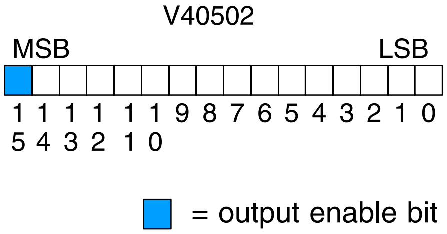
Bit	Value	Bit	Value
0	1	8	256
1	2	9	512
2	4	10	1024
3	8	11	2048
4	16	12	4096
5	32	13	8192
6	64	14	16384
7	128	15	32768



Output Enable Bit

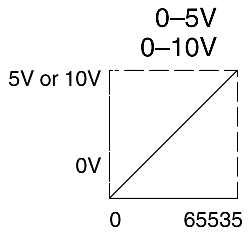
The most significant bit of the second word is the Output Enable Bit. Turning it on enables all four channels to be updated. Turning it off causes all output signal levels to go to 0V and clears the module's internal data registers for all channels.

After an off-to-on transition of this bit, each output stays at 0V until the channel and the CPU writes a non-zero value to it.



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 5V or 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.



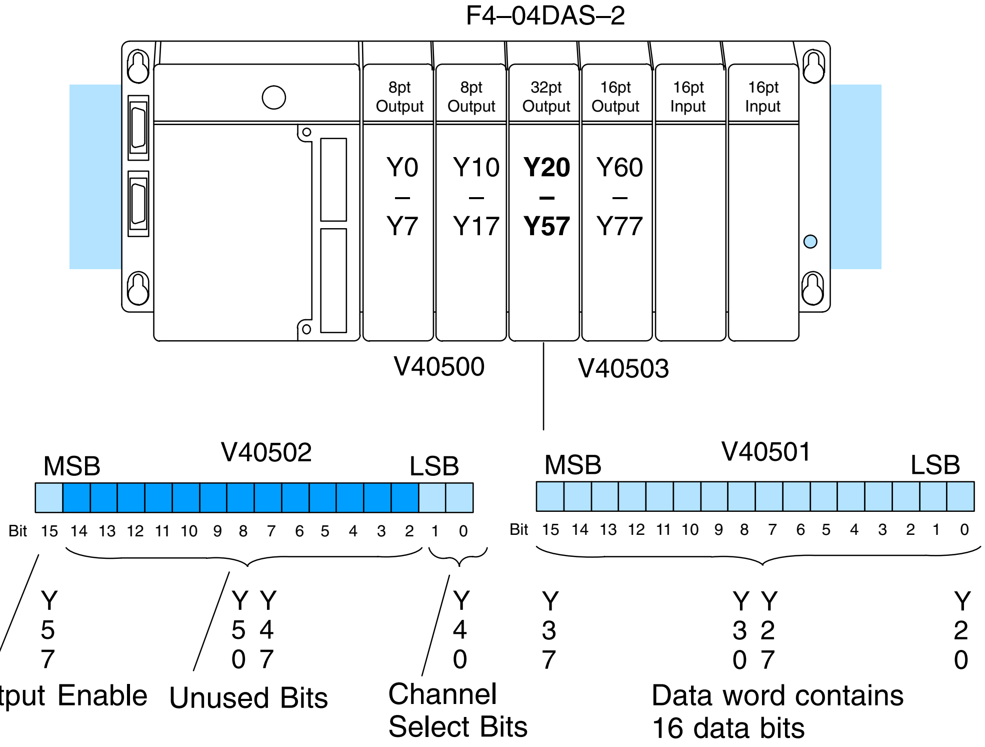
$$\text{Resolution} = \frac{H - L}{65535}$$

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

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).


F4–04DAS–2
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
MSB
V40502
LSB
MSB
V40501
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20
Output Enable Bit
Unused Bits
Channel Select Bits
Data word contains 16 data bits

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 almost all 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.

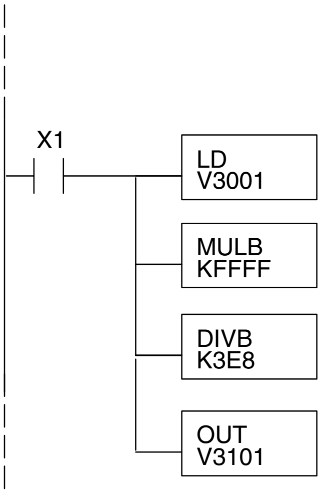
$$A = 10U \frac{65535}{10(H - L)} \qquad A = 494 \frac{65535}{1000 - 0} \qquad A = 32374$$

Engineering Unit Conversion

✓ ✓ ✓
430 440 450

Here is 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 V3001. Also, you have to perform this for all four channels if you are using different data for each channel.

NOTE: The DL405 offers various instructions that allow you to perform math operations using binary, BCD, etc. When using this module, it is usually easier to perform any math calculations in binary because of the large numbers involved.



When X1 is on, the engineering units (stored in V3001) are loaded into the accumulator. This example assumes the numbers are BIN.

Multiply the accumulator by 65535 (to start the conversion).

Divide the accumulator by 1000 (3E8 hex, because we used a multiplier of 10, we have to use 1000 instead of 100).

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 D4–450 CPUs.

V-Memory Register Addresses for 16-Point Output (Y) Locations

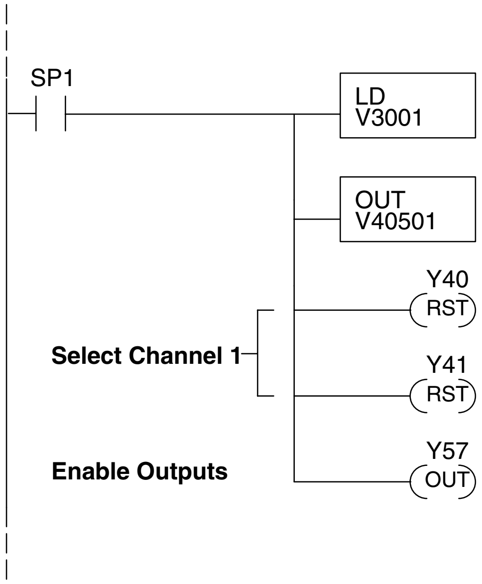
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

Sending Data to One Channel

✓ ✓ ✓
430 440 450

The following programs show you how to update a single channel. Notice that the BCD method uses a slightly different program than the binary method. Both examples assume you already have the data loaded in V3001.

Binary Example


SP1
LD V3001
OUT V40501
Y40 RST
Y41 RST
Y57 OUT
Select Channel 1
Enable Outputs

Data is in a range of 0–FFFF (hex).

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.

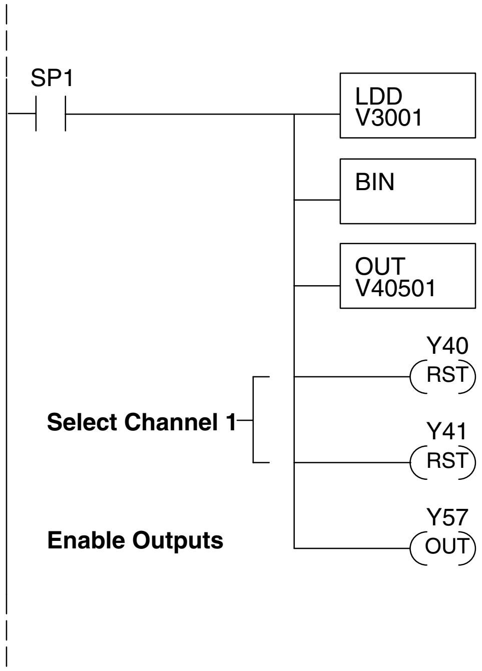
The OUT sends the 16 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.

Select channel 1 for updating.

Y41	Y40	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.

BCD Example


SP1
LDD V3001
BIN
OUT V40501
Y40 RST
Y41 RST
Y57 OUT
Select Channel 1
Enable Outputs

Data is in a range of 0–65535 (2 words).

The LDD 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.

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 1 for updating.

Y41	Y40	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 three example programs show you how to send digital values to the module when you have more than one channel. The first two examples will automatically update all four channels over four scans, while the last example updates all four channels in one scan.

The first sequencing example is fairly simple and will work in almost all situations. We recommend it for new users. It uses 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.

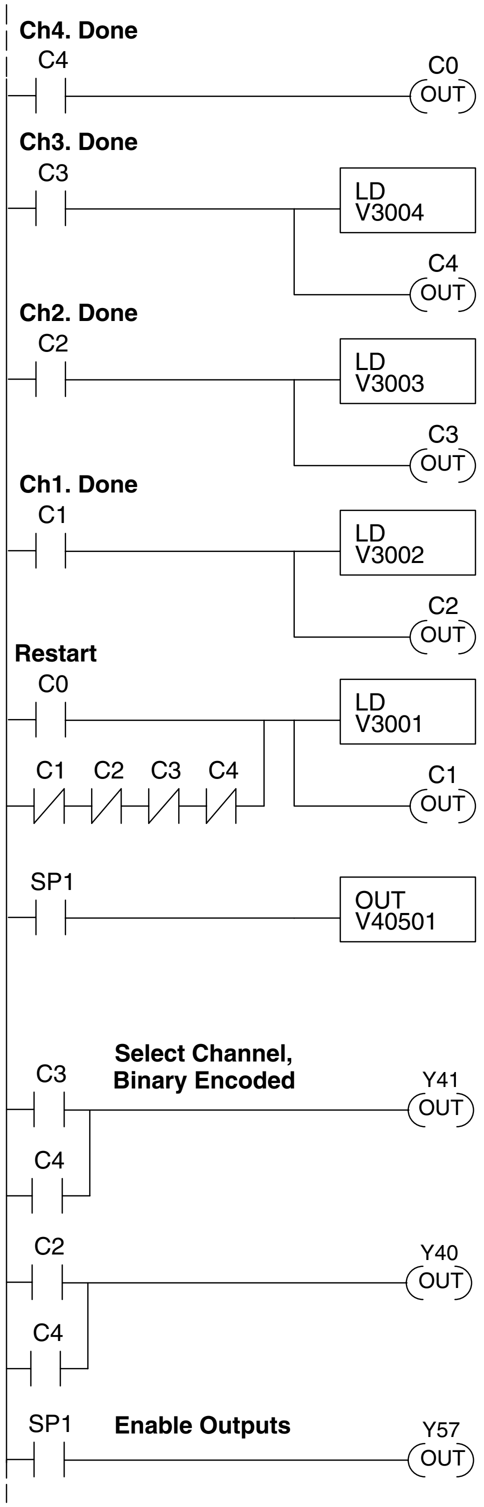
The second example is slightly more complex. However, it does not depend on the use of control relays to provide channel sequencing. Instead, it uses function boxes to increment a channel pointer value in V-memory. Then, other instructions perform bit manipulations to position the channel select bits properly in the output word to the module.

In the last example, we show you how you can update all four channels in the same scan with D4–440 and D4–450 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

✓	✓	✓
430	440	450

This 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 binary format in V3001, V3002, V3003, and V3004 for channels 1 – 4 respectively (note that these locations are in a range of 0–FFFF hex). It is important to use the rungs in the order shown for the program to work.


Ch4. Done
C4
C0
OUT
Ch3. Done
C3
LD
V3004
C4
OUT
Ch2. Done
C2
LD
V3003
C3
OUT
Ch1. Done
C1
LD
V3002
C2
OUT
Restart
C0
LD
V3001
C1 C2 C3 C4
C1
OUT
SP1
OUT
V40501
C3
Select Channel, Binary Encoded
Y41
OUT
C4
C2
Y40
OUT
C4
SP1
Enable Outputs
Y57
OUT

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

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).

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).

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).

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.

This rung 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.

Set Y41 and Y40 to select the output channel, based on the control relay status.

CR(on)	Y41	Y40	Channel
C1	Off	Off	Ch. 1
C2	Off	On	Ch. 2
C3	On	Off	Ch. 3
C4	On	On	Ch. 4

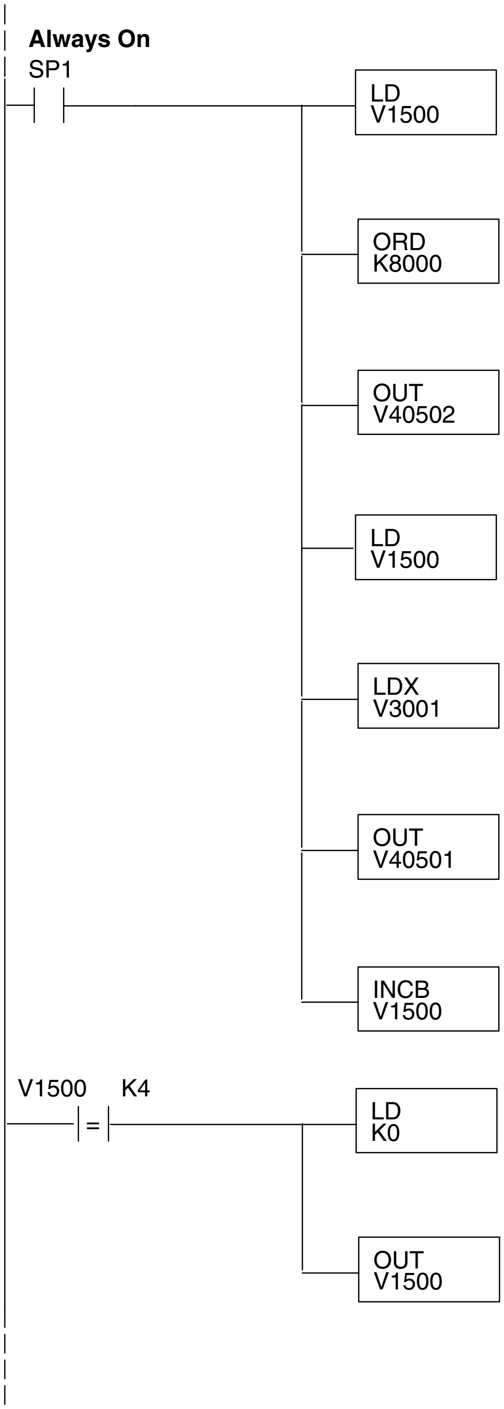
Enables all four output channels. SP1 is always on.

Sequencing Example 2

✓ ✓ ✓
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 have the data in binary format and are using the following data locations.

- V3001 – channel 1 data V3002 – channel 2 data
- V3003 – channel 3 data V3004 – channel 4 data
- V1500 – channel to update: 0 = ch. 1, 1 = ch. 2, 2 = ch. 3, 3 = ch. 4



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.

Logically ORs the value in the accumulator with the constant 8000, which sets the Output Enable Bit.

The result is stored in this location.

Again load the channel selection from V1500 back into the accumulator.

Use the channel selection value as an offset from V3001 to load the channel data into the accumulator.

Sends 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).

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

When channel 4 has been updated, this instruction resets the channel selection memory location to 0 (0 is for channel 1).

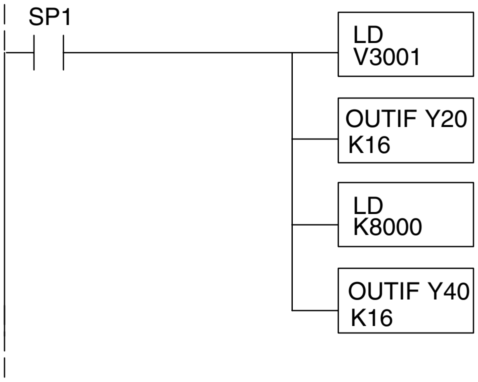
Updating all Channels in a Single Scan

☒	✓	✓
430	440	450

By using the Immediate instructions found in the D4–440 and D4–450 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 are using binary format and already have the data loaded in V3001, V3002, V3003, and V3004 for channels 1 – 4 respectively. This example will not work with D4–430 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



The LD instruction loads the data into the accumulator. Specifiying V3001 selects channel 1.

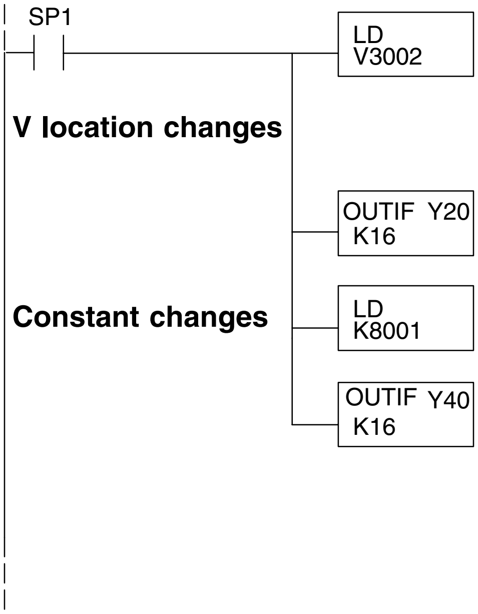
The OUTIF instruction sends 16 bits to the data word. Our example starts with Y20, but the actual value depends on the location of the module in your application.

Loads the constant 8000 into the accumulator.

The OUTIF instruction sends 16 bits to the channel select word. Our example starts with Y40, but the actual value depends on the location of the module in your application.

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 second LD instruction. The constant loaded with the second LD instruction is different for each channel. The following example shows where these differences occur.

Changes for channels 2 – 4



The LD instruction loads the data into the accumulator. Specifying V3002 selects channel 2. Here are the locations for each of the four channels.

Location	Channel
V3001	1
V3002	2
V3003	3
V3004	4

The OUTIF instruction 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.

Loads the constant 8001 into the accumulator.

The OUTIF instruction sends 16 bits to the channel select word. Our example starts with Y40, but the actual value depends on the location of the module in your application. The following constants are used.

Constant	Channel
K 8000	1
K 8001	2
K 8002	3
K 8003	4

Analog and Digital Value Conversions

Sometimes it is helpful to be able to quickly convert between the 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 ...
0–5VDC	0 to 65535	$A = \frac{5D}{65535}$	$D = \frac{65535}{5}A$
0–10VDC	0 to 65535	$A = \frac{10D}{65535}$	$D = \frac{65535}{10}A$

19

F4–08DA–2 8-Channel Analog Voltage Output

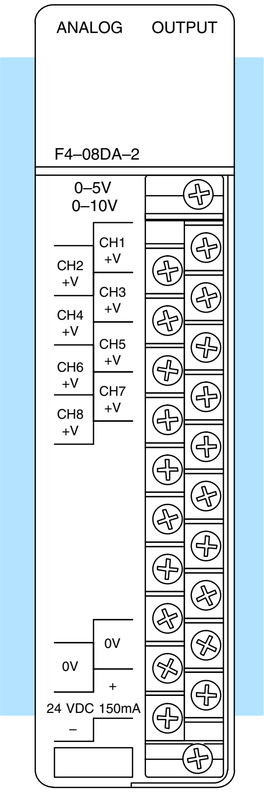
In This Chapter. . . .

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

Module Specifications

The F4–08DA–2 Analog Voltage Output Module provides several features and benefits.

- It provides eight channels of 0–5V or 0–10V single ended voltage outputs.
- 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.
- From one to eight analog outputs may be updated in one CPU scan (D4–440 and D4–450 CPUs only).



Analog Output Configuration Requirements

The F4–08DA–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 tables provide the specifications for the F4–08DA–2 Analog Voltage Output Module. Review these specifications to ensure the module meets your application requirements.

Output Specifications

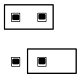
Number of Channels	8, single ended (one common)
Output Range	0–5VDC, 0–10VDC
Resolution	12 bit (1 in 4095)
Output Type	Voltage Sourcing 10mA max.
External Load	1kΩ maximum / 10kΩ minimum (for example: 10 volts at 1kΩ = 10mA load; 10 volts at 10kΩ = 1mA load)
Crosstalk	–70 dB, ±1 count maximum
Linearity Error (end-to-end) and Relative Accuracy	±1 count maximum (10VDC at 25°C)
Full Scale Calibration Error (offset error included)	±6 counts maximum (10VDC at 25°C)
Offset Calibration Error	±3 counts maximum (0VDC at 25°C)
Maximum Inaccuracy	±0.2% at 25°C (77°F) ±0.4% at 0° to 60°C (32° to140°F)
Conversion Time	400 μS maximum, for full scale change 4.5 to 9 mS for digital output to analog out

General Module Specifications

Digital Output Points Required	16 point (Y) outputs, 12 bits binary data, 3 bits channel select, 1 bit output enable
Power Budget Requirement	80 mA at 5 VDC (supplied by base power supply)
External Power Supply	21.6 to 26.4 VDC, 150 mA max., class 2
Accuracy vs. Temperature	±57 ppm / °C full scale calibration range (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

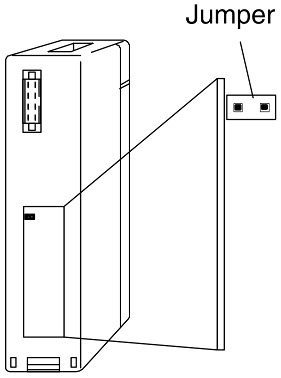
Setting the Module Jumper

Before installing and wiring the module, you may need to change the internal jumper setting. The module has one jumper, located in the open cutout at the rear of the housing. When the jumper is installed (which is the factory default setting), the module operates in 0–5VDC mode for all eight channels. When the jumper is removed, the module operates in 0–10VDC mode. When removed, store the jumper by placing it over one terminal as shown below to prevent losing it.



