

FactoryFloor to PAC Project Migration

Introduction

This technical note describes how to migrate a control system that uses Opto 22 FactoryFloor[®] controllers and software applications—specifically OptoControl[™] and OptoDisplay[™]—to Opto 22's PAC Project[™] Professional software and SNAP PAC controllers. Note that PAC Project Pro (not PAC Project Basic) is required for this migration.

Other migration documents:

If you're doing this	See this document	Form #
Migrating from ioProject to PAC Project, or integrating older SNAP systems with the SNAP PAC System	SNAP PAC System Migration Technical Note	1688
Migrating from Cyrano to PAC Project (requires migrating to FactoryFloor as an intermediate step)	Cyrano to OptoControl Migration Technical Note	1169

The following sections are included in this technical note:

- [“Considerations Before You Migrate” on page 5](#)
- [“Preparing for Migration” on page 9](#)
- [“Converting OptoControl Strategies to PAC Control Pro” on page 10](#)
- [“Converting OptoDisplay Projects to PAC Display Pro” on page 31](#)

Why Migrate to PAC Project?

PAC Project Professional software and Opto 22 SNAP PAC[™] controllers provide a migration path for legacy control systems based on FactoryFloor software and *mistic*[™] controllers. Moving to contemporary hardware and software can expand your control system's capabilities while maintaining your investment in I/O units, sensors/actuators, wiring, and other components.

PAC Project Professional software applications are designed for more complex projects, especially Internet of Things (IoT) applications or projects requiring OPC (OLE for Process Control), database connectivity, multiple protocol, multiple network, or legacy hardware support.

PAC Project Professional is designed for use with SNAP PAC controllers and takes advantage of their features, including a RESTful API and HTTP/HTTPS web server for secure data access, and dual independent Ethernet network interfaces for redundant Ethernet links or segmented networking.

PAC Control requires the use of a SNAP PAC controller; a PAC Control strategy cannot be downloaded to an OptoControl controller.

See the section [“Comparing Opto 22 FactoryFloor and PAC Project Professional” on page 2](#) for specific features and more information.

Comparing Opto 22 FactoryFloor and PAC Project Professional

The following table compares the features of PAC Project Professional and FactoryFloor software suites. For additional comparisons, including controller capabilities and hardware compatibility charts, see Opto 22 form #1693, *Legacy and Current Product Comparison and Compatibility Charts*.

Feature	PAC Project Professional 8.x and higher	FactoryFloor 4.1
Included software	<ul style="list-style-type: none"> • PAC Control Professional • PAC Display Professional • OptoOPCServer™ • Utilities • OptoDataLink™ • SoftPAC™ softwarecontroller for PC-based control • PAC Manager™ 	<ul style="list-style-type: none"> • OptoControl • OptoDisplay • OptoServer™ • OptoUtilities™ <p>NOTE: An earlier version of FactoryFloor contained OptoConnect for database connectivity.</p>
Network	<ul style="list-style-type: none"> • <i>Controller to host</i>: Ethernet or PPP over dial-up modem • <i>Controller to I/O</i>: S-series—Ethernet and serial; R-series—Ethernet only • Ethernet or Serial to third-party devices • Support for Ethernet link redundancy or segmented control network 	<ul style="list-style-type: none"> • <i>Controller to host and I/O</i>: Serial, Ethernet, or ARCNET (depending on controller) • Serial to third-party devices
I/O unit compatibility	<ul style="list-style-type: none"> • SNAP PAC EB brains • SNAP PAC SB brains (S-series controllers only) • SNAP PAC R-series controllers (R-series have built-in I/O processor) • Serial <i>mistic</i> I/O units (S-series controllers only): B3000, SNAP-BRS, B100, B200, G4D16R, G4D32RS, G4A8R • SNAP Ultimate, SNAP Ethernet, SNAP Simple, E1, and E2 I/O units (not all 8.x features are available) 	<ul style="list-style-type: none"> • <i>mistic</i> protocol I/O units including <i>mistic</i> bricks, B3000, B100, B200, and G4 high-density analog • SNAP Ethernet and SNAP Simple I/O units (M4 bus controllers only; requires Ethernet card) • ARCNET B3000-HA, SNAP-BRS-HA I/O units (G4LC32; or M4 bus or G4LC32SX controllers with ARCNET card)
Control software		
Name	PAC Control Professional	OptoControl
Compatible controllers	<ul style="list-style-type: none"> • SNAP PAC S-series • SNAP PAC R-series 	<ul style="list-style-type: none"> • SNAP-LCM4 • SNAP-LCSX, SNAP-LCSX-PLUS • G4LC32 series • M4 and M4RTU series
Main features	<ul style="list-style-type: none"> • Flowchart programming • OptoScript™ programming • Subroutines (debuggable) • Graphical debugger • Conversion utility for OptoControl strategies (from version 4.1) • Support for serial <i>mistic</i> I/O units • Additional features on SNAP Ethernet brains (ramping, etc.) • Ethernet link redundancy 	<ul style="list-style-type: none"> • Flowchart programming • OptoScript programming • Subroutines (no debugging within strategy) • Graphical debugger • Cyrano™ conversion utility • Support for serial <i>mistic</i> I/O units
Maximum charts running at once	<ul style="list-style-type: none"> • 16 on SNAP PAC R-series (plus host task) • 32 on SNAP PAC S-series (plus host task) • 64 on SoftPAC 	32 (including host task)

Feature	PAC Project Professional 8.x and higher	FactoryFloor 4.1
Proportional-integral derivative (PID) loops	<ul style="list-style-type: none"> • 4 PID algorithms for Ethernet I/O units (96 PIDs per SNAP PAC R-series controller and SNAP PAC brain; 32 PIDs per SNAP Ultimate I/O unit; 16 PIDs per SNAP Ethernet I/O unit) • 1 PID algorithm for <i>mistic</i> serial (8 PIDs per <i>mistic</i> analog I/O unit) • Graphical tool for manual tuning of Ethernet and <i>mistic</i> PIDs 	<ul style="list-style-type: none"> • 1 PID algorithm • 8 PIDs per analog brain
Ethernet link redundancy	<ul style="list-style-type: none"> • Primary and secondary IP addresses for SNAP PAC controllers • PAC Control commands can be used to control redundancy algorithm 	--
Additional developer tools	<ul style="list-style-type: none"> • RESTful API (SNAP PAC S-series & R-series controllers) • Allen-Bradley DF-1 Integration Kit • Modbus Integration Kit • EtherNet/IP • HART Software Development Kit • CAN Protocol Integration Kit • DNP3 Integration Kit • BACnet Protocol Integration Kit • TL1 Integration Kit • IEC60870-5 Integration Kit • OptoMMP™ Software Development Kits (.NET, C++) 	<ul style="list-style-type: none"> • Allen-Bradley DF-1 Integration Kit • Modbus Serial Integration Kit • OptoControl SDK
HMI software		
Name	PAC Display Professional	OptoDisplay
Main features	<ul style="list-style-type: none"> • Alarming • Trending • Security • 3000-graphic library • Conversion utility for OptoDisplay projects • OPC server redundancy (OptoOPCServer) • Scanner redundancy via OptoOPCServer 	<ul style="list-style-type: none"> • Alarming • Trending • Security • 3000-graphic library • Conversion utility from <i>mistic</i> MMI projects
HMI controller data	<ul style="list-style-type: none"> • PAC Project controllers • ioProject controllers • FactoryFloor controllers on Ethernet network 	<ul style="list-style-type: none"> • FactoryFloor controllers • Cyrano controllers
OPC server		
Name	OptoOPCServer	OptoServer
OPC version	OPC DA 2.0-compliant	OPC DA 1.0-compliant
Scanner redundancy	PAC Display primary and secondary OptoOPCServers	--
Database connectivity		
Name	OptoDataLink	[A previous version of FactoryFloor contained OptoConnect.]
Databases supported	Microsoft SQL Server Microsoft Access MySQL	[OptoConnect supported SQL 6 and Access 97]
Other file types	Text files	--

Opto 22 FactoryFloor and PAC Project Software Applications

The following table matches PAC Project software applications to their comparable FactoryFloor applications.

PAC Project	FactoryFloor
PAC Control	OptoControl
PAC Display	OptoDisplay
PAC Utilities™ PAC MessageViewer™ PAC Terminal™ and PAC Manager™ OptoVersion	OptoUtilities™ OptoSnif™ OptoTerm™ OptoVersion™
OptoOPCServer	OptoServer
OptoDataLink	OptoConnect (pre-4.1 version of FactoryFloor)
SoftPAC	-- no comparable application --

Documentation

See the following guides for additional information on using PAC Control, PAC Display, and related software applications. Make certain you obtain the legacy editions as noted below, because they contain all commands and information for older hardware:

For using	See these guides	Opto 22 form
PAC Control	<i>PAC Control User's Guide, Legacy Edition</i> <i>PAC Control Command Reference, Legacy Edition</i> <i>PAC Control Commands Quick Reference, Legacy Edition</i>	1710 1711 1713
PAC Display	<i>PAC Display User's Guide</i>	1712
OptoOPCServer	<i>OptoOPCServer User's Guide</i>	1439
OptoDataLink	<i>OptoDataLink User's Guide</i>	1705
PAC Manager	<i>PAC Manager User's Guide, Legacy Edition</i>	1714

For Help

If you are having trouble migrating your control system to PAC Project Professional and cannot find the help you need in this technical note or in the related documentation, contact Opto 22 Product Support.

