ISSN 0429-8284

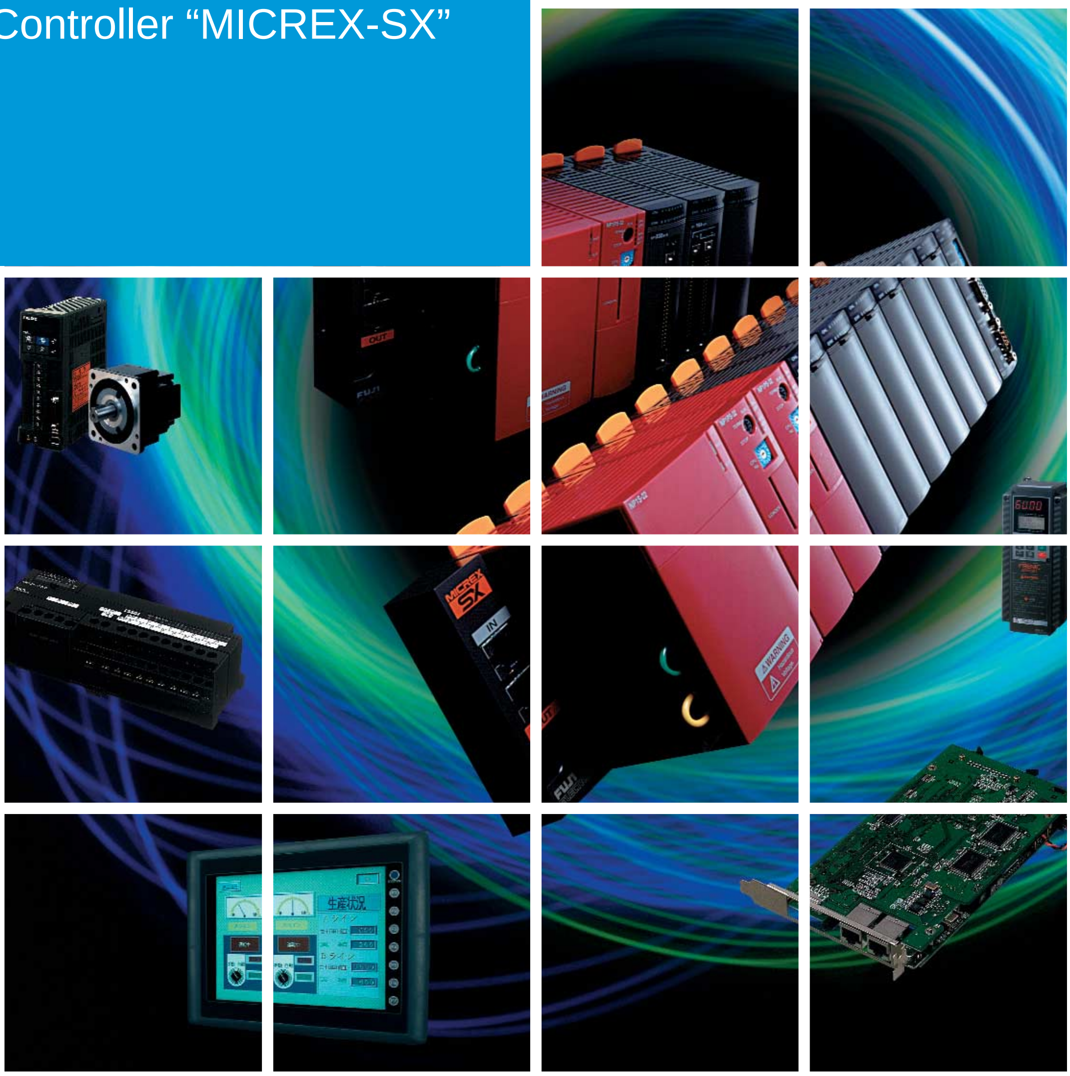
Whole Number 190

FUJI ELECTRIC REVIEW

3

2000 VOL.46

Integrated Controller “MICREX-SX”