Installed = 0–5VDC Mode

Removed = 0–10VDC Mode



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.
- 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 F4–08DA–2 requires a field-side power supply. The module requires 21.6 – 26.4 VDC, Class 2, 150mA max. current.

The D4–430/440/450 CPUs, D4–RS Remote I/O Controller, H4–EBC, and D4–EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. You may use one of these instead of a separate supply if there is only a couple of analog modules. The current required is 150mA max.: eight outputs driving 1kΩ loads to 10V (10mA x 8); plus 70mA for the module's internal circuitry.

Load Requirements

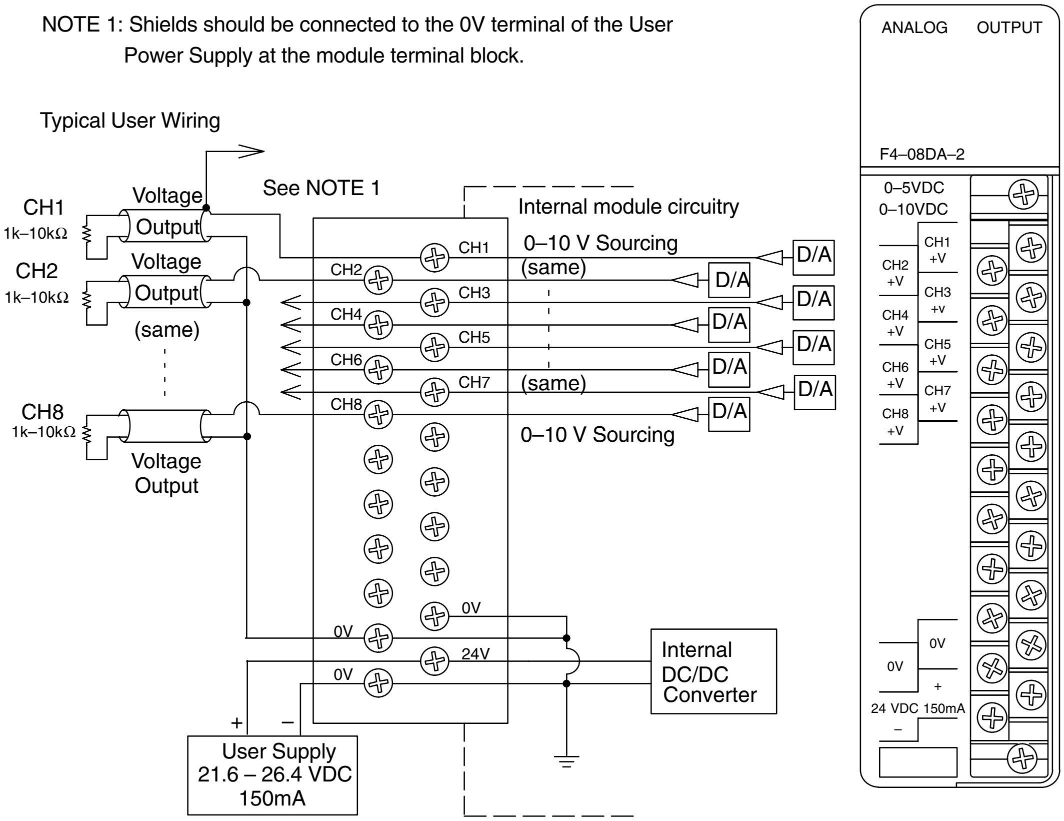
Each channel in use must have a load impedance of 1kΩ to 10kΩ. Unused channels must be left disconnected.

WARNING: If you are using the 24 VDC base power supply, 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.

Removable Connector

The F4–08DA–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

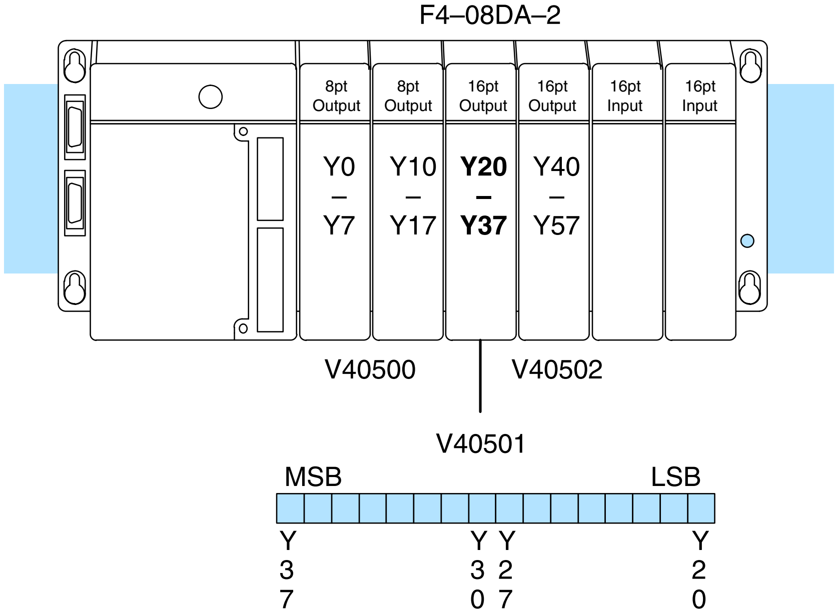

NOTE 1: Shields should be connected to the 0V terminal of the User Power Supply at the module terminal block.
Typical User Wiring
See NOTE 1
Internal module circuitry
CH1
1k–10kΩ
Voltage Output
CH2
1k–10kΩ
Voltage Output
(same)
CH8
1k–10kΩ
Voltage Output
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
0–10 V Sourcing
(same)
(same)
0–10 V Sourcing
D/A
0V
0V
24V
0V
Internal DC/DC Converter
+
–
User Supply 21.6 – 26.4 VDC 150mA
ANALOG OUTPUT
F4–08DA–2
0–5VDC
0–10VDC
CH1 +V
CH2 +V
CH3 +V
CH4 +V
CH5 +V
CH6 +V
CH7 +V
CH8 +V
0V
0V
+
24 VDC 150mA
–

Module Operation

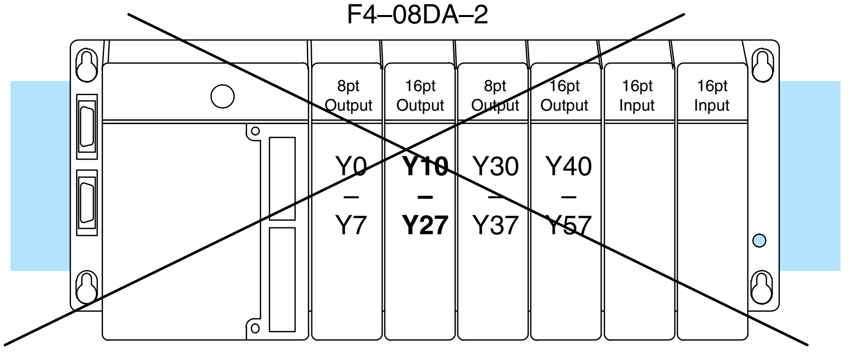
D4–430 Special Requirements

Even though the module can be placed in any slot, it is important to examine the configuration if you are using a D4–430 CPU. As you'll 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.

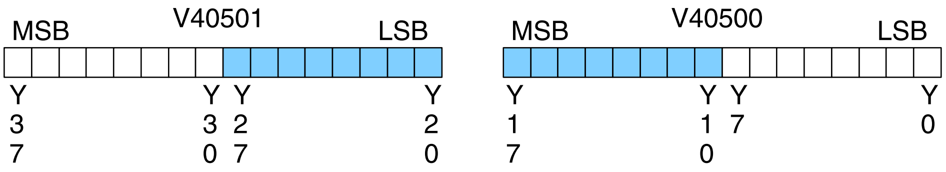
Correct!


F4–08DA–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!


F4–08DA–2
8pt Output
16pt Output
8pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y27
Y30 – Y37
Y40 – Y57

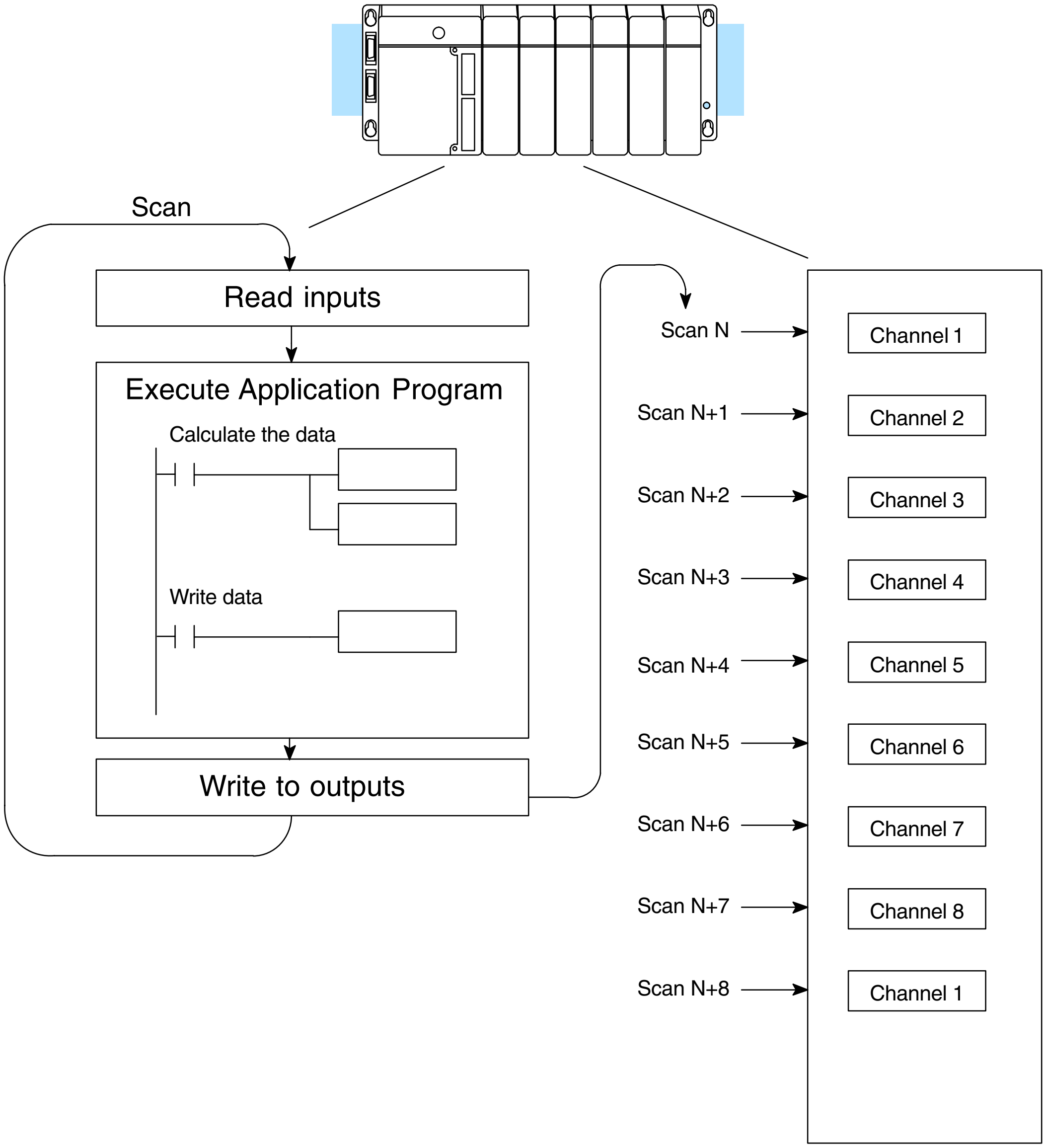
Data is split over two locations, so instructions cannot access data from a D4–430.


MSB
V40501
LSB
MSB
V40500
LSB
Y 3 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 F4–08DA–2 module allows you to update the channels in any order. Your control program determines which channels get updated on any given scan. A typical ladder program will update one channel per CPU scan. So, all eight channels can be updated every eight scans. With a D4–440 or D4–450 CPU, you can use Immediate instructions to update all eight channels in the same scan (we'll show you how to do this later).

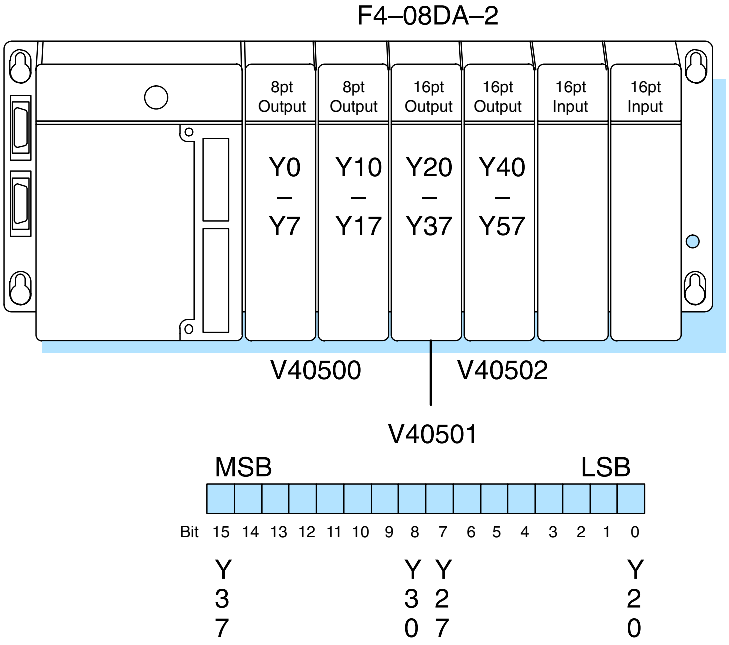

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Scan N+8
Channel 1
Channel 2
Channel 3
Channel 4
Channel 5
Channel 6
Channel 7
Channel 8
Channel 1

Output Bit Assignments

You may recall the F4–08DA–2 module requires 16 discrete output points from the CPU. These points provide:

- The digital representation of one analog signal per scan.
- Identification of the channel that is to receive the data.
- Output enable control for all channels.

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.

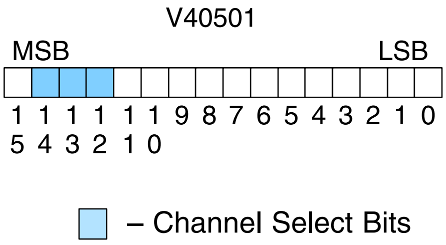

F4–08DA–2
8pt Output
8pt Output
16pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y37
Y40 – Y57
V40500
V40502
V40501
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y37
Y30
Y27
Y20

Within this V-memory location the individual bits represent specific information about the channel selected and the analog signal.

Channel Select Bits

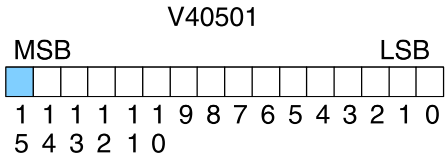
Output bits 12, 13, and 14 of the data word are the channel select outputs. They are binary encoded to select the channel from 1 to 8 that will be updated with the data.

Bit 14	Bit 13	Bit 12	Channel
Off	Off	Off	1
Off	Off	On	2
Off	On	Off	3
Off	On	On	4
On	Off	Off	5
On	Off	On	6
On	On	Off	7
On	On	On	8


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– Channel Select Bits

Output Enable Bit

Output bit 15 is the Output Enable control bit for all eight channels. When the bit is off, all eight channel output voltage levels drop to 0VDC. Disabling the outputs also clears all eight output data registers. To resume analog output levels, first the Output Enable control bit must turn on. Then, the CPU must write new data to each channel to restore the output voltage for that channel.



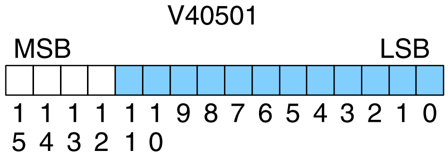
🟦 – Output Enable Bit

OFF = Disable (and clear)
ON = Enable

Analog Data Bits

The first twelve bits of the data word represent the analog data in binary format.

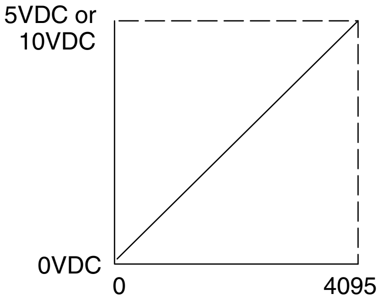
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048



🟦 – data bits

Module Resolution

Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0–4095 (2^{12}). For the 0 to 5V scale, sending a 0 produces a 0VDC signal, and 4095 sends a 5VDC signal. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The graph to the right shows the linear relationship between the data value and output signal level.



Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal change that occurs when the digital value is increased by 1 LSB.

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

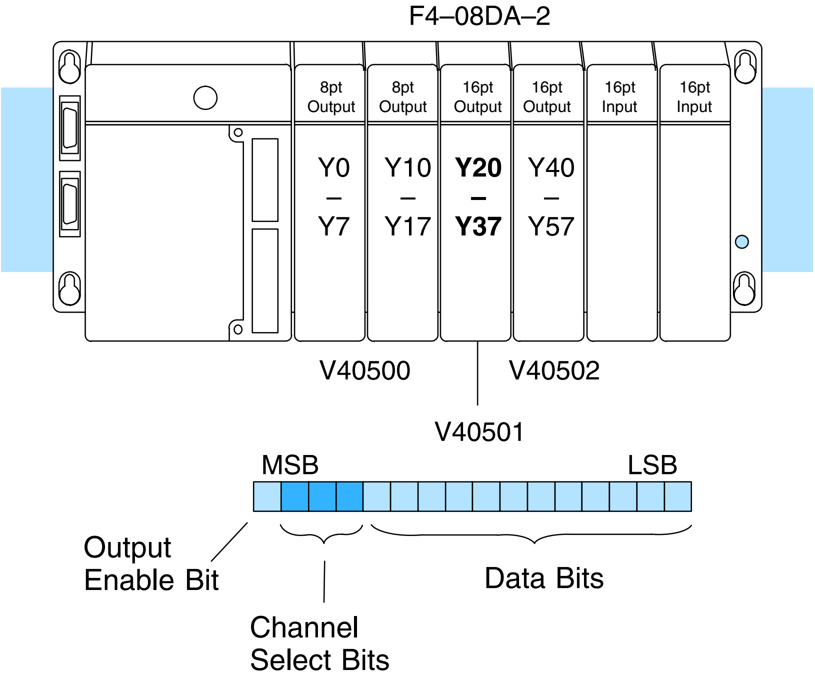
H = high limit of the signal range
L = low limit of the signal range

Signal Range	Span (H – L)	Divide By	Smallest Change
0 to 5VDC	5VDC	4095	1.221mV
0 to 10VDC	10VDC	4095	2.442mV

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 select outputs to determine which channel gets updated (more on this later).


F4–08DA–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
Output Enable Bit
Channel Select Bits
Data Bits

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.

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.

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

A = analog value (0 – 4095)
U = engineering units
H = high limit of the engineering unit range
L = low limit of the engineering unit range

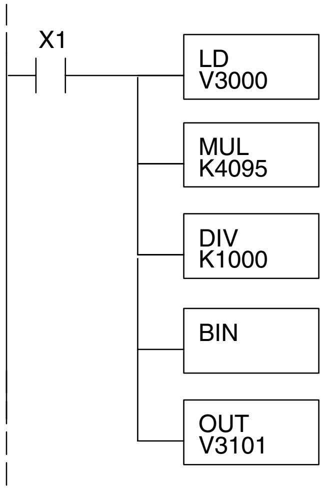
$$A = 10U \frac{4095}{10(H - L)}$$

$$A = 494 \frac{4095}{1000 - 0}$$

$$A = 2023$$

Here is 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 all eight channels if you are 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary.

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 slot of your analog module. See Appendix A for additional addresses available in the D4–450 CPU.

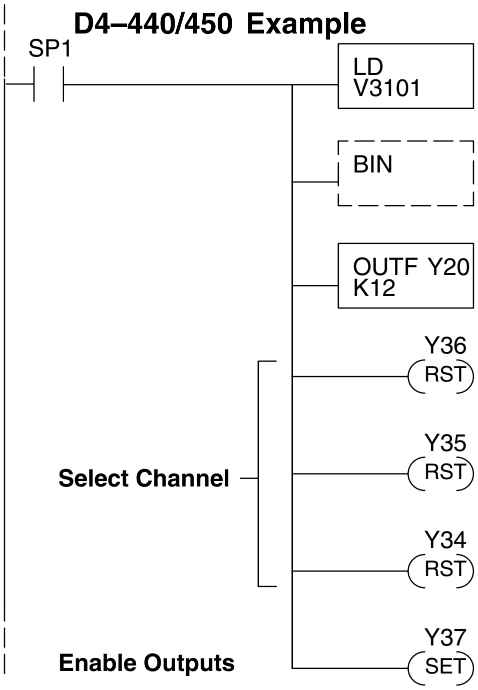
V-Memory Register Addresses for 16-Point Output (Y) Locations										
Y	000	020	040	060	100	120	140	160	200	220
V	40500	40501	40502	40503	40504	40505	40506	40507	40510	40511
Y	240	260	300	320	340	360	400	420	440	460
V	40512	40513	40514	40515	40516	40517	40520	40521	40522	40523

Sending Data to One Channel

The following programs show you how to update a single channel. Notice the D4–430 CPU requires a slightly different program than the D4–440/D4–450 CPUs. Since the D4–430 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. These examples assume you already have the data loaded in V3101.