Phone: 800-TEK-OPTO
(800-835-6786 toll-free
in the U.S. and Canada)
951-695-3080
Monday through Friday,
7 a.m. to 5 p.m. Pacific
Time

*NOTE: Email messages and phone
calls to Opto 22 Product Support
are grouped together and
answered in the order received.*

Fax: 951-695-3017

Email: support@opto22.com

Opto 22 website: www.opto22.com

Considerations Before You Migrate

As with any migration, it's wise to think ahead about how best to approach it. Here are some considerations to keep in mind.

Serial vs. Ethernet I/O Units

To achieve the best overall system performance and throughput, Opto 22 recommends using Ethernet rather than serial I/O units with PAC Project systems. Throughput between host software (like PAC Display) and controllers with serial I/O units will be much slower and in all likelihood won't meet your requirements.

If you need to use serial I/O units, you can improve throughput by using PAC Control commands that move data to and from I/O units and strategy tables, for example *Move I/O Unit to Numeric Table* and *Move Numeric Table to I/O Unit*. Then use PAC Display dynamic attributes that access the I/O data stored in the strategy tables rather than accessing the serial I/O tags directly. However, this method is usually too much work to consider.

Two Ways to Approach Migration

Approach #1: Recommended

- Replace OptoControl controller with SNAP-PAC-S2 (or -S1) controller.
- Keep existing Ethernet I/O units.
- Replace existing serial I/O units with SNAP PAC Ethernet I/O units (using SNAP-PAC-EB1 or EB2 brains, or SNAP-PAC-R1-B controller/brains).
- Migrate the OptoControl strategy and OptoDisplay project to PAC Project Pro.

Advantages: In this approach, you're moving to the new controller, I/O hardware, and software. PAC Control Professional is able to convert the bulk of the OptoControl strategy logic, and the PAC Display Professional conversion utility converts the OptoDisplay project. These conversions are a huge benefit compared with starting over from scratch. You will have work to be done after the initial conversion, but it's an excellent start. (The remainder of this technical note details the necessary post-conversion work.)

If the existing I/O units use SNAP I/O modules, you can replace only the brains and racks (and communication wiring to them) while keeping existing I/O modules and field wiring in place. If existing I/O units have serial B3000 or B3000-B brains, just replace the brains with SNAP-PAC-R1-B brain/controllers and use them as Ethernet I/O units. The SNAP-PAC-R1-B uses the same racks, modules, and field wiring.

This method provides the best overall system performance, because you're using high-performance Ethernet I/O units along with tightly integrated Ethernet communications in the SNAP PAC controller and PAC Display Pro scanners.

Disadvantages: You need to replace more hardware.

Approach #2: Not recommended

- Replace OptoControl controller with SNAP-PAC-S2 (or -S1) controller.
- Keep existing Ethernet and serial I/O units.
- Migrate the OptoControl strategy and OptoDisplay project to PAC Project Pro.

Advantages: You can move to the new controller hardware and software while keeping all existing I/O hardware and field wiring in place.

Disadvantages: You will experience lower overall system performance compared to the FactoryFloor system, including strategy throughput and host communication throughput with PAC Display or other host software. See ["Serial vs. Ethernet I/O Units"](#) on page 5.

Try it First

The best thing you can do to evaluate your migration options and the work involved is to try doing the initial conversion process. Convert the OptoControl strategy to PAC Control Pro and the OptoDisplay project to PAC Display Pro, but don't do the post-conversion work yet.

This “dry run” allows you to see how much post-conversion work will be necessary. The amount of post-conversion work depends on many factors; some migrations may be simple, but most require a significant amount of work after the initial conversion.

Host Communication

If you have been using OptoDisplay for host communication and are migrating from OptoDisplay to PAC Display, the migration procedure is explained in [“Converting OptoDisplay Projects to PAC Display Pro” on page 31](#). However, if you are using something other than OptoDisplay for host communication, consider the following:

If you used OptoConnect in FactoryFloor:

You will need to use OptoDataLink in PAC Project. However, the migration is not automatic because OptoConnect and OptoDataLink are fundamentally different products that accomplish the same types of tasks. Therefore, you will need to re-create your connection configuration between the database and the controller. For more information, see form 1705, the *OptoDataLink User’s Guide*.

If you used OptoServer in FactoryFloor with either the OPC or DDE interface:

The OLE for Process Control (**OPC**) in OptoServer interface supported only OPC Data Access (DA) 1.0, which did not include tag browsing. However, OptoOPCServer in PAC Project supports OPC DA 2.0 which includes tag browsing, and that makes configuration much easier for smaller tag database projects (though browsing a 3,000 tag PAC Control strategy is kind of difficult). When you use the OptoOPCServer, you will need to configure the OPC tags again. For more information, see form 1439, the *OptoOPCServer User’s Guide*, which is included with PAC Project Professional.

Dynamic Data Exchange (**DDE**) is an old communication standard that was also supported in OptoServer and by Microsoft Office. The current version of Office still supports DDE. However, the OptoOPCServer supports only OPC DA 2.0, and does not support DDE. If you are currently using the DDE component of OptoServer with your FactoryFloor controller, and you still want or need to use DDE after migrating to PAC Project, you will need to purchase an OPC to DDE converter. For example, you can obtain a converter from one of the following websites:

- For getting data into Microsoft Excel
www.softwaretoolbox.com/opcdatabus/html/opctoexcel.html
- For connecting OPC Client applications to legacy DDE server applications
www.toolboxopc.com/html/DriverDetailPage_Production.asp?pn=41233153

If you wrote your own host interface program using the OptoControl SDK:

You can use the [.NET Controller SDK for SNAP PAC](#). This SDK lets you access objects in a PAC Control strategy, including integer, float, and string variables and tables. You can also access analog and digital I/O points with this toolkit, but for best throughput, we recommend using the [.NET OptoMMP toolkit](#) to access I/O points directly from the I/O units. This architecture improves overall throughput because it is much faster to get data directly from the I/O Units than to scan I/O point data through the controller.

Be aware that these two .NET toolkits use different methods and functions than the OptoControl toolkit does, so you will need to rework the communication aspect of your program.

In addition, .NET programming languages are different from the older Microsoft Visual Studio languages. Converting a program from an older version of Microsoft Visual Studio to .NET is similar to converting from FactoryFloor to PAC Project—the conversion is just the initial step (and a huge time saver), but you will have post-conversion work to get the program to compile and run correctly.

Other host communication options from a PC:

You can also use the following options for accessing PAC controller tags from a PC:

- Use the SNAP PAC controller's RESTful API and any JSON-compatible programming language to exchange data with databases, hardware devices, web services, or software applications.
- Use OptoDataLink to get the data into a database, such as Access or SQL. Or, use OptoOPCServer and use an OPC client.
- Purchase a VB to OPC interface from a company such as Kepware. Visit their website at www.kepware.com. Then use the OptoOPCServer to access tags in the PAC controller.
- If your PC application interface has only a few integers, floats, or strings, you might be able to use the Opto MMP Toolkit and implement scratchpad registers to exchange data between the application and the controller. This will require modifying the PAC Control strategy to use the controller's scratchpad for exchanging data. For more information, see form 1465, the *OptoMMP Protocol Guide*.

If you used a third-party SCADA:

Most SCADA and HMI products support OPC, so you can access the PAC controller via the OptoOPCServer.

Peer-to-Peer Communications

If you used serial, Arcnet or TCP/IP to exchange data between two or more controllers, we recommend replacing the existing peer-to-peer communications with scratchpad registers, which are very efficient. For more information, see the user's guide that came with your controller.

NOTE: If you create an I/O unit for a SNAP-PAC-S controller (for scratchpad access), use the I/O unit type Generic OptoMMP Device.

Preparing for Migration

Requirements

- **PAC Project Professional software suite**—The FactoryFloor controllers used in the control system must be configured on the PC used for converting to PAC Control and PAC Display, because the controller configuration information is stored in the computer's registry.
- **Opto 22 FactoryFloor R4.1 software suite**—If you have a version of FactoryFloor that is older than R4.1, contact Opto 22 Product Support for assistance.
- **SNAP PAC controller(s)**—PAC Control requires the use of a SNAP-PAC controller; a PAC Control strategy cannot be downloaded to an OptoControl controller.

If the OptoControl strategy references serial I/O units on more than one serial port, use a SNAP-PAC-S2 controller. The S2 has four serial ports, all software configurable for RS-485. (The SNAP-PAC-S1 will work fine if you need only one RS-485 serial port.)

1—Inventory Your Existing Control System

Before starting to convert OptoControl strategies and OptoDisplay projects, we recommend that you list or map the hardware and software components of your control system. Doing this will help to identify unsupported hardware, functions that are not supported in PAC Project, and other potential obstacles to migrating your system.

Make sure to identify all controllers, I/O units, and the type of network (Ethernet, RS-485, ARCNET, etc.) used to connect them. Also list any PCs running OptoDisplay or other FactoryFloor applications.

