Engineering Support Tools for the MICREX-NX

Yoshitomo Takeuchi
Keisho Yamano
Takashige Mori

1. Introduction

The process automation system (PAS) market in Japan has transitioned in recent years from a period of growth, driven largely by new demand, to a mature market where most of the demand comes from system updating and renewal, and these conditions – under which significant market growth is not expected – are continuing. Meanwhile, PAS hardware components such as human-machine interfaces (HMIs), controllers, I/O devices, and the networks that connect them, have continued to adopt open standards, and it is becoming difficult to differentiate components made by different companies. Under these market conditions, system vendors are facing the difficult task of increasing the added value of their application software, i.e., system vendors must effectively utilize their accumulated wealth of control-related expertise in order to differentiate their products from those of competitors while, at the same time, realizing highly efficient and high quality engineering capable of surviving in the severe price competition of a mature market.

In other words, the importance of engineering will only increase for system vendors, and in order to succeed in the above-described task, a suitable platform for engineering support tools must be provided.

The engineering station (ES) engineering support tool for the MICREX-NX meets market needs for the above-described engineering support tools and also provides various functions.

2. ES Functions and Features

2.1 ES functions

Figure 1 shows a functional overview of the ES. Configured on a Windows*1 personal computer, the ES is a common platform for a variety of support tools, such as an HMIs and controllers. The ES is a collection of multiple types of engineering software programs, all of which are integrally controlled by the manager. The manager calls the various engineering support tools, and by means of data linked among software programs and the use of a common platform, provides a function for managing the support software.

*1: Windows is a registered trademark of Microsoft Corporation of the USA.

2.2 ES features

(1) Integrated engineering data

Data from the various engineering support tools called by the manager are integrally managed in a database. Because input data is under centralized control in the database, inputting data to one location will cause a plurality of related data to be updated automatically. As a result, the task of inputting data is made more efficient, and data inconsistency due to mistaken input or other human error is eliminated and software quality is improved.

(2) Hierarchical engineering

In an ES, project data and library data for control and monitoring software and the like is hierarchically managed from a Windows Explorer-like screen. As a result, an engineer is able to access desired data intuitively and speedily, and copy-and-modify operations can be readily performed for each hierarchical unit.

(3) View layout

The three views that form the core of MICREX-NX engineering enable integrated engineering data to be observed from different angles corresponding to each view. The appropriate view can be selected according to the situation and the phase. Figure 2 shows an example of the view screen.

① Component view

The component view is used to set hardware parameters relating to the automation system (AS) of the control system, I/O devices that are supplied to the AS, the operator station (OS), the ES, and the network that connects these hardware components. One special feature of the component view is that it enables monitoring of the resource usage status of the AS. Moreover, the hardware configuration (HW Config) launched from this view can also be used when migrating the existing I/O, and the process device manager (PDM) is able to perform centralized setting, failure diagnosis