FUJI ELECTRIC

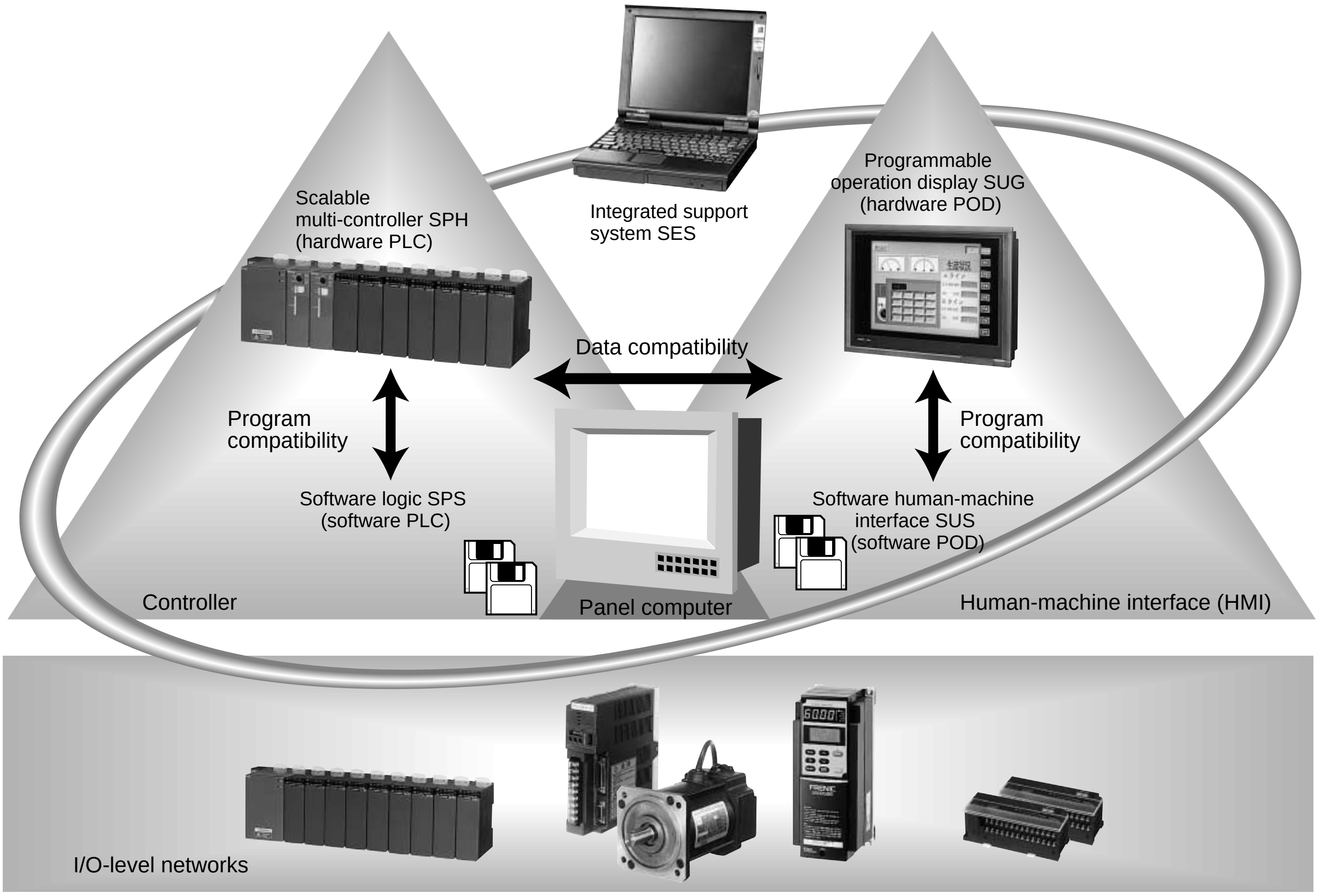
Creating a Better Tomorrow with Trustworthy Technology

FUJI ELECTRIC

Integrated Controller

MICREX-SX Series

meets Open Systems and unifies Control, Operation, and Monitoring.



The integrated controller MICREX-SX series realizes program compatibility, common use of data, and a unified operation environment.

● Scalable multi-controller SPH (hardware PLC)

Realizes a 1-ms control by load distribution with multi-CPU structure and high-speed processing with execution time of 20ns per instruction. This controller realizes the optimum system construction in every case of control and is the next generation controller far beyond the concept of existing programmable controllers.

● Software logic SPS (software PLC)

Uses IEC standard language common to the SPH. Can connect to the SX bus and various I/O-level networks.

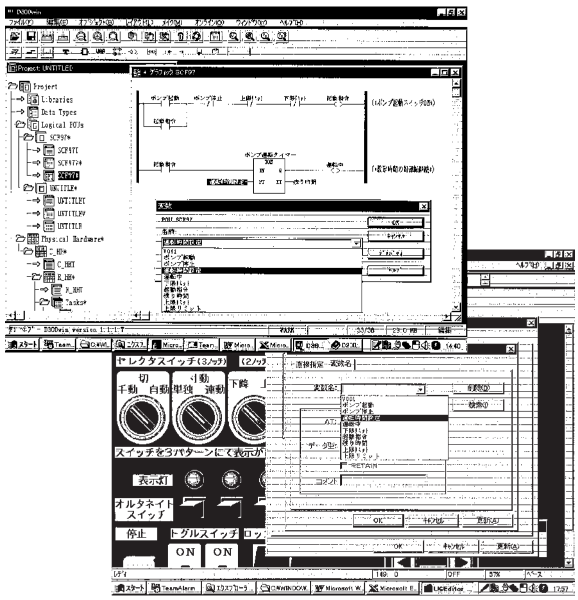
● Programmable operation display SUG/software human-machine interface SUS (hardware POD/software POD)

The two lines of exclusive-use hardware SUG and personal computer-use human-machine interface SUS are available. Graphic Display data is compatible between both types.