2—Back Up Your Current Strategies and Projects

Before making any changes to your existing OptoControl strategy or OptoDisplay project, back up all files to one or more secure locations.

3—Convert Strategies and Projects to FactoryFloor R4.1

PAC Control Professional and PAC Display Professional can import only strategies and projects saved in OptoControl R4.1f and OptoDisplay R4.1, respectively. If necessary, open older strategies and projects using OptoControl R4.1f and OptoDisplay R4.1 and then save the files. If you have an older version of FactoryFloor, contact Opto 22 Product Support for assistance.

For Cyrano and *mistic* MMI Users

Legacy Cyrano control strategies or *mistic* MMI projects must be saved in FactoryFloor R4.1 before starting the conversion steps described in this technical note. There are functional

differences between Cyrano and OptoControl strategies that you must consider when opening a Cyrano strategy in OptoControl. These differences and other information are described in *Cyrano to OptoControl Migration Technical Note (Opto 22 form 1169)*, which can be downloaded from the Opto 22 website (www.opto22.com). Also see the following documents for more information on using Cyrano and *mistic* MMI in OptoControl and OptoDisplay:

Application	See these guides	Opto 22 form
Cyrano	<i>OptoControl User's Guide</i>	724
	<i>OptoControl Command Reference</i>	725
<i>mistic</i> MMI	<i>OptoDisplay User's Guide</i>	723

4—Evaluate Serial I/O Unit Firmware and Upgrade if Necessary

When migrating from OptoControl (or Cyrano) to a PAC Control strategy and SNAP-PAC controller, if you will be using existing serial *mistic* I/O units (*mistic* remote bricks or the serial B3000), it is necessary to make sure the I/O units have the current version of firmware. Older versions of the firmware may not be compatible with the PAC controller.

I/O Unit Type	Minimum Firmware Version Recommended
B3000 (serial)	81.36 (or newer)
<i>mistic</i> Analog Brick	R3.0a (which is the same as 1.19) (or newer)
<i>mistic</i> Digital MF Brick	R3.0a (which is the same as 1.11) (or newer)

*NOTE: In order to upgrade firmware on a *mistic* brick or the serial B3000, the firmware chip must be replaced with the current version of the chip. Contact the Opto 22 Product Support Group with the I/O unit type and quantity of firmware chips needed. Be aware that B3000s with firmware v1.10 or lower cannot be upgraded and will need to be replaced.*

NOTE: Remember that you can replace a serial B3000/B3000-B brain with a SNAP-PAC-R1-B controller/brain to use Ethernet communications and keep racks, I/O, and field wiring.

Converting OptoControl Strategies to PAC Control Pro

PAC Control Professional handles the OptoControl-to-PAC Control conversion automatically. However, depending on the complexity and features of the OptoControl strategy, you may need to change or adjust the strategy both before and after conversion.

1—Assessing Major Differences in Features

Before converting an OptoControl strategy to PAC Control, determine if any features and functions used in the OptoControl strategy are unavailable or implemented differently in

PAC Control. For most features and functions, there are alternative approaches or workarounds to accomplish the same task using PAC Control. On [page 11](#), review the table of OptoControl features not available in PAC Control; also check “[Additional Related Differences](#)” on [page 11](#).

OptoControl Features Not Available in PAC Control

OptoControl Feature Not Available in PAC Control	Alternative, Workaround, or Notes												
Serial host support (except for PPP over a modem)	Use PPP and a modem.												
ARCNET networking	Evaluate your networking environment to see if Ethernet networking is an option.												
External instruction sets (also called “XIDs” and “Forth libraries”)	Most functions implemented using external instruction sets have been integrated into PAC Control. If necessary, recreate what the external instruction set does using OptoScript and/or subroutines:												
	<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 50%;">External Instructions</th> <th style="width: 50%;">PAC Control Equivalent</th> </tr> </thead> <tbody> <tr> <td>Save tables</td> <td>Persistent variables</td> </tr> <tr> <td>External pointer commands</td> <td>Typed pointer commands</td> </tr> <tr> <td>Modbus toolkit with XID</td> <td>Modbus toolkit</td> </tr> <tr> <td>Allen-Bradley toolkit with XID</td> <td>Allen-Bradley toolkit. Also IO4AB (with firmware 8.3 or newer)</td> </tr> <tr> <td>Case statements, if/then/else, looping logic</td> <td>OptoScript</td> </tr> </tbody> </table>	External Instructions	PAC Control Equivalent	Save tables	Persistent variables	External pointer commands	Typed pointer commands	Modbus toolkit with XID	Modbus toolkit	Allen-Bradley toolkit with XID	Allen-Bradley toolkit. Also IO4AB (with firmware 8.3 or newer)	Case statements, if/then/else, looping logic	OptoScript
	External Instructions	PAC Control Equivalent											
	Save tables	Persistent variables											
	External pointer commands	Typed pointer commands											
	Modbus toolkit with XID	Modbus toolkit											
Allen-Bradley toolkit with XID	Allen-Bradley toolkit. Also IO4AB (with firmware 8.3 or newer)												
Case statements, if/then/else, looping logic	OptoScript												
Binary/checksum or ASCII/checksum support for <i>mistic</i> brains	PAC Control supports only binary/CRC or ASCII/CRC format for <i>mistic</i> brains. Contact Opto 22 Product Support for current information on binary/checksum or ASCII/checksum options in PAC Control.												
HRD analog brick support	Consider using a SNAP I/O unit with SNAP analog modules, which have up to 32 channels per module. A maximum of 16 SNAP analog modules can be placed on one SNAP rack.												
Localbus support	Change the localbus port in OptoControl to a remote (serial) port. Resolve any potential I/O address conflicts if you are already using serial ports. You will need to replace the local I/O unit and points with compatible replacements. It may be possible to simply replace the local brains with “remote” brains. Also evaluate your application and networking environment to see if using Ethernet-based brains is an option.												

EtherNet/IP is supported by SNAP PAC controllers with 8.3 or higher firmware. Thus, SNAP PAC systems running PAC Control strategies can be integrated into an Allen-Bradley control environment.

Additional Related Differences

Also note the following important differences between OptoControl and PAC Control and the hardware used with each:

- **PIDs on SNAP Ethernet-based brains have more features and options than *mistic* PID loops.** PIDs have been divided into their own command groups in PAC

Control: PID Ethernet and PID Mystic. See Chapter 10 of the *PAC Control User's Guide, Legacy Edition* for more information.

Make sure to change the sign of the gain for each PAC brain Velocity PID algorithm. The sign is reversed between the two systems.

If part of your migration is to replace *mistic* bricks or serial B3000 I/O units with SNAP PAC Ethernet-based I/O units, plan for and schedule time to retune your PID loops, even if you use the same velocity PID algorithm. We do not have a method of calculating what the new PID tuning parameters should be in order to achieve the same tuning.

When you set a PID loop into manual mode, note that manually controlling the PID output is different on SNAP Ethernet-based brains and *mistic* brains.

- On a *mistic* brick or serial B3000, the PID loop stops calculating and also stops updating the analog PID output point. You can then manually control the analog point using normal commands, such as Move.
- On a SNAP Ethernet-based brain, the PID loop stops calculating but continues to update the PID output. To manually control the PID output, use the Set PID Output command.

- ***mistic* event/reactions are not available on SNAP Ethernet brains.** If your OptoControl strategy included *mistic* event/reactions, and you decide to change your *mistic* bricks or serial B3000 I/O units to SNAP PAC or Ethernet I/O units, be aware that there is no equivalent to *mistic* event/reactions in the newer brains.

SNAP Ethernet-based brains have features called events and reactions, but they are different; they are typically set up in PAC Manager and work separately from the PAC Control strategy.

If you change to a SNAP PAC controller and still use your *mistic* bricks or serial B3000 I/O units, your *mistic* event reactions will still work. But if you change to the newer brains, you'll need to change your strategy to accomplish the same thing in logic.

- **When setting a variable True, PAC Control uses positive 1; when setting a variable False, PAC Control uses 0.** This is different from OptoControl, which used -1 for true and 0 for false. This difference also affects true/false returns on some commands. Please note the following:

Command	OptoControl Result	PAC Control Result
Set Variable True	Sets integer variable to -1	Sets integer variable to +1
Set Variable False	Sets integer variable to 0	Sets integer variable to 0
Variable False?	Evaluates true if value of integer variable is 0 . False is defined as zero .	
Variable True?	Evaluates true if value of integer variable is not 0 . True is defined as any non-zero value (any non-false value).	

- **PAC Control string commands that use an Index parameter start at zero; equivalent commands in OptoControl start at 1.**

The following commands are affected:

- Find Character in String
- Find Substring in String
- Get Substring
- Get Nth Character
- Set Nth Character

If your strategy uses these commands, in order to get the correct character(s), you must change each occurrence of the command: subtract 1 from the Index value you're using.

