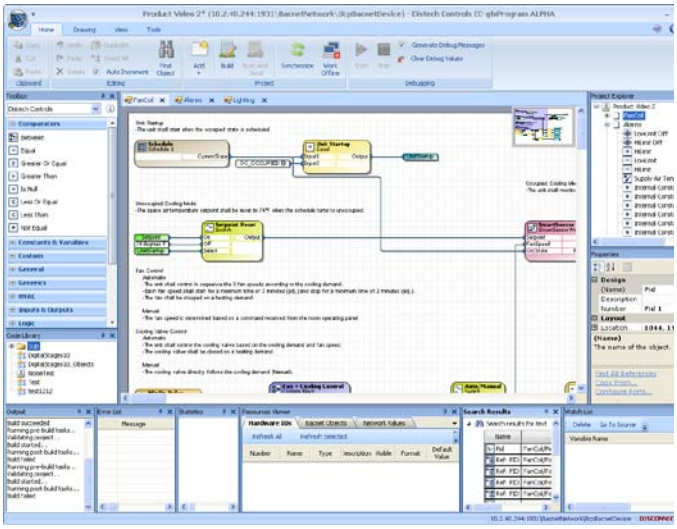
DISTECH CONTROLS®

Datasheet

EC-*gfx*Program

Graphical Programming Interface for Programmable Controllers



Overview

Distech Controls' EC-*gfx*Program Graphical Programming Interface (GPI) tool makes Building Automation System (BAS) programming effortless by allowing you to visually assemble building blocks together as necessary to create a custom control sequence for any HVAC / building automation application. By "dragging and dropping" a few block objects from the EC-*gfx*Program's vast library and connecting them with a simple "click, select, and release" process, you can quickly and easily assemble common control sequences and customized applications specific to your needs.

EC-*gfx*Program provides an intuitive and customizable programming environment with window panes that can be moved, docked, and hidden; it adapts to how you work. The programming area is where you visually compose your code and when two or more code sheets need to be managed, new programming sheets can be created and layered relative to each other. Coupled with a ribbon bar along with the project explorer pane, you have all the tools necessary to keep your code well-organized.

EC-*gfx*Program's block object toolbox provides you with an ample collection of components and functions that can be used to create simple to very complex control sequences. Use a Custom Block to keep your code clean by putting the specialized code that this block encapsulates on its own programming sheet. Block objects not only make coding clean and easy, but they also reduce basic errors that may arise when writing code conventionally. Furthermore, EC-*gfx*Program's smart code compiling, error list pane, Watch List, and live debugger allows you to execute code, view input/output values, and troubleshoot errors in real-time.

EC-*gfx*Program can be run from any multi-protocol software platform supporting BACnet® and LonWorks® devices such as Distech Controls' EC-Net^{AX} Pro, powered by the Niagara^{AX} Framework or from any LNS-based software such as Distech Controls' Lonwatcher.

Applications

- Designed to program Distech Controls' ECB, ECL and ECP programmable controllers.
- Furnished with *gfx*Applications, a diverse library of standard pre-coded, tested, and energy-efficient HVAC applications. These applications are modular, allowing you to easily customize them to your specific needs. Variable Air Volume applications are currently supported with a growing range of application types: Air Handling Unit, Roof top Unit, Fan Coil Unit, Heat Pump Unit, Chilled Ceiling Unit, Chiller plant, and more.
- Supports the configuration of a line of wireless battery-less devices[1] as well as the EC-Smart-Sensor and EC-Smart-Vue series of communicating sensors with LCD display.
- Supports large deployments with multiple device code download.

Features & Benefits

- Program both ECL/ECP Series LonWorks® and ECB Series BACnet® controllers with the same tool
- EC-*gfx*Program simplifies BAS programming:
  - Allows you to easily create a control sequence according to the engineer's specifications.
  - Uses Block-oriented programming that reduces your learning curve and results in fewer errors making it a faster and more intuitive programming method.
  - Reduces language barriers in international environments.
- Easily troubleshoot your application in real-time through live-debugging that shows block input and output values of the code being executed, and a Watch List to monitor specific process variables to detect errors as they occur.
- Supplied toolbox includes more than 100 pre-defined functions split into 14 categories including HVAC, Comparators, Logic, Math, Time, Custom, and Inputs & Outputs among others to simplify programming and reduce programming time.
- The EC-Net^{AX} wizard and LNS plug-ins are supplied as freeware: Program and configure the device with your preferred platform. There are no associated licensing costs.