● Integrated support system SES

Supports the controller and the human-machine interface in the unified operation environment.

A programming display by the scalable multi-controller



A programming display by the human-machine interface

Fuji Electric's Integrated Controller

MICREX-SX

FUJI ELECTRIC
REVIEW

3

2000 VOL.46

Integrated Controller "MICREX-SX"

CONTENTS

Cover Photo:

At present, in the production activity of the enterprise, globalization, total cost-down, total efficiency up for diversified and small-quantity production, etc. are urgently needed. Therefore, the production system has strongly asked for open system, standardization, flexibility, and linking to information technology (IT).

Fuji Electric's integrated controller "MICREX-SX" series is a controller aiming the opened, distributed and objective system, and can greatly contribute to innovation of the production system.

The cover photo shows an image of the expanses of peripheral devices such as inverter and servodrive around the central "MICREX-SX" series.

Fuji Electric Co.,Ltd. Head Office : No.11-2, Osaki 1-chome, Shinagawa-ku, Tokyo 141-0032, Japan

Increasing Applications of Integrated Controller “MICREX-SX”

Ken-ichiro Ide
Keiichi Tomizawa
Masayuki Noguchi

1. Introduction

One and a half years have elapsed since the “MICREX-SX” series was introduced to the market. This series has been favorably received due to its concept of anticipating future needs, its application to new fields using high-speed processing, and open architecture approach that match the needs of the time. To expand the application range further, we have improved the functions not only of the hardware level, but also of the software and usability.

In this paper, we will introduce a summary of these improvements. Their details and application examples will be described in other papers in this special issue.

2. Expansion of Application Range by Integration with Peripheral Modules

2.1 Integration of control, operation and monitoring realized on an open platform

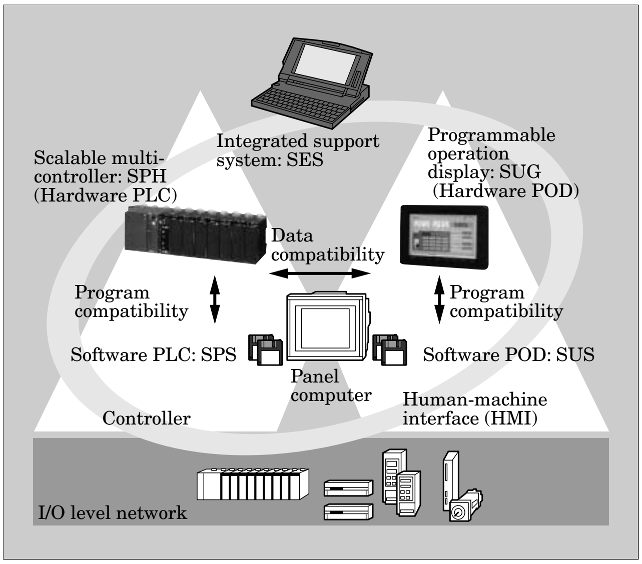
There are two types of programmable controllers (PLCs) and programmable operation displays (PODs), a hardware type which is realized with each specific hardware, and a software type, for which functions are realized with a personal computer on an open platform.

The programs of the hardware PLC (SPH) and software PLC (SPS), and the hardware POD (SUG) and software POD (SUS) are designed for compatibility with the integration of these devices. Therefore, when developing programs, effective development and device transition is possible without having to be conscious of the implementation system, and further extension of the application fields are intended.

Here, we have provided the following personal computer based ISA bus boards to utilize the expandability of personal computers and to construct integrated systems with control components such as PLC, POD, etc.

(1) OPCN-1 board which is an open network below PLC in rank
(2) SX bus board that connects the variety of devices supported by the SX bus
(3) PLC board which integrates the equivalent functions of the hardware PLC (SPH300) and realizes high-speed processing on a personal computer
(4) Other various boards to realize organic combinations with components of MICREX-SX

