

Modicon Quantum automation platform

Concept programming software

Presentation

Concept is a software configuration and application programming tool for the Quantum and Momentum automation platforms. It is Windows-based software that can be run on a standard PC. The configuration task can be carried out online (with the PC connected to the Quantum processor) or offline (PC only). Concept supports the configuration by recommending only permissible combinations. During online operation, the configured hardware is checked immediately for validity, and illegal statements are rejected.

When the connection between programming unit (PC) and Quantum CPU is established, the configured values are checked and compared with actual hardware resources. If a mismatch is detected, an error message is issued.

Concept editors support five IEC programming languages:

- Function Block Diagram (FBD) language
- Ladder (LD)
- Sequential Function Chart (SFC) and Grafcet language
- Instruction List (IL)
- Structured Text (ST)

as well as Modsoft-compatible ladder logic (LL984). IEC 61131-3 compliant data types are also available. With the data type editor, custom data types can be converted to and from the IEC data types.

The basic elements of the FBD programming language are functions and function blocks that can be combined to create a logical unit. The same basic elements are used in the LD programming language; LD provides contact and coil elements.

The Grafcet SFC programming language (SFC) uses basic step, transition, connection, branch, join and jump elements. The IL and ST text programming languages use instructions, expressions, and key words. The LL984 programming language uses an instruction set and contact and coil elements.

You can write your control program in logical segments. A segment can be a functional unit, such as conveyor belt control. Only one programming language is used within a given segment. You build the control program, which the automation device uses to control the process, by combining segments within one program. Within the program, IEC segments (written in FBD, LD, SFC, IL and ST) can be merged. The LL984 segments are always processed as a block by the IEC segments. Concept's sophisticated user interface uses windows and menus for easy navigation. Commands can be selected and executed quickly and easily using a mouse. Context-sensitive help is available at each editing step.

PLC configuration

Variables for linking basic objects within one section are not required by the graphic programming languages (FBD, LD, SFC and LL984).

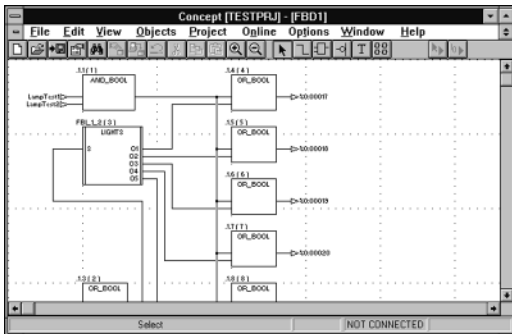
These connections are managed by the system, which eliminates any configuration effort. Other variables, such as variables for data transfers between different sections, are configured with the variables editor. With the data type editor, custom data types can be derived from existing data types.



Functions

Concept provides an editor for each programming language. These editors contain custom menus and tool bars. You can select the editor to be used as you create each program segment.

In addition to the language editors, Concept provides a data type editor, a variables editor and a reference data editor.



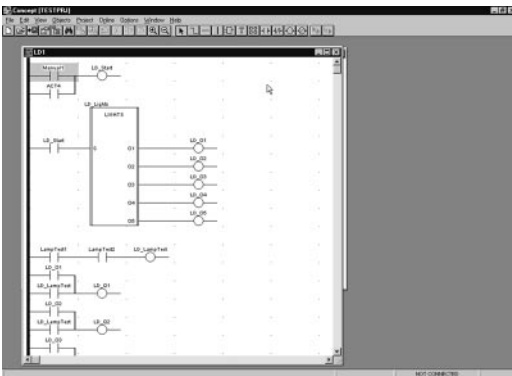
Function Block Diagram (FBD) language

With the IEC 61131-3 function block diagram language, you can combine elementary functions, elementary function blocks (EFBs) and derived function blocks (all three of which are known as FFBs) with variables in an FBD. FFBs and variables can be commented. Text can be freely placed within the graphic. Many FFBs offer an option for input extensions.

Concept provides various block libraries with predefined EFBs for programming an FBD. EFBs are grouped in the libraries according to application types to facilitate the search.

In the FBD editor, you can display, modify and load initial values; current values can be displayed. The CONT_CTL control library allows you to display animated diagrams of the FFBs and a graph of the current values.

For custom function blocks (DFBs), the Concept-DFB editor is used. In this editor, you can create your own function blocks from EFBs or existing DFBs. DFBs created in the FBD editor can be recalled in the LD, IL and ST editors, and DFBs created in the LD, IL and ST editors can be used in the FBD editor.