1. Only when the controller is combined with Open-to-Wireless Receiver.

Features & Benefits *(Continued)*

- Create your own standard code libraries and toolboxes from your own code to better manage your favorite or most commonly used code or code sections
- Standardize and reuse code in your organization by sharing code libraries and toolboxes.
- Complete jobs faster and simplify field support with the Toolbox Builder by providing technicians with tested, non-modifiable, application-specific blocks that are known to work.
- Send your terminal application code to multiple devices at once for easier deployment and update. This eliminates the tedious task of uploading code to each individual device one by one.
- Device firmware update wizard allows you to conveniently upgrade multiple devices at once[1].
- Automatically import point type, name and unit/enumeration into Niagara^{AX} thereby saving time normally required to import and configure a controller's Internal Points such as Inputs, Outputs, Constants, and Variables.
- Live Trend block allows you to view and optimize system response and Pid tuning by monitoring controlled variables in real time. This is ideal to view control loop effect on supply air temperature, chilled water temperature, CO_2 level, etc.
- Assisted troubleshooting:
  - Real-time error checking identifies programming errors during program sequence creation.
  - Quickly locate coding errors in a large project by double-clicking on an error in the Error List.
- Network Variable fan-in aggregates multiple network information sources into one Network Variable Input to retrieve the highest, lowest, average, and sum of all inputted values.
- Obtain optimal control system response accuracy with Pid loops.
- Customizable blocks enable you to create unique functions and programs.
- Open support for industry-standard hardware allows you to connect your preferred sensing or actuating device to the controller.
- Communicate and receive more information from a LCD-based Smart-Sensor device than from a typical sensor.
- Reduce installation/retrofit time by taking advantage of wireless battery-less technology.
- Persistently store values such as fan or pump run time or number of start/stop cycles in the device so that these values are not reset by a power failure.
- Schedule your system's start/stop period or use it as a back-up to the centralized scheduling device in case of network communication failure.
- Quick access to manage, monitor, and override the values of Inputs, Outputs, Constant, Variables, and Network Variables through the Resource Viewer.
- Backup / Restore function stores the complete code in the controller allowing the retrieval of all programming code features
- The following advanced features are available with the ECB and ECL Series controllers:
  - Advanced mathematical functions such as sin, cosine, power, exponential, logarithm, and so on
  - For loop can be used to find highest, lowest, and average values

1 Available with ECB Series controllers.

Related Products

ECB, ECL and ECP Programmable Controllers

ECB Series Controllers — BACnet line of controllers

ECL / ECP Series Controllers — LonWorks line of programmable controllers

For more information on these or other Distech Controls products please refer to our web site at www.distech-controls.eu.

Block Objects[1]

Comparators - Comparators are blocks that evaluate two numeric inputs using a particular function (=, ≠, <, ≤, >, and ≥).

- Equal
- Not Equal
- Less Or Equal
- Greater Or Equal
- Less Than
- Greater Than
- Between
- Is Null

Constants & Variables - Constants are blocks that are mainly used to configure set values (setpoints, delays, limits, etc.) that may need to be made available to an HMI. Variables are blocks that are mainly used to monitor changing values or calculate new values using old ones that may need to be made available to an HMI.

- Constant Numeric
- Constant Enum
- Variable Numeric
- Variable Enum
- Internal Constant
- Internal Variable
- Analog Value
- Binary Value
- Multi State Value
- Null Value

Custom - Custom blocks are used to simplify code representation on a Programming Sheet by creating a block that contains code that makes up a unique sequence, function, or logic. They are also used to create blocks that do not already exist in the standard Toolbox pane and they can be saved in the Code Library for easy reuse. A Custom block can also be converted into a toolbox with the Toolbox Builder.

- Custom Block
- Conditional Custom Block
- Exported Input
- Exported Output
- For Loop
- Loop Info

General - General blocks are used to perform various important control loop functions in a program to provide control and supervision of a process.