☒	✓	✓
430	440	450

D4–440/450 Example



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.

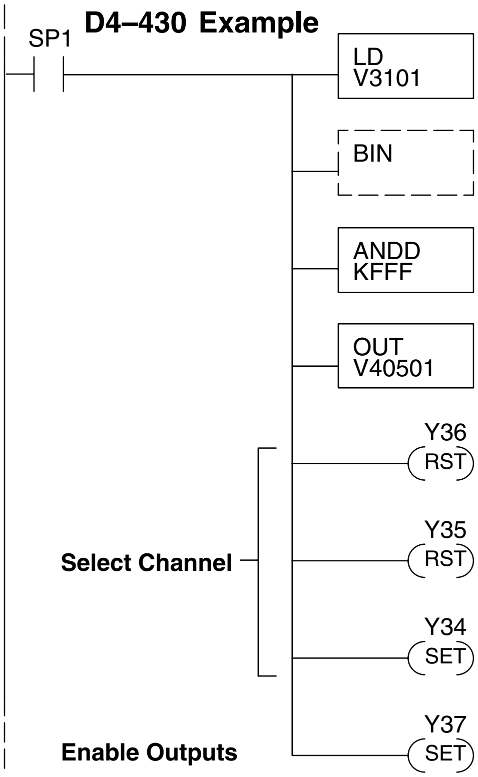
Turn Y36, Y35, and Y34 off to update Channel 1.

Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels.

✓	✓	✓
430	440	450

D4–430 Example



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 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.

Turn Y36, Y35, and Y34 off to update Channel 1.

Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels.

Sequencing the Channel Updates

The next four 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 eight channels over eight 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 C10 as index numbers corresponding to the channel updated on any particular scan. At the end of each scan, only one control relay C1 through C10 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 2 with D4–430 CPUs. Either example will work with D4–440 or D4–450 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 4 with D4–430 CPUs. Either example will work with D4–440 or D4–450 CPUs.

In the fifth example, we show you how you can update all eight channels in the same scan with D4–440 and D4–450 CPUs. However, this can increase the scan time and you may not always need to update all eight channels on every scan.

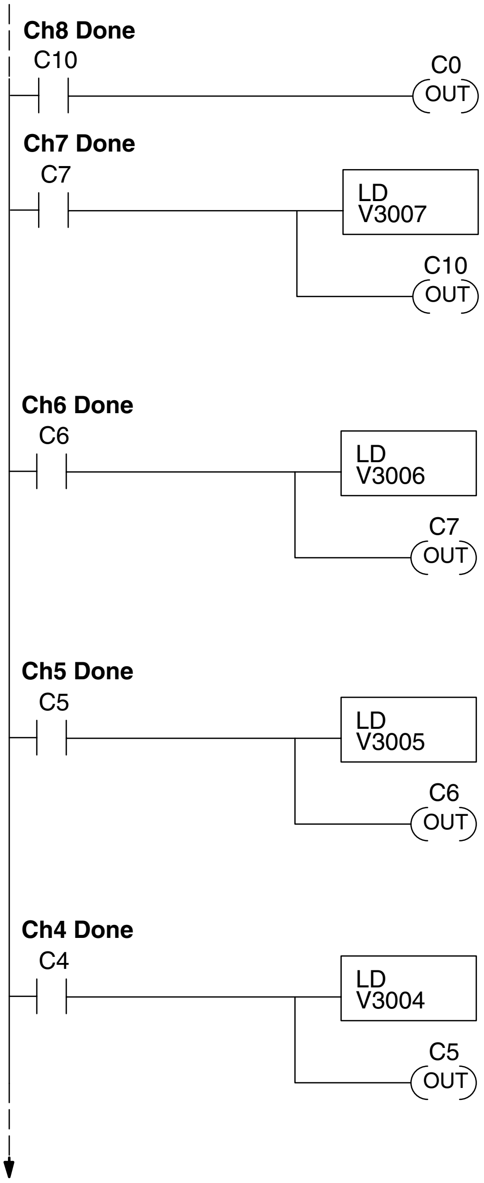
In the last example, we show you how you can update a single channel during the scan with D4–440 and D4–450 CPUs using the Immediate instructions.

Sequencing Example 1, D4–440/450

✕	✓	✓
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 have already loaded the data according to the following table. It is important to use the rungs in the order shown for the program to work. This example will not work with D4–430 CPUs.

V-Memory Locations for Output Data in Examples 1 and 2								
Channel Number	1	2	3	4	5	6	7	8
V–Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007



When channel 8 is updated, C0 restarts the update sequence.

When channel 7 has been updated, this rung loads the data for channel 8 into the accumulator.

Turning on C10 triggers the channel update (see the channel select rungs).

When channel 6 has been updated, this rung loads the data for channel 7 into the accumulator.

Turning on C7 triggers the channel update (see the channel select rungs).

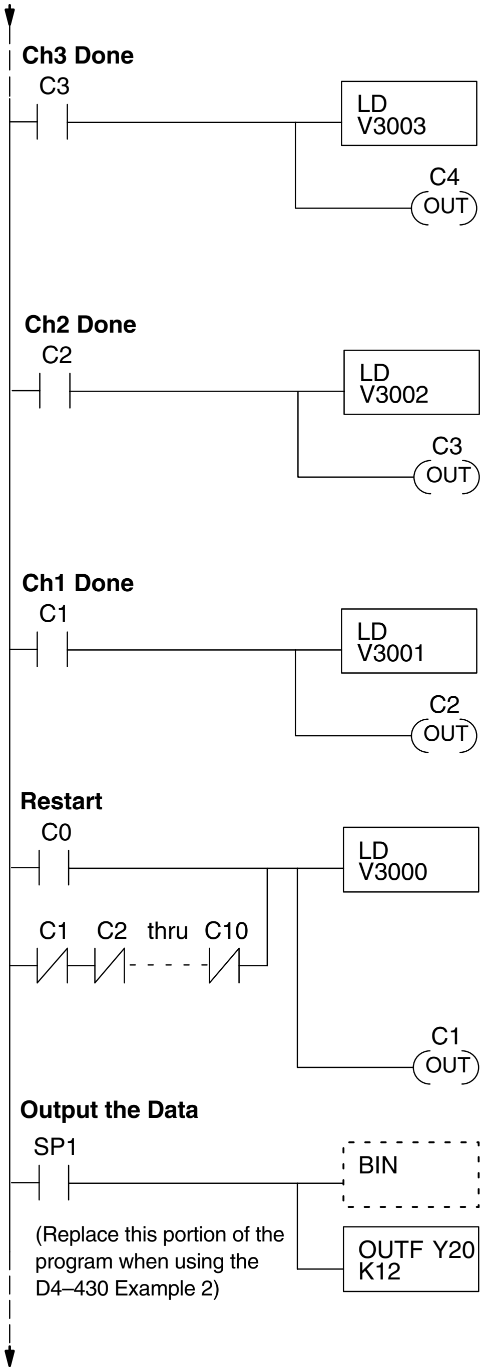
When channel 5 has been updated, this rung loads the data for channel 6 into the accumulator.

Turning on C6 triggers the channel update (see the channel select rungs).

When channel 4 has been updated, this rung loads the data for channel 5 into the accumulator.

Turning on C5 triggers the channel update (see the channel select rungs).

Example 1 Continued


Ch3 Done
C3
LD
V3003
C4
OUT
Ch2 Done
C2
LD
V3002
C3
OUT
Ch1 Done
C1
LD
V3001
C2
OUT
Restart
C0
LD
V3000
C1 C2 thru C10
C1
OUT
Output the Data
SP1
BIN
OUTF Y20
K12
(Replace this portion of the program when using the D4–430 Example 2)

When channel 3 has been updated, this rung loads the data for channel 4 into the accumulator.

Turning on C4 triggers the channel update (see the channel select rungs).

When channel 2 has been updated, this rung loads the data for channel 3 into the accumulator.

Turning on C3 triggers the channel update (see the channel select rungs).

When channel 1 has been updated, this rung loads the data for channel 2 into the accumulator.

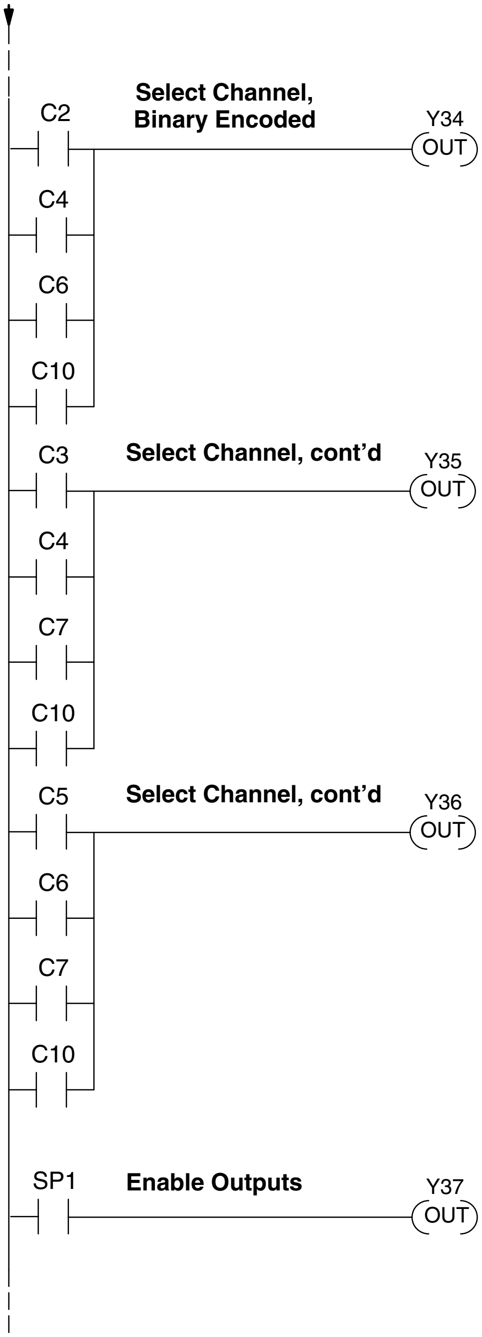
Turning on C2 triggers the channel update (see the channel select rungs).

This rung loads the data for channel 1 into the accumulator. C0 restarts the sequence after channel 8 is done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C1 through C10 are off (all eight contacts not shown here due to space constraints). In this case, we also start the sequence with channel 1.

Turning on C1 triggers the channel update (see the channel select rungs).

This rung converts the accumulator data for all channels (one per scan) 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.

Example 1 Continued


C2
Select Channel,
Binary Encoded
Y34
OUT
C4
C6
C10
C3
Select Channel, cont'd
Y35
OUT
C4
C7
C10
C5
Select Channel, cont'd
Y36
OUT
C6
C7
C10
SP1
Enable Outputs
Y37
OUT

Set Y36, Y35, and Y34 to the binary code which selects the output channel 1 through 8, based on the control relay status.

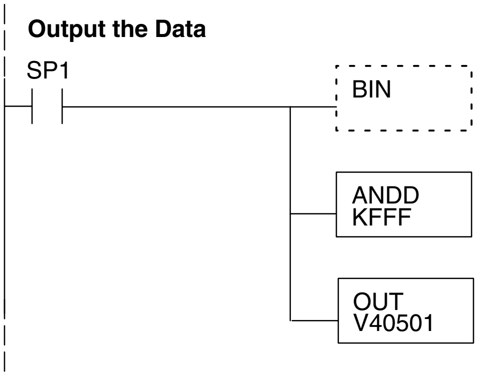
CR(on)	Y36	Y35	Y34	Channels
C1	Off	Off	Off	Ch. 1
C2	Off	Off	On	Ch. 2
C3	Off	On	Off	Ch. 3
C4	Off	On	On	Ch. 4
C5	On	Off	Off	Ch. 5
C6	On	Off	On	Ch. 6
C7	On	On	Off	Ch. 7
C10	On	On	On	Ch. 8

Enable all channels. SP1 is always on.

Sequencing Example 2, D4–430

✓	✓	✓
430	440	450

Since the D4–430 does not support the OUTF instruction, the previous program must be modified to make sure the channel select bits or the output enable bits are not accidentally changed by the data in the accumulator. Replace the “Output the Data” rung in the middle of Example 1 with the new rung below. Be sure to retain the original order of the rungs shown in Example 1 for the program to work. This example will also work with D4–440 and D4–450 CPUs.



This rung converts the accumulator data for channels 1 through 8 (one per scan) 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 for the first bank of 8 channels, but the actual value depends on the location of the module in your application.

Sequencing Example 3, D4–440/450

☒ ☑ ☑
430 440 450

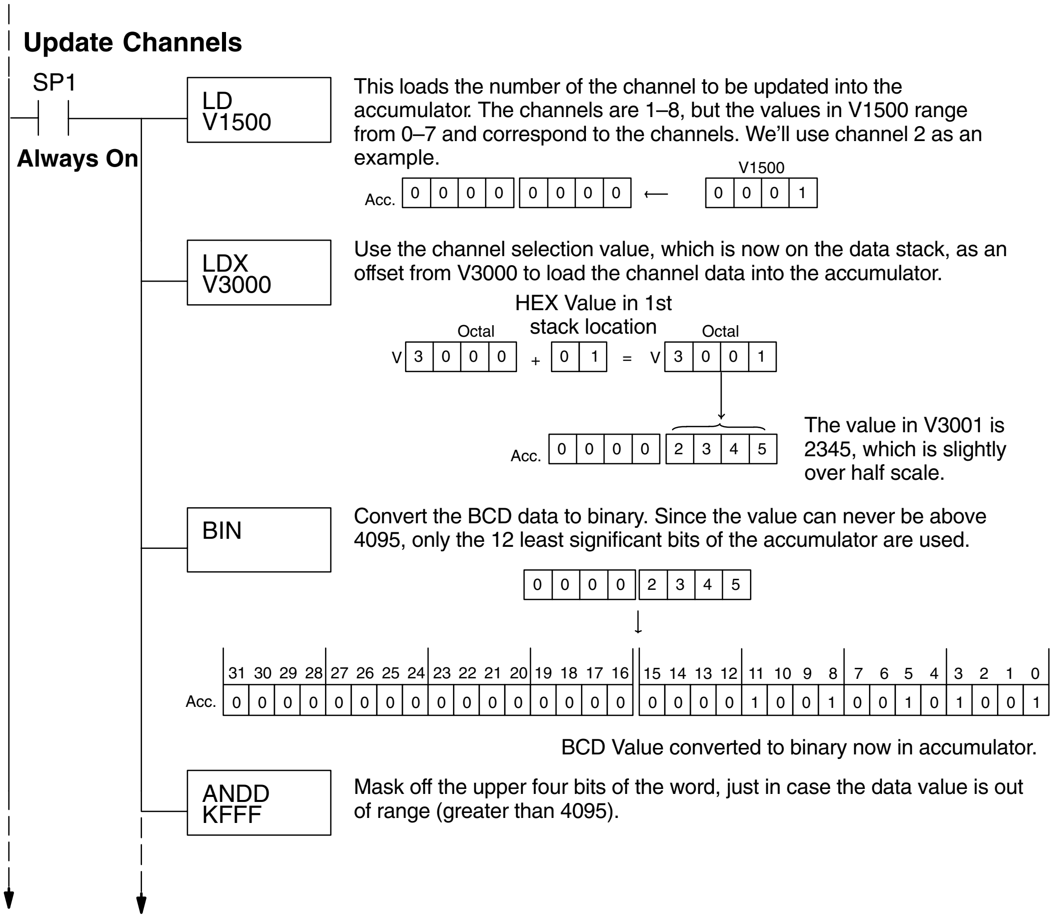
The following program example shows how to send digital values to the module when you have more than one channel. This example works only for D4–440 and D4–450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 3

Channel Number	1	2	3	4	5	6	7	8
V-Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007

The channel index is stored in V1500. It varies from 0 to 7, pointing to channels as shown: 0 – Ch. 1, 1 – Ch. 2, and 7 – Ch. 8. This example assumes V1500 is initialized to “0” earlier in the program.

This example program updates one channel during each scan. The program comments for this portion also shows the accumulator status at each step. The last portion of the program increments the channel index number and resets it after eight scans.


Update Channels
SP1
Always On
LD
V1500
Acc. 0 0 0 0 0 0 0 0 ⟵ V1500 0 0 0 1
LDX
V3000
HEX Value in 1st stack location
Octal
V 3 0 0 0 + 0 1 = V 3 0 0 1
Octal
Acc. 0 0 0 0 2 3 4 5
BIN
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
ANDD
KFFF

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

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

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.

BCD Value converted to binary now in accumulator.

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

Example 3 Continued

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

ORD
K8000

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 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	1	0	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	1	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
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	1	0	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

OUTF Y20
K16

Send the lower 16 bits stored in the accumulator to the analog module. The lowest 12 bits contain the analog data. Bits 12, 13, and 14 are the channel selection bits. Bit 15 is the Output Enable bit.

Increment Channel Index

INCB
V1500

Increment the channel index value. This allows the logic to cycle through all eight channels.

Reset Channel Index

V1500 K8
─┤ = ├─

LD
K0

When channel 8 has been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1).

OUT
V1500

Sequencing Example 4, D4–430

✓ ✓ ✓
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 works for D4–430, D4–440 or D4–450 CPUs. It assumes you are using the following data locations.

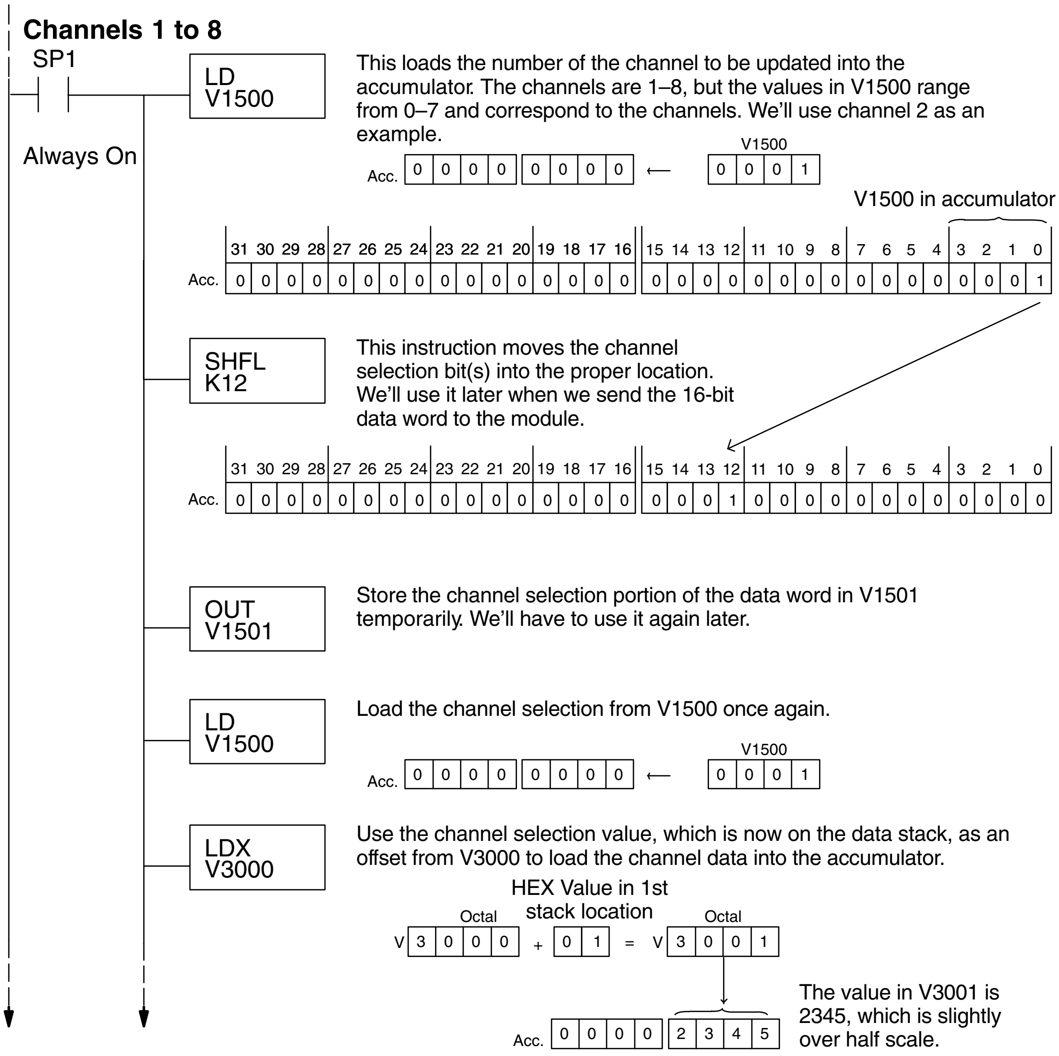
V-Memory Locations for Output Data in Example 4

Channel Number	1	2	3	4	5	6	7	8
V–Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007

The channel index is stored in V1500. It varies from 0 to 7, pointing to channels as shown: 0 – Ch. 1, 1 – Ch. 2, and 7 – Ch. 8. This example assumes V1500 is initialized to “0” earlier in the program.

