Allen-Bradley

DeviceLogix Technology for Industrial Applications

Additional Resources

These documents contain additional information concerning related products from Rockwell Automation.

Resource	Description
Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1	Provides general guidelines for installing a Rockwell Automation industrial system.
Product Certifications website, http://ab.rockwellautomation.com/	Provides declarations of conformity, certificates, and other certification details.
DeviceLogix System User Manual, publication RA-UM003	Provides Information on the DeviceLogix™ Editor Tool.
Armorstart® LT User Manuals, publications 290E-UM001; 290D-UM001	Provides installation and wiring instructions for the ArmorStart motor controllers. Also provides information on how to set parameters and troubleshoot the device.
Getting Results with RSNetWorx for DeviceNet, publication DNET-GR001G-EN-E	Provides information on RSNetWorx™ for DeviceNet software and the DeviceLogix Editor Tool.
E3 and E3 Plus Solid-State Overload Relay User Manual, publication 193-UM002	Provides mounting and installation instructions for E3™ and E3 Plus™ overload relays.
SMC™-50 Solid-State Smart Motor Controller User Manual, publication 150-UM011	Provides mounting and installation instructions for SMC-50 soft starter module.

You can view or download publications at http://www.rockwellautomation.com/global/literature-library/overview.page. To order paper copies of technical documentation, contact your local Allen-Bradley distributor or Rockwell Automation sales representative.

Introduction

There is an emerging need for highly flexible and adaptable 'smart' devices in control system architectures—devices that can execute simple control functions by making decisions independent of a central processor. You can achieve this local decision-making ability at the component level with DeviceLogix™ component technology from Rockwell Automation. DeviceLogix technology allows simple Boolean logic execution, signal conditioning, event detection, and alarms. The result is higher-performance and lower-cost distributed control.

What Is DeviceLogix Technology?

DeviceLogix technology is a platform-independent logic engine that is embedded into several Rockwell Automation devices, such as push button stations, I/O blocks, motor starters, and drives. This logic engine controls outputs and manages status information locally within a device.

The configuration of the DeviceLogix functionality is accomplished through the DeviceLogix editor that runs on the workstation and can be accessed through an RSLogix 5000® Add-On profile or other Rockwell Automation software. (Table 1 on page 8).

The DeviceLogix editor includes two kinds of logic configuration tools for DeviceLogix devices. You can use either one, depending on your configuration preferences:

- Function Block Editor – Provides a graphical interface for configuring functional blocks to provide local control within for devices capable of DeviceLogix technology. (Figure 1)
- Ladder Editor – Provides a ladder-style configuration tool for for devices capable of DeviceLogix technology. (Figure 2)

Figure 1 - Function Block Diagram

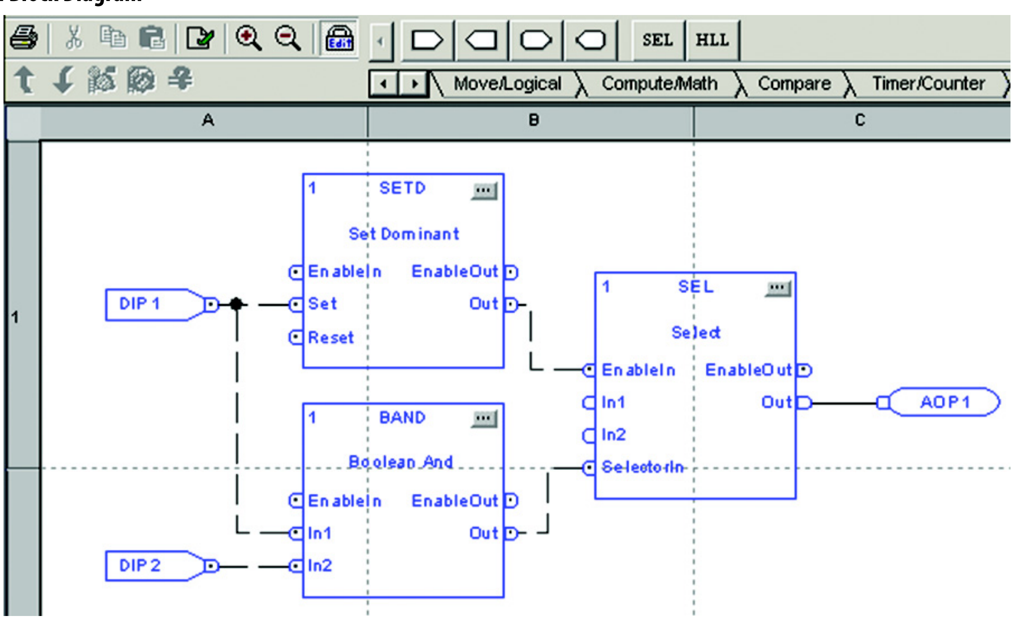
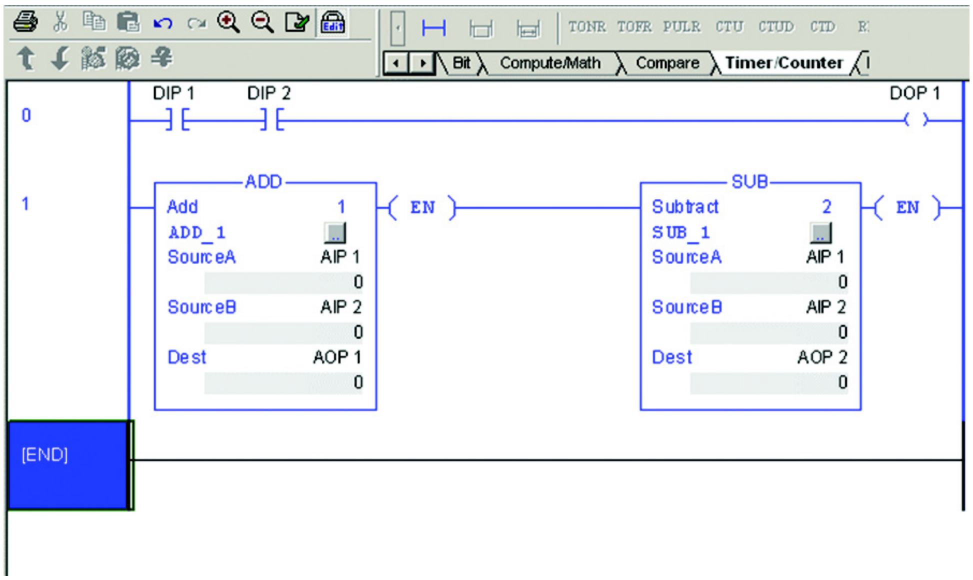
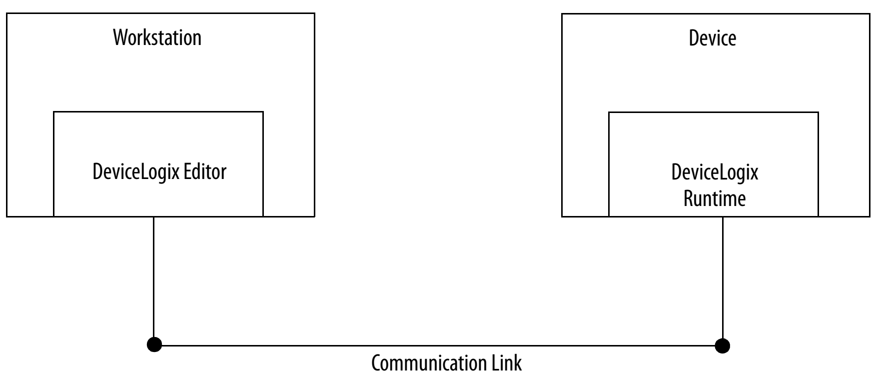


Figure 2 - Ladder Logic Diagram



The DeviceLogix run-time engine (Runtime) embedded into the device firmware is what executes DeviceLogix functionality. This connection is illustrated in Figure 3.

Figure 3 - DeviceLogix Technology Communication



You can describe the working process of using DeviceLogix-enabled products as follows:

1. You program the application logic in the editor.
2. The editor compiles the graphic logic into downloadable code that can be downloaded to the device through a communication link.
3. The device run-time engine executes the downloaded logic.

In addition to creating a run-time code in the Editor, you must also configure device parameters. Each DeviceLogix-enabled device contains a series of input, output, scratchpad, and datalink parameters. These device parameters are used to map physical I/O to DeviceLogix inputs and output.