Fig.1 ES functional overview

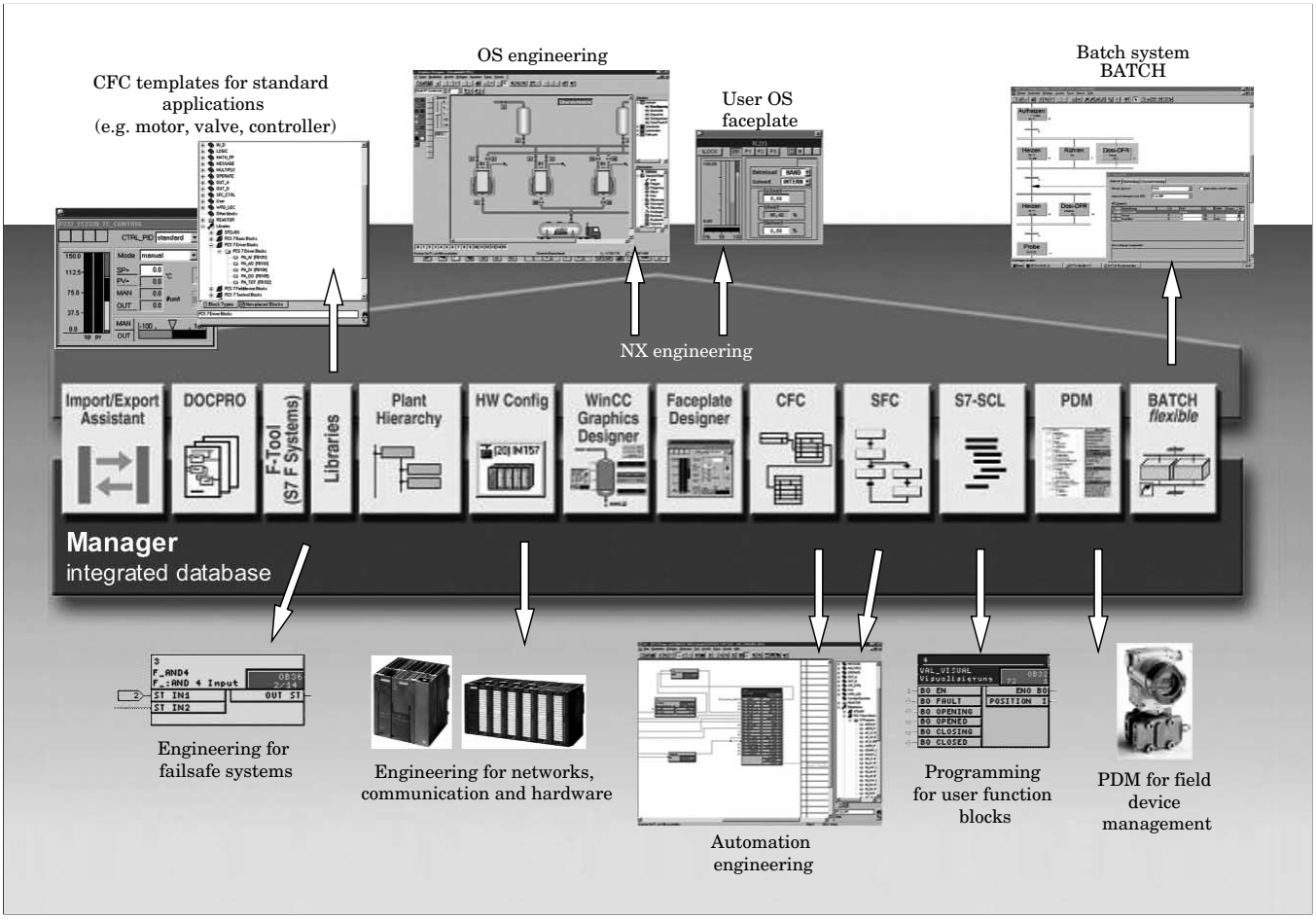


Fig.2 View layout

and loop checks for intelligent field devices.

② Plant view

Plant view shows the hierarchical management architecture for the purpose of providing an easy to understand overview of the plant to which control will be applied. This hierarchical structure complies with IEC61512 (ISA S88), the international standard for batch control. Operation from a Windows Explorer-like screen to create a hierarchy that matches that hierarchical structure of the actual plant enables efficient management of the controller control program and the HMI screen. Moreover, the plant hierarchy defined here is automatically evolved to the OS screen hierarchy.

③ Process object view

Though it is difficult to grasp the overall aspect of scattered objects clearly by using hierarchical management of plant view, process object view can provide an environment to manage process objects intensively. The various parameters for process objects such as valves and motors scattered throughout an individual control program or HMI screen can be set or modified all at once by operation from an Excel*2-like screen, without having to open individual files. Integration with Office*3 facilitates copy-and-paste operations from various lists such as an I/O list received from a customer.

*2: Excel is a registered trademark of Microsoft Corporation of the USA.

*3: Office is a registered trademark of Microsoft Corporation of the USA.