The first portion of the program updates one channel during each scan. The program comments show the accumulator status at each step. The last portion of the program increments the channel index number and resets it after eight scans.

Channels 1 to 8


SP1
Always On
LD V1500
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
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
OUT V1501
LD V1500
Acc. 0 0 0 0 0 0 0 0 ← V1500 0 0 0 1
LDX V3000
Octal
V 3 0 0 0 + 0 1 = V 3 0 0 1
Octal
Acc. 0 0 0 0 2 3 4 5

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

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

Load the channel selection from V1500 once again.

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

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

Example 4 Continued

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.

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 send 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 K8000

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 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	1	0	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.

Increment Channel Index

INCB V1500

Increment the channel index value. This allows the logic to cycle through all eight channels.

Reset Channel Index

V1500K8 ─┤ = ├─

LD K0

When channel 8 has been updated, then reset the channel index memory location to 0 (remember, 0 represents channel 1).

OUT V1500

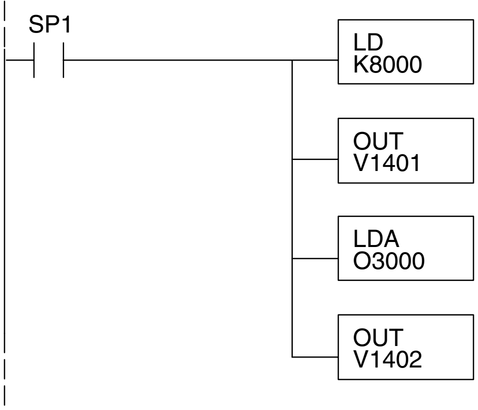
Updating all Channels in a Single Scan, D4–440/450

☒ ✓ ✓
430 440 450

By using the Immediate instructions found in the D4–440 and D4–450 CPUs, you can easily update all eight channels in a single scan. Before choosing this method, remember it slows CPU scan time (approximately 12 mS). 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 V3000 to V3007 for channels 1 to 7 respectively.

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

Initialize the Immediate Analog Output Pointers



The LD instruction loads the data into the accumulator. The constant K8000 represents channel 1 selected, Output Enable bit is on.

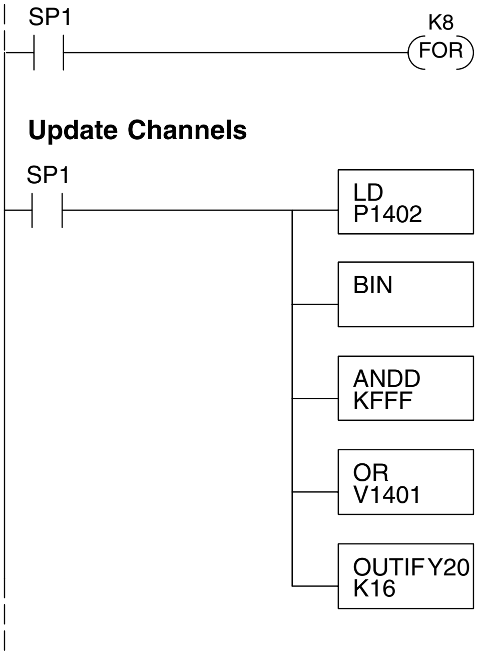
We store the channel index pointer in V1401. The program increments this in each pass, resetting it after eight scans.

The Load Address instruction takes the octal 3000 and converts it to hex, then puts it in the accumulator. V3000 is the location which contains the data for channel 1.

V1402 contains the pointer for channel 1 to 8 data.

The following FOR–NEXT loop updates all eight channels in a single scan.

FOR–NEXT Loop



The following FOR–NEXT loop updates channels 1 through 8. The valid range for the constant K is 1 to 8 for the module. For example, a value of 1 will update channel 1.

Load the analog output value into the accumulator. V3000 = Ch. 1, V3001 = Ch. 2, ... V3007 = Ch. 8. The number at V1402 is a pointer to the address of the value.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y36, Y35, and Y34 with the data in the accumulator.

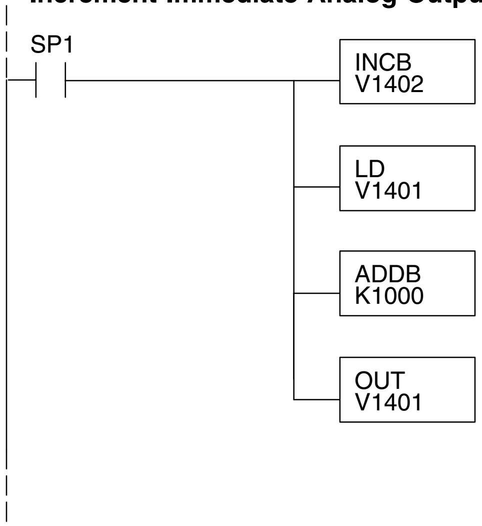
Immediately write the lower 16 bits of the accumulator to the module's lower data word. This updates channels 1 through 8 during the FOR–NEXT loop.

Updating all Channels in a Single Scan, Continued

☒ ✓ ✓
430 440 450

Now we increment the Immediate Analog Output Pointers for the channel, before the next pass through the FOR–NEXT loop.

Increment Immediate Analog Output Pointers



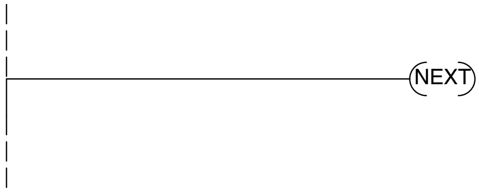
Increment the analog output data pointer.

Load the curent channel index count into the accumulator.

Add 1000 hex to the channel index value, since the channel select field is in the most significant four bits.

Save the incremented channel index value for the next pass through the FOR–NEXT loop.

Here is the end of the FOR–NEXT loop.

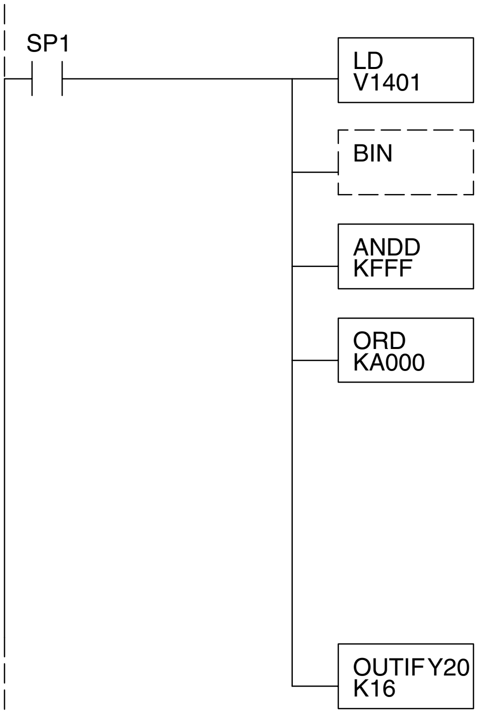


Execute the loop above the number of times specified in the FOR instruction.

Updating a Single Channel During a Scan, D4–440/450

☒	✓	✓
430	440	450

You can also update just a single channel during a ladder logic scan by using the Immediate instructions found in the D4–440 and D4–450 CPUs. By removing the FOR–NEXT rungs and a couple of other rungs, we create the example below. This example assumes the data is already loaded in V1401.



Load the analog output value for the channel from V1401 into the accumulator.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits (channel index) with the desired channel with the data in the accumulator. We chose channel 3 here.

Index	Channel
8000	Ch. 1
9000	Ch. 2
A000	Ch. 3
B000	Ch. 4
C000	Ch. 5
D000	Ch. 6
E000	Ch. 7
F000	Ch. 8

Immediately write the lower 16 bits of the accumulator to the module's lower data word. This updates the selected channel.

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.

Range	If you know the digital value ...	If you know the analog signal level ...
0 to 5VDC	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}A$
0 to 10VDC	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}A$

For example, if you need a 3V signal level with the module set for 0–5V, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

$$D = \frac{4095}{5}A$$

$$D = \frac{4095}{5}(3V)$$

$$D = (819)(3)$$

$$D = 2457$$

F4–16DA–2 16-Channel Analog Voltage Output

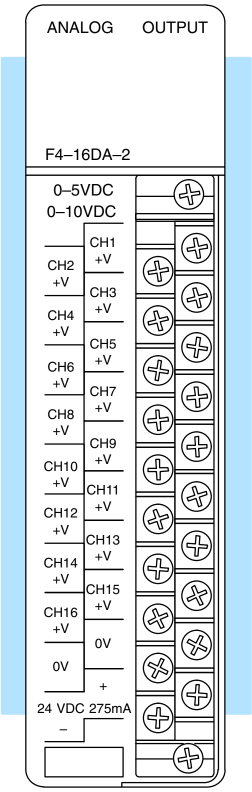
In This Chapter. . . .

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

Module Specifications

The F4–16DA–2 Analog Voltge Output module provides several features and benefits.

- It provides sixteen channels of 0–5VDC or 0–10VDC single ended voltage outputs, or a combination of 8 channels of 0–5VDC and 8 channels of 0–10VDC outputs.
- 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.
- From two to sixteen analog outputs may be updated in one CPU scan (D4–440 and D4–450 CPUs only).


ANALOG OUTPUT
F4–16DA–2
0–5VDC
0–10VDC
CH1 +V
CH2 +V
CH3 +V
CH4 +V
CH5 +V
CH6 +V
CH7 +V
CH8 +V
CH9 +V
CH10 +V
CH11 +V
CH12 +V
CH13 +V
CH14 +V
CH15 +V
CH16 +V
0V
0V
+
24 VDC 275mA
–

Analog Output Configuration Requirements

The F4–16DA–2 Analog Output module requires 32 discrete output 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 output points.
- For remote I/O systems, the available power budget and remote I/O points.

The following tables provide the specifications for the F4–16DA–2 Analog Output Module. Review these specifications to ensure the module meets your application requirements.

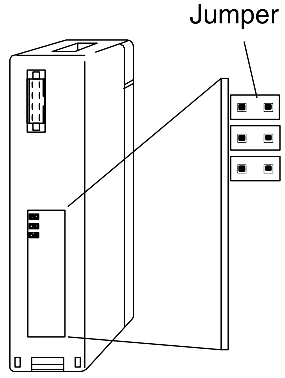
Output Specifications

Number of Channels	16, single ended (one common)
Output Range	0–5VDC, 0–10VDC, or 8 channels of 0–5VDC and 8 channels of 0–10VDC
Resolution	12 bit (1 in 4095)
Output Type	Voltage sourcing 10mA max.
External Load	1kΩ maximum / 10kΩ minimum (for example: 10 volts at 1kΩ = 10mA load; 10 volts at 10kΩ = 1mA load)
Crosstalk	–70 dB, ±1 count maximum
Linearity Error (end-to-end) and Relative Accuracy	±1 count maximum (10VDC at 25˚C)
Full Scale Calibration Error (offset error included)	±6 counts maximum (10VDC at 25˚C)
Offset Calibration Error	±3 counts maximum (0VDC at 25˚C)
Maximum Inaccuracy	±0.2% at 25˚C (77˚F) ±0.4% at 0˚ to 60˚C (32˚ to140˚F)
Conversion Time	400 µS maximum, for full scale change 4.5 to 9 mS for digital output to analog out

General Module Specifications

Digital Output Points Required	32 point (Y) outputs, 2 sets each of 12 bits binary data, 3 bits channel select, 1 bit output enable
Power Budget Requirement	80 mA @ 5 VDC (supplied by base)
External Power Supply	21.6 – 26.4 VDC, 275 mA. max. class 2
Accuracy vs. Temperature	±57 ppm / ˚C full scale calibration range (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



Before installing and wiring the module, you may need to change the internal jumper settings. The module has three pairs of terminals, located in the open cutout at the rear of the housing. Two jumpers are supplied. These jumper settings determine whether the outputs are 16 channels of 0–5VDC (setting #1 below), 16 channels of 0–10VDC (setting #2) or eight channels of each (setting #3).

There are three possible settings. When a jumper is removed, store the jumper by placing it over one terminal as shown below to prevent losing it.

Setting #1	**Setting #2**	**Setting #3**
16 channels 0–5VDC (factory setting)	16 channels 0–10VDC	8 channels (ch.1–8) at 0–10VDC and 8 channels (ch. 9–16) at 0–5VDC
Installed	Removed	Removed
Installed	Installed	Removed
Removed	Removed	Installed

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.
- 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 F4–16DA–2 requires a field-side power supply. The module requires 21.6 to 26.4 VDC, Class 2, at 275mA max. current.

The D4–430/440/450 CPUs, D4–RS Remote I/O Controller, H4–EBC, and D4–EX Expansion Units have built-in 24 VDC power supplies that provide up to 400mA of current. You may use one of these instead of a separate supply if there is only one F4–16DA–2 module in use.

Load Requirements

Each channel in use must have a load impedance within a range of 1kΩ to 10kΩ. Unused channels must be left disconnected.

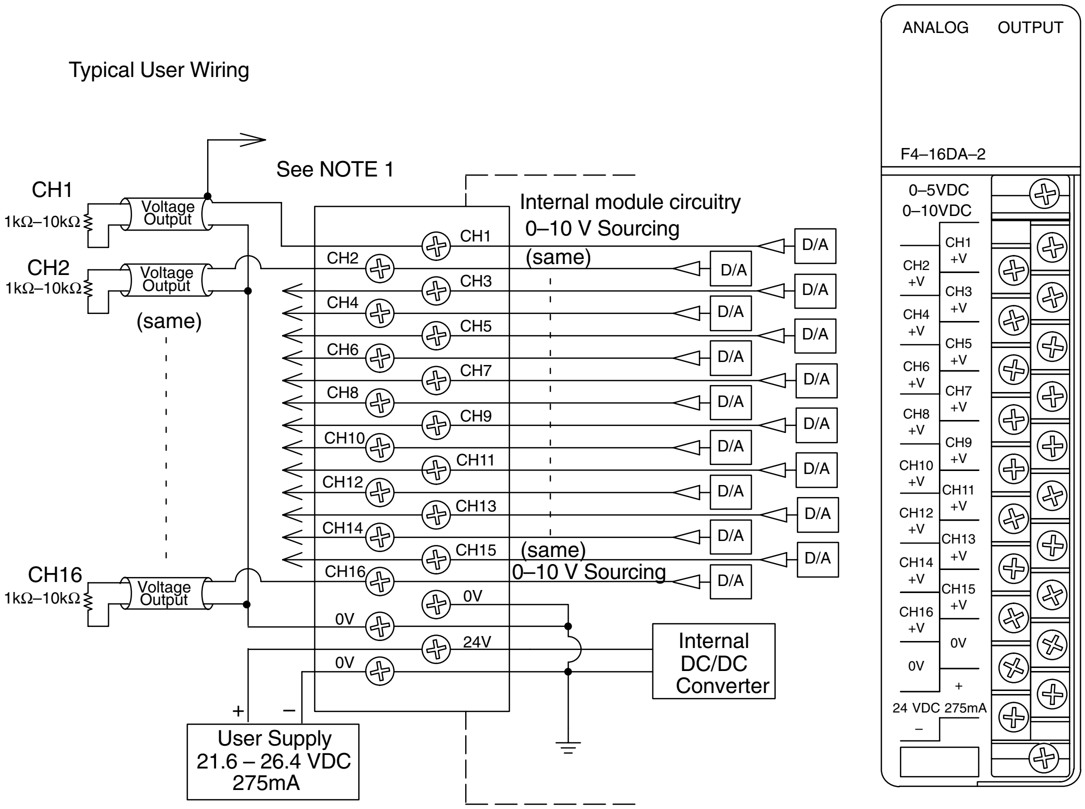
WARNING: If you are using the 24 VDC base power supply, make sure you calculate the power budget. Exceeding the power budget can cause unpredictable system operation that can lead to a risk of personal injury or damage to equipment.

Removable Connector

The F4–16DA–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 User Power Supply at the module terminal block.

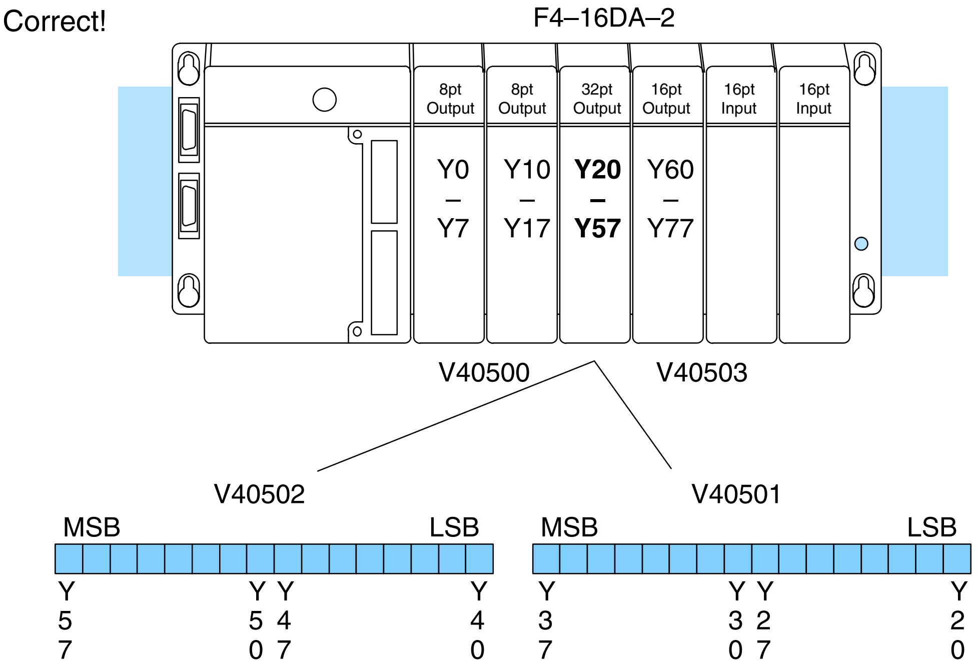


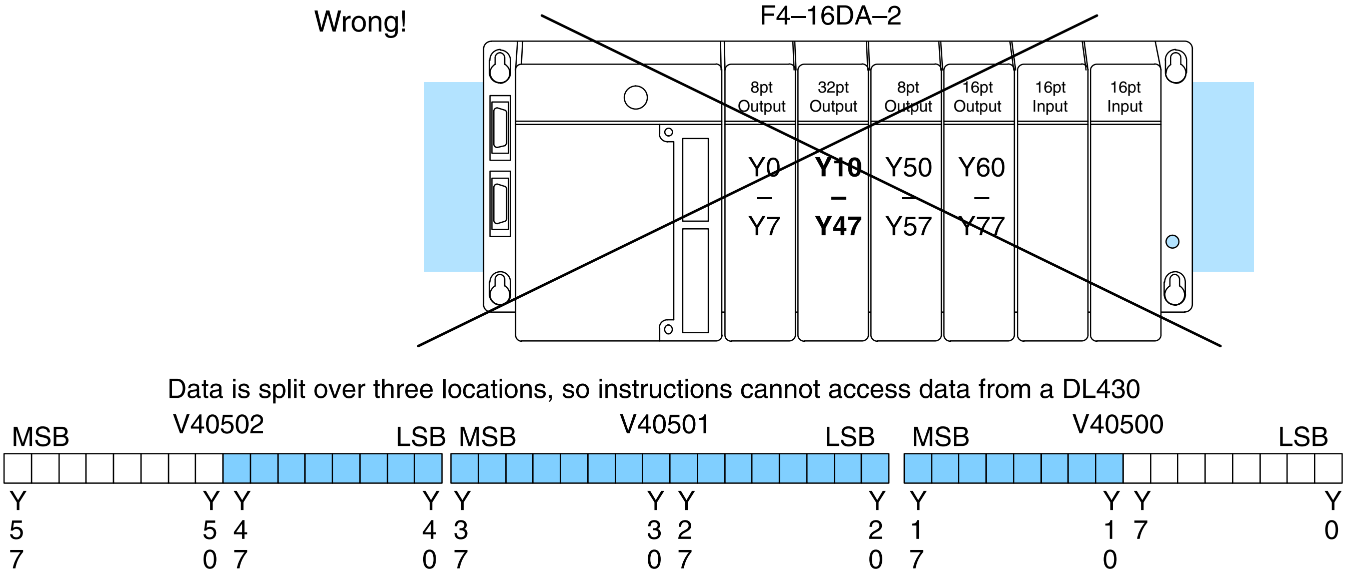
Typical User Wiring
See NOTE 1
CH1
1kΩ–10kΩ
Voltage Output
CH2
1kΩ–10kΩ
Voltage Output
(same)
CH16
1kΩ–10kΩ
Voltage Output
Internal module circuitry
0–10 V Sourcing
(same)
CH1
CH2
CH3
CH4
CH5
CH6
CH7
CH8
CH9
CH10
CH11
CH12
CH13
CH14
CH15
CH16
0V
0V
24V
0V
(same)
0–10 V Sourcing
D/A
Internal DC/DC Converter
+
−
User Supply 21.6 – 26.4 VDC 275mA
ANALOG OUTPUT
F4–16DA–2
0–5VDC
0–10VDC
CH1 +V
CH2 +V
CH3 +V
CH4 +V
CH5 +V
CH6 +V
CH7 +V
CH8 +V
CH9 +V
CH10 +V
CH11 +V
CH12 +V
CH13 +V
CH14 +V
CH15 +V
CH16 +V
0V
0V
+
24 VDC 275mA
–

Module Operation

D4–430 Special Requirements

Even though the module can be placed in any slot, it is important to examine the configuration if you're using a D4–430 CPU. As you'll 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 can't access the data.