You can configure DeviceLogix technology to operate under specific situations. The DeviceLogix program runs only if the logic has been enabled and unswitched power is present. You can do this within the Logic Editor. Operation configuration is accomplished by setting the "Network Override" and "Communication Override" parameters. The following describes the varying levels of operation:

- If both overrides are disabled and the logic is enabled, DeviceLogix **only** runs if there is an active I/O connection with a master, that is, the master is in Run mode. At all other times DeviceLogix runs the logic but does NOT control the state of the outputs.
- If the Network Override is enabled and the logic is enabled, then DeviceLogix controls the state of the outputs when the PLC is in Run mode and if a network fault occurs.
- If the Communications Override is enabled and the logic is enabled, the device does not need any I/O connection to run the logic. As long as there are switched and unswitched power sources connected to the device, the logic controls the state of the outputs.

Benefits of using DeviceLogix Technology

DeviceLogix technology provides local control at the component level, and brings a level of intelligence to devices that was not previously available. This allows the capability of having simple local control functions without the need to rely on the programmable logic and automation controllers.

Improve Control System Performance

- Faster input sense to out actuation

  DeviceLogix-enabled smart components are well suited for applications that require a quick, local response to an input without the performance constraints and sense-to-activation delays that are imposed by network round-trip times with a centralized controller and a communications network. You can achieve input-sense to output actuation times in the range of 2 ms with several product families; this yields significant improvements in system performance.
- Reduction in overall network traffic

  Performing simple repetitive logic function at the Device level reduces the amount of communications traffic, which in turn reduces network bandwidth delays.

Increase System Reliability

- Improved diagnostics and reduced troubleshooting-

  Processing much of the logic at the component level makes troubleshooting much easier. You can take a device offline without affecting other devices in the process, and you can do it without needing an operator to modify logic within a master controller that isolates the device that needs troubleshooting.
- Continue to run processing the event of network interruptions

  In a use case where a failure in the central programmable logic controller (PLC) could occur, the component device can identify the failure by detecting an interruption in the communication with it. You can design the DeviceLogix program running on this device to respond to such a failure by turning the field process under its local control into a pre-defined state. This could either allow the controlled process to keep running, with the failure of the central PLC, or shut down the process appropriately to meet the design requirements of your application. For more time-sensitive applications such as mixing cement, this type of functionality is imperative because a machine shutdown during the process could be costly.

- Reduce potential points of system-level failure

  On-board logic processing capability of DeviceLogix technology eliminates the need for extra components such as timers or relays that could not only introduce an additional delay to the signal processing, but also become a potential point of failure. Reducing extra components greatly reduces the complexity of the system, while also lowering its cost.

Increase System /Machine Modularity

- Smaller programs in the controller

  Smaller programs in controller (PLC or PC-based), saves memory and reduces scan times.

- Distributing control on the machine or process

  You can copy and paste programs that are written with DeviceLogix functionality to other devices and essentially use them as “blocks” that you can reuse over multiple devices of the same type. You can build machine sections in a modular fashion, which speeds installation while reducing diagnostic and troubleshooting times.

Following are two examples that further illustrate the benefit of having a control system with DeviceLogix-enabled devices.

In a conventional setup, the input module receives the sensor signals. The module processes and waits for a time slot for the data transmission via the fieldbus. Once transferred over the network a scanner card picks up the information and sends it to the PLC. The PLC scans the stored program to execute the logic at the scheduled timing cycle. Then the appropriate output data travels back to the scanner, onto the network, and to the output module, finally energizing output devices. When you add the time required at the beginning of each logic scan for the input data and the time required updating the output data after the scan, the total time the PLC requires becomes significant. This is particularly true when the PLC is fully loaded running the application in the control system and as the number of nodes on the network increases.

DeviceLogix functionality significantly reduces reaction time. Input signals still travel to the input terminal. But because program logic resides at the DeviceLogix node, data is processed and output is generated much more quickly. Local processing time is a fraction of the time required for data to travel back and forth through the network connection. You can gain improvements of approximately 90…95% of the response time when you use DeviceLogix-based local control instead of a traditional PLC-based field bus control system.

Another system benefit is the advantage of greater flexibility for designing and constructing a simple control application system.

In a typical architecture for an application that uses a conventional fieldbus control system with DeviceLogix-enabled component devices, you would implement the so-called hierarchy control strategy. The component device running the local DeviceLogix program can control the local loop with faster response time as well as off-loading the PLC, which could also reduce the size and complexity of the PLC control program. The amount of savings is application specific.

Applications for DeviceLogix Functionality

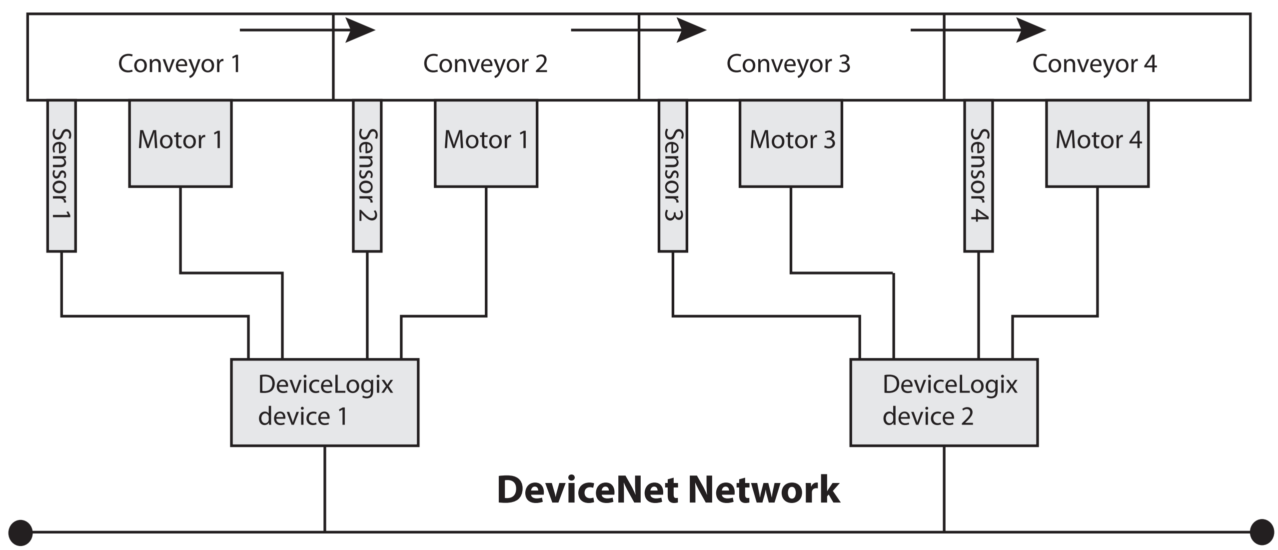
DeviceLogix functionality has many applications and the implementation is typically only limited to the imagination of the programmer. The application of DeviceLogix functionality is only designed to handle simple logic routines.

Applications can be simple, using digital I/O for signal actuation, or perform more complex functions by using Analog signals and user-defined Macro functions. Some of the typical applications for DeviceLogix functionality are listed below.

- Material Counting for Batch Processes
- Encoder Functions
- Selector Switch Functions
- Signal Conditioning
- Signal Scaling
- Fault Handling
- Complex Math Function/ Integration
- Temperature Control
- Speed Reference Selection
- Motor Control Applications

This material handling application uses DeviceLogix-enabled devices to construct a truly distributed, simple, control system with no central controllers, such as a PLC or PAC. Figure 4 demonstrates the design flexibility of simple control system that uses DeviceLogix technology.

Figure 4 - Typical Material Handling Application



To demonstrate the use of DeviceLogix functionality in the material handling application in Figure 4, the installation structure is simplified to show only the two control segments of the system. In reality, the whole system contains many more segments. However, the principle of using DeviceLogix functionality is the same regardless of the scale of the system.

In this application, each segment is composed of two conveyors that are used to transfer the material downstream. A DeviceLogix-enabled device controls each segment. For each conveyor in the segment, a DeviceLogix-enabled controller is installed along with a sensor to detect the existence of the material on the conveyor, and with a motor to drive the conveyor while it turns and moves the material. Between the segments, the DeviceLogix devices are connected by a DeviceNet network for the data exchange in a peer-to-peer communication manner.

The purpose of this application is to reduce the amount of energy consumed by the motors. The design calls for the motor to work only when material is detected on the conveyor to transfer. When the material is delivered to next conveyor, the motor stops.

Figure 5 gives the illustrative algorithm using DeviceLogix technology and is executed in both DeviceLogix device1 and DeviceLogix device2 for this application. Figure 6 shows the completed DeviceLogix functionality for this material handling application.