Fig.1 Concept of integration of control, operation and monitoring

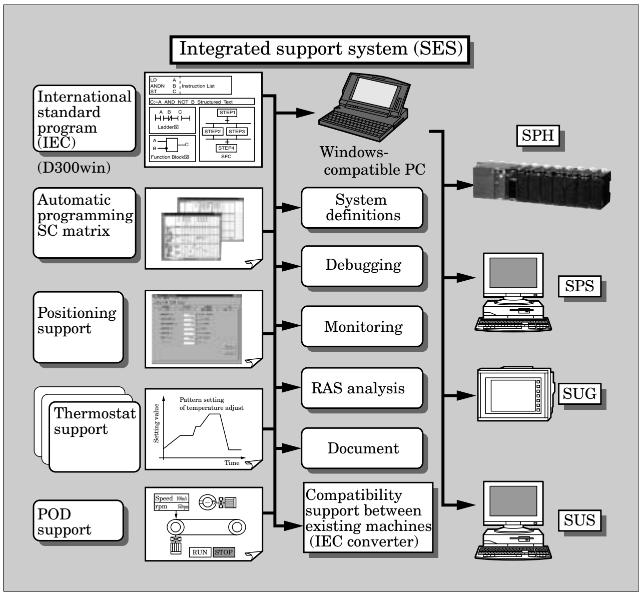


As mentioned above, owing to the support from both the software and hardware due to the superior environment for program development and the provision of a wide variety of ISA bus boards, an optimal system construction is possible. In Fig. 1, the concept of the integration of control, operation and monitoring is shown.

2.2 Enriched integrated support system

The integrated support system (SES) of the MICR EX-SX series is implemented under an integrated environment that centers around PLC support with an international standard language, and integrates POD support and function module support. The SES improves efficiency of control program development for high functionality and improves the efficiency of peripheral device support. Figure 2 shows a functional configuration of the integrated support system.

Fig.2 Functional configuration of integrated support system



We will introduce specific means to make each support tool efficient, from software development to maintenance.

2.2.1 D300win in accordance with international standard languages

The MICREX-SX series provides a programming support tool (D300win) that is completely based on the international standard languages (IEC61131-3), that are utilized in the major advanced nations such as Europe and North America, and can prepare IEC compliant programs with a minimum knowledge. Since being brought to market, improvements to the D300win have been planned to further enhance usability, and version 2 is introduced here. [For details, refer to the other paper titled, “New Features of the Loader D300winVer.2” in this special issue.] Some of features of the D300win including a list of new features of this latest version are described below.

(1) Programming with each language

The D300win fully supports 5 languages (IL, ST, LD, FBD and SFC) and programs can be developed using the most suitable language depending on the application.

(2) Automatic allocation of memory address

Conventionally, the memory addresses of variables were consciously allocated by a programmer when a program was being created. With the D300win, this allocation is automatically performed only by designating the data type and property of the variables according to the specifications of IEC. Consequently, the control of memory address allocation is reduced and redundant allocations can be prevented.

(3) Backup of program and data

In software development, the MICREX-SX is based on general source management, but provides the following two methods of backup of programs and data in consideration of lost source files in case of an emergency.

(a) Programs in the controller can be uploaded to the D300win. In addition to uploading as an IL language program, conversion from IL language to LD language is possible.

(b) The entire source may be stored in and read out from memory cards in the controller. The user can edit documents with the same content as was created.

(4) Creation of documents

The objects to be made efficient are not only software development, but from the viewpoint of supplying a device to an user, also include complete documents attached to the product. D300win realizes a print level equivalent to that of commercial applications running on the Windows[*1], and facilitates the task of and saves power in creating documents.

2.2.2 Linking of variables between PLC and POD support tools

In general, the POD editor directly specifies the PLC memory address when transferring data and signals between the PLC and the POD. With connection to the POD in the MICREX-SX, variables in both support tools are automatically linked, and can be shared. Therefore, the task of address allocation in both these support tools can be made remarkably more efficient.

The shared variables can be allocated with 30 characters, preventing careless mistakes such as redundant allocation.

2.2.3 Integrated function module support

The operating environment and connection form of support tools for function modules such as PLC position control, fault diagnosis and communication are made common with the D300win. Functions can be realized simply by implementing changes at the software level, enabling integration of the support tools and facilitating management of peripheral devices.

Further, in consideration of working offline, the parameters that were set in the various support tools are saved in the D300win project, and can be forwarded together with the program to PLC.

2.3 Position control function implemented in software

Conventionally, since a position control system was generally configured using a custom module (hardware) for each control function, it was necessary to prepare hardware for each position control function on the user side. In the MICREX-SX, the majority of the position control function is realized with software using high-speed CPU processing and a high-speed bus (SX bus). Therefore, the user can configure various position control systems with minimum types of hardware. In Fig. 3, a configuration of the position control module and software for position control is shown.

---

*1 Windows: A registered trademark of Microsoft Corp., USA

This time, we have attempted to enrich the software of position control by preparing approximately 80 types of software. Table 1 lists the software for position control. Application programs can be simply configured by combining these types of software.

On other hand, as hardware, a servo-system directly coupled to the SX bus (FALDIC-α) is newly provided, enabling application to fields requiring high-speed response.

In addition to the above provided products, we are planning to provide motion control, which is a higher-grade position control, and to comply with JEM1473, "Function block for motion control of general-purpose programmable controller" of the Japan Electrical Manufacturer's Association.