Correct!
F4–16DA–2
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
MSB
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20

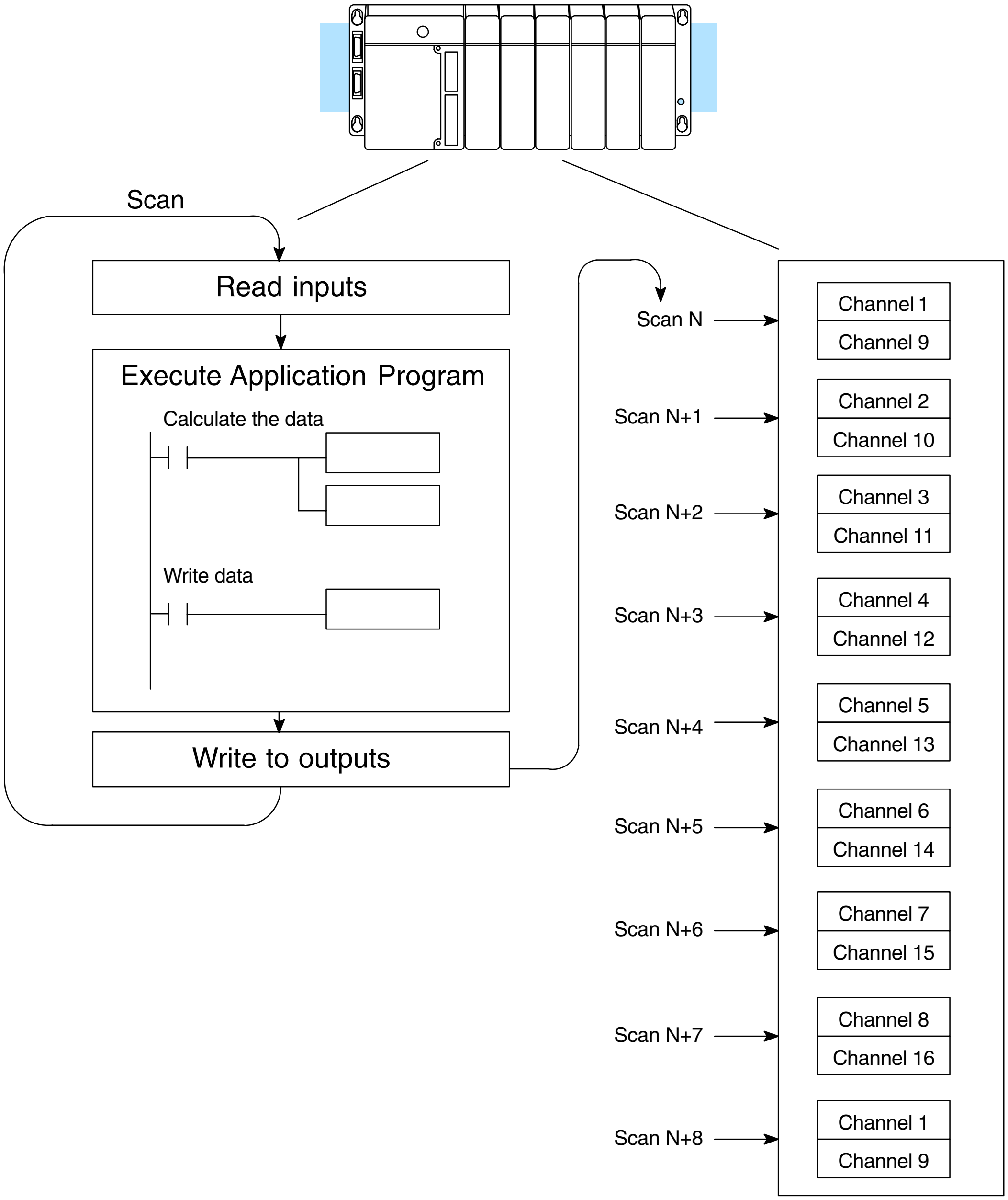

Wrong!
F4–16DA–2
8pt Output
32pt Output
8pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y47
Y50 – Y57
Y60 – Y77
MSB
V40502
LSB
MSB
V40501
LSB
MSB
V40500
LSB
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20
Y17
Y10
Y7
Y0

Data is split over three locations, so instructions cannot access data from a DL430

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 F4–16DA–2 module allows you to update the channels in any order. Your control program determines which channels get updated on any given scan. The channels are organized as two banks of outputs, eight channels each. A typical ladder program will update one channel in each bank, or two channels per CPU scan. So, all sixteen channels can be updated every eight scans. With a D4–440 or D4–450 CPU, you can use Immediate instructions to update all sixteen channels in the same scan (we'll show you how to do this later).

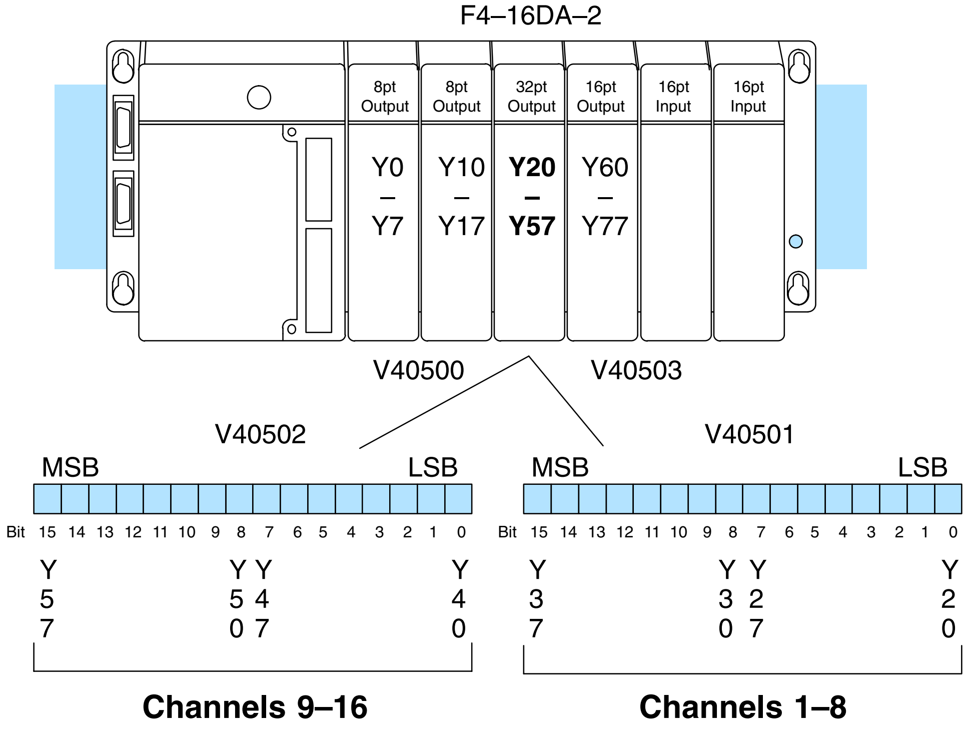

Scan
Read inputs
Execute Application Program
Calculate the data
Write data
Write to outputs
Scan N
Scan N+1
Scan N+2
Scan N+3
Scan N+4
Scan N+5
Scan N+6
Scan N+7
Scan N+8
Channel 1
Channel 9
Channel 2
Channel 10
Channel 3
Channel 11
Channel 4
Channel 12
Channel 5
Channel 13
Channel 6
Channel 14
Channel 7
Channel 15
Channel 8
Channel 16
Channel 1
Channel 9

Output Bit Assignments

You may recall the F4–16DA–2 module requires 32 discrete output points from the CPU. These points provide:

- The digital representation of two analog signals per scan.
- Identification of the two channels that are to receive the data.
- Output enable control for all channels.

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

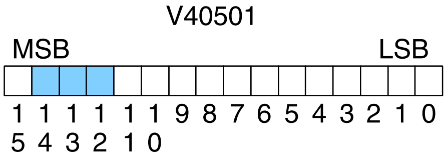

F4–16DA–2
8pt Output
8pt Output
32pt Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
MSB
LSB
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Y57
Y50
Y47
Y40
Y37
Y30
Y27
Y20
Channels 9–16
Channels 1–8

Within these V-memory locations the individual bits represent specific information about the channels selected and the analog signals.

Channel Select Bits

Output bits 12, 13, and 14 of both data words are the channel select outputs. They are binary encoded to select the channel that will be updated with the data. The bits of the lower data word select channels 1 to 8 as follows.

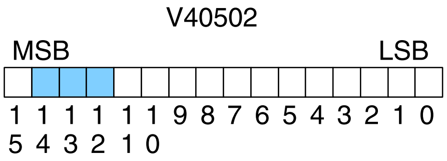
Bit 14	Bit 13	Bit 12	Channel
Off	Off	Off	1
Off	Off	On	2
Off	On	Off	3
Off	On	On	4
On	Off	Off	5
On	Off	On	6
On	On	Off	7
On	On	On	8


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

🟦 – Channel Select Bits, channels 1 through 8

The bits of the upper data word select channels 9 to 16 as follows.

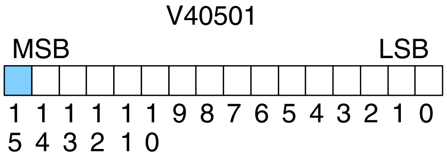
Bit 14	Bit 13	Bit 12	Channel
Off	Off	Off	9
Off	Off	On	10
Off	On	Off	11
Off	On	On	12
On	Off	Off	13
On	Off	On	14
On	On	Off	15
On	On	On	16


V40502
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

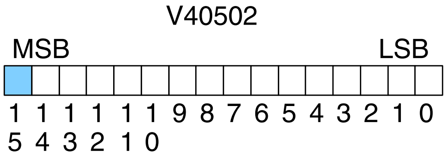
🟦 – Channel Select Bits, channels 9 through 16

Output Enable Bits

Output bit 15 is the Output Enable control bit in each data word for all eight channels of each bank. When either bit is off, the corresponding group of eight channel output voltages decrease to their lowest level, which is 0V for connected loads. Disabling the outputs also clears all eight output data registers for each bank. To resume analog output levels, first the appropriate Output Enable control bit must turn on. Then, the CPU must write new data to each channel to restore the output voltage for that channel.


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

🟦 – Output Enable Bit, channels 1 through 8


V40502
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

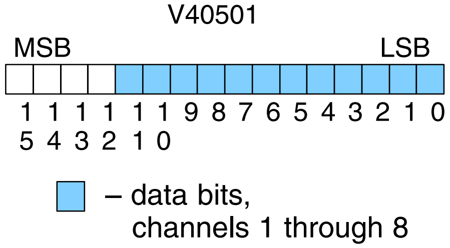
🟦 – Output Enable Bit, channels 9 through 16

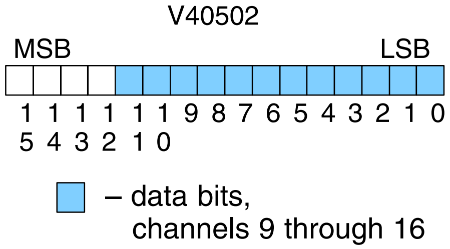
OFF = Disable (and clear)
ON = Enable

Analog Data Bits

The first twelve bits of both V-memory locations represent the analog data in binary format.

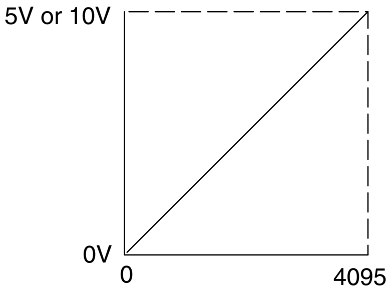
Bit	Value	Bit	Value
0	1	6	64
1	2	7	128
2	4	8	256
3	8	9	512
4	16	10	1024
5	32	11	2048


V40501
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– data bits,
channels 1 through 8


V40502
MSB
LSB
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
– data bits,
channels 9 through 16

Module Resolution

Since the module has 12-bit resolution, the analog signal is made of 4096 counts ranging from 0 – 4095 (2^{12}). For the 0 to 5V scale, sending a 0 produces a 0V signal, and 4095 gives a 5V signal. This is equivalent to a binary value of 0000 0000 0000 to 1111 1111 1111, or 000 to FFF hexadecimal. The graph to the right shows the linear relationship between the data value and output signal level.



Each count can also be expressed in terms of the signal level by using the equation shown. The following table shows the smallest signal change that occurs when the digital value is increased by 1 LSB.

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

H = high limit of the signal range

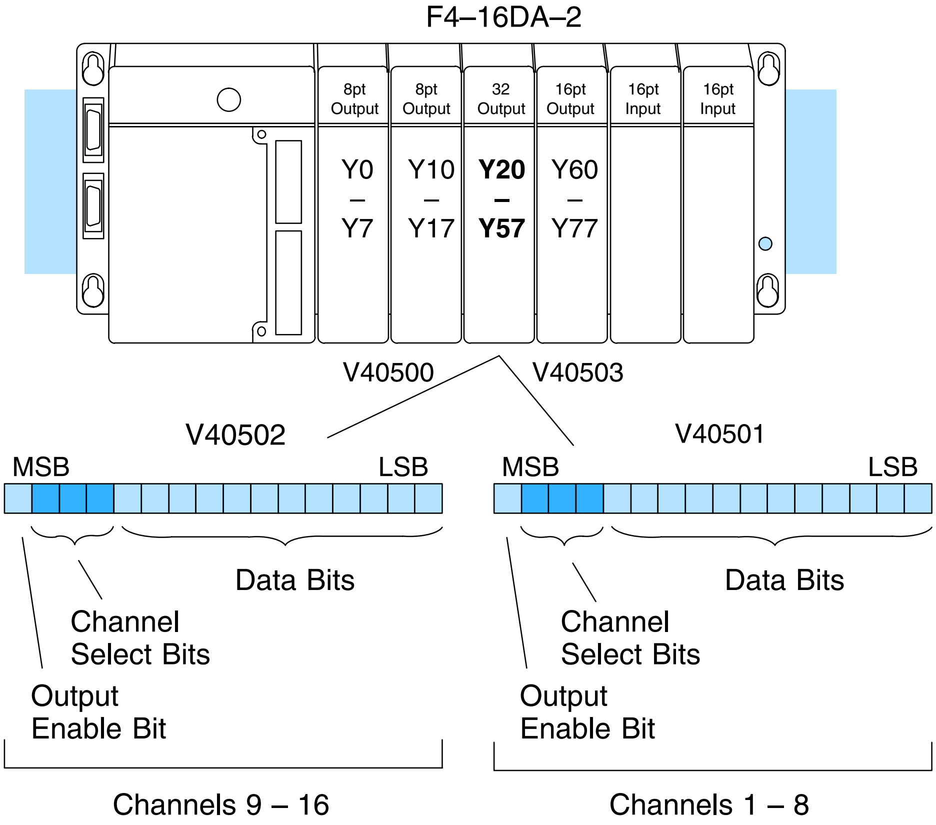
L = low limit of the signal range

Signal Range	Span (H – L)	Divide By	Smallest Change
0 to 5VDC	5VDC	4095	1.221mV
0 to 10VDC	10VDC	4095	2.442mV

Writing the Control Program

Update Any Channel

As mentioned earlier, you can update two channels 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 select outputs to determine which channels get updated (more on this later).


F4–16DA–2
8pt Output
8pt Output
32 Output
16pt Output
16pt Input
16pt Input
Y0 – Y7
Y10 – Y17
Y20 – Y57
Y60 – Y77
V40500
V40503
V40502
V40501
MSB
LSB
MSB
LSB
Data Bits
Channel Select Bits
Output Enable Bit
Data Bits
Channel Select Bits
Output Enable Bit
Channels 9 – 16
Channels 1 – 8

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.

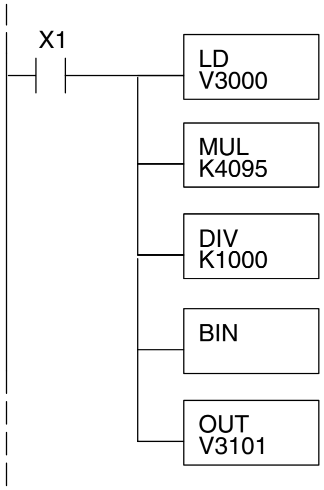
$$A = 10U \frac{4095}{10(H - L)}$$

$$A = 494 \frac{4095}{1000 - 0}$$

$$A = 2023$$

Here is 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 all sixteen 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.



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

Multiply the accumulator by 4095 (to start the conversion).

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

Convert the BCD number to binary (you must omit this step if you have converted the data elsewhere).

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 slot of your analog module. See Appendix A for additional addresses for D4–450 CPUs.

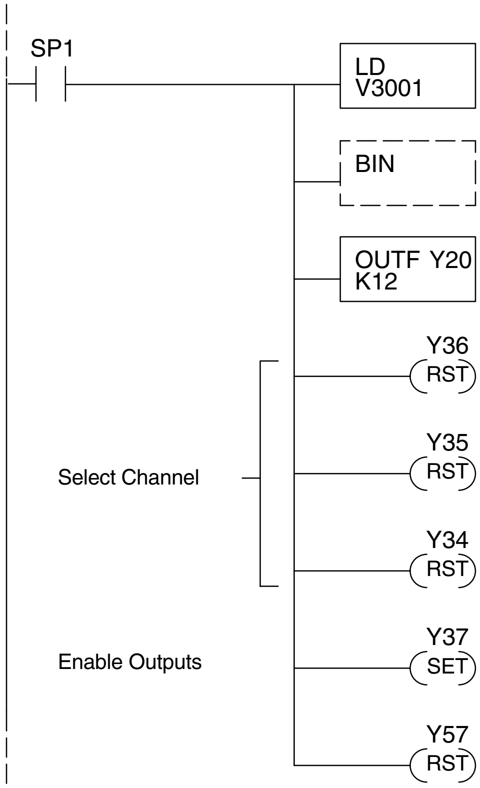
V-Memory Register Addresses for 16-Point Output (Y) Locations

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

Sending Data to One Channel, D4–440/450

☒ ✓ ✓
430 440 450

The following two programs show you how to update a single channel. Notice the D4–440/D4–450 and D4–430 CPUs use slightly different programs, although the D4–430 example will also work with D4–440 and D4–450 CPUs. Since the D4–430 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. This example assumes you already have the data loaded in V3001.



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.

Turn Y36, Y35, and Y34 off to update Channel 1.

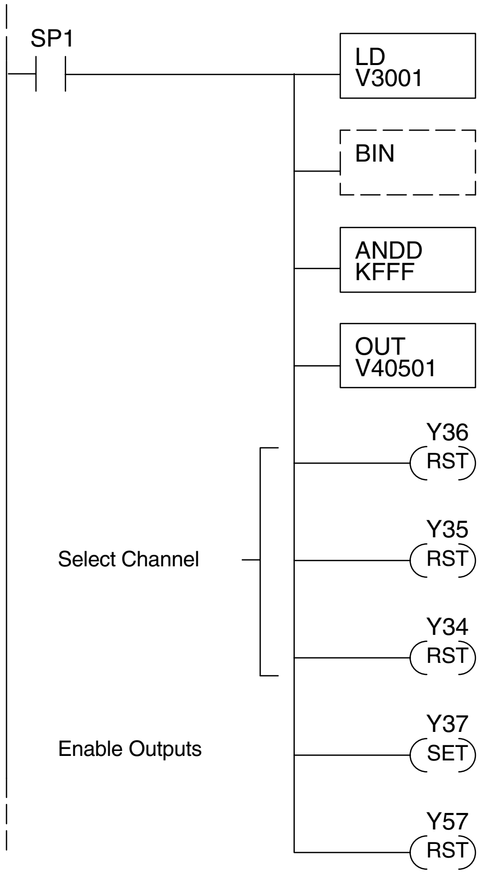
Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels of the first bank.

Turn off Y57 to disable the second bank of channels, 9 through 16.

Sending Data to One Channel, D4–430

✓	✓	✓
430	440	450



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 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.

Turn Y36, Y35, and Y34 off to update Channel 1.

Y36	Y35	Y34	Channel
Off	Off	Off	Ch. 1
Off	Off	On	Ch. 2
Off	On	Off	Ch. 3
Off	On	On	Ch. 4
On	Off	Off	Ch. 5
On	Off	On	Ch. 6
On	On	Off	Ch. 7
On	On	On	Ch. 8

Turn on Y37 to enable all eight output channels of the first bank.

Turn off Y57 to disable the second bank of channels, 9 through 16.

Sequencing the Channel Updates

The next four example programs show you how to send digital values to the module when you have more than one channel. These examples automatically update all sixteen channels over eight scans. Channel updates occur in pairs, such as channels 1 and 9, channels 2 and 10, and so on. This method uses the same binary code for the channel select bits of both groups of outputs, simplifying the programs. You may recall we stated earlier you could update all sixteen channels in the same scan. That's true, you can do that if you have a D4–440/450 CPU; however, this can increase the scan time and you may not always need to update all sixteen channels on every scan. We show you how to do this later in this chapter.

