



Omniflex ISaGraf Application Workbench

Model CC030A Omniflex ISaGraf IEC61131 Software Programmers Workbench.

DATASHEET

- IEC61131 PLC Programming Tools
- Remote Programming Support
- Windows 98/NT/2000/XP compatible
- Support for Ladder Diagram, Sequential Flow Chart, Function Block Diagram, Structured Text and Instruction List Programming methods.
- Online monitoring and debugging
- Bundled with the Omniset Pro Configuration Software to create a powerful integrated configuration, programming and debugging environment.



The Omniflex ISaGraf Application Workbench is development software that runs under Microsoft Windows for developing control application programs using the IEC61131-3 standard programming languages.

These application programs can be run on a range of Omniflex products such as the 'P' range of Maxiflex CPU Process Automation Controllers. Consult your Omniflex distributor for a list of available products.

For program development, the Application Workbench provides powerful and intuitive Windows based graphical and textual editors with drag-and-drop, and cut-and-paste to enhance ease of use.

Bundled with the Omniflex ISaGraf Workbench is the powerful but easy to use Omniset Pro configuration utility allowing complete control of software programming and configuration extensions for your entire network without leaving the Workbench environment.

The Workbench communicates with the target processor through a communications server installed with the workbench. This communications server provides the ability to download, monitor and debug programs remotely over any of the supported networks.

The Omniflex ISaGraf Application Workbench offers the following features:

- Project Management
- I/O Definition
- Modular Programming
- Simulation
- Real-time on-line debugging
- Document Generation
- Remote Programming over the network
- Simultaneous Target Register Viewing and Editing using "Omniset", the powerful companion configuration editing tool.



Omniflex ISaGraf Application Workbench

Model CC030A Omniflex ISaGraf IEC61131 Software Programmers Workbench.

THE PROGRAMMING ENVIRONMENT

Overview

The Application Workbench is a complete programming environment used to develop complex control algorithms. The Workbench fully supports six automation languages: the five IEC 1131-3 languages plus Flow Chart. This flexibility enables developers to choose the language that best suits their knowledge, style and application. The Workbench provides tools for editing, debugging, code generation, documentation, library management, archiving, on-line monitoring, off-line simulation and on-line changes.

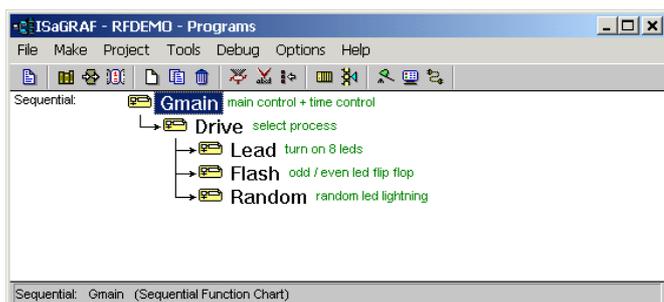
The Application Workbench uses the IEC61131 industrial standard PLC programming methodology for designing powerful applications without requiring the programmer to know complex, high-level computer languages. Designed to make it easier and faster to write applications, the Workbench imposes a simple but structured methodology and catches syntactic errors at during program writing. The result is a much more robust application code in the shortest possible development time.

The programming languages supported are:

- SFC – Sequential Function Charts
- FC – Flow Charting
- FBD –Function Blocks
- LD – Graphical Ladder Diagram
- ST – Structured Text
- IL – Instruction List

Program Structure

The components of a project are called programs. A program is a logical entity that describes one part of the control execution. Global variables (such as I/O variables) can be used by any program in the application. Local variables may be used by only one program. Programs are listed in a hierarchy tree, divided into different logical sections. The window below shows the programs and the links between them. The "Top level" programs appear on the left side of the hierarchy tree. Each program and sub-program may be programmed in a different language. Child programs may be started and stopped by the parent program.



Program Structure on the Application Workbench

The hierarchy of programs is divided into four main sections or groups:

Begin: Programs executed at the beginning of a cycle.

Sequential: Programs following SFC or FC dynamic rules

End: Programs executed at the end of a cycle.