Thus, the supported range of the MICREX-SX is further expanded, and depending on the position control function and accuracy, the user can change the module and software combination without having to select a device from different device groups. Scalable expansion can be achieved without wasting hardware and software resources.

Fig.3 Configuration of position control module and software for position control

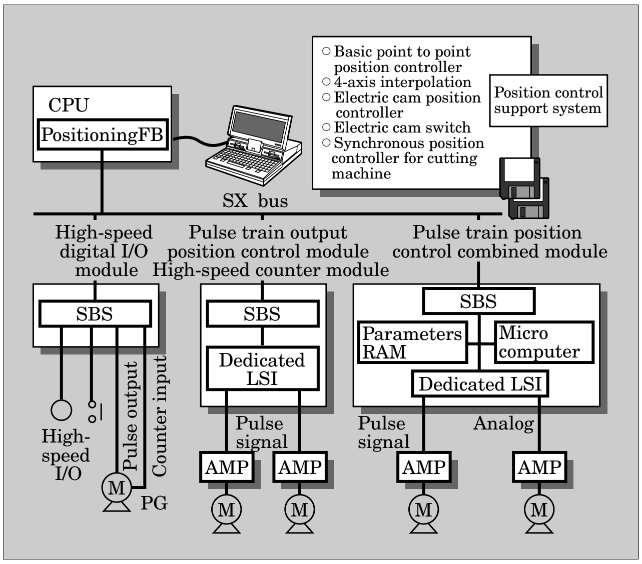


Table 1 Software for position control

Operation system	Software for position control	Kind
1-axis PTP operation	Software for simple position control	2
	Software for compact 1-axis PTP (include simple linear interpolation)	5
	Software for 1-axis PTP (include simple linear interpolation)	22
	Software for multi-function 1-axis PTP	28
Special synchronous operation	Software for special synchronous Software for multi-function 1-axis PTP	13
2- to 4-axis interpolation	Software for 2- to 4-axis interpolation Software for multi-function 1-axis PTP	9
Total		79

3. Expansion of Application Range by Block Engineering and IEC Language

3.1 Block engineering and hierarchical design of IEC

In the MICREX-SX, the whole control program can be hierarchically divided into individual functions, from large class to small class, and made as component parts. Fuji Electric has packaged these divided programs of each function unit as function blocks (FB), and configured a whole system by hierarchically combining these function blocks. We call this method "block engineering."

Figure 4 shows an overview of the hierarchical design and block engineering.

The packaged FBs are IEC compliant and can easily be made with D300win by the user himself.

3.2 Effect of block engineering

The following effects are expected with this block engineering.

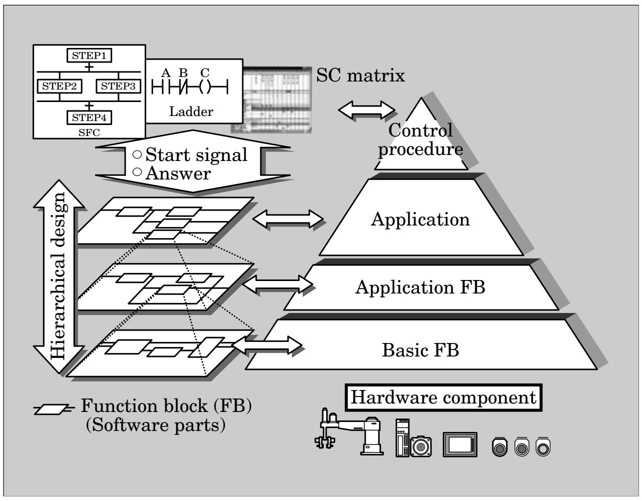
(1) The composition of an original library supporting various controls for each user, improvement of program productivity by implementing hierarchical design, and reduction of the system construction cost are possible.

(2) System construction cost can be minimized by assembling the most suitable specific controller through integrating the original library into the MICREX-SX.

(3) Systems conventionally realized with a specific controller can be replaced with the MICREX-SX by means of customized combination of the library and the high-speed and high-performance of the MICREX-SX.

Further, the parameter setting and data transfer functions of various function modules, communication modules, etc. were made into FBs, and the product series of basic components is nearly complete.

Fig.4 Overview of hierarchical design and block engineering



In the future, it is planned to make standard control functions that are similar to applications, such as speed control, torque control, etc., into FBs.