Figure 5 - Illustrative Algorithm in Devices for Material Handling Application

If *Sensor 1* is ON Turn on Motor 1; If *Sensor 2* is ON Turn off Motor 1; Turn on Motor 2; If *P2P_Sensor 3* is ON	If *Sensor 3* is ON Turn on Motor 3; If *Sensor 4* is ON Turn off Motor 3; Turn on Motor 4; Send out the data of *Sensor 3*

Figure 6 - Completed DeviceLogix Functionality for Material Handling Application

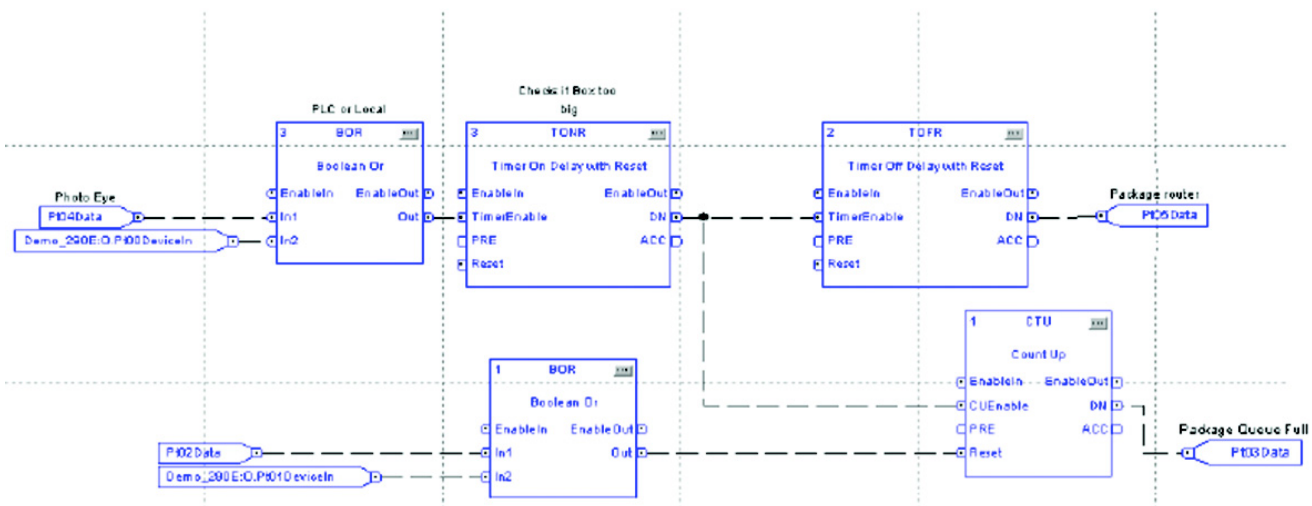


Table 1 - Devices Capable of DeviceLogix Technology

	RSNetWorx™ for DeviceNet Software	DriveExplorer™ Software	DriveTools™ SP/ DriveExecutive™ Software	RSLogix 5000 Add-On Profile	Connected Components Workbench™ Software
800T Push Button Station	vX.01 (and higher)	—	—	—	—
E3 Plus™ Electronic Overload Relay	vX.01 (and higher)	—	—	—	—
E300 Electronic Overload Relay	—	—	—	V5.01 (and higher)	—
ArmorStart Bulletin 280/ 281/284	—	—	—	vX.01 (and higher)	—
ArmorStart Bulletin 290/ 291/294	—	—	—	vX.01 (and higher)	—
SMC™-50 v4.02	—	—	V5.01 (and higher)	v4.01 (and higher)	V6.01 (and higher)
PowerFlex™ 755 v1.xx	—	v6.01 (and higher)	v5.01 (and higher)	v2.01 (and higher)	v1.02 (and higher)
PowerFlex 753 v1.xx – x5.xx PowerFlex 755 v2.xx – x5.xx	—	v6.02 (and higher)	v5.02 (and higher)	v2.02 (and higher)	v1.02 (and higher)
PowerFlex 753 v6.xx – x7.xx PowerFlex 755 v6.xx – x7.xx	—	v6.04 (and higher)	v5.05 (and higher)	v4.02 (and higher)	v1.02 (and higher)

Available Functions in DeviceLogix Technology

With the development evolution of DeviceLogix technology, nowadays it can support the most frequently used instructions in the industry to build the logic program; specifically, the overall instruction set available for integration with device includes the following elements.

Table 2 - Process Category Toolbar

Function	Description
ALM	Alarm Instruction
TDG	Timing Diagnosis Instruction
PID	PID Instruction

Table 3 - Filter Category Toolbar

Function	Description
LPF	Low Pass Filter Instruction

Table 4 - Select/Limit Category Toolbar

Function	Description
SEL	Select Instruction
HLL	High Low Limit Instruction
LPF	Low Pass Filter Instruction

Table 5 - Statistical Category Toolbar

Function	Description
MAVE	Moving Average Instruction

Table 6 - Timer/Counter Category Toolbar

Function	Description
PULR	Pulse Timer Instruction
TONR	On-Delay Timer Instruction
TOFR	Off-Delay Timer Instruction
CTU	Up Counter Instruction
CTUD	Up Down Counter Instruction

TIP When using the ACC feature, you may notice some minor differences in the Timer/Counter instructions. Depending on the firmware implementation, some devices consider this an output and bind it to an analog output tag, while other devices consider it a parameter.

Table 7 - Compare Category Toolbar

Function	Description
GRT	Greater Than Instruction
GEQ	Greater Than or Equal To Instruction
EQU	Equal Instruction
NEQ	Not Equal Instruction
LES	Less Than Instruction
LEQ	Less Than or Equal Instruction
MEQ	Mask Instruction

Table 8 - Compute/Math Category Toolbar

Function	Description
ADD	Add Instruction
MUL	Multiply Instruction
SUB	Subtract Instruction
DIV	Divide Instruction
MOD	Modulus (DINT) Instruction
MOD	Modulus (REAL) Instruction
ABS	Absolute Instruction
NEG	Negative Instruction
SQR	Square Root Instruction
EXP	Power Instruction

Table 9 - Move/Logical Category Toolbar

Function	Description
BAND	AND Instruction
BNAND	Not AND Instruction
BOR	OR Instruction
BNOR	Not OR Instruction
BXOR	Exclusive OR Instruction
BXNOR	Exclusive Not OR Instruction
BNOT	NOT Instruction
SETD	Set Latch Instruction
RSTD	Reset Latch Instruction

In addition to the common instructions listed above, DeviceLogix functionality can also support user-customized instructions that encapsulate logic you develop with the built-in instruction set. This feature is called a Macro instruction, with which you can design customized application instructions for the target device.

Not all functions are available for each device. Each device type can support just enough of the overall DeviceLogix functionality as that device functionality requires.

For example, a device with only a few digital physical inputs and outputs is required to support only the logic instructions using digital data as its inputs and outputs, including Boolean, Timer, and Counter. Instructions that use analog data are not needed and moved out of the scope supported by this device. This can reduce the resource requirements in the run-time, and create a more cost-effective and easier-to-use product with just enough functionality.

DeviceLogix Configuration Tool

The configuration of the DeviceLogix functionality is accomplished through the DeviceLogix Editor. The DeviceLogix Editor includes two kinds of logic configuration tools for DeviceLogix devices to meet different configuration preferences:

- Function Block Editor - provides a graphical interface for configuring function blocks to provide local control within for devices capable of DeviceLogix technology.
- Ladder Editor - provides a ladder-style configuration tool for devices capable of DeviceLogix technology. Beginning with firmware revision 4, two Ladder Editors are available within DeviceLogix software. The Old Ladder Editor supports DeviceLogix firmware revision 3 and earlier. The New Ladder Editor supports DeviceLogix firmware revision 4.

With devices capable of DeviceLogix technology, you can enable a logic operation using the DeviceLogix Editor to provide local control over the device's operation. A DeviceLogix device consists of:

- A specific number of inputs and/or outputs.
- Local logic that determines its behavior.