Functions: The set of non-dedicated sub-programs

Graphic and Symbolic Debugging

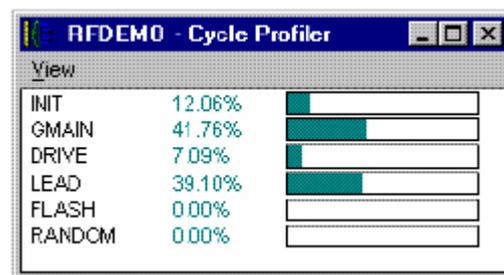
ISaGRAF includes a complete graphic and symbolic debugger. Comprehensive debugging capability is provided including cycle-by-cycle execution, breakpoints and variable monitoring and forcing.

Using the Workbench Debugger, it is possible to lock I/O while the debugger is connected, and to force an I/O point to a known state. Variable locking is a dynamic operation, and is not memorised when the application restarts. The lock operation applies to only one variable (one I/O channel) at a time.

Defining an I/O module as "virtual" disconnects the processing of the physical I/O channels allowing debugging of your application without needing to drive the I/O. In this mode, inputs/outputs of that module are not updated, and it is possible to use the Debugger to modify the input values. The virtual attribute is a static feature, and is stored when the application is stopped and restarted.

Cycle Profiling

The ISaGRAF Cycle Profiler is a powerful diagnostic tool that shows how cycle time is distributed between various programs, functions and function blocks of an application. This tool is very useful to have a quick diagnostic on the application performances, and leads the programmer to the parts of code which may need optimisations.



Development Configuration Management

A diary file is attached to each program. This is a text file, which contains all the notes about the modifications made to the program during its lifetime. The diary file can be edited, freely modified or printed at any time. When leaving the editor used to modify the source code of a program, a window is automatically opened to enter notes for the diary list. Such notes are inserted with the correct date and time into the diary file.



Omniflex ISaGRAF Application Workbench

Model CC030A Omniflex ISaGRAF IEC61131 Software Programmers Workbench.

Function Blocks

The user has the ability create custom function blocks that can be saved in a user library and used in multiple locations in many programs. This allows complex tasks to be designed once and used many times over.

Dictionary Editor

The ISaGRAF dictionary is an editing tool for the declaration of the internal variables, I/O variables, function block instances, and "defined words" of the application. The dictionary groups together the declared variables and function block instances of the application, and the words defined as constant strings.

Cross Reference Editor

The ISaGRAF workbench includes a cross-reference editor which provides the user with a total view of the declared variables in the project's programs, and where they are used. Cross-references are very useful for a global view of one variable life cycle. They help localise side effects, and reduce the time to understand the project during the maintenance. The cross-references may also be used for a global view of the complete dictionary of a project, so unused variables are easily found and the complexity of the project measured.

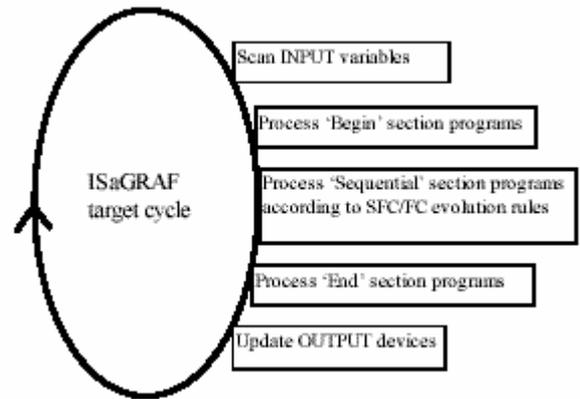
Program Flow

The program executes in a strict cycle:

First the Inputs are scanned

Then the Begin, Sequential and End sections of program are executed in turn.

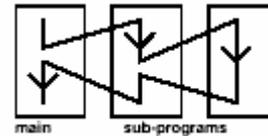
Then the Outputs are updated:



This sequence ensures that

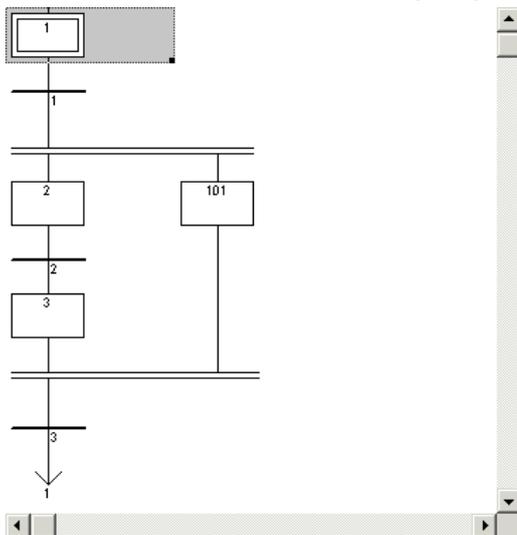
- Inputs keep their same value throughout a program cycle
- Outputs are updated only once a cycle
- The same global variables can be worked on from different programs