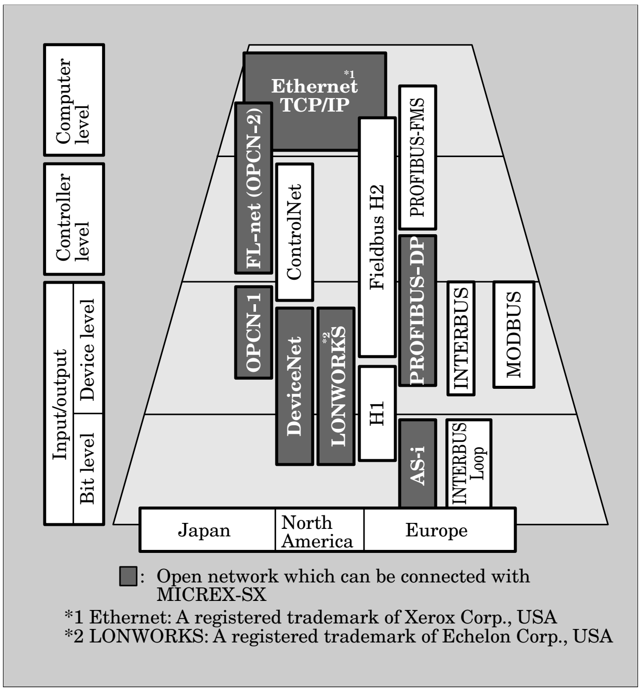
4. Expansion of Application Range by Open System

4.1 Support of standards and open network

4.1.1 Global standard product group according to main standards

The MICREX-SX is complies with main international standards and supports the vigorous overseas expansion of enterprises. The MICREX-SX has received approval for CE marking, and UL/cUL and NK/LR standards. Therefore, the user does not have to select a device according to the destination, and consequently the unification of manufacturing drawings and reduction of spare parts is possible.

Fig.5 Open networks corresponding to the MICREX-SX



4.1.2 System configuration having the optimum expandability

To properly configure the production systems of each enterprise, a wide variety of open network modules from computer level to bit level are prepared. Adoption of an open network enables the selection of an optimal device among manufacturers and the minimization of cost. In Fig. 5, open networks corresponding to the MICREX-SX are shown.

In addition to the open network modules, support of the open networks provides a wide variety of corresponding devices. Therefore, by supporting an open network, the troublesome selection of devices among vendors can be suppressed to a minimum. Table 2 lists equipment for open networks.

4.2 Excellent linking to general-purpose applications

4.2.1 Automatic programming using Excel*2

The MICREX-SX is provided with an automatic programming tool, the SC MATRIX, that automatically generates PLC language from a description of the control process schedule in Excel 97*3 spreadsheet format. The SC matrix realizes the process schedule mechanism with a pin board on Excel. The user can readily understand the straightforwardness of the matrix and the clear visualization of the working state

*2 Excel: A registered trademark of Microsoft Corp., USA
*3 Excel 97: A registered trademark of Microsoft Corp., USA

Fig.6 Overview of SC MATRIX

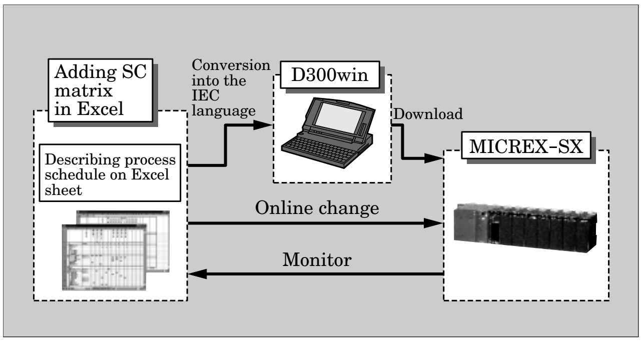
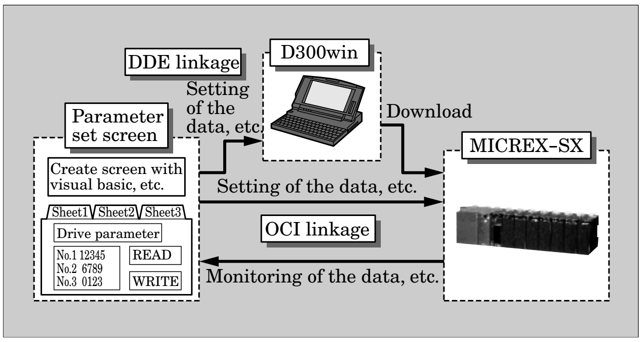


Table 2 Equipment for open networks

Network category \ Item	Corresponding network	Master module	Equipment for open networks: Interface module to expansion	Terminal remote I/O	POD	Inverter	Servo system
Computer level	Ethernet	○	–	–	○	–	–
Controller level	FL-net (OPCN-2)	○	–	–	△	–	–
Device level	OPCN-1	○	○	○	○	–	△
	DeviceNet	○	△	△	–	○	△
	PROFIBUS-DP	△	△	△	△	○	△
	LONWORKS	△	–	△	–	–	–
Bit (sensor) level	AS-i	○	△	○	–	–	–

○: Compatible △: In development or planning

Fig.7 Linkage at DDE and OCI levels



using Excel's editing function without being conscious of the programming language. In Fig. 6, an overview of the SC matrix is shown. [For details, refer to the other paper titled, "Automatic Program Generation Software Package "SC MATRIX"" in this special issue.]