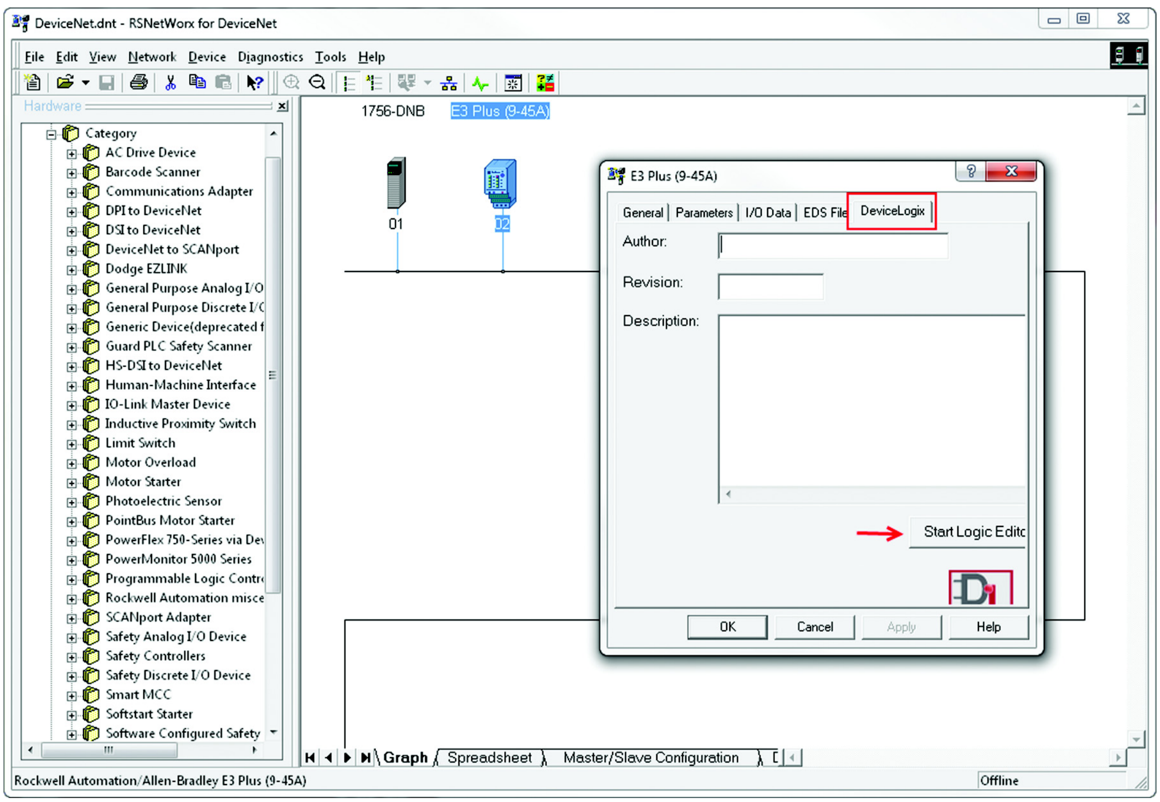
The DeviceLogix Editor is an applet of RSNetWorx for DeviceNet software, Drive Tools software, and Connected Components Workbench software, and it can be launched directly from those hosts.

With the appropriate EDS File, the DeviceLogix Editor Tool can also be launched from RSLogix 5000 software.

The examples that follow show where to access the DeviceLogix Editor Tool.

The DeviceLogix Editor Tool through RSNetWorx for DeviceNet software can be accessed by selecting the device node while in the network view or RSNetWorx software. The Device Properties window pops up. If the device is a DeviceLogix-enabled device, the DeviceLogix tab is present. Click on the DeviceLogix tab and select the Start Logic Editor.

Figure 7 - Start Logic Editor



You can also access the DeviceLogix Editor tool through the Connected Component Workbench software. In this software, add the DeviceLogix-enabled device to the Project Organizer by searching for it under the Device Toolbox Catalog or by using the Discover feature. Once the device is in the Project Organizer, double-click on it to display the Device Menus. If the device is DeviceLogix enabled, the DeviceLogix icon appears. Double-click on it to launch the DeviceLogix Editor.

Figure 8 - Launching the DeviceLogix Editor Tool through the Connected Component Workbench Software

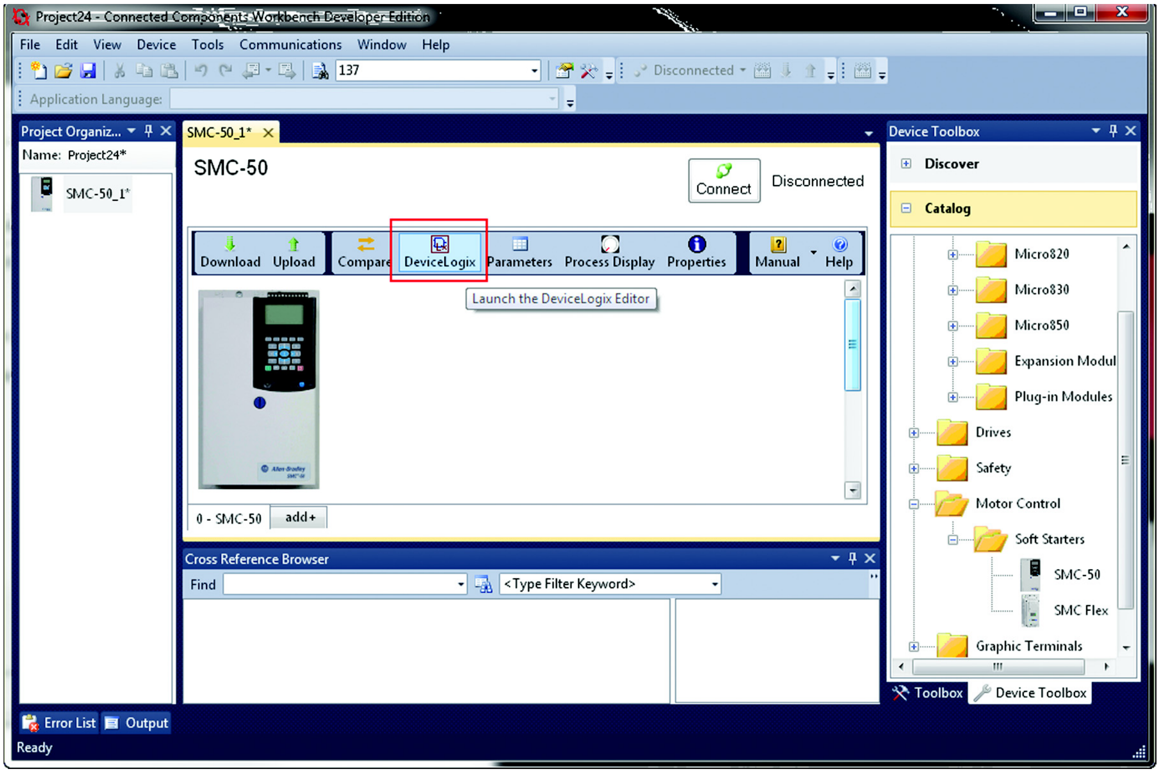


Figure 9 - Launching the DeviceLogix Editor Tool through the RSLogix Add-On Profile