A sub-program or a function execution is driven by its parent program. The execution of the parent program is suspended until the sub-program or the function ends:



IEC61131 PROGRAMMING LANGUAGES

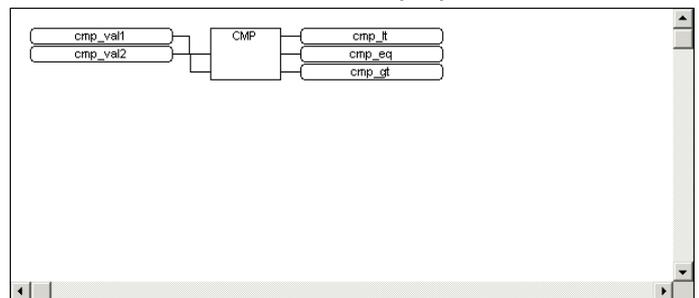
SEQUENCE FUNCTION CHART (SFC)



Sequential Function Chart (SFC), the core language of the IEC 61131-3 standard, divides the process cycle into

a number of well-defined steps, separated by transitions. The other languages are used to describe the actions performed within the steps and the logical conditions for the transitions. Parallel processes can easily be described using SFC

FUNCTION BLOCK DIAGRAM (FB)

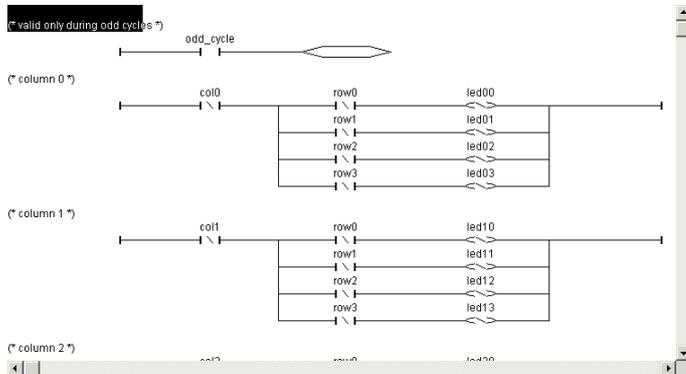


Function Block Diagram (FBD) is a graphical language that allows the user to build complex procedures by



taking existing function blocks from the library, and wiring them together on screen.

LADDER DIAGRAM



The Ladder Diagram (LD) is one of the most familiar methods of representing logical equations and simple actions, particularly in the United States. Contacts represent input arguments and coils represent output results. The Workbench's Quick LD editor provides the best compromise between high-level graphic capabilities and easy-to-use keyboard driven programming. LD and FBD programming can be mixed in the same chart.

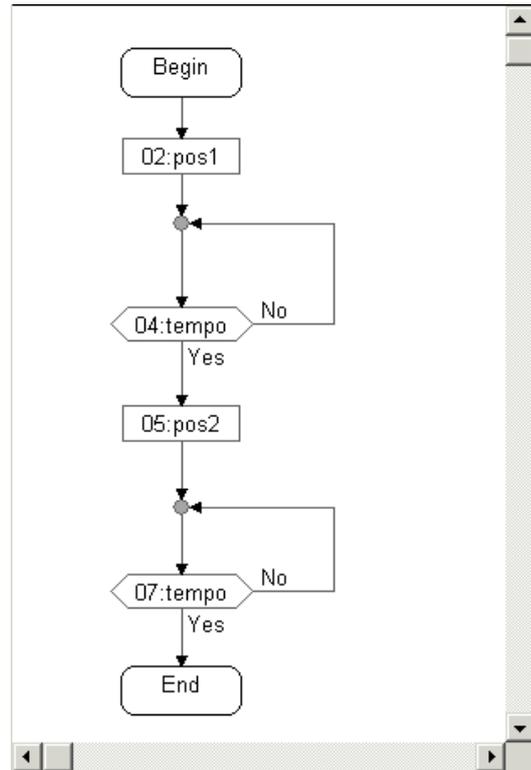
STRUCTURED TEXT

Structured Text (ST) is a high level structured language with a syntax similar to Pascal but more intuitive to the automation engineer. This language is primarily used to implement complex procedures that cannot be easily expressed with graphical languages (e.g. IF / THEN / ELSE, FOR, WHILE...). The ST editor guides the user to the correct syntax and punctuation. To further facilitate and speed development, highly useful validation and programmer assistance facilities are included.