Ladder (LD)

With the IEC 61131-3 ladder diagram language, you can build an LD program with elementary functions, function blocks and derived function blocks (all of which are known as FFBs), along with contacts, coils and variables. FFBs, contacts, coils and variables can be commented. Text can be freely placed within the graphic. Many FFBs offer an option for input extensions.

The structure of an LD segment corresponds to that of a current path for relay circuits. On its left side is a left bus bar, which corresponds to the phase (L conductor) of a current path. As with a current path, only the LD objects (contacts, coils) connected to a power supply (i.e., connected to the left bus bar) are processed in LD programming. The right bus bar, which corresponds to the neutral conductor, is not visible. However, all coils and FFB outputs are internally connected to it in order to create a current flow.

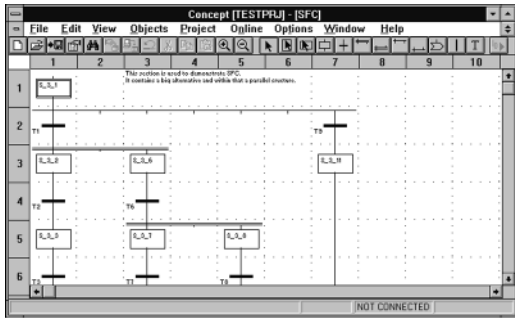
The same EFB block libraries available for the FBD editor can be used in the LD editor to program a ladder diagram.

In the FBD editor, you can display, modify and load initial values; current values can be displayed. The CONT_CTL control library, for example allows you to display animated diagrams of the FFBs, along with a graph of the current values.

For custom function blocks (DFBs), the Concept-DFB editor is used. In this editor, you can create your own function blocks from EFBs or existing DFBs. DFBs created in the LD editor can be recalled in the FBD, IL and ST editors, and DFBs created in the FBD, IL and ST editors can be used in the LD editor.

Modicon Quantum automation platform

Concept programming software



Functions (continued)

Sequential Function Chart (SFC) language

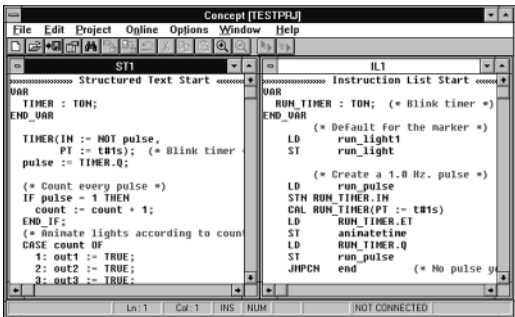
With the IEC 61131-3 sequential function chart (SFC) language, you can define a series of SFC objects that comprise a control sequence. Steps, transitions and jumps in the sequence can be commented. Text can be freely placed within the graphics. You can assign any number of actions to every step. A series of monitoring functions – e.g., maximum and minimum monitoring time – can be integrated into each step’s characteristics. The actions can be assigned an attribute symbol (as required by IEC) to control the action’s performance after it has been activated – e.g., a variable can be set to remain active after exiting.

Instruction List (IL)

With the IEC 61131-3 IL language, you can call entire functions and function blocks conditionally or unconditionally, execute assignments and make conditional and unconditional jumps within a program segment.

IL is a text-based language, and standard Windows word processing tools can be used to generate code. The IL editor also provides several word processing commands. Keywords, separators and comments are spell-checked automatically as they are entered. Errors are highlighted in color.

For custom function blocks (DFBs), the Concept-DFB editor is used. In this editor, you can create your own function blocks from EFBs or existing DFBs. DFBs created in the IL editor can be recalled in the ST, LD, and FBD, DFBs created in the ST, LD, and FBD editors can be recalled in the IL editor.



Structured Text (ST)

With the IEC 61131-3 ST language, you can call function blocks, execute functions and assignments and conditionally execute and repeat instructions. The ST programming environment is similar to Pascal. IL is a text-based language, and therefore standard Windows word processing tools can be used to generate code. The ST editor also provides several word processing commands. Keywords, separators and comments are spell-checked automatically as they are entered. Errors are highlighted in color.

Custom function blocks (DFBs) created with the ST editor can be called in the IL, LD and FBD editors; DFBs created in the IL, LD and FBD editors can be used in the ST editor.