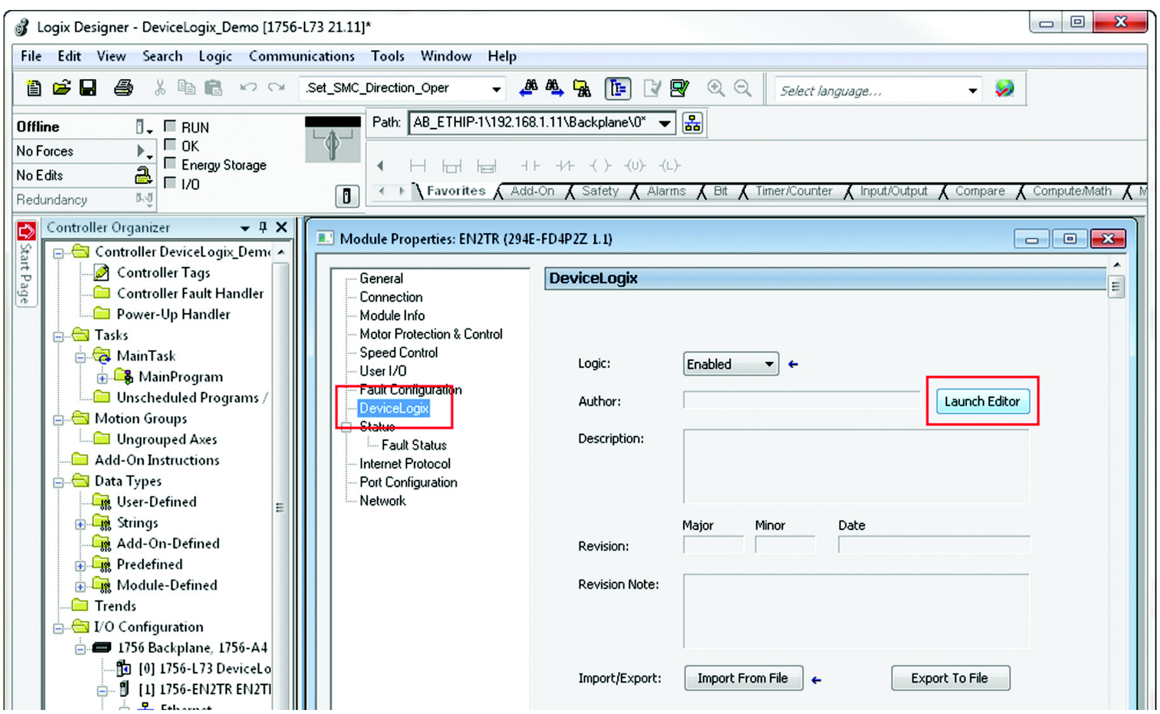
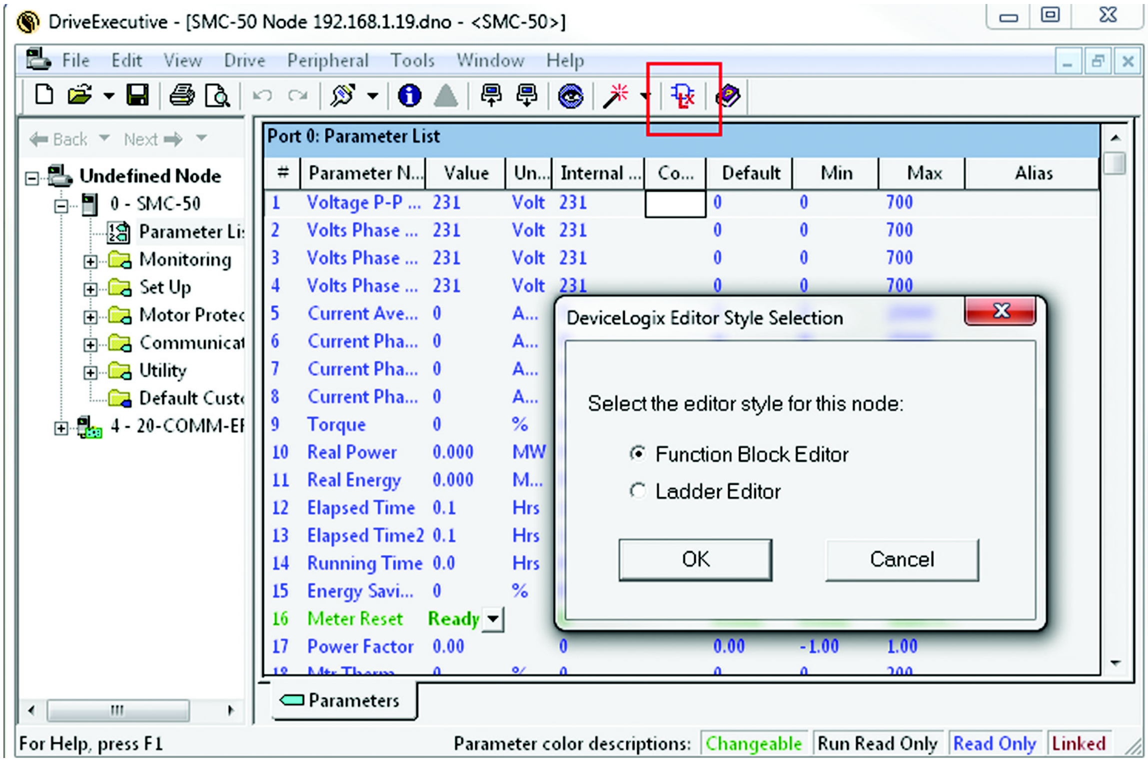


Figure 10 - Launching the DeviceLogix Editor Tool through the DriveTools DriveExecutive



Once you select the Icon for the DeviceLogix Editor Tool (Figure 11), the Style Selection box appears and prompts you to select whether you would like to proceed with a Function block (Figure 12) or Ladder program.

Figure 11 - DeviceLogix Editor Style Selection

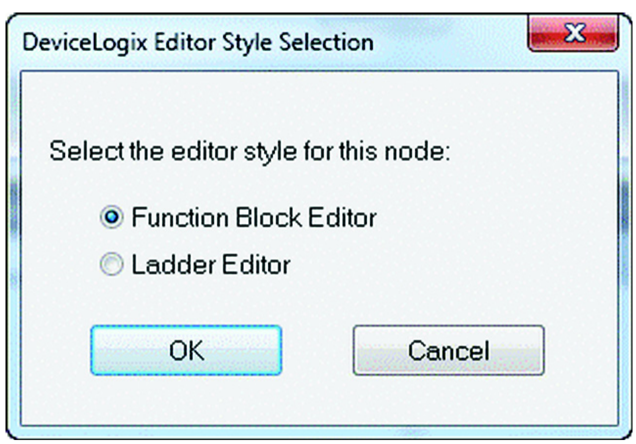
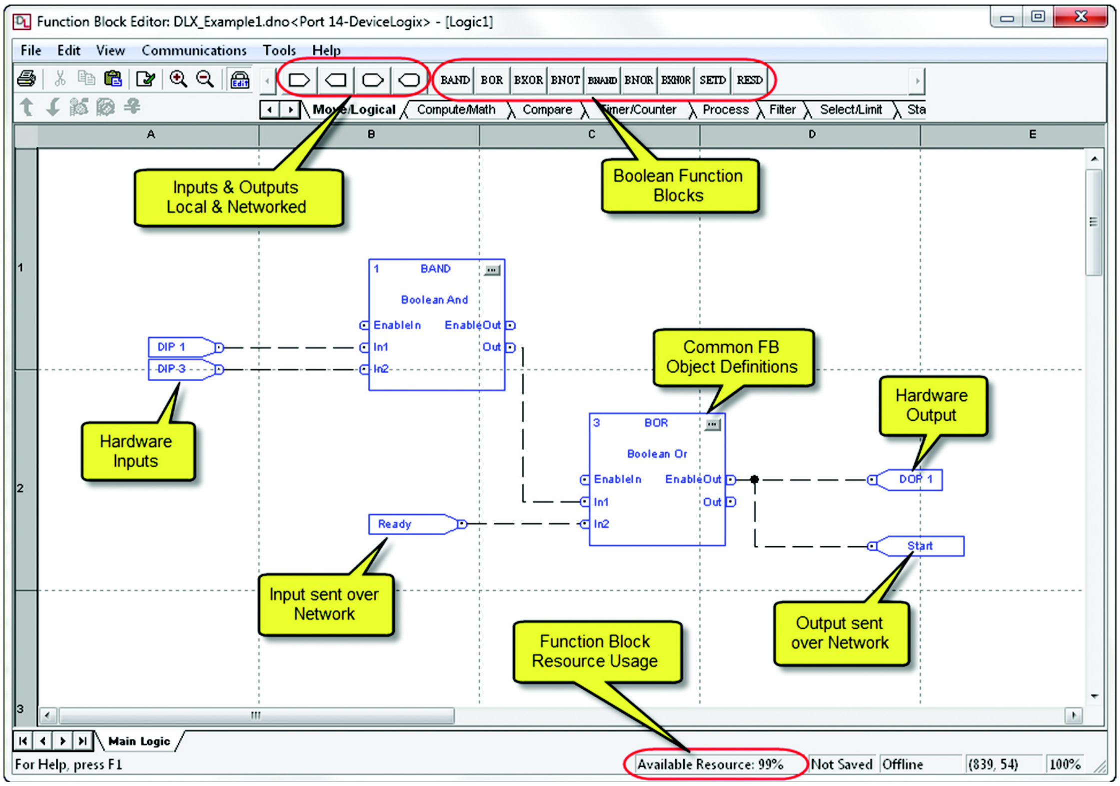