INSTRUCTION LIST

The Application Workbench also includes Instruction List (IL), a low-level Boolean language similar to the simple textual PLC languages that are programmed at the register level.

FLOW CHART



Recognising that virtually every engineer graduating from college today has programmed in Flow Chart, the Workbench fully supports graphical Flow Chart programming. The Flow Chart is an easy to read decision diagram where actions are organised in a graphic flow. Binary decisions are used to control the flow. The Flow Chart Editor has full support for connectors and sub-programs. Actions and tests can be programmed in LD, ST or IL. The graphical editor allows each symbol to be re-sized independently, and automatically arranges the chart during development. The Level 2 code is displayed in a resizable editor window.

FUNCTION BLOCKS

In addition to the IEC 61131-3 languages and Flow Chart, the ISaGRAF Application Workbench includes a library with more than 60 ready-to-use blocks. Users can enlarge this library by writing functions and function blocks in LD/FBD/ST/IL languages or "C". The enhanced Library Manager is completed with import/export commands between the library and applications, so that new developed functions can easily be stored in library, and are directly ready for future applications.



Omniflex ISaGraf Application Workbench

Model CC030A Omniflex ISaGraf IEC61131 Software Programmers Workbench.

GENERAL SPECIFICATIONS

Minimum System Requirements

Disk Space Required	15Mbytes
Operating System	Microsoft Windows 98SE, ME, NT(SP6a), 2000(SP2), or XP.
Communications interfaces supported	RS232 Port Ethernet Port Conet Port (requires plug in card) (Options selected for use depends upon the hardware and application).

Supplied Components

- The Omniflex ISaGraf Application Workbench is supplied on CD with the following components ready to install:
 - Omniflex ISaGraf Programmer's Workbench
 - Conet Network Communications Server Engine
 - Omniset Pro Configuration Software
 - Full documentation in pdf format.

Documentation

- On-Line accessible documentation from application interface
- Multiple Tools to access Documentation
 - Table of Contents
 - Index
 - Search engine
- Paper version of documentation

Development Tools

- Project management tools including project groups
- Variable Dictionary management including importing and exporting of variables.
- Library Tools for the general management of I/O Structures, common functions blocks and user defined functions.
- Project Development independent from implementation using built in simulator
- On-line help including complete language reference.
- Graphical Development Interfaces
 - Language Editors (IEC61131, Flowchart)
 - Network Configuration
- Complete IEC61131 language development support
 - Ladder Logic
 - Structure Text
 - Function Blocks
 - Sequential Function Chart
 - Instruction List
- Flow Chart Development
- Supported Advanced Functions
 - Bit manipulation
 - Advanced mathematics
 - String manipulation
 - DCS Functions (Autotuning PID, Ramps etc.)
 - User-defined Functions written using any of the standard languages
- Supported Variable Types
 - Boolean
 - Integer
 - Real
 - Timers
 - Multi-dimensional arrays
- Multiple language segments can co-exist in the same application.
- Import/Export Facility allows variable configuration in standard CSV formats:
 - Tab delimited
 - Comma delimited

Debugging Tools

- Compilation Error and Warning message display
- Compile programs individually for faster debugging. (A program is a part of a single application (sub-routine) developed with one type of language.)
- Compile entire application at once
- Applications Simulation:
 - On development computer without I/O
 - On target system with I/O simulator
- On-line application monitoring (including simulation mode)
 - Diagnostic screen with run-time information
 - Graphically monitor languages on line
 - Monitor and modify variables on line
- I/O Forcing
- Run applications in cycle-by-cycle mode
- Start and Stop applications from the workbench at any time.
- User definable cycle time, on and off line

Remote Programming

- Access Target controllers over any supported network type.
- Addressing of targets from the Workbench allows multiple targets to be programmed from a single network connection
- Indirect access of remote targets on sub-networks using Conet Routing technology.
-

Omniset Pro Bundled

- The Omniflex ISaGraf Workbench is supplied bundled with Omniset Pro, the powerful utility for configuring and monitoring most Omniflex products.
- Run multiple copies of Omniset Pro together with the Omniflex ISaGraf Workbench for powerful monitoring and debugging of networked control systems.