The first two sequencing examples are fairly simple and will work in most situations. We recommend these for new users. They use control relays C1 through C10 as index numbers corresponding to the pair of channels updated on any particular scan. At the end of each scan, only one control relay C1 through C10 is on. On each subsequent scan, the next control relay energizes. The channel sequencing automatically begins with channels 1 and 9 on the first scan, or after any disruption in the logic.

The last 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 words to the module.

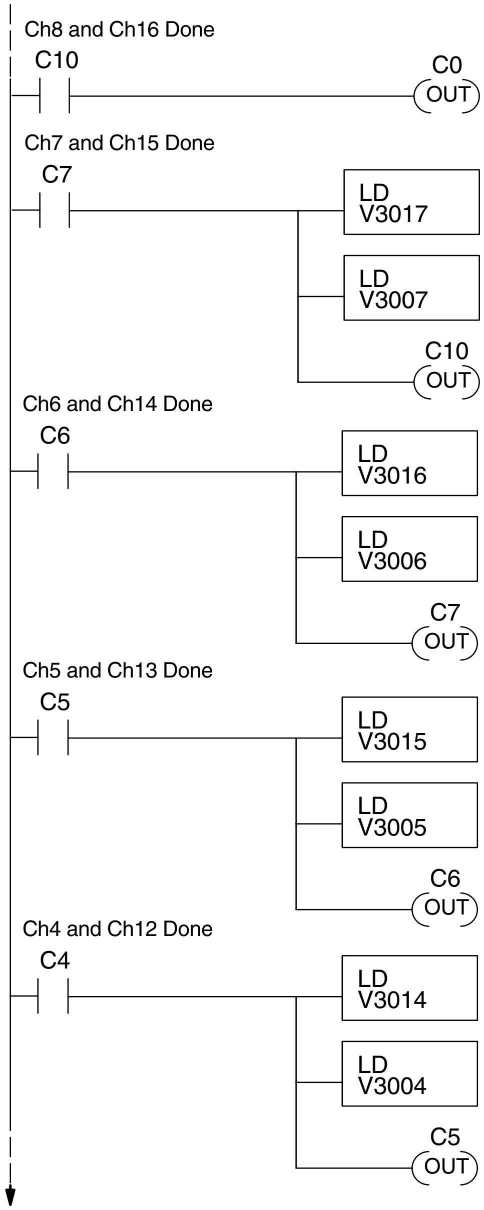
Sequencing Example 1, D4–440/450

✗ ✓ ✓
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 have already loaded the data according to the following table. It is important to use the rungs in the order shown for the program to work.

V-Memory Locations for Output Data in Examples 1 and 2

Channel Number	1	2	3	4	5	6	7	8
V-Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007
Channel Number	9	10	11	12	13	14	15	16
V-Memory Storage	3010	3011	3012	3013	3014	3015	3016	3017



When channels 8 and 16 are updated, C0 restarts the update sequence.

When channels 7 and 15 have been updated, this rung loads the data for channel 16 into the accumulator. The second LD instruction places the data for channel 8 in the accumulator and pushes channel 16’s data onto the stack.

Turning on C10 triggers the channel update (see the channel select rungs).

When channels 6 and 14 have been updated, this rung loads the data for channel 15 into the accumulator. The second LD instruction places the data for channel 7 in the accumulator and pushes channel 15’s data onto the stack.

Turning on C7 triggers the channel update (see the channel select rungs).

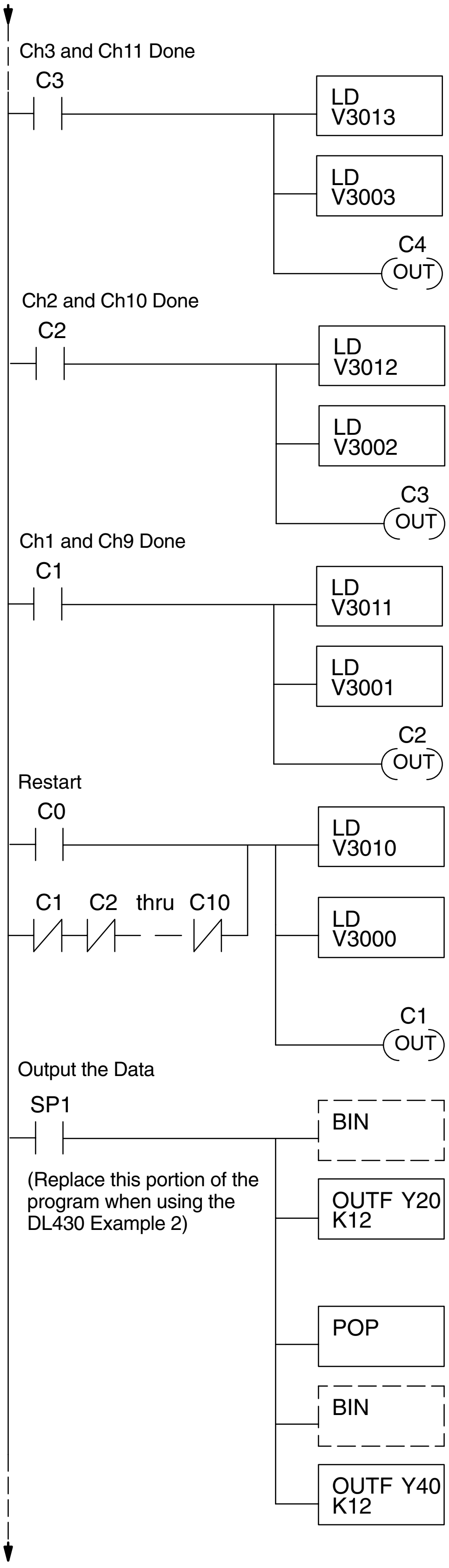
When channels 5 and 13 have been updated, this rung loads the data for channel 14 into the accumulator. The second LD instruction places the data for channel 6 in the accumulator and pushes channel 14’s data onto the stack.

Turning on C6 triggers the channel update (see the channel select rungs).

When channels 4 and 12 have been updated, this rung loads the data for channel 13 into the accumulator. The second LD instruction places the data for channel 5 in the accumulator and pushes channel 13’s data onto the stack.

Turning on C5 triggers the channel update (see the channel select rungs).

Example 1 Continued


Ch3 and Ch11 Done
C3
LD V3013
LD V3003
C4
OUT
Ch2 and Ch10 Done
C2
LD V3012
LD V3002
C3
OUT
Ch1 and Ch9 Done
C1
LD V3011
LD V3001
C2
OUT
Restart
C0
C1 C2 thru C10
LD V3010
LD V3000
C1
OUT
Output the Data
SP1
(Replace this portion of the program when using the DL430 Example 2)
BIN
OUTF Y20 K12
POP
BIN
OUTF Y40 K12

When channels 3 and 11 have been updated, this rung loads the data for channel 12 into the accumulator. The second LD instruction places the data for channel 4 in the accumulator and pushes channel 12’s data onto the stack.

Turning on C4 triggers the channel update (see the channel select rungs).

When channels 2 and 10 have been updated, this rung loads the data for channel 11 into the accumulator. The second LD instruction places the data for channel 3 in the accumulator and pushes channel 11’s data onto the stack.

Turning on C3 triggers the channel update (see the channel select rungs).

When channels 1 and 9 have been updated, this rung loads the data for channel 10 into the accumulator. The second LD instruction places the data for channel 2 in the accumulator and pushes channel 10’s data onto the stack.

Turning on C2 triggers the channel update (see the channel select rungs).

This rung loads the data for channel 9 into the accumulator. The second LD instruction places the data for channel 1 in the accumulator and pushes channel 9’s data onto the stack. C0 restarts the sequence after channels 8 and 16 are done (see the top rung). The first scan or any interruption in control relay sequencing is detected when control relays C1 through C10 are off (all eight contacts are not shown here due to space constraints). In this case, we also start the sequence with channels 1 and 9.

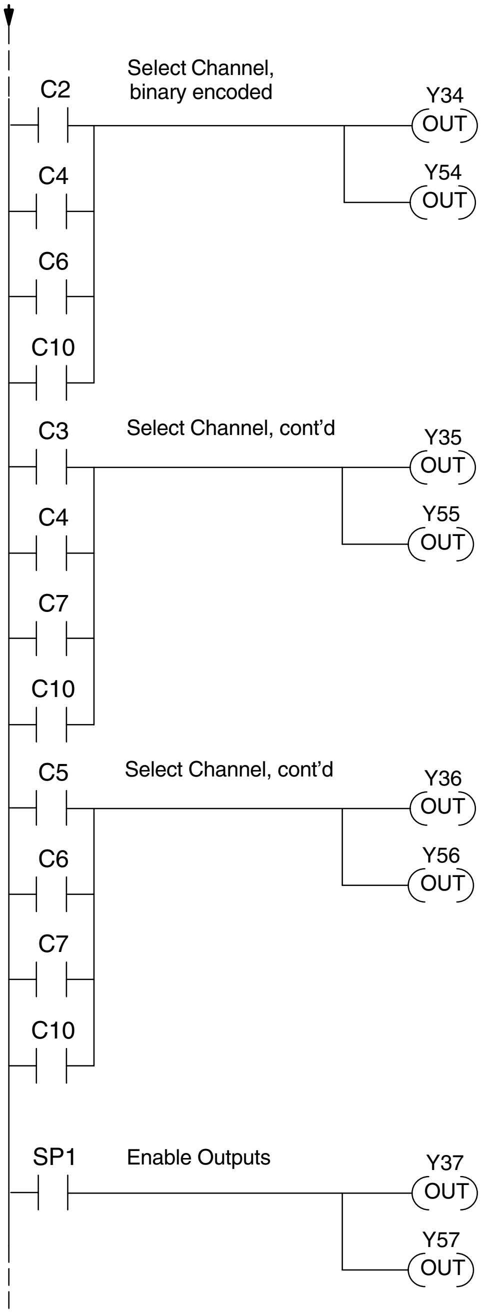
Turning on C1 triggers the channel update (see the channel select rungs).

This rung converts the accumulator data for channels 1 through 8 (one per scan) 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 for the first bank of eight channels. Our example starts with Y20, but the actual value depends on the location of the module in your application.

We move the data for channels 9 through 16 (one per scan) from the stack to the accumulator.

This rung converts the accumulator data for channels 9 through 16 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 for the second bank of eight channels. In our example, it is Y40, but the actual value depends on the location of the module in your application.

Example 1 Continued


C2
Select Channel, binary encoded
Y34
OUT
C4
Y54
OUT
C6
C10
C3
Select Channel, cont’d
Y35
OUT
C4
Y55
OUT
C7
C10
C5
Select Channel, cont’d
Y36
OUT
C6
Y56
OUT
C7
C10
SP1
Enable Outputs
Y37
OUT
Y57
OUT

Set Y36, Y35, and Y34 to the binary code which selects the output channel 1 through 8, based on the control relay status. We use the same binary code to select channels 9 through 16.

CR(on)	Y36,Y56	Y35,Y55	Y34,Y54	Channels
C1	Off	Off	Off	Ch. 1 and 9
C2	Off	Off	On	Ch. 2 and 10
C3	Off	On	Off	Ch. 3 and 11
C4	Off	On	On	Ch. 4 and 12
C5	On	Off	Off	Ch. 5 and 13
C6	On	Off	On	Ch. 6 and 14
C7	On	On	Off	Ch. 7 and 15
C10	On	On	On	Ch. 8 and 16

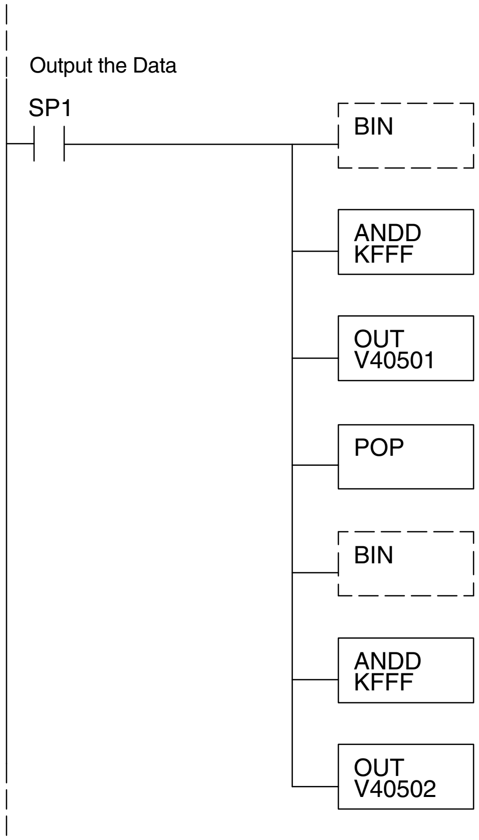
Enable channels 1 through 8. SP1 is always on.

Enable channels 9 through 16.

Sequencing Example 2, D4–430

✓ ✓ ✓
430 440 450

Since the D4–430 does not support the OUTF instruction, the previous program must be modified to make sure the channel select bits or the output enable bits are not accidentally changed by the data in the accumulator. Replace the “Output the Data” rung in the middle of Example 1 with the new rung below. Be sure to retain the original order of the rungs shown in Example 1 for the program to work. This will also work with D4–440 and D4–450 CPUs.



This rung converts the accumulator data for channels 1 through 8 (one per scan) 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 for the first bank of 8 channels, but the actual value depends on the location of the module in your application.

We move the data for channels 9 through 16 (one per scan) from the stack to the accumulator.

This rung converts the accumulator data for channels 9 through 16 (one per scan) 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 uses V40502 for the second bank of 8 channels, but the actual value depends on the location of the module in your application.

Sequencing Example 3, D4–440/450

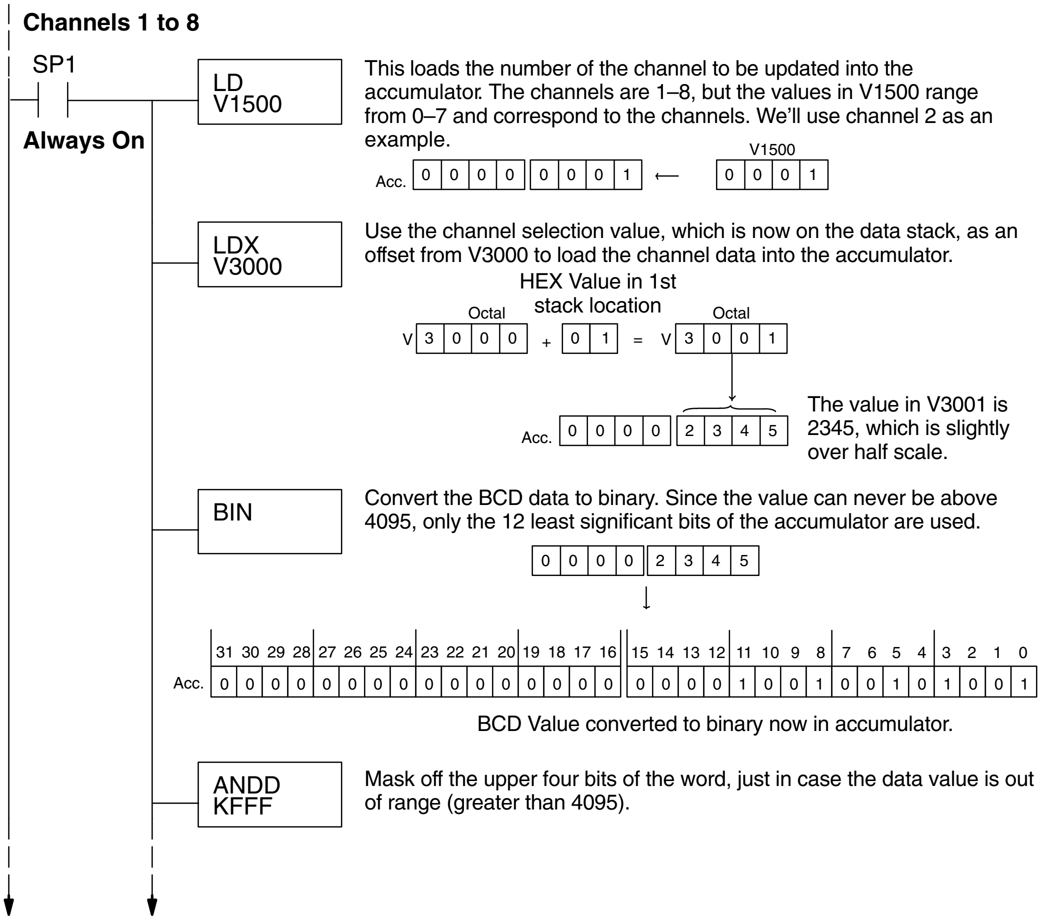
☒ ☑ ☑
430 440 450

The following program example shows how to send digital values to the module when you have more than one channel. This example works only for D4–440 and D4–450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 3								
Channel Number	1	2	3	4	5	6	7	8
V-Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007
Channel Number	9	10	11	12	13	14	15	16
V-Memory Storage	3010	3011	3012	3013	3014	3015	3016	3017

The channel index is stored in V1500. It varies from 0 to 7, pointing to pairs of channels as shown: 0 = ch. 1 and 9, 1 = ch. 2 and 10, ... and 7 = ch. 8 and16. This example assumes V1500 is initialized to “0” earlier in the program.

The first portion of the program updates one channel from the first bank of eight channels (1 to 8) during each scan. The program comments for this portion also shows the accumulator status at each step. The second part of the program updates one channel from the second bank of eight channels (9 to 16) during each scan (accumulator status not shown). So, there are two channels which are updated on each scan. The last portion of the program increments the channel index number and resets it after eight scans.


Channels 1 to 8
SP1
Always On
LD
V1500
Acc. 0 0 0 0 0 0 0 1
V1500
0 0 0 1
LDX
V3000
HEX Value in 1st stack location
Octal
V 3 0 0 0
+ 0 1
=
Octal
V 3 0 0 1
Acc. 0 0 0 0 2 3 4 5
BIN
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
ANDD
KFFF

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

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.

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

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.

BCD Value converted to binary now in accumulator.

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

Example 3 Continued

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

ORD
K8000

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to "1", enabling channels 1 through 8.

	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	1	0	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	1	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
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

↓

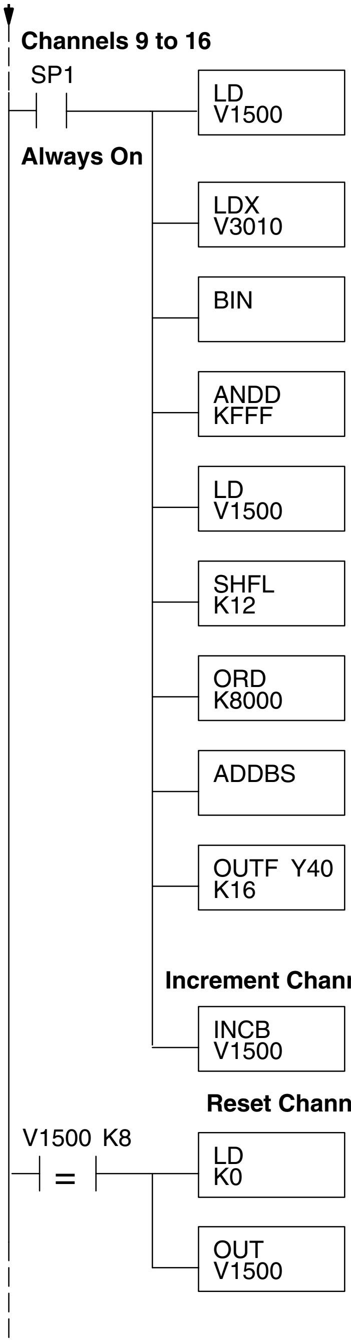
	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Acc.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	0	0	1	0	1	0	0	1

Data for Analog Module

OUTF Y20
K16

Send the lower 16 bits stored in the accumulator to the analog module.. The lowest 12 bits contain the analog data. Bits 12, 13, and 14 are the channel selection bits. Bit 15 is the Output Enable bit. This example uses Y20, but the actual value depends on the location of the modules in your application.

Example 3 Continued



This loads the number of the channel to be updated into the accumulator. The channels are 9–16, but the values in V1500 range from 0–7 and correspond to the channels. We'll use channel 10 as an example.

Use the channel selection value, which is now on the data stack, as an offset from V3010 to load the channel data into the accumulator.

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.

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

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).

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.

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to "1", enabling channels 9 through 16.

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.

Send the lower 16 bits stored in the accumulator to the analog module. The lowest 12 bits contain the analog data. Bits 12, 13, and 14 are the channel selection bits. Bit 15 is the Output Enable bit. This example uses Y40, but the actual value depends on the location of the modules in your application.

Increment Channel Index

Increment the channel index value. This allows the logic to cycle through all sixteen channels (8 pairs).

Reset Channel Index

When channels 8 and 16 have been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1 and channel 9).