4.2.2 Linking of data to personal computers

The trend of open systems is expanding the application of personal computers into the control field and is increasing the need for users to link on-site controllers and general-purpose applications in the management system's personal computers. The internal information of the MICREX-SX is opened to the public to enable the linking of general-purpose applications to a PLC and the D300win at the DDE*4 (dynamic data exchange) level, etc. The user can create a parameter setting screen using a programming language such as Visual Basic*5 and closely link the data to a PLC and the D300win. Figure 7 shows linkage at the DDE and OCI*6 (open communication interface) levels.

*4 DDE: Function to exchange data and messages between applications

*5 Visual Basic: A registered trademark of Microsoft Corp., USA

*6 OCI: A registered trademark of Microsoft Corp., USA

4.3 Realization of tight coupling by opening the specification of the high-speed SX bus

It is possible with an open network to realize the optimum system configuration sought by the users, but in a position control system and in the human-machine interface connection that displays this data, there are the cases where higher-speed processing and higher response speed are required and even tighter coupling is sought. To meet these user needs, Fuji Electric has opened to the public the SX bus of the MICREX-SX, which realizes high-speed and facilitated distributed control (such as a T-branch). Therefore, high-grade custom systems can be configured on the user side, and open network systems also can be configured easily.

Furthermore, from the viewpoint of system expandability, an optical SX bus has been realized, enabling the construction of reliable systems with high noise resistance.

Presently, devices for the open SX bus are the servo-system, inverter, human-machine interface and temperature control unit. Hereafter, we intend to vigorously open the specifications of those devices (venders) requiring high-speed processing.

5. Conclusion

As described above, together with the enriched hardware and software, the application range of the MICREX-SX has expanded. The objective of system components is not only in the performance of the hardware, but the larger issue is in the usefulness of the surrounding utilities, and we think that the users also expect this usefulness.

Opening and standardizing systems is the trend of the times, and we intend to advance the development of the devices as a basic concept of the MICREX-SX.

New Features of the Loader "D300winV2"

Akihide Hamada
Masashi Yamada

1. Introduction

D300win is a program development support tool in compliance with IEC61131-3 and provides full support of the programmable controller (PLC) for the first time in Japan. Functionality and performance have been improved continuously since the first edition, incorporating the demands of many users. Fuji Electric now offers D300win Version 2 (D300winV2) which, based on those demands, greatly improves the ease of operation by using the latest Windows*1 technology. In this report, the main features of D300winV2 are introduced.

2. New Features of D300winV2

D300winV2 shifts completely to a 32-bit system, and provides many new features and high performance. Some of the main features are described below.

2.1 More comfortable programming support environment

An API (application program interface) for a 32-bit system is used to provide an interface with optimal look and feel. The main features are enumerated below.

(1) Quick window selection (dockable window, over view window, etc.)
(2) Toolbar customizing for desired functions.
(3) Keyword emphasized display to prevent careless mistakes.
(4) User can set short cuts to each menu item.

Figure 1 shows the overall display of D300winV2.

2.2 Project handling

To make the IEC61131-3 structure more user friendly, the handling of projects was simplified by the following functions.

(1) New project view

The new project view consists of the project, POU (program organization unit), libraries, physical hardware, and the instances (execution object) as shown in

Fig.1 Overall display of D300winV2

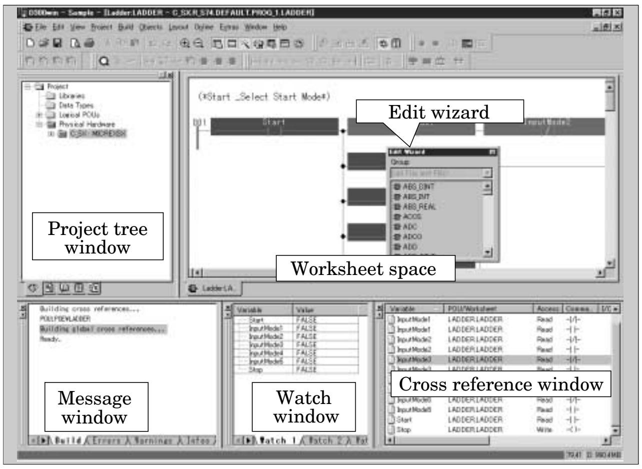


Fig. 2. These corresponded well with the concept of the IEC61131-3, and consequently, the overall view and partial views can be switched quickly.

(2) Easy and quick insertion of the IEC61131-3 structure

The menu and the toolbar can insert the POU, data type, library, and the configuration (configuration of PLC) easily and quickly.