OPC Compatible

- Compatible with Conet OPC server running on same machine (see separate OPC Server documentation for details)

Ordering Information

Model Description	Order Code
Omniflex ISaGraf Application Workbench for 32 I/O maximum	CC030A-032
Omniflex ISaGraf Application Workbench for 64 I/O maximum	CC030A-064
Omniflex ISaGraf Application Workbench for 75 I/O maximum	CC030A-075
Omniflex ISaGraf Application Workbench for 128 I/O maximum for 32 I/O maximum	CC030A-128
Omniflex ISaGraf Application Workbench for 256 I/O maximum	CC030A-256
Omniflex ISaGraf Application Workbench for unlimited I/O Systems.	CC030A-UL



WHAT IS IEC61131?

The IEC61131 Standard

The IEC61131 standard was created in the 1990's in recognition of the need for some form of standardisation in PLC programming languages.

With the proliferation of PLC manufacturers, the number of PLC language dialects grew to unmanageable proportions. It became impossible to retain hardware independence when each PLC manufacturer was demanding special knowledge and training in each PLC language dialect.

The IEC61131 standard was created to create a common language toolset that control engineers could learn that would allow them to program any PLC hardware that supports the standard. This removed the single supplier stranglehold that many suppliers had on their customers.

The IEC61131 standard is divided into a number of parts:

Part 1 General information Definition of basic terminology and concepts.

Part 2 Equipment requirements and tests Electronic and mechanical construction and verification tests. - published 1992

Part 3 Programmable languages PLC software structure, languages and program execution.

Part 4 User guidelines Guidance on selection, installation, maintenance of PLCs.

Part 5 Messaging service specification Software facilities to communicate with other devices using communications based on MAP Manufacturing Messaging Services.

Part 6 Communications via fieldbus Software facilities of PLC communications using IEC fieldbus

Part 7 Fuzzy control programming Software facilities, including standard function blocks for handling fuzzy logic within PLCs - published 1997

Part 8 Guidelines for the implementation of languages for programmable controllers Application and implementation guidelines for the IEC61131-3 languages.

The part applicable to PLC programming is IEC61131-3. The Omniflex ISaGraf Application Workbench conforms to this IEC standard for programming languages.

The IEC61131-3 programming Languages

The IEC61131-3 standard defines 5 programming languages:

- Sequential Flow Chart (SFC)
A graphical language for depicting sequential behaviour of a control system. It is used for defining control sequences that are time- and

event-driven.

Sequential Function Chart (SFC), the core language of the IEC 61131-3 standard, divides the process cycle into a number of well-defined steps, separated by transitions. The other languages are used to describe the actions performed within the steps and the logical conditions for the transitions. Parallel processes can easily be described using SFC.

- Function Block Diagram (FBD)
A graphical language for depicting signal and data flows through function blocks - re-usable software elements. FBD is very useful for expressing the interconnection of control system algorithms and logic.
- Ladder Diagram (LD)
A graphical language that is based on the relay ladder logic - a technique commonly used to program current generation PLCs. However, the IEC Ladder Diagram language also allows the connection of user defined function blocks and functions and so can be used in a hierarchical design.
- Structured Text (ST)
A high level textual language that encourages structured programming. It has a language structure (syntax) that strongly resembles PASCAL and supports a wide range of standard functions and operators.
This language is primarily used to implement complex procedures that cannot be easily expressed with graphical languages (e.g. IF / THEN / ELSE, FOR, WHILE...).
- Instruction List (IL)
A low level 'assembler like' textual language that is based on similar instruction list languages found in a wide range of today's PLCs.

The Application Workbench supports all 5 of these IEC61131-3 languages as well as a sixth language called "Flow Chart".

- Flow Chart (FC)
Recognizing that virtually every engineer graduating from college today has programmed in Flow Chart, the Workbench fully supports graphical Flow Chart programming. The Flow Chart is an easy to read decision diagram where actions are organized in a graphic flow. Binary decisions are used to control the flow. The Flow Chart Editor has full support for connectors and sub-programs. Actions and tests can be programmed in LD, ST or IL.