- **The Convert Number to String command (NumberToString in OptoScript) behaves differently in OptoControl and PAC Control.** For more information, see [OptoKB article #KB80115](#).
- **Error codes, both returned and queued, are different in OptoControl and PAC Control.** See note 4 in [“Notes for OptoControl and PAC Control Commands” on page 29](#) for more information on differences in return values.
- **The sign used on quadrature counting is different in *mistic* and SNAP Ethernet brains.** See “Digital Point Commands” in Chapter 10 of the *PAC Control User's Guide, Legacy Edition* for more information.
- **Repeated use of the commands Start Chart or Call Chart may be an issue in your PAC Control strategy due to the execution speed of these commands (100 ms per instance).** If your OptoControl strategy uses these commands repeatedly, contact Opto 22 Product Support to discuss alternate strategy logic.
- **The Get Year command returns a different value.** OptoControl returns the least significant 2 digits (for example, 2008=08). PAC Control returns the actual year (for example, 2008=2008).
- **Allen-Bradley DF1 Integration Kit and Modbus Integration Kits are available for both OptoControl and PAC Control.** However, their implementation differs: in OptoControl, the integration kits use external commands; in PAC Control, they do not. Conversion of strategy logic and subroutines related to these toolkits is *not* automatic; the logic will need to be rewritten.
- **EtherNet/IP is supported by SNAP PAC controllers with 8.3 or higher firmware.** Therefore SNAP PAC systems running PAC Control strategies can be integrated into an Allen-Bradley control environment.

2—Preparing the Strategy

Before converting your OptoControl strategy to PAC Control, do the following:

- **Back up the current, working OptoControl strategy and subroutines to one or more secure locations.**

- **For SNAP-PAC-S1 controllers, check for I/O units configured on more than one serial port.** In the strategy tree, under I/O Units, double-click the I/O Units folder to see a list of all I/O units for the strategy, including Port and Address field. If more than one port is used, you must use a SNAP-PAC-S2 rather than a SNAP-PAC-S1 controller. (The S1 has only one RS-485 port; the S2 has four ports configurable for RS-485.)
- **Check for any ARCNET I/O units.** ARCNET I/O units, such as the B3000-HA and the SNAP-BRS-HA, are not supported in PAC Project. You'll need to replace these brains with Ethernet or serial brains. In the OptoControl strategy, create new replacement I/O units. Then move I/O points from the old ARCNET units to the new units.
- **Check for external instructions.** In the strategy tree there's a folder called "Ext. Instruction Files Included." If this folder is not empty, you'll need to figure out what the function of those commands is and then implement it using PAC Control commands.
- **Check subroutines.** OptoControl and PAC Control subroutines support different parameters. Any unsupported parameters passed to a subroutine will need to be handled differently.

OptoControl Subroutine Data Types Supported in PAC Control Pro

- Numeric variables, literals, and tables
- String variables, literals, and tables
- Communication handles
- I/O points, I/O units, and pointer tables
- Pointer variables as local parameters

OptoControl Subroutine Data Types Not Supported in PAC Control



- Unsupported I/O units and the I/O points on them, such as G4 or HRD I/O units
- Event/reactions
- Charts
- PID loops
- **If you have *mistic* local-bus I/O,** you need to move the points to a SNAP-B3000 brain. However, not all *mistic* brick points are movable, so first you must change any non-movable *mistic* brick points to movable point types.

The following table shows which G4 point types can and cannot be moved. Only a point with *no* scaling can be moved, and it must be moved into a SNAP module that offers the same range. These rules also apply to similar B200 points.

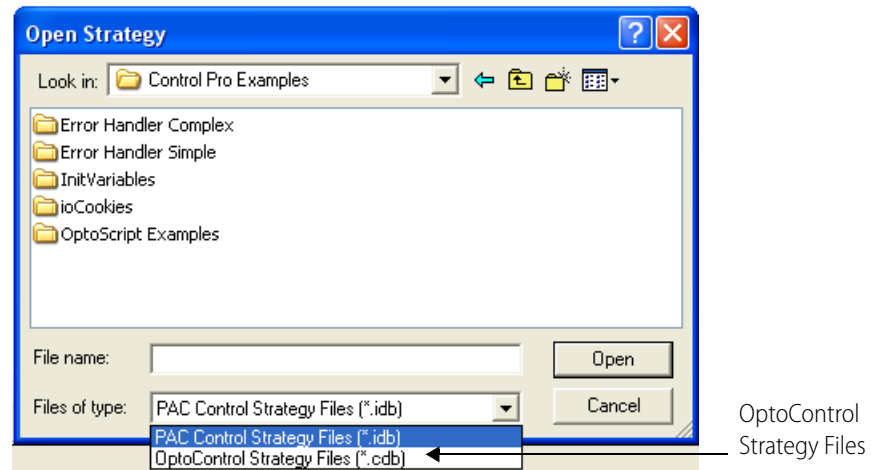
Point Type	Movable	Not Movable
G4AD3: 4 - 20 mA		■
G4AD4: ICTD Temp. Probe	■	
G4AD5: Type J Thermocouple	■	
G4AD6: 0 - 5 VDC		■

Point Type	Movable	Not Movable
G4AD6HS: 0 - 5 VDC High-Speed		■
G4AD7: 0 - 10 VDC		■
G4AD7HS: 0 - 10 VDC High-Speed		■
G4AD8: Type K Thermocouple	■	
G4AD9: 0 - 50 mV	■	
G4AD10: 100 Ohm RTD	■	
G4AD11: -5 - +5 VDC	■	
G4AD12: -10 - +10 VDC	■	
G4AD13: 0 - 100 mV		■
G4AD16: 0 - 5 AAC/DC		■
G4AD17: Type R Thermocouple	■	
G4AD18: Type T Thermocouple	■	
G4AD19: Type E Thermocouple	■	
G4AD20: Rate (Frequency)		■
G4AD22: 0 - 1 VDC		■
G4AD23: Type S Thermocouple	■	
G4AD24: Type B Thermocouple	■	
G4AD25: 0 - 100 VAC/VDC		■
G4AD26: Velocity		■
G4DA3: 4 - 20 mA	■	
G4DA4: 0 - 5 VDC		■
G4DA5: 0 - 10 VDC	■	
G4DA6: -5 - +5 VDC		■
G4DA7: -10 - +10 VDC	■	
G4DA8: 0 - 20 mA		■
G4DA9: 5 - 60 VDC TPO		■
G4DA10: 12 - 280 VAC TPO		■

3—Converting the Strategy

- To open PAC Control:
 - In Windows 7 and Windows Vista, press the Windows Start key , and then click Programs > Opto 22 > PAC Project 9.6 > PAC Control.
 - In Windows 10 and Windows 8.1, press the Windows Start key , type **PAC Control 9.6** and then press the Enter key.

2. In PAC Control, select File > Open Strategy.
3. In the Open Strategy dialog box that appears, select “OptoControl Strategy Files (*.cdb)” from the “Files of type” drop-down menu.



4. Navigate to the OptoControl strategy file you want to convert, select it, and click Open.
 - If there are no problems converting the strategy, a message box will prompt you to choose a location where the converted strategy will be saved. Continue to [step 5](#) on [page 16](#).
 - If there is a problem that prevents the OptoControl strategy from being converted, the conversion process stops. Read the error message describing the problem, modify the OptoControl strategy as needed, and repeat [step 4](#).

If the conversion process stops while converting the OptoControl strategy, one of the following error messages will appear:

Error message	Description
Cannot open from OptoControl version <versionnumber>.	PAC Control can open only OptoControl files saved in OptoControl R4.1f. Contact Opto 22 Product Support if you do not have this version of the software.
I/O unit <unitname> is using ARCNET or a local port. Strategy cannot be converted.	Review the information in “ 1—Assessing Major Differences in Features ” on page 10 and “ 2—Preparing the Strategy ” on page 13 for workarounds or other approaches.

5. Click OK to close the message box, and in the dialog box that opens, select a location to save the converted strategy and then click Open.

The automatic part of the conversion process is now complete.

4—Saving Strategies to Flash

Be sure to save your strategy to the SNAP PAC’s flash memory after downloading.

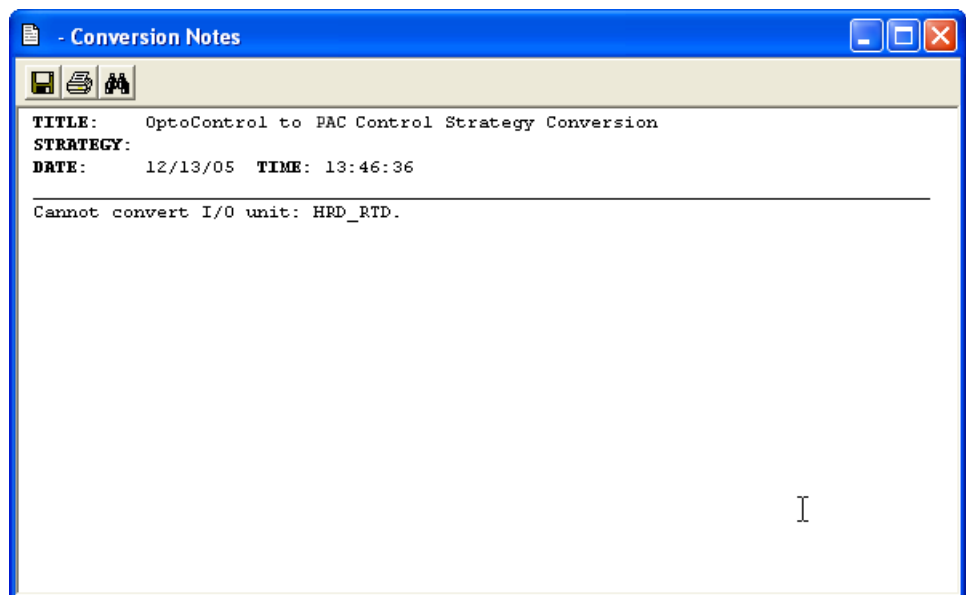
On the SNAP PAC controller, persistent variables and variables initialized on download are automatically saved to battery-backed RAM, but strategies are not. The SNAP PAC S-series controller has about 7.5 MB available in flash memory for strategy storage. See the *PAC Control User's Guide, Legacy Edition* (form 1710) for instructions.