Functions (continued)

Data type editor

The data type editor defines new derived data types. Any elementary data types and derived data types already existing in a project can be used for defining new data types. With derived data types, various block parameters can be transferred as one set. Within the program, this set is divided again into single parameters, processed, then output as either a parameter set or individual parameters. Derived data types are defined in text format, and standard Windows word processing tools can be used. The data type editor also provides several word processing commands.

Variables editor

The variables editor contains input options for:

- The variable type (located variable, unlocated variable, constant).
- The symbolic name
- The data type
- Direct address (explicit, if desired)
- Comments
- Identification as human-machine interface (HMI) variable for data exchange

Animation tables editor

In online mode, the reference data editor displays, forces and controls variables.

The editor contains the following options:

- Default values for the variable
- Status display for the variable
- Various format definitions
- The ability to isolate the variable from the process

Functions (continued)

Libraries

IEC library

The IEC library contains the EFBs defined in IEC 61131-3 (calculations, counters, timers, etc).

Extended library

The extended library contains useful supplements to various libraries. It provides EFBs for mean value creation, maximum value selection, negation, triggering, converting, building a traverse with interpolation of the first order, edge detection and determination of the neutral range for process variables.

System library

The system library contains EFBs in support of system functions. It provides EFBs for cycle time detection, utilization of various system clocks, control of SFC sections and system status display.

CLC and CLC_PRO library

The Continuous Control library can be used to set up process-specific control loops. In particular, it offers controller, derivative and integral control functions. The CLC_PRO library contains the same EFBs as the CLC library along with data structures.

Communication library

The communication libraries of built-in function blocks provide easy integration of programs which allow communication between PLCs or HMI devices from within the PLC's application program. Like other function blocks, these EFBs can be used in all languages to share data, or provide data to the HMI device for display to the operator.

Diagnostics library

The diagnostics library is used for troubleshooting the control program. It contains EFBs for action, reaction, interlocking, and process prerequisite diagnostics, along with signal monitoring.

LIB984 library

The LIB984 library provides common function blocks used in both the 984 ladder logic editor and the IEC languages. This allows for easy transition of portions of application code from the 984LL environment to the IEC environment.

Fuzzy logic library

The fuzzy library contains EFBs for fuzzy logic.

Analog I/O library

The ANA_IO library is used to process analog values.

Modicon Quantum automation platform

Concept programming software

References			
Concept packages			
Description	Licence type	Reference	Weight kg
Concept S Version 2.6	Single (1 station)	372 SPU 471 01 V26	–
Concept M Version 2.6	Single (1 station)	372 SPU 472 01 V26	–
Concept XL Version 2.6	Single (1 station)	372 SPU 474 01 V26	–
	Group (3 stations)	372 SPU 474 11 V26	–
	Team (10 stations)	372 SPU 474 21 V26	–
	Site (network)	372 SPU 474 31 V26	–
Concept EFB Toolkit Version 2.6	Single (1 station)	332 SPU 470 01 V26	–
HVAC Function Blocks Library	Site (network)	372 HVA 160 30V25	–
Concept maintenance and diagnostics packages			
Description	Licence type	Reference	Weight kg
Concept Application Loader Version 2.6	Single (1 station)	372 SPU 477 01 V26	–
Concept Updates			
Description	Licence type	Reference	Weight kg
Concept XL Version ●● to Concept XL V 2.6	Single (1 station)	372 ESS 474 01	–
	Group (3 stations)	372 ESS 474 03	–
	Team (10 stations)	372 ESS 474 10	–
	Site (network)	372 ESS 474 00	–
Concept S Version ●● to Concept S V 2.6	Single (1 station)	372 ESS 471 01	–
Concept M Version ●● to Concept M V 2.6	Single (1 station)	372 ESS 472 01	–
Modsoft version ●● to Concept XL V 2.6	Depends on number of users	372 ESS 485 01	–
Concept EFB Toolkit version ●● to version 2.6	–	372 ESS 470 01	–
Concept Documentation			
Description	Number of volumes	Reference (1)	Weight kg
Installation	1	840 USE 492 0●	–
Programming	3	840 USE 493 0●	–
Concept IEC Block library	13	840 USE 494 0●	–
Concept 984 LL Block Library	2	840 USE 496 0●	–
Concept EFB Tool User Manual	1	840 USE 495 01	–

(1) ● = Defines the documentation language: 0 English, 1 French, 2 German, and 3 Spanish.