Sequencing Example 4, D4–430

✓ ✓ ✓
430 440 450

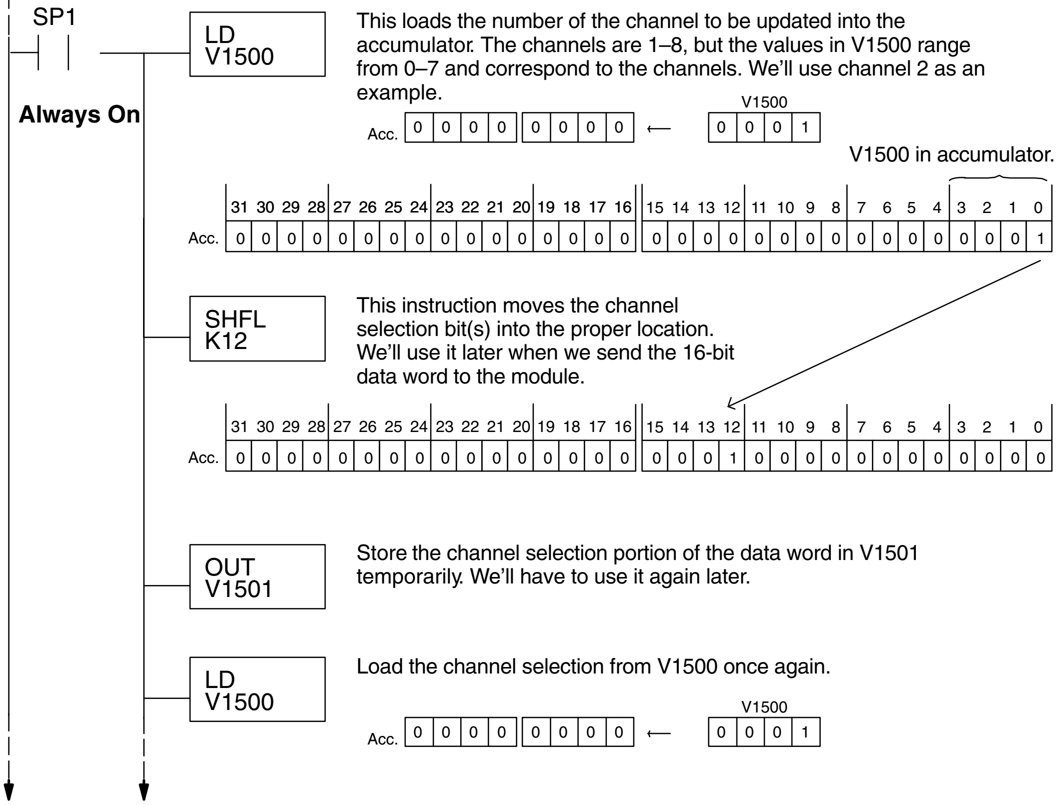
The following program example shows how to send digital values to the module when you have more than one channel. This example also works for D4–440 and D4–450 CPUs. It assumes you are using the following data locations.

V-Memory Locations for Output Data in Example 4								
Channel Number	1	2	3	4	5	6	7	8
V–Memory Storage	3000	3001	3002	3003	3004	3005	3006	3007
Channel Number	9	10	11	12	13	14	15	16
V–Memory Storage	3010	3011	3012	3013	3014	3015	3016	3017

The channel index is stored in V1500. It varies from 0 to 7, pointing to pairs of channels as shown: 0 = ch. 1 and 9, 1 = ch. 2 and 10, ... and 7 = ch. 8 and16. This example assumes V1500 is initialized to “0” earlier in the program.

The first portion of the program updates one channel from the first bank of eight channels (1 to 8) during each scan. The program comments for this portion also show the accumulator status at each step. The second part of the program updates one channel from the second bank of eight channels (9 to 16) during each scan (accumulator status not shown, for brevity). So, there are two channels which are updated on each scan. The last portion of the program increments the channel index number and resets it after eight scans.

Channels 1 to 8


SP1
Always On
LD
V1500
Acc. 0 0 0 0 0 0 0 0
V1500 0 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 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
SHFL
K12
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
OUT
V1501
LD
V1500
Acc. 0 0 0 0 0 0 0 0
V1500 0 0 0 1

This loads the number of the channel to be updated into the accumulator. The channels are 1–8, but the values in V1500 range from 0–7 and correspond to the channels. We’ll use channel 2 as an example.

V1500 in accumulator.

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.

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

Load the channel selection from V1500 once again.

Example 4 Continued

```
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 V 3 0 0 0 + 0 1 = Octal 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 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.

```
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 send 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