(4) Library

The MICREX-NX contains a library for each project, and libraries may be created easily by copying and pasting control programs created by engineers. Thus, control expertise can be accumulated and readily utilized in future engineering tasks. Also provided are library packages, equipped with many technical blocks that contain a set of function blocks such as linearization, control blocks, and block icons and faceplates, enabling the engineer to realize a high level of control.

3. Engineering Functions and Features

3.1 Overview of MICREX-NX engineering

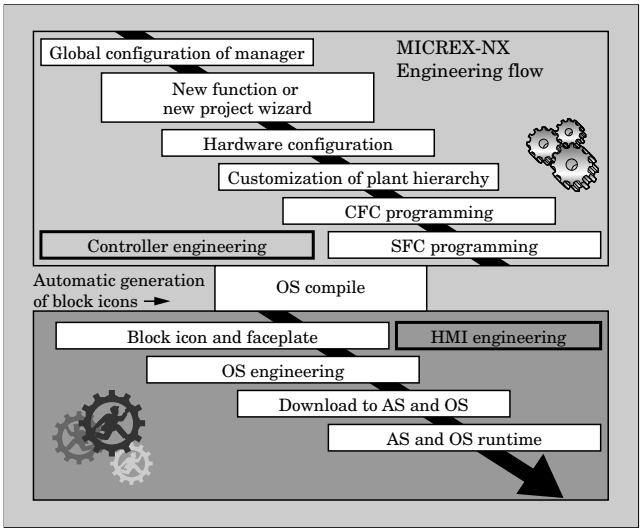
Figure 3 shows the MICREX-NX engineering flow. With the MICREX-NX, systems for performing controller engineering and HMI engineering are linked to one another to increase the engineering efficiency.

3.2 Controller engineering functions and features

A control program is generated by combining the following three languages in accordance with the desired goal. Regardless of the language used to generate a control program, the memory that stores parameters used in each program is allocated automatically to permit use from only that program, and therefore the engineer does not have to worry about the problem of duplex writing in which multiple control programs write to the same physical memory. Moreover, this arrangement enables a library to be operated easily. Compared to previous model types, the functionality has been increased significantly at this point.

Additionally, control program debugging is implemented by either connecting an AS to an ES, or by running a PLCSIM control simulator on an ES to enable online monitoring of each control program, and thereby increasing the efficiency of control program debugging.

Fig.3 MICREX-NX engineering flow



(1) Continuous function chart (CFC)

CFC that conforms to IEC61131 FBD is used to generate a continuous control program in the MICREX-NX. Figure 4 shows a screen shot of the CFC editor. This screen shot shows an example of motor operation control, and consists of a protection interlock function, an operating function and a control function. The control program can be created simply on a CFC editor by dragging and dropping control blocks known as function blocks (FBs) and function calls (FCs) and then by linking the control blocks together with connection lines. Many various control blocks are available, including instrumental loop control, sequential control, numeric processing, logic processing, and the like, and these program components can be combined freely on the same chart according to the desired control to generate a control program. The result is a change from complex, vertically segmented engineering for each control function, as seen in Fuji Electric's previous models, to sophisticated and horizontally structured engineering. Moreover, the computational sequence may be customized and unused pins of control blocks may be hidden in order to support various detailed needs. Additionally, a functionally organized chart may be converted into control blocks to provide an environment that enables the easy reuse of control programs. The above features interact organically to enhance the engineering efficiency.

(2) Sequential function chart (SFC)

Conforming to IEC61131, an SFC is used to generate a sequential control program. Figure 5 shows an example screen shot of the SFC editor. The layout of signals involved in the generation of steps and transitions can be accomplished by simply selecting the required signals from a control block parameter list. Moreover, the use of an SFC visualization package enables online SFC status display and operation from an OS, without any special engineering.