5—Followup

The converted OptoControl strategy will open in PAC Control. Before using the strategy, you must review and correct conversion errors, test-compile any OptoScript blocks, and finally configure the strategy to be saved to the controller's flash memory upon download.

Reviewing Conversion Errors

If errors occur that do not stop the conversion process, they are listed in the error log file "Conversion_Notes.rtf". The log file is saved in the same directory as the converted strategy and opens in its own PAC Control window.



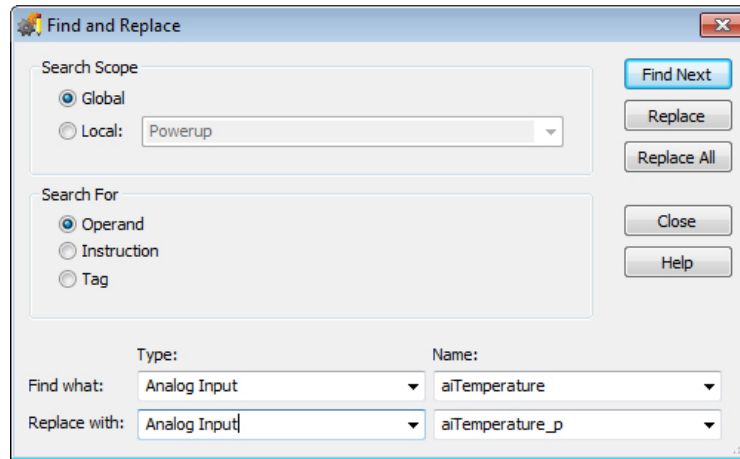
Check each message listed in the error log against the list in ["Conversion Error Log Messages"](#) on page 20 and correct the converted strategy as needed.

Migrating *mistic* I/O Points to SNAP PAC I/O

When migrating analog points from *mistic* to SNAP-PAC I/O, there are many differences between *mistic* I/O points and SNAP PAC I/O points, so you will need to re-create each point in PAC Control.

1. Open the strategy in PAC Control.
2. For each old point, create a new point that has the same name as the old one plus a suffix such as "_p" (the p is for PAC). For example, aiTemperature becomes aiTemperature_p.

3. Choose Edit > Replace to open the Find and Replace dialog box.



4. For "Find what:", select the Type and Name for the point being replaced.
5. For "Replace with:", select the Type and Name of the new point, and then click Replace All.
6. Repeat the previous steps for each point being replaced.
7. In the strategy tree, delete all the *mistic* points.
8. Rename the new points by deleting the suffix so that they have the original names.
This will be important if you convert a related OptoDisplay project to PAC Display. When you use AutoCorrect Tags, PAC Display will be able to match the names. Otherwise, you would need to manually change each dynamic attribute in your Display project in order to connect to the new points.

Test Compiling OptoScript Blocks

Because OptoScript blocks in the OptoControl strategy remain unchanged through the conversion process, you'll need to check them for necessary command and error handling changes. If OptoScript commands are different in OptoControl and PAC Control, the OptoScript block will not compile.

Select Compile > Compile All to check all the OptoScript blocks. Or you can check individual OptoScript blocks by clicking the Test Compile button in each OptoScript block; if the block does not compile, the exact lines of code that need to be updated will be listed. Click the error description to jump directly to the line of code that needs modification. See [page 24](#) for a list of OptoControl commands and PAC Control equivalents. Also remember that any PID command used in an OptoControl strategy now has "mistic" in its command name in PAC Control to distinguish it from Ethernet PID commands.

Saving Strategies to Flash

IMPORTANT: Be sure to configure your strategy to be saved to the SNAP PAC's flash memory upon download.

On the SNAP PAC controller, persistent variables and variables initialized on download are automatically saved to battery-backed RAM, but strategies are not. The SNAP PAC controller has about 7.5 MB available in flash memory for strategy storage. See the *PAC Control User's Guide, Legacy Edition* (form 1710) for instructions.

Converting Tag Names

Previous to OptoControl 4.0, tag names of different types could have the same name. However, in OptoControl 4.0 and newer, tag names of different types cannot have the same name.

When updating a strategy from a version prior to 4.0 to version 4.0 or newer, OptoControl searches for and changes any duplicate names. It changes each variable that has a duplicate name by adding a suffix that has two underscore characters, an indication of the tag type, and a tag ID number.

For example, if a digital input, an I/O unit, and an integer variable all had the name "Test," the new names would be similar to the following:

Test__dig_17

Test__iou_4

Test__num_132

The new names can cause the following problems:

- User *initialization files* related to the OptoControl strategy are not automatically updated
- Any *recipe files* related to the OptoDisplay project are not automatically updated
- The *dynamic attributes* of the OptoDisplay project are not automatically updated

What You Should Do. If your OptoControl strategy downloads any user *initialization files*, be sure to review these files to determine if any of the duplicate tags (whose names were changed) are used. If so, update the initialization files with the new names.

Because of the effect on *recipes*, we recommend that you use OptoControl to edit the names of any tables used in recipes so that they have the original names. By doing this, you won't have to edit the recipe files. The names of the other tags that conflicted with the table names can remain with the name automatically changed by OptoControl during the update.

After you have finished editing tag names in OptoControl, use OptoDisplay Configurator to run AutoCorrect Tags from the Tools menu. This will automatically update any tag name changes related to *dynamic attributes*.

Conversion Error Log Messages

Error Message	Description
Cannot convert instruction <instructionname> in chart <chartname>, block <blocknumber>.	An instruction used in a chart in the OptoControl strategy isn't available in PAC Control. An invalid instruction is turned into a "Comment (OptoControl Conversion Issue)" instruction. This instruction's description string contains the name of the old instruction and the parameters used.
Cannot convert subroutine <subname>, it contains pointer variables.	A subroutine uses pointers as internal variables, which PAC Control doesn't support. All other subroutine variable types used in OptoControl are allowed in PAC Control Pro.
Cannot convert subroutine <subname>, it uses invalid parameter types.	<p>OptoControl allows many subroutine parameter types that PAC Control does not support. These subroutine parameter types will not convert to PAC Control:</p> <ul style="list-style-type: none"> • Analog Event Reaction • B100 Digital Multifunction I/O Unit • B200 Analog Multifunction I/O Unit • B3000 SNAP Analog • B3000 SNAP Digital • B3000 SNAP Mixed I/O • Chart • Digital Event/Reaction • Event/Reaction Group • G4 Analog Multifunction I/O Unit • G4 Digital Local Simple I/O Unit • G4 Digital Multifunction I/O Unit • G4 Digital Remote Simple I/O Unit • HRD Analog Current Output I/O Unit • HRD Analog RTD Input I/O Unit • HRD Analog Thermocouple/mV Input I/O Unit • HRD Analog Voltage Output I/O Unit • HRD Analog Voltage/Current Input I/O Unit • Local Simple Digital Input • Local Simple Digital Output • PID Loop • Pointer Table • SNAP Digital 64 • SNAP Remote Simple Digital • SNAP Simple 64 • TPO
Cannot convert instruction <instructionname> in subroutine <subname>, block <blocknumber>.	An instruction used in a subroutine in the OptoControl strategy isn't available in PAC Control. An invalid instruction is turned into a "Comment (OptoControl Conversion Issue)" instruction. This instruction's description string contains the name of the old instruction and the parameters used.