Figure 12 - Function Block Editor

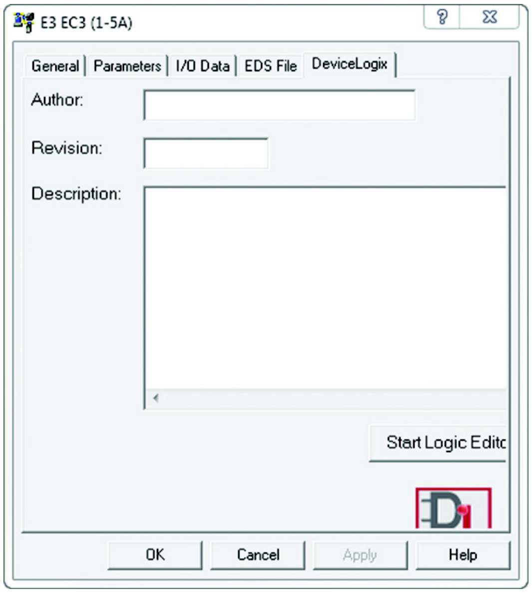


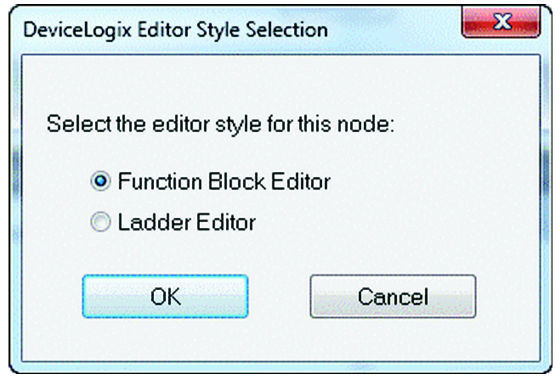
For more information on the DeviceLogix Editor, refer to the DeviceLogix System User Manual, publication RA-UM003B.

DeviceLogix Programming Example

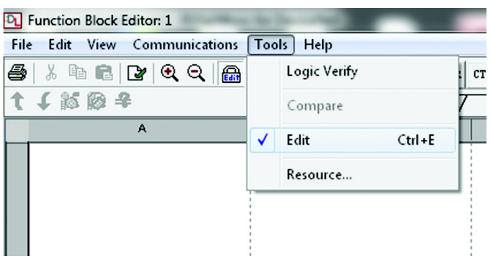
The following example shows how to program a simple logic routine to control the E3 Plus Overload Relay outputs based on the condition of input signals. OUT A control is defined by the states of IN1 and IN2 processed through a Boolean OR gate. OUT B control is defined by the states of IN3 and IN4 processed through a separate Boolean OR gate. This example is using RSNetWorx for DeviceNet software, version 8.00.01, and an E3 Plus Series C overload relay.

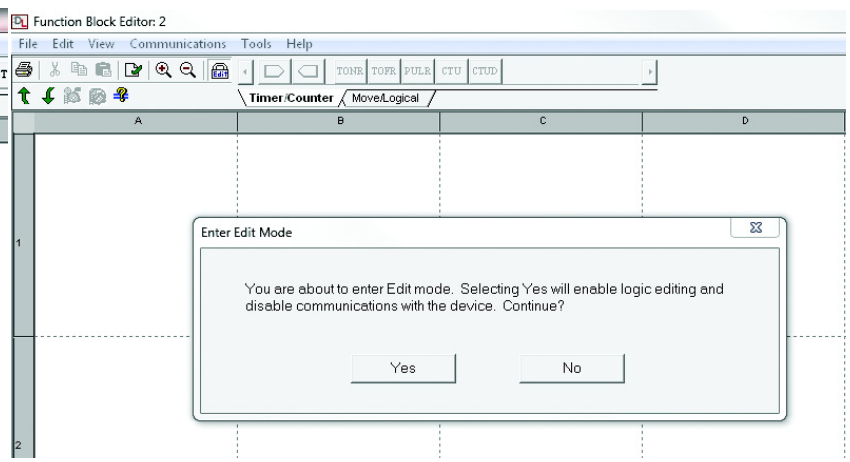
1. While in RSNetWorx for DeviceNet software, double-click the E3 Plus relay, select the DeviceLogix Tab.
2. If you are prompted with a dialog box while online with the E3 Plus, then select Upload.
3. Select Start Logic Editor, select the Function Block Editor, and press OK.



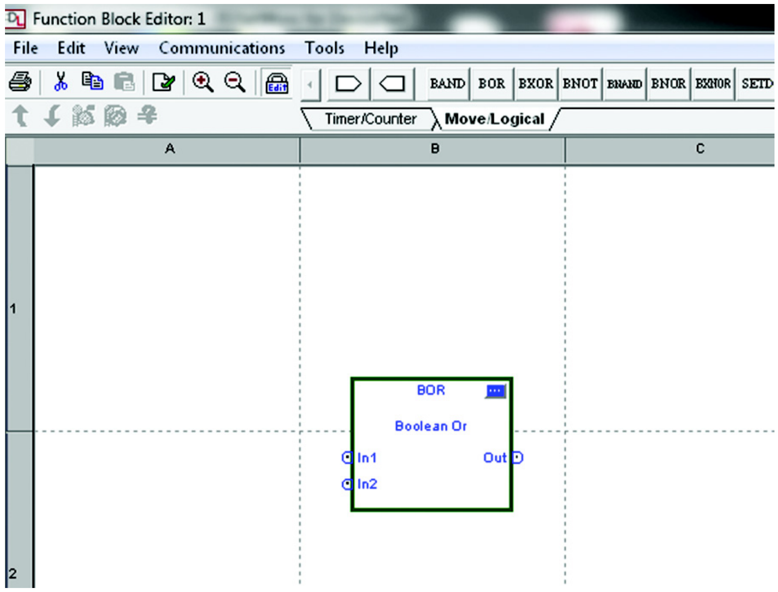


4. If you are programming offline, then continue to Step 5. If you are programming online, place the E3 Plus relay into Edit mode by selecting Edit from the Tools pull-down menu or by selecting the button. Select **Yes** to enter Edit mode.

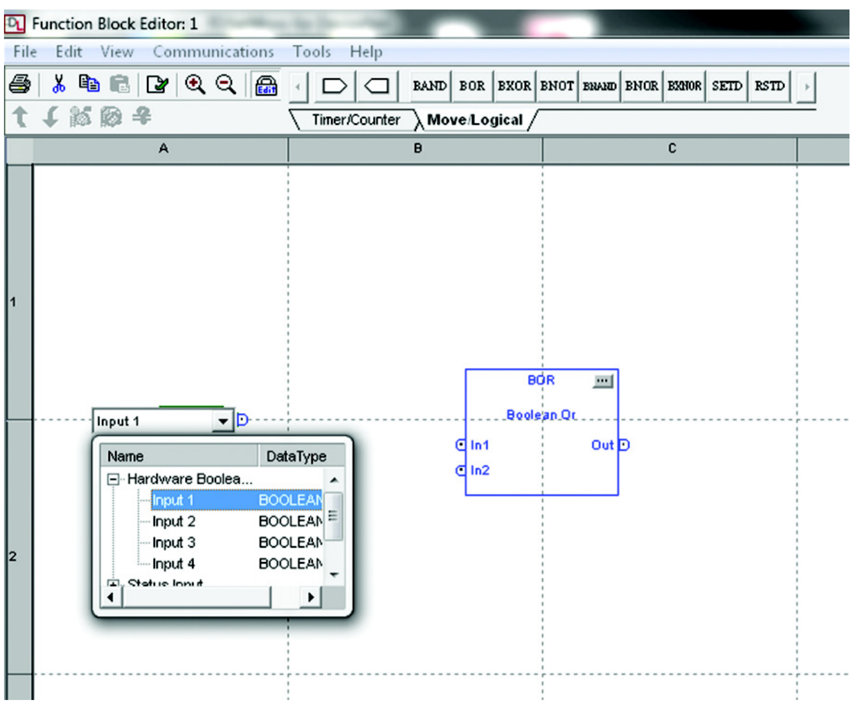




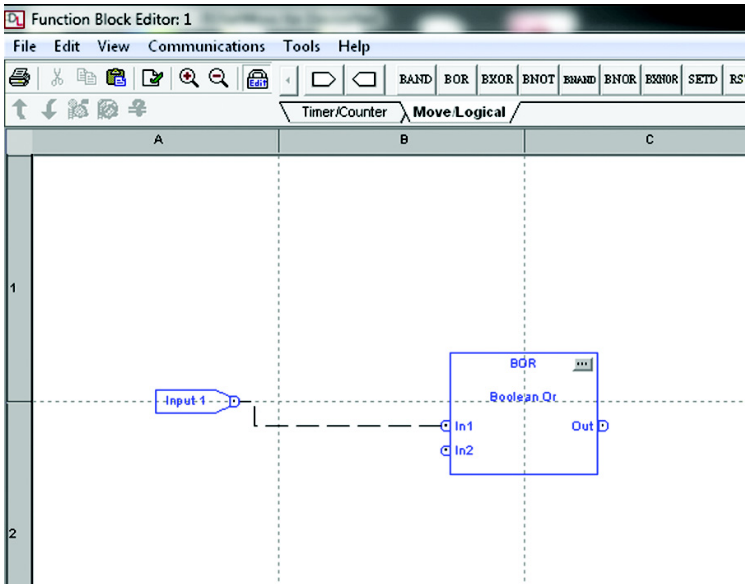
5. Using the left mouse button, select the Boolean OR (BOR) function block from the Move/Logical tab and drag it onto the display.