(3) Structured control language (SCL)

SCL conforms to the structured text (ST) of IEC61131. Since this programming language can

Fig.4 CFC editor screen shot

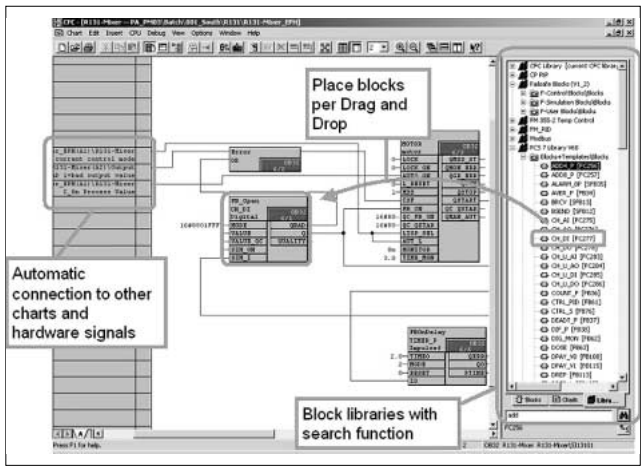


Fig.5 SFC editor screen shot

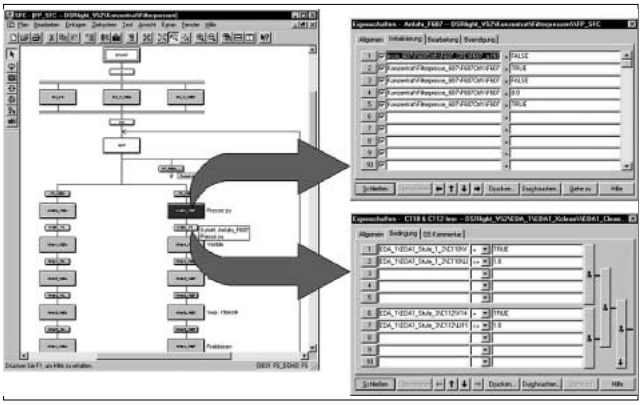
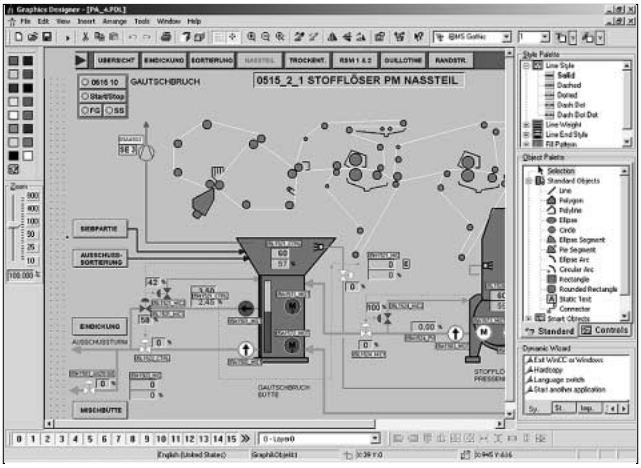


Fig.6 Graphic designer screen shot



provide a program with control structures, SCL is mainly used when generating the FBs and FCs in a CFC. SCL can be applied to generate more flexible control programs since it enables a user to generate required functions by him/herself and then to use those functions.

3.3 HMI engineering functions and features

As has been described above, HMI engineering in a MICREX-NX system is closely related to the controller's control program. When a control block, for example PID control, is created in the controller, two HMI components are automatically generated. One is a block icon that appears as a symbol on the HMI screen. Another one is a faceplate window, is called when a block icon is clicked and is used to implement detailed status display, operation and settings. On the other hand, a graphic designer is used to create an HMI screen according to the process image, and then automatically generated block icons are repositioned according to that image to complete the HMI screen. A faceplate designer may also be used to generate a faceplate according to user needs.

(1) WinCC Explorer

The WinCC Explorer is a common platform for HMI engineering. In addition to the abovementioned graphic designer, various other functions such as a global script and user manager are also provided.

(2) Graphic designer

Figure 6 shows a screen shot of the graphic designer. The graphic designer is a software program for creating HMI screens. The graphic designer supports HMI screen creation with the same operability and expressive ability as general-purpose image processing software. Moreover, because users are able to create their own libraries, previously created HMI parts can be accumulated and reused, providing an arrangement that leads to more efficient screen creation and to improved screen operability and the like.