- Latch
- Toggle
- Hysteresis
- Limit
- Digital Fault
- Numeric Fault
- Linear
- Ramp
- Rising Edge
- Falling Edge
- Count Up
- Count Down
- Startup
- Pid

Generics - Generic blocks allow a resource instance to be dynamically selected from the EC-*gfx*Program code. This is mainly used with the For Loop block.

- Generic Analog Value
- Generic Binary Value
- Generic Hardware Input
- Generic Hardware Output
- Generic Internal Variable
- Generic Multi State Value
- Generic Network Value
- Generic Timer
- Generic Pid Loop
- Generic ComSensor Condition
- Generic ComSensor Value

HVAC - HVAC blocks are used for standard HVAC requirements such as stage control.

- Analog Stages
- Digital Stages
- Digital Stages + Delay
- Smart Stages
- Stages With Modulation
- Optimum Start/Stop
- Thermostat

Inputs & Outputs - Inputs and Outputs are blocks used to interface with various types of physical inputs and outputs, as well as network variable inputs (NVIs) and network variable outputs (NVOs).

- Hardware Input
- Network Variable Input
- ComSensor
- Wireless Sensor
- Hardware Output
- Network Variable Output
- Floating Output
- Led Output
- Network Value
- ComSensor Condition
- ComSensor Value
- Smart Sensor Module
- Wireless Module

Logic - Logic blocks evaluate the binary values at two or more inputs according to the block's Boolean logic and to perform Boolean operations.

- And
- Or
- Xor
- Multiplexer
- Switch
- Not

Logic Binary - Logic blocks that operate on values at the bit level according to the block's Boolean logic.

- Bitwise And
- Bitwise Or
- Left Bit Shift
- Right Bit Shift

Math - Math blocks evaluate the values at two or more inputs according to the block's mathematical or trigonometric operator.

- Add
- Subtract
- Multiply
- Divide
- Absolute
- Modulus
- Summation
- Square Root
- Minimum
- Maximum
- Average
- Min / Max / Average
- Muldiv
- Sine
- Cosine
- Tangent
- Inverse Sine
- Inverse Cosine
- Inverse Tangent
- Power
- Ln
- Log

SNVT Conversions - The SNVT Conversion blocks are used to process structured 2 byte long SNVT types.

- SNVT_scene Demux
- SNVT_scene Mux
- SNVT_state Demux
- SNVT_state Mux
- SNVT_switch Demux
- SNVT_switch Mux

Block Objects[1]

Psychrometric - Psychrometric blocks are for psychrometric calculations.

- Dew Point
- Actual Vapor Pressure
- Enthalpy
- Wet Bulb
- Air Density
- Heat Index
- Humidity Ratio
- Relative Humidity
- Saturation Vapor Pressure

Time - Time blocks are used to configure delays, schedules, and time events.

- Min On Time
- Min Off Time
- Min On Off Time
- Real Time Clock
- Start Delay
- Stop Delay
- Start Stop Delay
- Timer

Tools - Tools are blocks that are used to help program developers keep their code organized.