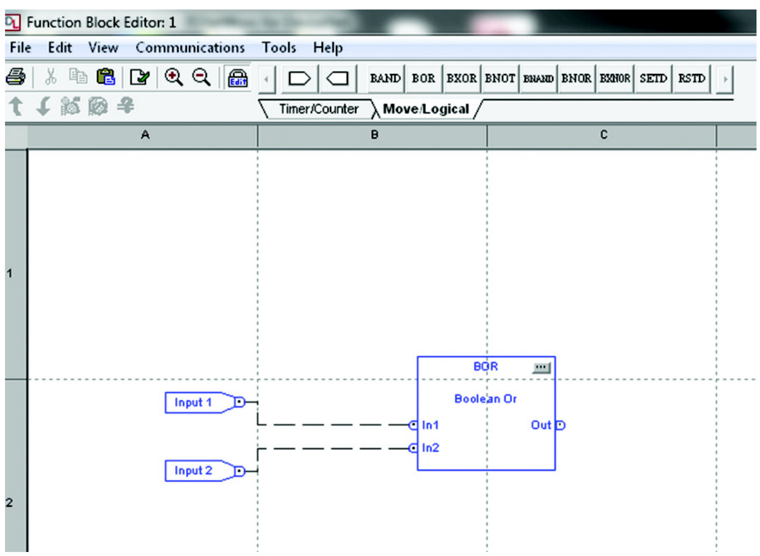
6. Using the left mouse button, select the Bit Input block and drag it to the left of the BOR function block. Double-click on the Bit Input block to select Input 1 of the E3 Plus relay under the Hardware Boolean Input set.



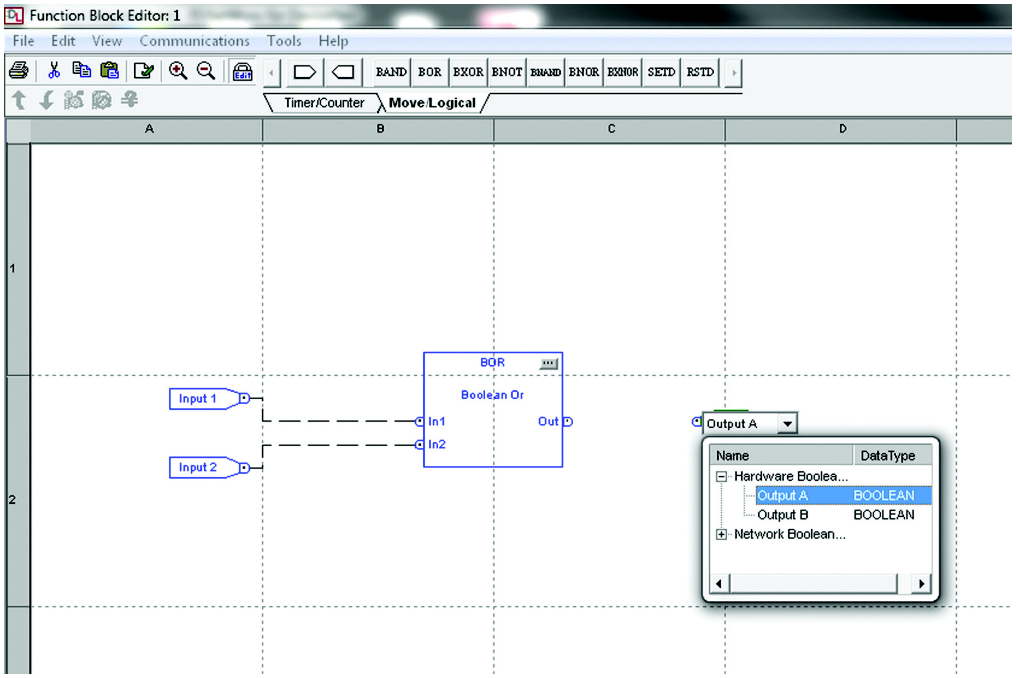
7. Place the cursor to the right of the Bit Input block and press the left mouse button. Draw a line from the Bit Input block to the In1 of the BOR function block. Double-click the left mouse button to establish a connection.



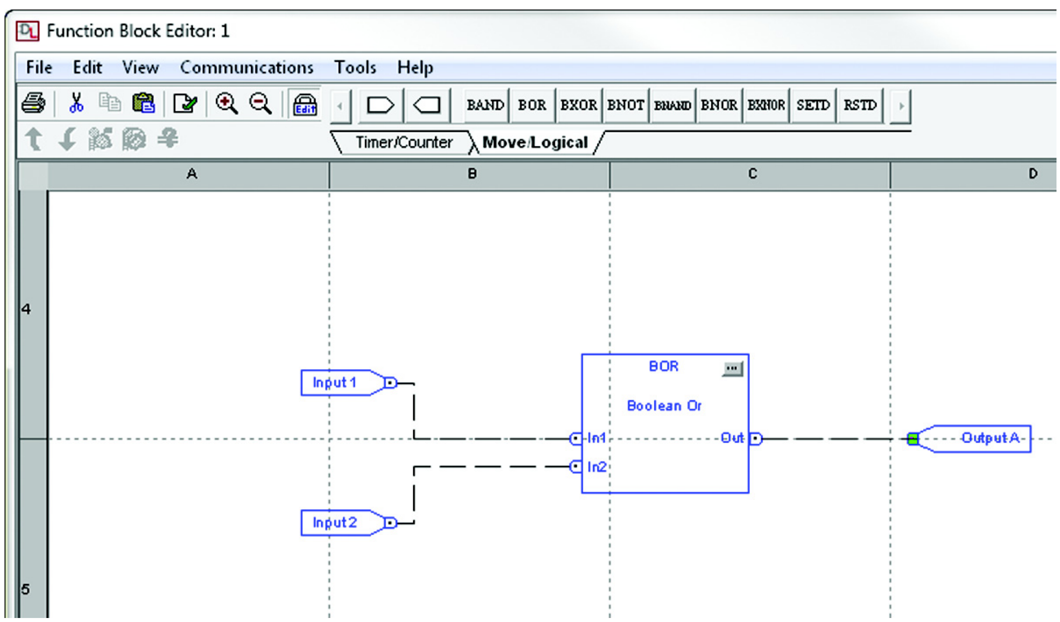
8. Repeat steps 6 and 7 to add Input 2 of the E3 Plus to In2 of the BOR function block.



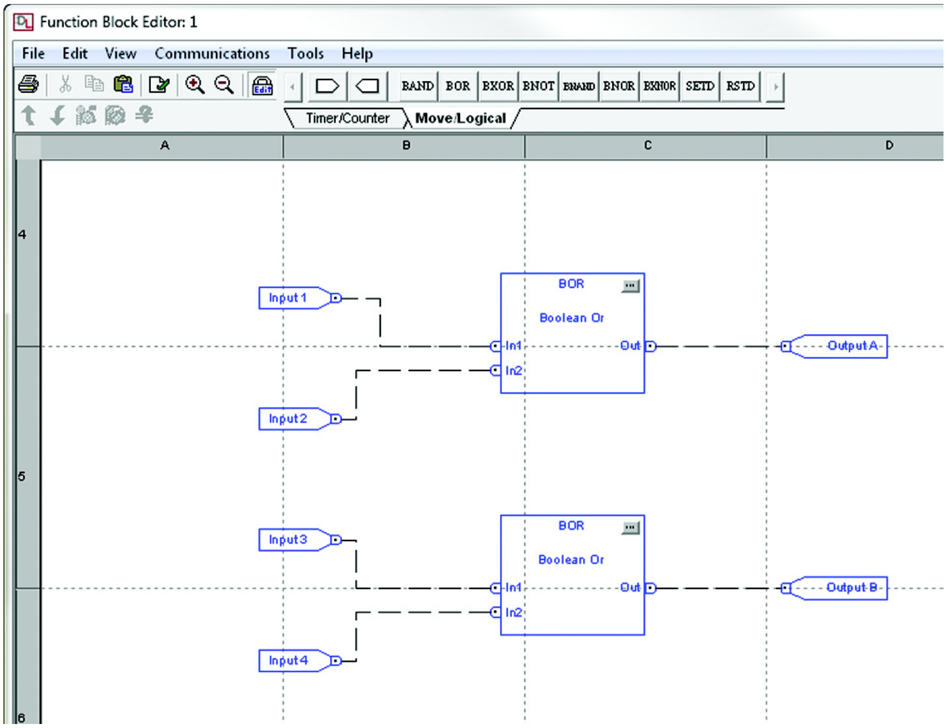
9. Using the left mouse button, select the Bit Output block and drag it to the right of the BOR function block. Double-click the Bit Input block to select Output A of the E3 Plus relay under the Hardware Boolean Output set.