Error Message	Description
<p>Cannot convert I/O unit: <unitname>.</p>	<p>An I/O unit definition used in the OptoControl strategy isn't available in PAC Control. The following I/O units are not supported in PAC Control:</p> <ul style="list-style-type: none"> • G4 Digital Local Simple (G4D16LS) • HRD Analog RTD Input (G4AIRTD) • HRD Analog Thermocouple/mV Input (G4AITM) • HRD Analog Voltage/Current Input (G4AIVA) • HRD Analog Current Output (G4AOA) • HRD Analog Voltage Output (G4AOV) • Motion Servo and Stepper boards (NOTE: The SNAP PAC Motion Control Subsystem can provide stepper motor control.)
<p>Cannot convert analog point: <pointname>.</p>	<p>An analog I/O point on an HRD (high density) I/O unit will not be available because this I/O unit cannot be converted into PAC Control. The following analog inputs and outputs on an HRD I/O unit cannot be converted:</p> <p>Inputs</p> <ul style="list-style-type: none"> • Type B Thermocouple (Note that SNAP-AITM2: Type B Thermocouple will convert.) • Type C Thermocouple • Type D Thermocouple • Type E Thermocouple • Type G Thermocouple • Type J Thermocouple • Type K Thermocouple • Type N Thermocouple • Type R Thermocouple • Type S Thermocouple • Type T Thermocouple • 0 - 100 mV (Scalable) • -50 - +50 mV • -150 - +150 mV • 0 - 20 mA • 4 - 20 mA (Scalable) • 4 - 20 mA • 0 - 1.25 VDC • 0 - 5 VDC • 0 - 10 VDC (Scalable) • 0 - 10 VDC • -5 - +5 VDC • -10 - +10 VDC • 0 - 12500 Ohms (Scalable) • 100 Ohm Pt: Alpha=385 3-wire • 100 Ohm Pt: Alpha=385 2/4-wire • 120 Ohm Ni: Alpha=672 3-wire • 120 Ohm Ni: Alpha=672 2/4-wire • 604 Ohm NiFe: Alpha=518 3-wire • 604 Ohm NiFe: Alpha=518 2/4-wire • 100 Ohm Pt: Alpha=3923 3-wire SAMA • 100 Ohm Pt: Alpha=3923 2/4-wire SAMA • 100 Ohm Pt: Alpha=3928 3-wire ITS90 • 100 Ohm Pt: Alpha=3928 2/4-wire ITS90 • 100 Ohm Pt: Alpha=391 3-wire OIML • 100 Ohm Pt: Alpha=391 2/4-wire OIML • 100 Ohm Pt: Alpha=3916 3-wire JIS

Error Message	Description
<p>Cannot convert analog point: <pointname>. (Continued)</p>	<ul style="list-style-type: none"> • 100 Ohm Pt: Alpha=3916 2/4-wire JIS • 100 Ohm Pt: Alpha=3916 3-wire JEMIMA • 100 Ohm Pt: Alpha=3916 2/4-wire JEMIMA • 100 Ohm Pt: Alpha=391 3-wire GOST • 100 Ohm Pt: Alpha=391 2/4-wire GOST <p>Outputs</p> <ul style="list-style-type: none"> • 0 - 5 VDC • 0 - 10 VDC (Scalable) • 0 - 10 VDC • -5 - +5 VDC • -10 - +10 VDC • 0 - 20 mA • 4 - 20 mA (Scalable) • 4 - 20 mA
<p>Cannot convert digital point: <pointname>.</p>	<p>A digital I/O point will not be available if the I/O unit for that point was not converted into PAC Control.</p>
<p>Cannot convert PID Loop: <pidname>. I/O Unit was not converted.</p>	<p>A PID loop will not be available if the I/O unit for that PID loop was not converted into PAC Control.</p>
<p>Cannot convert Event/Reaction: <ername>. I/O Unit was not converted.</p>	<p>An event/reaction (E/R) will not be available if the I/O unit for that E/R was not converted into PAC Control.</p>
<p>Strategy has external action definitions which won't be converted.</p>	<p>External actions are not supported in PAC Control.</p>
<p>Strategy has external condition definitions which won't be converted.</p>	<p>External conditions are not supported in PAC Control.</p>
<p>Added new numeric variable: <variablename> for returning subroutine status.</p>	<p>This message will appear for any OptoControl strategy that includes subroutines. In PAC Control, all subroutines return a status parameter that reports the overall success or errors for the subroutine call. Because this parameter does not exist in OptoControl, a new numeric variable "SubroutineReturnStatus" is automatically created.</p>
<p>A subroutine and chart are both named <suband-chartname>. This will cause problems when you run the strategy.</p>	<p>Subroutines and charts in an OptoControl strategy can share the same name (for example, a chart and a subroutine can both be named Valve_Control). In PAC Control, subroutines and charts cannot share the same name. You must change the name of either the chart or the subroutine before running the converted strategy.</p>
<p>A string command was found that may need to be adjusted in <chartname> at block <blocknumber>. OptoControl strings use a 1-based index but PAC Control strings use a 0-based index.</p>	<p>OptoControl string commands that use an index start with index 1; equivalent PAC Control commands start with index 0.</p> <p>Five commands are affected by this change:</p> <ul style="list-style-type: none"> • Get Substring [no error code] • Get Nth Character • Set Nth Character • Find Substring in String • Find Character in String <p>Find each instance of these commands in your strategy. For each one, subtract 1 from the index value you used in OptoControl.</p>

Handling Interrupt Charts from OptoControl

FactoryFloor Interrupt charts are converted in PAC Control 8.2 and later, and they are fully supported in a properly configured system. However, here are some questions that you may have when converting strategies that use an Interrupt chart:

What if I want to continue using an Interrupt chart from OptoControl (from a PAC Control 8.2 migration or later).

1. Verify the operation of the Interrupt chart.
2. Make sure that the I/O unit interrupt line is wired properly to the controller.

NOTE: If the I/O unit interrupt line is wired incorrectly or improperly biased or terminated, this could result in the IRQ signal being activated accidentally. The interrupt chart will run as long as the IRQ signal is active or asserted, which can cause the strategy to become very unresponsive.

What if there is an Interrupt chart that I don't use?

If the strategy does not implement the interrupt chart, just delete the chart. If you need an Interrupt chart later, you can create it then. If you would like to retain the contents of the Interrupt chart, create a new chart that is not named Interrupt. Cut and paste the functionality of the Interrupt chart into the new chart, and then delete the Interrupt chart.

In OptoControl, an Interrupt chart was created automatically when the strategy was created. In PAC Control, you must manually create an Interrupt chart.

How do I resurrect an Interrupt chart from OptoControl . If a strategy is converted from FactoryFloor using PAC Control 8.1 or earlier, the *mistic* interrupt handling commands are converted to comments. These interrupt commands are responsible for detecting which I/O unit has asserted the IRQ and (more importantly) responsible for negating (or clearing) the IRQ signal.

To resurrect an Interrupt chart:

1. Search for all instructions named *Comment (OptoControl Conversion Issue)* in the Interrupt chart.
2. Update these comments with their equivalent functions.
3. Make sure that the I/O unit interrupt line is wired properly to the controller.

NOTE: If the I/O unit interrupt line is wired incorrectly or improperly biased or terminated, this could result in the IRQ signal being activated accidentally. The interrupt chart will run as long as the IRQ signal is active or asserted, which can cause the strategy to become very unresponsive.

NOTE: If an I/O unit's configuration is stored in flash, when certain conditions are met, the I/O unit may be capable of asserting an IRQ causing the controller to invoke the Interrupt chart. However, if the mistic interrupt handling commands were converted to comments, when an I/O unit asserts an IRQ, the Interrupt chart will be activated, but it won't be able to handle the IRQ signal. The Interrupt chart will continue to run indefinitely resulting in severe degradation of performance.

OptoControl and PAC Control Commands

Most commands convert automatically. However, some OptoControl commands are different or unavailable in PAC Control, and you'll need to change your strategy as noted in the following table. For each OptoControl command, the comparable PAC Control command and any applicable notes are listed.

Command	OptoControl Result	PAC Control Result
Set Variable True	Sets integer variable to -1	Sets integer variable to +1
Set Variable False	Sets integer variable to 0	Sets integer variable to 0
Variable False?	Evaluates true if value of integer variable is 0 . False is defined as zero .	
Variable True?	Evaluates true if value of integer variable not 0 . True is defined as any non-zero value (any non-false value).	

Table footnotes follow on [page 29](#).

OptoControl Command	Comparable PAC Control Command(s)	Notes
Accept Session On TCP Port	Accept Incoming Communication	11
ARCNET Connected?	n/a	5
ARCNET Message Address Equal to?	n/a	5
ARCNET Node Present?	n/a	5
Calculate & Store Strategy CRC	n/a	This command is not needed in PAC Control. PAC Control includes options for backing up strategies and saving them on the PC or controller. In PAC Control, see Strategy Options under the File menu.
Characters Waiting at Serial Port?	Get Number of Characters Waiting	11
Clear PC Byte Swap Mode (ISA Only)	n/a	3
Clear Quadrature Counter	Clear Counter	1 —The PAC Control command can access quadrature points.
Clear Receive Buffer	Use "Get Number of Characters Waiting" and "Receive N Characters" to achieve the same result. For TCP communication handles, use "Clear Communication Receive Buffer."	11
Close Ethernet Session	Close Communication	11
Configure I/O Unit	n/a	In PAC Control, I/O units are automatically configured when an I/O unit becomes enabled.
Configure Port	Set Communication Handle Value	11