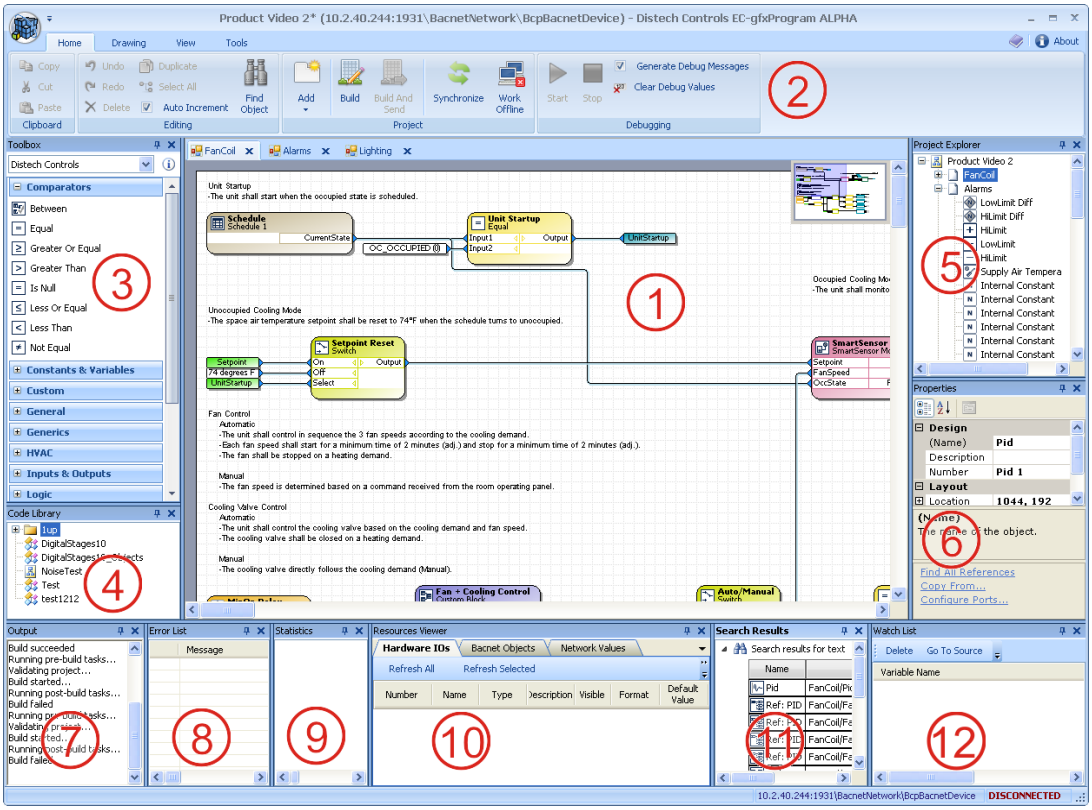
- Text
- Monitor
- Reference Hub
- Reference Target
- Live Trend Log

VAV - VAV blocks are used to interface with the flow sensor and actuator of a programmable VAV controller for single-duct and dual-duct applications.

- Damper Control
- Flow Sensor
- Actuator Control
- Flow Calculation
- Diff Pressure
- Internal Actuator

1 Block objects availability varies according to controller type. Refer to the EC-*gfx*Program User Guide for more information.

User Interface



1. **Programming Sheet:** This area is the main section of the user interface and is where device programming is done. “Drag and drop” block objects from the *Toolbox* then connect them together with a “click, select, and release” to build a control sequence.
2. **Ribbon Bar:** EC-*gfx*Program comes with a ribbon bar that allows for easy access to commonly-used functions.
3. **Toolbox Pane:** This library contains the block objects that can be “dragged and dropped” into the *Programming Sheet* to build a control sequence. The block objects are organized into 12 categories. You can select purpose-built toolboxes you have created with the Toolbox Builder to apply standard control methods to your project.
4. **Code Library:** This library contains saved code drawings (snippets) and projects that can be “dragged and dropped” into the *Programming Sheet*.
5. **Project Explorer Pane:** This tree-view list allows for easy navigation throughout the block objects and drawing documents of a project.
6. **Properties Pane:** This pane is used to define the properties of each block object, drawing documents, projects, etc. in the *Programming Sheet*. The properties are then dynamically adjusted according to the block object(s) selected.
7. **Output Pane:** This pane displays information and progress of the build.
8. **Error List Pane:** This list indicates errors when compiling the control sequence to the controller. This helps you to troubleshoot and debug problems.
9. **Statistics Pane:** After a control sequence is compiled, this pane displays certain statistics such as memory usage, resource usage, compiling time, etc.
10. **Resources Viewer Pane:** This pane displays information about all Hardware IOs, Wireless Inputs, Smart Sensor Outputs, Network Variables, Constants and Variables, such as name, value, and mode.
11. **Search Results Pane:** Search for objects based on text entered in the object’s properties, the type of block, or port names.
12. **Watch List:** Monitor a selection of process values during debug mode for troubleshooting.

Product Warranty & Total Quality Commitment

All Distech Controls product lines are built to meet rigorous quality standards and carry a two-year warranty. Distech Controls is an ISO 9001 registered company.

Specifications subject to change without notice.