(3) Global script

A script language conforming to C or visual basic (VB) is provided, enabling general-purpose SCADA-like screens to be customized flexibly.

(4) User management

For the purpose of ensuring plant security and traceability, all operators are given detailed authorization rights. Operation levels such as operational authorization can be specified per plant.

3.4 Support software

A group of tools that improve the efficiency and quality of MICREX-NX engineering are introduced below.

(1) Import/export assistant

The import/export assistant provides the capability to generate multiple control programs all at once, and all having the same functionality. A control program template is created, and then the program destination, I/O parameters, and other data that are unique for each control program are specified in a single operation from an Excel-like screen and expanded to multiple programs. The expanded settings file can be stored in an ES so that when a control program requires modification, a pre-existing setting file can be used to implement that change instantaneously in multiple control programs. Figure 7 shows an overview of the import/export assistant.

(2) Version cross checker

The version cross checker provides the capability for version management of the generated control programs. This software is provided with a function that highlights the display of project contents that have been added, deleted and/or changed before and after modification, thereby enabling graphical verification of that modification. The version cross checker can be used in a wide variety of applications such as control program version control and failure analysis. Figure 8 shows an example screen shot of comparative results obtained by the version cross checker.

(3) Control simulator (PLCSIM)

With the MICREX-NX, an ES personal computer can run a simulation of a controller. The simulation environment is easily configured, and a single ES personal computer can achieve the equivalent debug-

Fig.7 Import/export assistant

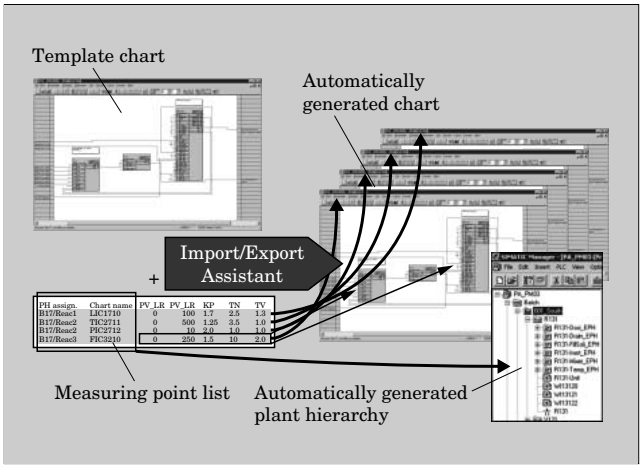
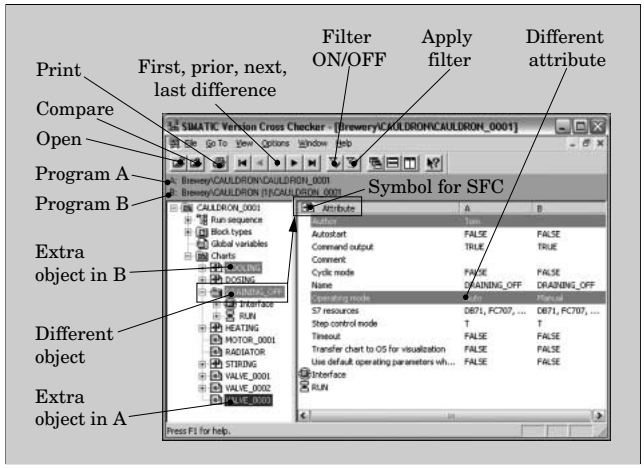


Fig.8 Version cross-checker screen shot



ging performance as one OS and one AS connected to each other. Accordingly, the debugging efficiency of a controller and HMI screen is dramatically increased. Moreover, an all-in-one personal computer enhances portability and enables performance to be verified at any location.

4. Conclusion

With the sophisticated MICREX-NX engineering support tools, previously acquired control expertise can be fully utilized to realize highly efficient and high quality engineering that satisfies market needs. In the future, Fuji Electric intends to continue to supply ideal engineering platforms and to contribute to the development of process automation systems.