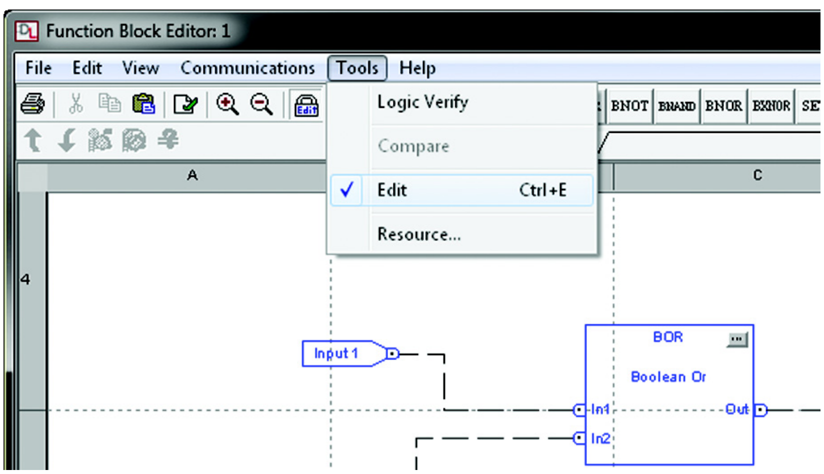
10. Place the cursor on the left of the Bit Output block and press the left mouse button. Draw a line from the Bit Output block to the Out of the BOR function block and double-click the left mouse button to establish a connection.



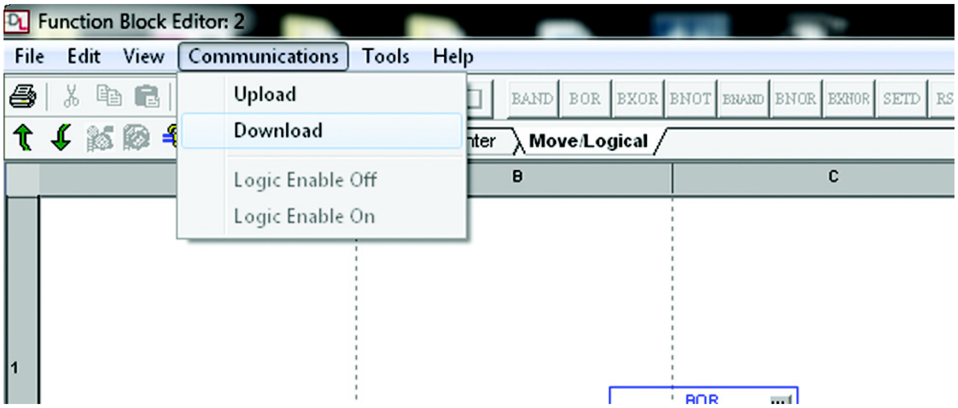
11. Repeat steps 5 through 10 to add a second BOR function block that monitors Input 3 and 4 to control Output B of the E3 Plus relay.



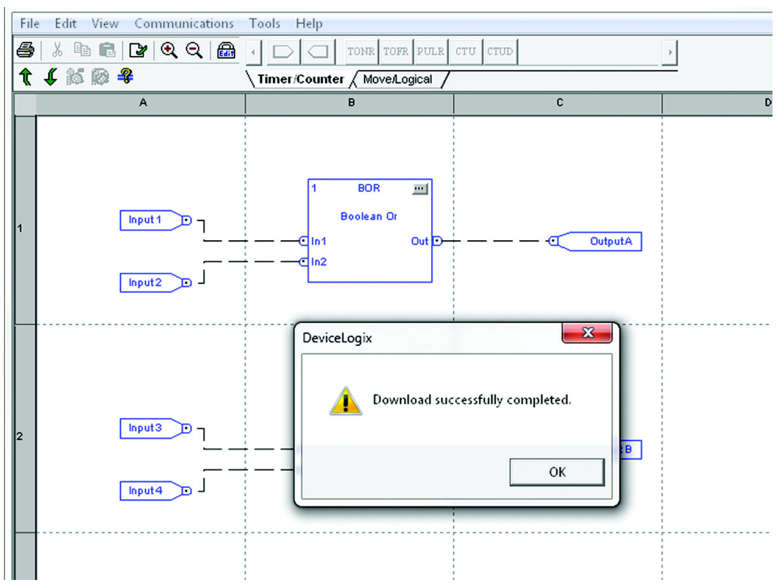
12. Disable Edit mode by de-selecting the Edit mode button or through the Tools menu.



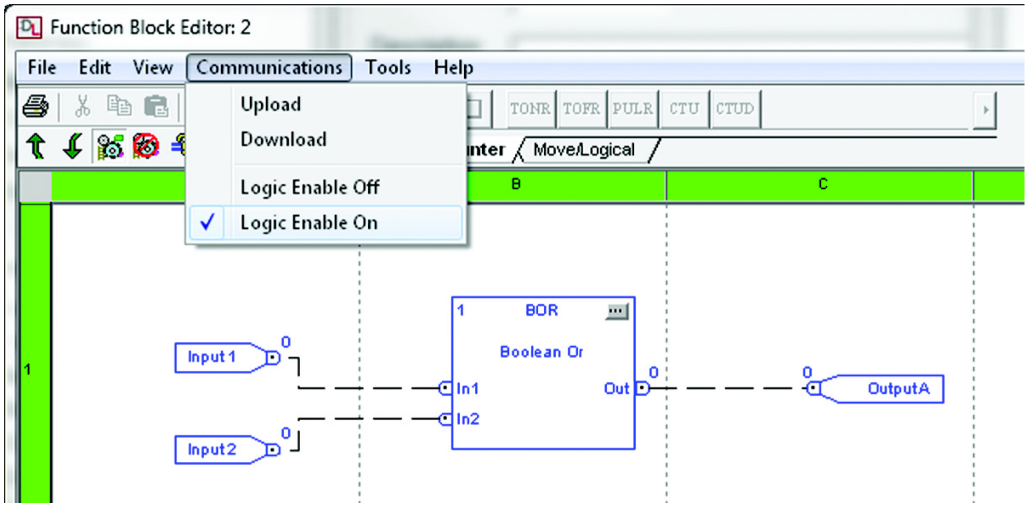
13. If you are programming the function block offline, exit the Function Block editor and go online with the Device Network. Download the parameters, including the DeviceLogix Function Blocks, to the E3 Plus relay and proceed to Step 12. If you are programming the function block online, download the Function Blocks to the E3 Plus relay by selecting the Download button or through the Communications menu.



A dialog box appears when the Function Blocks are successfully downloaded to the E3 Plus relay. Click OK to continue.



14. Enable the DeviceLogix Function Blocks. You can do this through a DeviceNet Configuration Terminal (Catalog Number 193-DNCT) or through RSNetWorx software. If you use RSNetWorx software to enable the DeviceLogix function blocks, select the Logic Enable On button or enable the blocks through the Communications menu.



15. Verify the functionality of the DeviceLogix Function Blocks by enabling Input 1 on the E3 Plus relay. When Input 1 is enabled, Output A energizes.

Notes:

Important User Information

Read this document and the documents listed in the additional resources section about installation, configuration, and operation of this equipment before you install, configure, operate, or maintain this product. Users are required to familiarize themselves with installation and wiring instructions in addition to requirements of all applicable codes, laws, and standards.

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice.

If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete this form, publication RA-DU002, available at http://www.rockwellautomation.com/literature/.

Rockwell Automation maintains current product environmental information on its website at http://www.rockwellautomation.com/rockwellautomation/about-us/sustainability-ethics/product-environmental-compliance.page.

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 193-AT001B-EN-P - July 2016
Supersedes Publication 193-AT001A-EN-P December 2015