OR

	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
K8000
```

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to “1”, enabling channels 1 through 8.

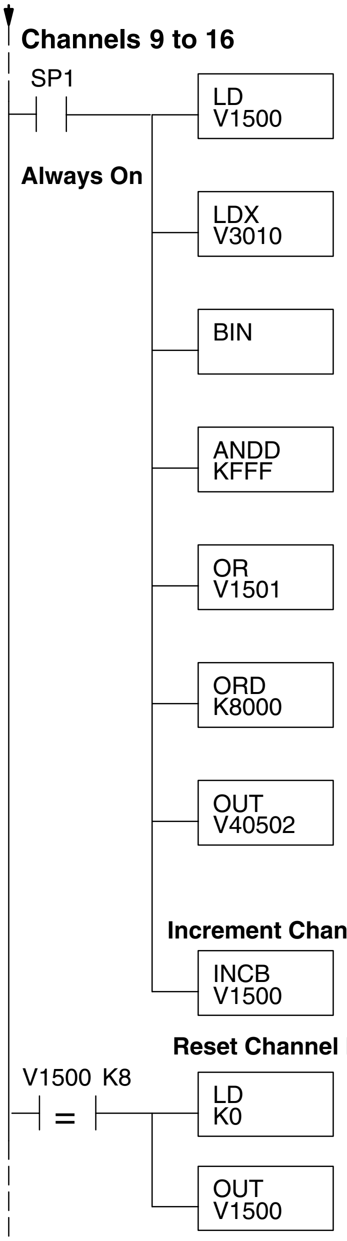
↓

	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	1	0	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. This example uses V40501, but the actual value depends on the location of the module in your application.

Example 4 Continued



Load the channel selection from V1500 once again.

Use the channel selection value, which is now on the data stack, as an offset from V3010 to load the channel data into the accumulator.

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.

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

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 send to the analog module.

Set the Output Enable bit, by combining the value of 8000 hex with the accumulator value. This sets bit 15 to “1”, enabling channels 9 through 16.

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 and the output enable bit. The remaining 12 bits contain the analog data. This example uses V40502, but the actual value depends on the location of the module in your application.

Increment the channel index value. This allows the logic to cycle through all sixteen channels (8 pairs).

When channels 8 and 16 have been updated, then reset the channel selection memory location to 0 (remember, 0 represents channel 1 and channel 9).

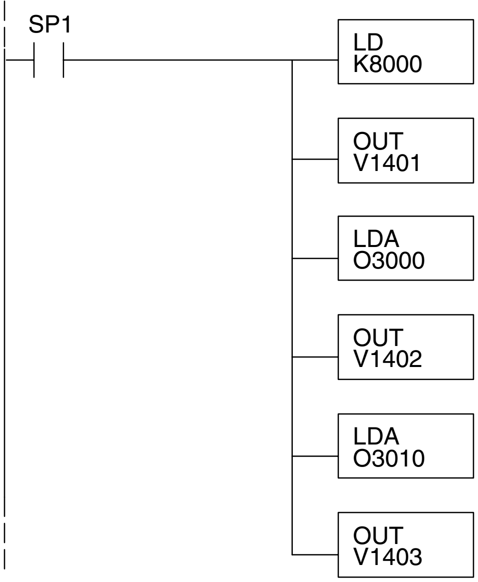
Updating All Channels in a Single Scan, D4–440/450

☒ ✓ ✓
430 440 450

By using the Immediate instructions found in the D4–440 and D4–450 CPUs, you can easily update all sixteen channels in a single scan. This method slows 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 V3000 to V3007 for channels 1 to 8 respectively, and data loaded in V3010 to V3017 for channels 9 to 16 respectively.

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

Initialize the Immediate Analog Output Pointers



The LD instruction loads the data into the accumulator. The constant K8000 represents channel 1 and 9 selected, Output Enable bit is on.

We store the channel index pointer in V1401. The program increments this in each pass, resetting it after eight scans.

The Load Address instruction takes the octal 3000 and converts it to hex, then puts it in the accumulator. V3000 is the location wich contains the data for channel 1.

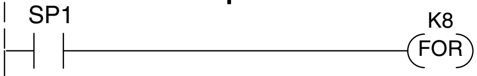
V1402 contains the pointer for channel 1 to 8 data.

The Load Address instruction takes the octal 3010 and converts it to hex, then puts it in the accumulator. V3010 is the location which contains the data for channel 9.

V1403 contains the pointer for channel 9 to 16 data.

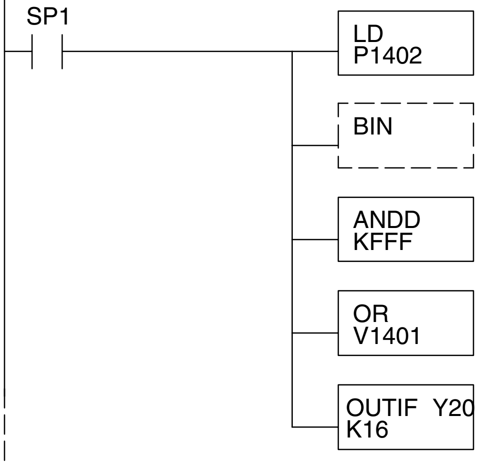
The following FOR–NEXT loop updates all 16 channels in a single scan. The first portion of the loop updates channels 1 through 8.

FOR-NEXT Loop



The following FOR–NEXT loop updates channels 1 through 16. The valid range for the constant K is 2 to 8 for the module. For example, a value of 2 will update channels 1 and 9, then 2 and 10.

Update Channels 1 - 8



Load the analog output value into the accumulator. V3000 = Ch. 1, V3001 = Ch. 2, ... V3007 = Ch. 8. The number at V1402 is a pointer to the address of the value.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

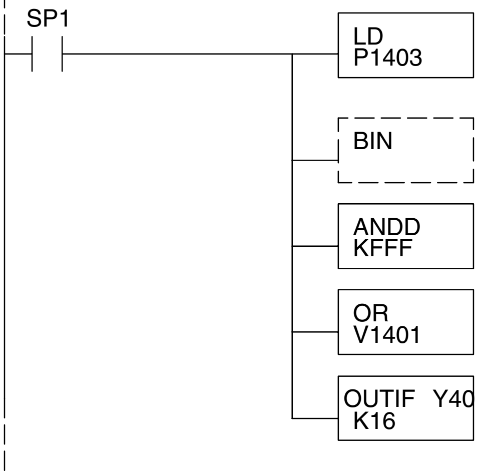
Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y36, Y35, and Y34 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module's lower data word. This updates channels 1 through 8 during the FOR–NEXT loop. This example uses Y20, but the actual value depends on the location of the modules in your application.

The second portion of the FOR–NEXT loop updates channels 9 through 16.

Update Channels 9 - 16



Load the analog output value into the accumulator. V3010 = Ch. 9, V3011 = Ch. 10, ... V3017 = Ch. 16. The number at V1403 is a pointer to the address of the value.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

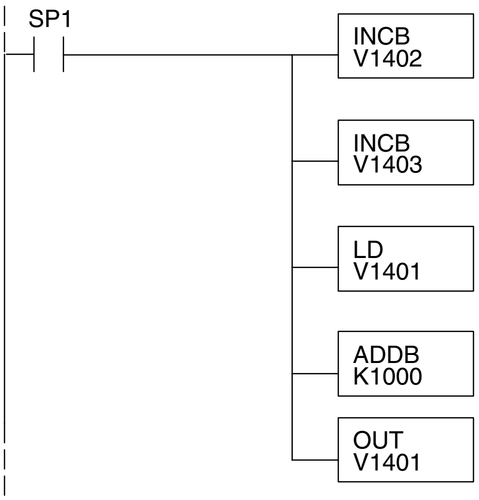
Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y56, Y55, and Y54 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module's upper data word. This updates channels 9 through 16 during the FOR–NEXT loop. This example uses Y40, but the actual value depends on the location of the modules in your application.

Now we increment the Immediate Analog Output Pointers for the next pair of channels, before the next pass through the FOR-NEXT loop.

Increment Immediate Analog Output Pointers



Increment the analog output data pointer for channels 1 through 8.

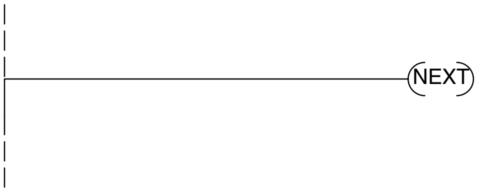
Increment the analog output data pointer for channels 9 through 16.

Load the current channel index count into the accumulator.

Add 1000 hex to the channel select field is in the most significant four bits.

Save the incremented channel index value for the next pass through the FOR–NEXT loop.

Here is the end of the FOR-NEXT loop.



Execute the loop above the number of times specified in the FOR instruction.

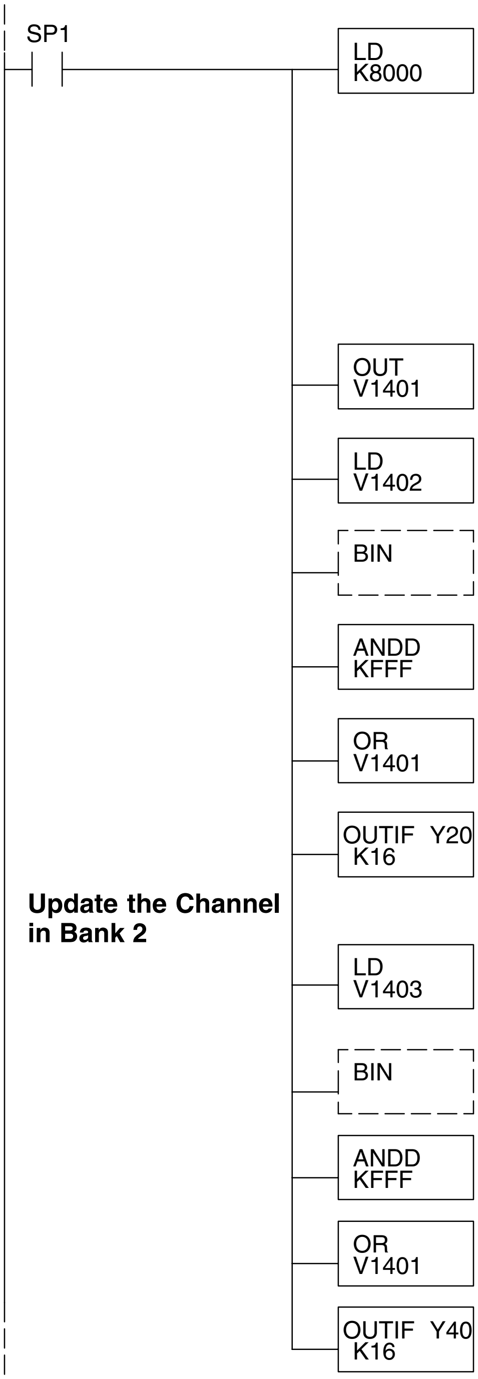
The program above can be modified to immediately output any number of channel pairs by changing the constant in the FOR instruction.

Updating a Pair of Channels in a Single Scan, D4–440/450

☒	✓	✓
430	440	450

You can also update just a pair of channels in a single scan by using the Immediate instructions found in the D4–440 and D4–450 CPUs. By removing the FOR–NEXT rungs and a couple of other rungs, we create the example below. This example assumes you will use one channel from the first eight, and a corresponding channel from the second bank of eight. Also, the data must already be loaded in V1401 and V1402 respectively.

Update the Channel in Bank 1



Load a constant representing the channel pair to be updated into the accumulator.

Index	Channels
8000	Ch. 1 and 9
9000	Ch. 2 and 10
A000	Ch. 3 and 11
B000	Ch. 4 and 12
C000	Ch. 5 and 13
D000	Ch. 6 and 14
E000	Ch. 7 and 15
F000	Ch. 8 and 16

We store the channel index in V1401.

Load the analog output value for the channel in bank 1 from V1402 into the accumulator.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y36, Y35, and Y34 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module's lower data word. This updates a channel in the first bank of eight. This example uses Y20, but the actual value depends on the location of the modules in your application.

Load the analog output value for the channel in bank 2 from V1403 into the accumulator.

Convert the data to binary. This step is optional, and your program may do this conversion elsewhere.

Mask off the channel select and output enable bits, so they are not corrupted by data out of range.

Combine the channel select and output enable bits. for Y56, Y55, and Y54 with the data in the accumulator.

Immediately write the lower 16 bits of the accumulator to the module's upper data word. This updates a channel in the second bank of eight. This example uses Y40, but the actual value depends on the location of the modules in your application.

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.

Range	If you know the digital value ...	If you know the analog signal level ...
0 to 5VDC	$A = \frac{5D}{4095}$	$D = \frac{4095}{5}A$
0 to 10VDC	$A = \frac{10D}{4095}$	$D = \frac{4095}{10}A$

For example, if you need a 2.5V signal level with the module set for 0–5V, you would use the following formula to determine the digital value that should be stored in the V-memory location that contains the data.

$$D = \frac{4095}{5}A$$

$$D = \frac{4095}{5}(2.5V)$$

$$D = (819)\ (2.5)$$

$$D = 2048$$

DL405 Discrete I/O Memory Map

In This Chapter. . . .

- — X Input / Y Output Bit Map
- — Control Relay Bit Map
- — Remote I/O Bit Map
- — Stage Control / Status Bit Map

The following tables provide the discrete memory map for the DL405 CPUs. These are especially helpful since DL405 Analog modules utilize discrete input and output points. Also, there are some occasions where you use control relays to sequence the program logic that updates the analog data. If you need additional memory map information, see the DL405 User Manual.

X Input / Y Output Bit Map

This table provides a listing of individual Input and Output points associated with each V-memory address bit for the DL430, DL440, and DL450 CPUs (for DL450 CPUs, also see the following table).

MSB				**DL430 / DL440 / DL450 Input (X) and Output (Y) Points**											**LSB**	**X Input Address**	**Y Output Address**
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40400	V40500
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40401	V40501
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40402	V40502
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40403	V40503
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40404	V40504
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40405	V40505
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40406	V40506
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40407	V40507
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40410	V40510
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40411	V40511
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40412	V40512
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40413	V40513
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40414	V40514
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40415	V40515
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40416	V40516
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40417	V40517
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40420	V40520
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40421	V40521
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40422	V40522
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40423	V40523

MSB				**DL450 Additional Input (X) and Output (Y) Points**											**LSB**	**X Input Address**	**Y Output Address**
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40424	V40524
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40425	V40525
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40426	V40526
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40427	V40527
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40430	V40530
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V40431	V40531
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40432	V40532
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40433	V40533

MSB			DL450 Additional Input (X) and Output (Y) Points (cont'd)												LSB	X Input Address	Y Output Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40434	V40534
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40435	V40535
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40436	V40536
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40437	V40537
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V40440	V40540
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V40441	V40541
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V40442	V40542
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V40443	V40543
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V40444	V40544
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V40445	V40545
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V40446	V40546
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V40447	V40547
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V40450	V40550
1237	1236	1235	1234	1233	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	V40451	V40551
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V40452	V40552
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V40453	V40553
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V40454	V40554
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V40455	V40555
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V40456	V40556
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V40457	V40557
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V40460	V40560
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V40461	V40561
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V40462	V40562
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V40463	V40563
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V40464	V40564
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V40465	V40565
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V40466	V40566
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V40467	V40567
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V40470	V40570
1637	1636	1635	1634	1633	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	V40471	V40571
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V40472	V40572
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V40473	V40573
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V40474	V40574
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V40475	V40575
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V40476	V40576
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V40477	V40577

Control Relay Bit Map

This table provides a listing of the individual control relays associated with each V-memory address bit.

MSB	DL430 / DL440 / DL450 Control Relays (C)														LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40600
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40601
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40602
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40603
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40604
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40605
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40606
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40607
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40610
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40611
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40612
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40613
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40614
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40615
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40616
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40617
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40620
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40621
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40622
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40623
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40624
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40625
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40626
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40627
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40630
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V40631
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40632
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40633
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40634
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40635

This portion of the table shows additional Control Relays points available with the DL440 and DL450.

MSB	DL440 / DL450 Additional Control Relays (C)														LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40636
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40637
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V40640
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V40641
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V40642
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V40643
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V40644
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V40645
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V40646
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V40647
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V40650
1237	1236	1235	1234	1233	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	V40651
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V40652
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V40653
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V40654
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V40655
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V40656
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V40657
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V40660
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V40661
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V40662
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V40663
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V40664
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V40665
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V40666
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V40667
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V40670
1637	1636	1635	1634	1633	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	V40671
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V40672
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V40673
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V40674
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V40675
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V40676
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V40677

This portion of the table shows additional Control Relays points available with the DL450.

MSB	**DL450 Additional Control Relays (C)**														**LSB**	**Address**
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
2017	2016	2015	2014	2013	2012	2011	2010	2007	2006	2005	2004	2003	2002	2001	2000	V40700
2037	2036	2035	2034	2033	2032	2031	2030	2027	2026	2025	2024	2023	2022	2021	2020	V40701
2057	2056	2055	2054	2053	2052	2051	2050	2047	2046	2045	2044	2043	2042	2041	2040	V40702
2077	2076	2075	2074	2073	2072	2071	2070	2067	2066	2065	2064	2063	2062	2061	2060	V40703
2117	2116	2115	2114	2113	2112	2111	2110	2107	2106	2105	2104	2103	2102	2101	2100	V40704
2137	2136	2135	2134	2133	2132	2131	2130	2127	2126	2125	2124	2123	2122	2121	2120	V40705
2157	2156	2155	2154	2153	2152	2151	2150	2147	2146	2145	2144	2143	2142	2141	2140	V40706
2177	2176	2175	2174	2173	2172	2171	2170	2167	2166	2165	2164	2163	2162	2161	2160	V40707
2217	2216	2215	2214	2213	2212	2211	2210	2207	2206	2205	2204	2203	2202	2201	2200	V40710
2237	2236	2235	2234	2233	2232	2231	2230	2227	2226	2225	2224	2223	2222	2221	2220	V40711
2257	2256	2255	2254	2253	2252	2251	2250	2247	2246	2245	2244	2243	2242	2241	2240	V40712
2277	2276	2275	2274	2273	2272	2271	2270	2267	2266	2265	2264	2263	2262	2261	2260	V40713
2317	2316	2315	2314	2313	2312	2311	2310	2307	2306	2305	2304	2303	2302	2301	2300	V40714
2337	2336	2335	2334	2333	2332	2331	2330	2327	2326	2325	2324	2323	2322	2321	2320	V40715
2357	2356	2355	2354	2353	2352	2351	2350	2347	2346	2345	2344	2343	2342	2341	2340	V40716
2377	2376	2375	2374	2373	2372	2371	2370	2367	2366	2365	2364	2363	2362	2361	2360	V40717
2417	2416	2415	2414	2413	2412	2411	2410	2407	2406	2405	2404	2403	2402	2401	2400	V40720
2437	2436	2435	2434	2433	2432	2431	2430	2427	2426	2425	2424	2423	2422	2421	2420	V40721
2457	2456	2455	2454	2453	2452	2451	2450	2447	2446	2445	2444	2443	2442	2441	2440	V40722
2477	2476	2475	2474	2473	2472	2471	2470	2467	2466	2465	2464	2463	2462	2461	2460	V40723
2517	2516	2515	2514	2513	2512	2511	2510	2507	2506	2505	2504	2503	2502	2501	2500	V40724
2537	2536	2535	2534	2533	2532	2531	2530	2527	2526	2525	2524	2523	2522	2521	2520	V40725
2557	2556	2555	2554	2553	2552	2551	2550	2547	2546	2545	2544	2543	2542	2541	2540	V40726
2577	2576	2575	2574	2573	2572	2571	2570	2567	2566	2565	2564	2563	2562	2561	2560	V40727
2617	2616	2615	2614	2613	2612	2611	2610	2607	2606	2605	2604	2603	2602	2601	2600	V40730
2637	2636	2635	2634	2633	2632	2631	2630	2627	2626	2625	2624	2623	2622	2621	2620	V40731
2657	2656	2655	2654	2653	2652	2651	2650	2647	2646	2645	2644	2643	2642	2641	2640	V40732
2677	2676	2675	2674	2673	2672	2671	2670	2667	2666	2665	2664	2663	2662	2661	2660	V40733
2717	2716	2715	2714	2713	2712	2711	2710	2707	2706	2705	2704	2703	2702	2701	2700	V40734
2737	2736	2735	2734	2733	2732	2731	2730	2727	2726	2725	2724	2723	2722	2721	2720	V40735
2757	2756	2755	2754	2753	2752	2751	2750	2747	2746	2745	2744	2743	2742	2741	2740	V40736
2777	2776	2775	2774	2773	2772	2771	2770	2767	2766	2765	2764	2763	2762	2761	2760	V40737

MSB				DL450 Additional Control Relays (C) (cont'd)											LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
3017	3016	3015	3014	3013	3012	3011	3010	3007	3006	3005	3004	3003	3002	3001	3000	V40740
3037	3036	3035	3034	3033	3032	3031	3030	3027	3026	3025	3024	3023	3022	3021	3020	V40741
3057	3056	3055	3054	3053	3052	3051	3050	3047	3046	3045	3044	3043	3042	3041	3040	V40742
3077	3076	3075	3074	3073	3072	3071	3070	3067	3066	3065	3064	3063	3062	3061	3060	V40743
3117	3116	3115	3114	3113	3112	3111	3110	3107	3106	3105	3104	3103	3102	3101	3100	V40744
3137	3136	3135	3134	3133	3132	3131	3130	3127	3126	3125	3124	3123	3122	3121	3120	V40745
3157	3156	3155	3154	3153	3152	3151	3150	3147	3146	3145	3144	3143	3142	3141	3140	V40746
3177	3176	3175	3174	3173	3172	3171	3170	3167	3166	3165	3164	3163	3162	3161	3160	V40747
3217	3216	3215	3214	3213	3212	3211	3210	3207	3206	3205	3204	3203	3202	3201	3200	V40750
3237	3236	3235	3234	3233	3232	3231	3230	3227	3226	3225	3224	3223	3222	3221	3220	V40751
3257	3256	3255	3254	3253	3252	3251	3250	3247	3246	3245	3244	3243	3242	3241	3240	V40752
3277	3276	3275	3274	3273	3272	3271	3270	3267	3266	3265	3264	3263	3262	3261	3260	V40753
3317	3316	3315	3314	3313	3312	3311	3310	3307	3306	3305	3304	3303	3302	3301	3300	V40754
3337	3336	3335	3334	3333	3332	3331	3330	3327	3326	3325	3324	3323	3322	3321	3320	V40755
3357	3356	3355	3354	3353	3352	3351	3350	3347	3346	3345	3344	3343	3342	3341	3340	V40756
3377	3376	3375	3374	3373	3372	3371	3370	3367	3366	3365	3364	3363	3362	3361	3360	V40757
3417	3416	3415	3414	3413	3412	3411	3410	3407	3406	3405	3404	3403	3402	3401	3400	V40760
3437	3436	3435	3434	3433	3432	3431	3430	3427	3426	3425	3424	3423	3422	3421	3420	V40761
3457	3456	3455	3454	3453	3452	3451	3450	3447	3446	3445	3444	3443	3442	3441	3440	V40762
3477	3476	3475	3474	3473	3472	3471	3470	3467	3466	3465	3464	3463	3462	3461	3460	V40763
3517	3516	3515	3514	3513	3512	3511	3510	3507	3506	3505	3504	3503	3502	3501	3500	V40764
3537	3536	3535	3534	3533	3532	3531	3530	3527	3526	3525	3524	3523	3522	3521	3520	V40765
3557	3556	3555	3554	3553	3552	3551	3550	3547	3546	3545	3544	3543	3542	3541	3540	V40766
3577	3576	3575	3574	3573	3572	3571	3570	3567	3566	3565	3564	3563	3562	3561	3560	V40767
3617	3616	3615	3614	3613	3612	3611	3610	3607	3606	3605	3604	3603	3602	3601	3600	V40770
3637	3636	3635	3634	3633	3632	3631	3630	3627	3626	3625	3624	3623	3622	3621	3620	V40771
3657	3656	3655	3654	3653	3652	3651	3650	3647	3646	3645	3644	3643	3642	3641	3640	V40772
3677	3676	3675	3674	3673	3672	3671	3670	3667	3666	3665	3664	3663	3662	3661	3660	V40773
3717	3716	3715	3714	3713	3712	3711	3710	3707	3706	3705	3704	3703	3702	3701	3700	V40774
3737	3736	3735	3734	3733	3732	3731	3730	3727	3726	3725	3724	3723	3722	3721	3720	V40775
3757	3756	3755	3754	3753	3752	3751	3750	3747	3746	3745	3744	3743	3742	3741	3740	V40776
3777	3776	3775	3774	3773	3772	3771	3770	3767	3766	3765	3764	3763	3762	3761	3760	V40777

Remote I/O Bit Map

This table provides a listing of the individual remote I/O points associated with each V-memory address bit. The DL430 and DL440 CPUs use the GX point type for both remote input and output point types. The DL450 CPU has the additional GY point type for use as remote output point references.

MSB	**DL430 / DL440 / DL450 Remote I/O (GX) and (GY) Points**														**LSB**	**GX Address**	**GY Address** (DL450)
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V40000	V40200
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V40001	V40201
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V40002	V40202
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V40003	V40203
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V40004	V40204
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V40005	V40205
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V40006	V40206
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V40007	V40207
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V40010	V40210
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V40011	V40211
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V40012	V40212
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V40013	V40213
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V40014	V40214
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V40015	V40215
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V40016	V40216
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V40017	V40217
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V40020	V40220
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V40021	V40221
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V40022	V40222
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V40023	V40223
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V40024	V40224
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V40025	V40225
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V40026	V40226
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V40027	V40227
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V40030	V40230
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V40031	V40231
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V40032	V40232
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V40033	V40233
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V40034	V40234
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V40035	V40235
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V40036	V40236
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V40037	V40237

This portion of the table shows additional Remote I/O (GX) points available with the DL440 and DL450. The (GY) remote output point type is available only with the DL450 (the GX type points works as both input and output point types for the DL440).

MSB	DL440 / DL450 Additional Remote I/O (GX) Points														LSB	GX Address	GY Address (DL450)
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V40040	V40240
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V40041	V40241
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V40042	V40242
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V40043	V40243
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V40044	V40244
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V40045	V40245
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V40046	V40246
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V40047	V40247
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V40050	V40250
1237	1236	1235	1234	1233	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	V40051	V40251
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V40052	V40252
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V40053	V40253
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V40054	V40254
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V40055	V40255
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V40056	V40256
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V40057	V40257
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V40060	V40260
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V40061	V40261
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V40062	V40262
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V40063	V40263
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V40064	V40264
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V40065	V40265
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V40066	V40266
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V40067	V40267
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V40070	V40270
1637	1636	1635	1634	1633	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	V40071	V40271
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V40072	V40272
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V40073	V40273
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V40074	V40274
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V40075	V40275
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V40076	V40276
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V40077	V40277

This portion of the table shows additional Remote I/O (GX and GY) points available with the DL450.

MSB	DL450 Additional Remote I/O (GX) and (GY) Points														LSB	GX Address	GY Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0		
2017	2016	2015	2014	2013	2012	2011	2010	2007	2006	2005	2004	2003	2002	2001	2000	V40100	V40300
2037	2036	2035	2034	2033	2032	2031	2030	2027	2026	2025	2024	2023	2022	2021	2020	V40101	V40301
2057	2056	2055	2054	2053	2052	2051	2050	2047	2046	2045	2044	2043	2042	2041	2040	V40102	V40302
2077	2076	2075	2074	2073	2072	2071	2070	2067	2066	2065	2064	2063	2062	2061	2060	V40103	V40303
2117	2116	2115	2114	2113	2112	2111	2110	2107	2106	2105	2104	2103	2102	2101	2100	V40104	V40304
2137	2136	2135	2134	2133	2132	2131	2130	2127	2126	2125	2124	2123	2122	2121	2120	V40105	V40305
2157	2156	2155	2154	2153	2152	2151	2150	2147	2146	2145	2144	2143	2142	2141	2140	V40106	V40306
2177	2176	2175	2174	2173	2172	2171	2170	2167	2166	2165	2164	2163	2162	2161	2160	V40107	V40307
2217	2216	2215	2214	2213	2212	2211	2210	2207	2206	2205	2204	2203	2202	2201	2200	V40110	V40310
2237	2236	2235	2234	2233	2232	2231	2230	2227	2226	2225	2224	2223	2222	2221	2220	V40111	V40311
2257	2256	2255	2254	2253	2252	2251	2250	2247	2246	2245	2244	2243	2242	2241	2240	V40112	V40312
2277	2276	2275	2274	2273	2272	2271	2270	2267	2266	2265	2264	2263	2262	2261	2260	V40113	V40313
2317	2316	2315	2314	2313	2312	2311	2310	2307	2306	2305	2304	2303	2302	2301	2300	V40114	V40314
2337	2336	2335	2334	2333	2332	2331	2330	2327	2326	2325	2324	2323	2322	2321	2320	V40115	V40315
2357	2356	2355	2354	2353	2352	2351	2350	2347	2346	2345	2344	2343	2342	2341	2340	V40116	V40316
2377	2376	2375	2374	2373	2372	2371	2370	2367	2366	2365	2364	2363	2362	2361	2360	V40117	V40317
2417	2416	2415	2414	2413	2412	2411	2410	2407	2406	2405	2404	2403	2402	2401	2400	V40120	V40320
2437	2436	2435	2434	2433	2432	2431	2430	2427	2426	2425	2424	2423	2422	2421	2420	V40121	V40321
2457	2456	2455	2454	2453	2452	2451	2450	2447	2446	2445	2444	2443	2442	2441	2440	V40122	V40322
2477	2476	2475	2474	2473	2472	2471	2470	2467	2466	2465	2464	2463	2462	2461	2460	V40123	V40323
2517	2516	2515	2514	2513	2512	2511	2510	2507	2506	2505	2504	2503	2502	2501	2500	V40124	V40324
2537	2536	2535	2534	2533	2532	2531	2530	2527	2526	2525	2524	2523	2522	2521	2520	V40125	V40325
2557	2556	2555	2554	2553	2552	2551	2550	2547	2546	2545	2544	2543	2542	2541	2540	V40126	V40326
2577	2576	2575	2574	2573	2572	2571	2570	2567	2566	2565	2564	2563	2562	2561	2560	V40127	V40327
2617	2616	2615	2614	2613	2612	2611	2610	2607	2606	2605	2604	2603	2602	2601	2600	V40130	V40330
2637	2636	2635	2634	2633	2632	2631	2630	2627	2626	2625	2624	2623	2622	2621	2620	V40131	V40331
2657	2656	2655	2654	2653	2652	2651	2650	2647	2646	2645	2644	2643	2642	2641	2640	V40132	V40332
2677	2676	2675	2674	2673	2672	2671	2670	2667	2666	2665	2664	2663	2662	2661	2660	V40133	V40333
2717	2716	2715	2714	2713	2712	2711	2710	2707	2706	2705	2704	2703	2702	2701	2700	V40134	V40334
2737	2736	2735	2734	2733	2732	2731	2730	2727	2726	2725	2724	2723	2722	2721	2720	V40135	V40335
2757	2756	2755	2754	2753	2752	2751	2750	2747	2746	2745	2744	2743	2742	2741	2740	V40136	V40336
2777	2776	2775	2774	2773	2772	2771	2770	2767	2766	2765	2764	2763	2762	2761	2760	V40137	V40337

Stage Control / Status Bit Map

This table provides a listing of the individual stage control bits associated with each V-memory address bit.

MSB	DL430 / DL440 / DL450 Stage (S) Control Bits														LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
017	016	015	014	013	012	011	010	007	006	005	004	003	002	001	000	V41000
037	036	035	034	033	032	031	030	027	026	025	024	023	022	021	020	V41001
057	056	055	054	053	052	051	050	047	046	045	044	043	042	041	040	V41002
077	076	075	074	073	072	071	070	067	066	065	064	063	062	061	060	V41003
117	116	115	114	113	112	111	110	107	106	105	104	103	102	101	100	V41004
137	136	135	134	133	132	131	130	127	126	125	124	123	122	121	120	V41005
157	156	155	154	153	152	151	150	147	146	145	144	143	142	141	140	V41006
177	176	175	174	173	172	171	170	167	166	165	164	163	162	161	160	V41007
217	216	215	214	213	212	211	210	207	206	205	204	203	202	201	200	V41010
237	236	235	234	233	232	231	230	227	226	225	224	223	222	221	220	V41011
257	256	255	254	253	252	251	250	247	246	245	244	243	242	241	240	V41012
277	276	275	274	273	272	271	270	267	266	265	264	263	262	261	260	V41013
317	316	315	314	313	312	311	310	307	306	305	304	303	302	301	300	V41014
337	336	335	334	333	332	331	330	327	326	325	324	323	322	321	320	V41015
357	356	355	354	353	352	351	350	347	346	345	344	343	342	341	340	V41016
377	376	375	374	373	372	371	370	367	366	365	364	363	362	361	360	V41017
417	416	415	414	413	412	411	410	407	406	405	404	403	402	401	400	V41020
437	436	435	434	433	432	431	430	427	426	425	424	423	422	421	420	V41021
457	456	455	454	453	452	451	450	447	446	445	444	443	442	441	440	V41022
477	476	475	474	473	472	471	470	467	466	465	464	463	462	461	460	V41023
517	516	515	514	513	512	511	510	507	506	505	504	503	502	501	500	V41024
537	536	535	534	533	532	531	530	527	526	525	524	523	522	521	520	V41025
557	556	555	554	553	552	551	550	547	546	545	544	543	542	541	540	V41026
577	576	575	574	573	572	571	570	567	566	565	564	563	562	561	560	V41027

MSB	DL440 / DL450 Additional Stage (S) Control Bits														LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
617	616	615	614	613	612	611	610	607	606	605	604	603	602	601	600	V41030
637	636	635	634	633	632	631	630	627	626	625	624	623	622	621	620	V41031
657	656	655	654	653	652	651	650	647	646	645	644	643	642	641	640	V41032
677	676	675	674	673	672	671	670	667	666	665	664	663	662	661	660	V41033
717	716	715	714	713	712	711	710	707	706	705	704	703	702	701	700	V41034
737	736	735	734	733	732	731	730	727	726	725	724	723	722	721	720	V41035
757	756	755	754	753	752	751	750	747	746	745	744	743	742	741	740	V41036
777	776	775	774	773	772	771	770	767	766	765	764	763	762	761	760	V41037

MSB	DL440 / DL450 Additional Stage (S) Control Bits (continued)														LSB	Address
17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0	
1017	1016	1015	1014	1013	1012	1011	1010	1007	1006	1005	1004	1003	1002	1001	1000	V41040
1037	1036	1035	1034	1033	1032	1031	1030	1027	1026	1025	1024	1023	1022	1021	1020	V41041
1057	1056	1055	1054	1053	1052	1051	1050	1047	1046	1045	1044	1043	1042	1041	1040	V41042
1077	1076	1075	1074	1073	1072	1071	1070	1067	1066	1065	1064	1063	1062	1061	1060	V41043
1117	1116	1115	1114	1113	1112	1111	1110	1107	1106	1105	1104	1103	1102	1101	1100	V41044
1137	1136	1135	1134	1133	1132	1131	1130	1127	1126	1125	1124	1123	1122	1121	1120	V41045
1157	1156	1155	1154	1153	1152	1151	1150	1147	1146	1145	1144	1143	1142	1141	1140	V41046
1177	1176	1175	1174	1173	1172	1171	1170	1167	1166	1165	1164	1163	1162	1161	1160	V41047
1217	1216	1215	1214	1213	1212	1211	1210	1207	1206	1205	1204	1203	1202	1201	1200	V41050
1237	1236	1235	1234	1233	1232	1231	1230	1227	1226	1225	1224	1223	1222	1221	1220	V41051
1257	1256	1255	1254	1253	1252	1251	1250	1247	1246	1245	1244	1243	1242	1241	1240	V41052
1277	1276	1275	1274	1273	1272	1271	1270	1267	1266	1265	1264	1263	1262	1261	1260	V41053
1317	1316	1315	1314	1313	1312	1311	1310	1307	1306	1305	1304	1303	1302	1301	1300	V41054
1337	1336	1335	1334	1333	1332	1331	1330	1327	1326	1325	1324	1323	1322	1321	1320	V41055
1357	1356	1355	1354	1353	1352	1351	1350	1347	1346	1345	1344	1343	1342	1341	1340	V41056
1377	1376	1375	1374	1373	1372	1371	1370	1367	1366	1365	1364	1363	1362	1361	1360	V41057
1417	1416	1415	1414	1413	1412	1411	1410	1407	1406	1405	1404	1403	1402	1401	1400	V41060
1437	1436	1435	1434	1433	1432	1431	1430	1427	1426	1425	1424	1423	1422	1421	1420	V41061
1457	1456	1455	1454	1453	1452	1451	1450	1447	1446	1445	1444	1443	1442	1441	1440	V41062
1477	1476	1475	1474	1473	1472	1471	1470	1467	1466	1465	1464	1463	1462	1461	1460	V41063
1517	1516	1515	1514	1513	1512	1511	1510	1507	1506	1505	1504	1503	1502	1501	1500	V41064
1537	1536	1535	1534	1533	1532	1531	1530	1527	1526	1525	1524	1523	1522	1521	1520	V41065
1557	1556	1555	1554	1553	1552	1551	1550	1547	1546	1545	1544	1543	1542	1541	1540	V41066
1577	1576	1575	1574	1573	1572	1571	1570	1567	1566	1565	1564	1563	1562	1561	1560	V41067
1617	1616	1615	1614	1613	1612	1611	1610	1607	1606	1605	1604	1603	1602	1601	1600	V41070
1637	1636	1635	1634	1633	1632	1631	1630	1627	1626	1625	1624	1623	1622	1621	1620	V41071
1657	1656	1655	1654	1653	1652	1651	1650	1647	1646	1645	1644	1643	1642	1641	1640	V41072
1677	1676	1675	1674	1673	1672	1671	1670	1667	1666	1665	1664	1663	1662	1661	1660	V41073
1717	1716	1715	1714	1713	1712	1711	1710	1707	1706	1705	1704	1703	1702	1701	1700	V41074
1737	1736	1735	1734	1733	1732	1731	1730	1727	1726	1725	1724	1723	1722	1721	1720	V41075
1757	1756	1755	1754	1753	1752	1751	1750	1747	1746	1745	1744	1743	1742	1741	1740	V41076
1777	1776	1775	1774	1773	1772	1771	1770	1767	1766	1765	1764	1763	1762	1761	1760	V41077