OptoControl Command	Comparable PAC Control Command(s)	Notes
Configure Port Timeout Delay	Set Communication Handle Value	11 —For individual serial I/O units, timeout values can be changed from the 1 sec. default. Timeout values can also be changed for FTP communication handles. TCP timeouts have the following default values: <ul style="list-style-type: none"> • Open: 4 sec. • Selects: 1 sec. • (all other timeouts): 10 sec.
Copy Date to String (DD/MM/YYYY)	Copy Date to String (DD/MM/YYYY)	Make sure the string is long enough.
Copy Date to String (MM/DD/YYYY)	Copy Date to String (MM/DD/YYYY)	Make sure the string is long enough.
CTS Off?	n/a	10
CTS On?	n/a	10
Disable Communication To Analog Point	Disable Communication to Point	1
Disable Communication To Digital Point	Disable Communication to Point	1 —Does not convert if parameter is a pointer or a simple digital point.
Enable Communication To Analog Point	Enable Communication to Point	1
Enable Communication To Digital Point	Enable Communication to Point	1 —Does not convert if parameter is a pointer or a simple digital point.
Ethernet Session Open?	Communication Open?	11
Get & Clear Digital I/O Unit Latches	n/a	8
Get & Clear Digital-64 I/O Unit Latches	n/a	8
Get & Clear Quadrature Counter	Get & Clear Counter	1
Get & Clear Simple-64 I/O Unit Latches	n/a	8
Get Address Of I/O Unit Causing Current Error	Get Name of I/O Unit Causing Error	4
Get Analog Lower Clamp	n/a	1
Get Analog Upper Clamp	n/a	1
Get ARCNET Destination Address On Port	n/a	5
Get ARCNET Host Destination Address	n/a	5
Get ARCNET Peer Destination Address	n/a	5
Get Controller Address	"Get Control Engine Address" and "Convert IP Address String to Integer 32" (if needed)	1
Get Default Host Port	n/a	PAC Control controllers do not have jumpers.
Get Digital I/O Unit As Binary Value	Get I/O Unit As Binary Value	1 —Does not convert if original "Put In" parameter was a float variable.
Get Digital I/O Unit Latches	n/a	8
Get Digital-64 I/O Unit As Binary Value	Get I/O Unit As Binary Value	1
Get Mixed I/O Unit As Binary Value	Get I/O Unit As Binary Value	1

OptoControl Command	Comparable PAC Control Command(s)	Notes
Get Simple-64 I/O Unit As Binary Value	Get I/O Unit As Binary Value	1
Get Digital-64 I/O Unit Latches	n/a	8
Get Ethernet Session Name	Get Communication Handle Value	11
Get Number Of Characters Waiting On Ethernet Session	Get Number of Characters Waiting	11
Get Number Of Characters Waiting On Serial Or ARCNET Port	Get Number of Characters Waiting	5, 11
Get Port Of I/O Unit Causing Current Error	Get Name of I/O Unit Causing Error	4
Get Priority	n/a	6
Get Priority Of Host Task	n/a	6
Get Quadrature Counter	Get Counter	1
Get RTU/M4IO Temperature	n/a	7
Get RTU/M4IO Voltage	n/a	7
Get Simple-64 I/O Unit Latches	n/a	8
Get Year	Get Year	OptoControl returns the least significant 2 digits (e.g. 2008=08). PAC Control returns the actual year (e.g. 2008=2008).
Host Task Received a Message?	n/a	9
Interrupt On Port0?	Get Active Interrupt Mask	2
Interrupt On Port1?	Get Active Interrupt Mask	2
Interrupt On Port2?	Get Active Interrupt Mask	2
Interrupt On Port3?	Get Active Interrupt Mask	2
Interrupt On Port6?	n/a	2
IVal Set Analog From Table	IVal Move Numeric Table to I/O Unit	8
IVal Set Quadrature Counter	IVal Set Counter	1
Low RAM Backup Battery?	n/a	7
Move Analog I/O Unit To Table	Move I/O Unit to Numeric Table	1—Does not convert if parameter is a pointer.
Move Digital I/O Unit To Table	Move I/O Unit to Numeric Table	1
Move Digital I/O Unit To Table Element	Get I/O Unit as Binary Value	1—Use OptoScript to move value(s) to a table element using Table[index] syntax. Example: Int32table[0] = GetIoUnitAsBinaryValue(UIO);
Move Mixed I/O Unit To Table	Move I/O Unit to Numeric Table	1
Move Simple-64 I/O Unit To Table	Move I/O Unit to Numeric Table	1
Move Table Element To Digital I/O Unit	Set Digital I/O Unit From MOMO Masks	1
Move Table To Analog I/O Unit	Move Numeric Table to I/O Unit	1—Does not convert if parameter is a pointer.

OptoControl Command	Comparable PAC Control Command(s)	Notes
Move Table To Digital I/O Unit	Move Numeric Table to I/O Unit	1
Move Table To Simple-64 I/O Unit	Move Numeric Table to I/O Unit	1
Open Ethernet Session	Open Outgoing Communication	11
Read Byte From PC Memory (ISA Only)	n/a	3
Read Byte From PC Port (ISA Only)	n/a	3
Read Word From PC Memory (ISA Only)	n/a	3
Read Word From PC Port (ISA Only)	n/a	3
Receive Character Via Serial Port	"Receive Character" or "Receive N Characters"	11
Receive N Characters Via ARCNET	n/a	5
Receive N Characters Via Ethernet	"Receive Character" or "Receive N Characters"	11
Receive N Characters Via Serial Port	"Receive Character" or "Receive N Characters"	11
Receive String Via ARCNET	n/a	5
Receive String Via Ethernet	"Receive String" or "Receive String Table"	11
Receive String Via Serial Port	"Receive String" or "Receive String Table"	11
Receive Table Via ARCNET	n/a	5
Receive Table Via Ethernet	"Receive Numeric Table"	11
Receive Table Via Serial Port	"Receive Numeric Table"	11
Reset Controller	Write Number to I/O Unit Memory Map	This can be done by writing the value 5 to the address 0xF038000 in the controller's memory map. See Opto 22 form 1465 for more information on the memory map.
Set ARCNET Destination Address On Port	n/a	5
Set ARCNET Host Destination Address	n/a	5
Set ARCNET Mode Raw	n/a	5
Set ARCNET Mode Standard	n/a	5
Set ARCNET Peer Destination Address	n/a	5
Set Day Of Week	n/a	The day of the week is automatically calculated based on the date.
Set End-of-message Terminator	Set End-Of-Message Terminator	11
Set I/O Unit Configured Flag	n/a	In PAC Control, I/O units are automatically configured when an I/O unit becomes enabled.
Set Mixed I/O Unit From MOMO Masks	Set Mixed I/O Unit from MOMO Masks	1
Set Number Of Retries To All I/O Units	n/a	Retries are set on an I/O unit when the I/O unit is configured.

OptoControl Command	Comparable PAC Control Command(s)	Notes
Set PC Byte Swap Mode (ISA Only)	n/a	3
Set Priority	n/a	6
Set Priority Of Host Task	n/a	6
Set Simple-64 I/O Unit From MOMO Masks	Set Simple-64 I/O Unit From MOMO Masks	1
Set Variable True	Set Variable True	OptoControl: Sets integer variable to -1 PAC Control: Sets integer variable to +1
Set Variable False	Set Variable False	Both OptoControl and PAC Control: Sets integer variable to 0
Start Continuous Square Wave	Start Continuous Square Wave	1 —On Ethernet-based units, new square waves take effect as soon as a Start Continuous Square Wave instruction is sent.
Start Default Host Task	n/a	9
Start Host Task (Ascii)	n/a	9
Start Host Task (Binary)	n/a	9
Start Quadrature Counter	Start Counter	1
Stop Host Task	n/a	9
Stop Quadrature Counter	Stop Counter	1
Suspend Default Host Task	n/a	9
Transmit Character Via Serial Port	Transmit Character	11 —The OptoControl command transmits the character, whereas the PAC Control command puts the character in the serial port transmit buffer; when you are ready to send the packet, you must use Transmit NewLine. For details, see the <i>PAC Control Command Reference</i> (form 1701).
Transmit Newline Via Serial Port	Transmit Newline	11
Transmit String Via ARCNET	n/a	5
Transmit String Via Ethernet	Transmit String	11
Transmit String Via Serial Port	Transmit String	11
Transmit Table Via ARCNET	n/a	5
Transmit Table Via Ethernet	Transmit Numeric Table	11
Transmit Table Via Serial Port	Transmit Numeric Table	11
Transmit/receive Mistic I/O Hex String With Checksum	n/a	Checksum is not correctly supported.
Transmit/receive Mistic I/O Hex String With CRC	Transmit/Receive Mistic I/O Hex String	1 —See the <i>PAC Control Command Reference, Legacy Edition</i> (form 1711) for additional options for this command.

OptoControl Command	Comparable PAC Control Command(s)	Notes
Transmit/receive Optomux String	Transmit/Receive String	1 —The “Transmit/Receive String” command may be used to send a serial Optomux (or other protocol) string using a serial communication handle. B3000 brain users may consider reconfiguring their brain’s jumpers for <i>mistic</i> mode. B1 and B2 brain board users should consider upgrading to an E1 or E2 Ethernet-capable brain board.
Transmit/receive String Via ARCNET	n/a	5
Transmit/receive String Via Ethernet	Transmit/Receive String	11
Transmit/receive String Via Serial Port	Transmit/Receive String	11
Turn Off RTS	n/a	10
Turn Off RTS After Next Character	n/a	10
Turn On RTS	n/a	10
Variable False?	Variable False?	Evaluates true if value of integer variable is 0 . False is defined as zero .
Variable True?	Variable True?	Evaluates true if value of integer variable not 0 . True is defined as any non-zero value (any non-false value).
Write Byte To PC Memory (ISA Only)	n/a	3
Write Byte To PC Port (ISA Only)	n/a	3
Write Word To PC Memory (ISA Only)	n/a	3
Write Word To PC Port (ISA Only)	n/a	3

Notes for OptoControl and PAC Control Commands

1. The OptoControl command is directly converted to a comparable PAC Control command except as noted.
2. *mistic* interrupts are supported in PAC Control Professional version 8.2 and higher. Port 6 is not a valid port on SNAP-PAC controllers.
3. Legacy Opto 22 ISA controllers are not supported in PAC Control. Consider moving your control logic to SNAP PAC standalone and rack-mounted controllers.
4. Error handling commands are different and more extensive in PAC Control.

IMPORTANT: Keep in mind that the status and error codes that are returned by many PAC Control commands and that appear in the message queue often differ from the comparable command in OptoControl. Error handling charts and code may need to be modified accordingly.

If you are currently checking for errors by error number, Opto 22 recommends that you check instead for 0 (success) versus non-0 (error).

To search for places in your strategy that check for an exact match, try searching for the “Equal?” condition.

- a. In Configure Mode, choose Edit > Find.

The Find dialog box appears.

- b.** Under Search Scope, select Global.
- c.** Under Search For, select the Instruction and Condition radio button.
- d.** Under Instruction, select Equal?

Check each occurrence for any references to specific error codes that may have changed between OptoControl and PAC Control.

- 5.** ARCNET is not supported by PAC Project software. B3000-HA brains may be replaced by B3000-B (serial) brains or one of several SNAP Ethernet brains without having to change the rack, modules, and field wiring. SNAP PAC SB (serial) and EB (Ethernet) brains can also be used as replacements; they use the same modules but require a different rack or a rack adapter.
- 6.** "Priority" commands are not supported in PAC Control. You can make PAC Project charts more efficient by careful use of subroutines and "Delay" commands. For more information on this, see the *PAC Control User's Guide, Legacy Edition* (form 1710).
- 7.** M4RTU and M4IO controllers had special maintenance commands which do not apply to PAC Project controllers.
- 8.** To simultaneously read multiple values from a brain, or other values not directly supported by built-in PAC Control commands, you can construct your own low-level protocol commands to send directly to the I/O unit. This method is more difficult and error-prone than using built-in PAC Control commands for accessing I/O, but may be useful in certain cases such as speed-sensitive applications.

For example, you can construct a *mistic* command to read multiple latches (command 'S'), or to get analog lower/upper clamp values (command 'r'). *mistic* commands constructed manually within a strategy can be sent using the PAC Control command "Transmit/Receive Mistic I/O Hex String". For more information, see the *Mistic Protocol User's Guide* (Opto 22 form 270). For Ethernet-based brains, multiple addresses in the memory map can be read at once using the PAC Control commands in the "I/O Unit – Memory Map" group (for example, "Read Numeric Table from I/O Unit Memory Map"). See the *OptoMMP Protocol Guide* (Opto 22 form 1465) for information about the memory map.

You can also read or set multiple values conveniently in PAC Control by looping through similarly named and sequential variables. This is done by using pointers and the command "Get Pointer From Name." For example, to use a "for loop," create the point names (for example: "Point0", "Point1", "Point2," etc.) and then use the command "Get Pointer From Name" to update a pointer which you can then use to get/set latches for that point in the loop.

- 9.** PAC Control does not support manually manipulating the host task(s). Ethernet is used for host communications, supporting up to 31 sessions on the host port for SNAP PAC S-series controllers. Visit the Opto 22 website (www.opto22.com) or contact Opto 22 Product Support for current information about this feature.
- 10.** RS-232 handshaking using Communication Handles is not supported in PAC Control. Visit the Opto 22 website (www.opto22.com) or contact Opto 22 Product Support for current information about this feature.

11. Rather than referencing ports by number, data can now be transmitted and received using variables called Communication Handles. More generic open, close, transmit, receive, and other commands can be used with a variety of Communication Handle types, including TCP, serial, local files, and FTP. For more information, see Chapter 10, “Communication Commands,” in the *PAC Control User’s Guide, Legacy Edition* (Opto 22 form 1710).

Converting OptoDisplay Projects to PAC Display Pro

An OptoDisplay project is converted into PAC Display using a separate conversion utility.

NOTE: PAC Display Professional can access Ethernet-enabled FactoryFloor controllers running an OptoControl strategy. Specifically, PAC Display Pro can access SNAP-LCM4 or M4 series controllers via Ethernet when the controller has an M4SENET-100 Ethernet adapter card installed.



However, PAC Display is not optimized to communicate with FactoryFloor controllers, so the throughput will be much slower than it was using OptoDisplay to communicate to the OptoControl controller.

We highly recommend that you run tests with your system to determine if the throughput will be acceptable before you commit to using PAC Display with an OptoControl controller.

1—Preparing the Project

1. **If you have not already done so, back up the current, functioning OptoDisplay project to one or more separate locations.**
2. Confirm that the OptoDisplay project is saved in OptoDisplay R4.1. If you have an older version of OptoDisplay, contact Opto 22 Product Support.
3. Confirm that the FactoryFloor controllers used by the OptoDisplay project are configured on the PC used for converting to PAC Display.

2—Converting the Project

1. To open OptoDisplay Converter:
 - In Windows 7 and Windows Vista, press the Windows Start key , and then click Programs > Opto 22 > PAC Project 9.6 > Tools > OptoDisplay Converter.
 - In Windows 10 and Windows 8.1, press the Windows Start key , type **OptoDisplay Converter 9.6** and then press the Enter key.

You can also select File > Convert OptoDisplay Project in PAC Display Configurator Pro when no project is loaded.

2. From the File menu of the OptoDisplay Converter, select Open.
3. In the dialog box, navigate to the location of the OptoDisplay project to be converted.
4. Select the OptoDisplay project (the .MMI file) and click Open.

The OptoDisplay project is converted. If any errors occur, a warning will be displayed.

5. In the dialog box, navigate to the folder where you want to save the new PAC Display project and then click OK.

3—Followup

Updating the List of Controllers

After converting the OptoDisplay project to PAC Display, open and save the converted file, and then update the list of controllers which the project references.

1. Open PAC Display Configurator.

IMPORTANT: Do not try to open the converted project in PAC Display Runtime.

2. Select File > Open. In the Open File dialog box, navigate to the location where the converted file was saved.
3. Select the file and click OK.
4. If the project's windows do not appear in PAC Display Configurator, go to Windows > Open, select the windows you want to open, and then click OK.
5. Select Configure > Control Engine(s).
6. If the project uses multiple controllers, note the order of the FactoryFloor controllers listed in the Control Engines dialog box.

You will need to replace these controllers with PAC Project controllers in the same order.

7. Select one controller in the list and click Replace.
8. Click Browse and select the appropriate PAC Project controller(s) to be used.
9. Next to Strategy, select the appropriate PAC Control strategy location by clicking the Browse button to open the selection dialog box. Then select the strategy and click Open.

This is described in "Chapter 5, Configuring Control Engines and Tags" in the *PAC Display User's Guide, Legacy Edition* (form 1712).

If the project uses multiple controllers, **each controller must be replaced using the previous steps 7, 8, and 9.**

10. Click OK to close the Control Engines dialog box.
11. Select File > Save to save the converted project.
12. Choose Tools > Autocorrect Tags.
13. Choose Tools > Regenerate IO Scanner Tag Names.

Setting or Evaluating a Value

OptoDisplay and PAC Display both use a value of 0 for False. However, when setting a tag True, PAC Display uses -1 and OptoDisplay used +1. When evaluating a tag, 0 is evaluated as False and any non-zero value is evaluated as True. (False is defined as 0, and True is defined as any non-False value).

Reading or Changing Chart States

OptoDisplay and PAC Display both allow reading or changing chart states by Value dynamic attribute. The possible values are as follows:

Value	Chart State
1	Stopped
2	Suspended
3	Running

Value	Chart State
1	Stop
2	Suspend
3	Start
4	Continue

OptoDisplay also allows reading or changing chart states by Discrete (set/clear) dynamic attribute. When using Discrete, possible chart fields are Run, Stop, Suspend, and Continue.

If your OptoDisplay project used the Discrete dynamic attribute to read or change chart states, after converting your project to PAC Display, change the dynamic attribute to use Value instead (see [Changing Chart States](#) and [Reading Chart States](#) below).

The conversion report identifies all the charts that contain discrete chart state references. However, it only shows the *names* of the charts that have discrete references. It does *not* indicate how many references or where they are located. To locate the charts after the project is converted, use Edit > Find.

Changing Chart States. Changing chart states in PAC Display requires sending a value to the chart field called *State*.

1. Using the Select tool, double-click a graphic object to open the Graphic Dynamic Attributes dialog box.
2. Under Operator Driven Attributes, double-click the Send Value attribute.
3. In the Send Value dialog box, click the Tag Selection button (?) to open the Value Tag Selection dialog box.
4. Select Chart as the Item Type.
5. In the Field drop-down menu, select State.

Reading Chart States. Reading chart states in PAC Display requires reading a value from the chart field called *State*.

1. In the Graphic Dynamic Attributes dialog box, under Control Engine Driven Attributes double-click Visibility/Blink.
2. In the Send Value dialog box, click the Tag Selection button (?) to open the Value Tag Selection dialog box.
3. Select Chart as the Item Type.
4. In the Field drop-down menu, select State.