(3) Project wizard

The wizard is a function to simplify the operation by providing selectable guidance choices.

Knowledge of the IEC61131-3 was indispensable in the creation of new projects. New projects can be created by using templates already prepared for the former method, but with D300winV2, new products can be created more easily by using the project wizard.

A new project outline can be created simply with several selections according to the dialog shown in Fig. 3.

2.3 Quick IEC programming with assistance from the wizard

The edit wizard is a useful tool which facilitates the insertion and exchange of functions and function blocks of all languages, the keywords and statements of ST (structured text) language, and the operands of IL language. By using this tool, the user need not

*1 Windows: A registered trademark of US Microsoft Corp., USA

Fig.2 New project view

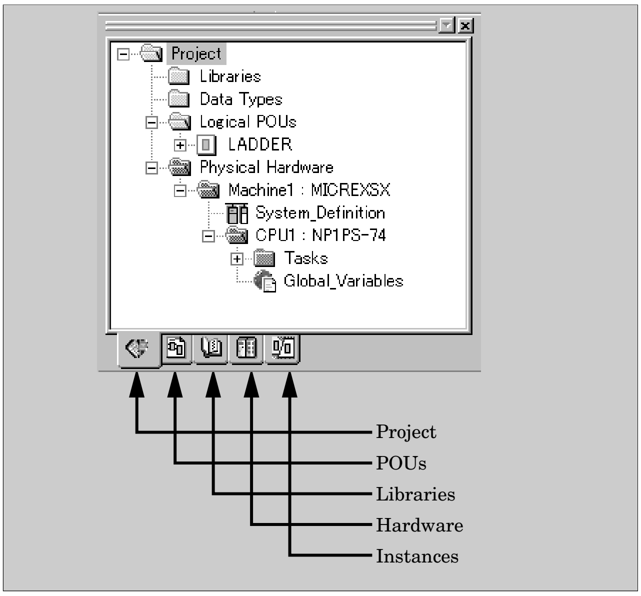


Fig.3 Project wizard

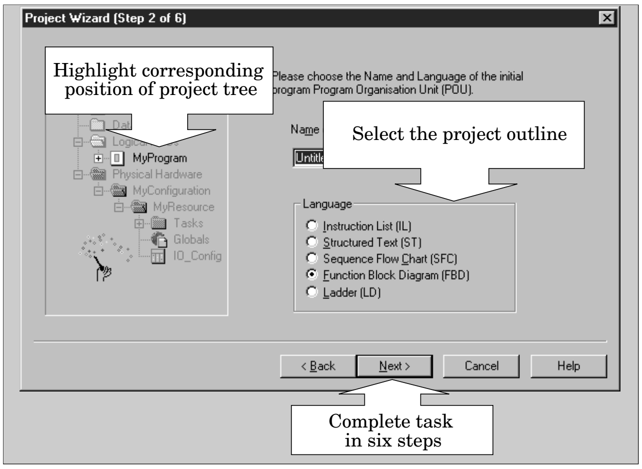


Fig.4 Example of using edit wizard

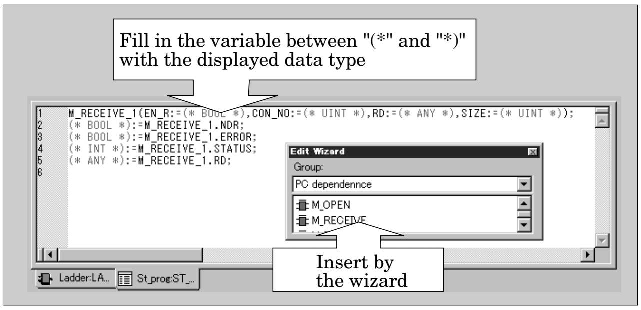
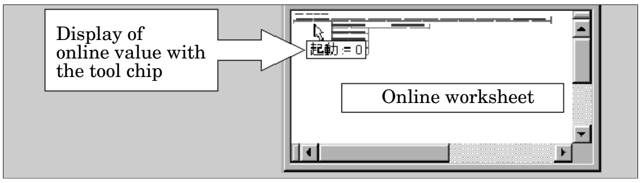


Fig.5 Example of displaying online value using tool chip



know the syntax for all statements. Moreover, the use of the edit wizard with the text editor prevents syntax errors such as a missing semicolon or a missing end-statement for a selection/repetition statement.

Figure 4 shows the Fuji Electric original function block with many parameters having been inserted into the ST worksheet using the edit wizard. The call of the function block can be described by simply filling in the variables according to the data type displayed between "(*" and "*)".

2.4 Graphical editor

To improve the description, the visual check, and the productivity of the graphic editor most often used for the program development, the following functions have been added to D300winV2.

(1) Automatic arrangement of graphical elements

In a graphical language worksheet, the LD (ladder diagram), FBD (function block diagram), and SFC (sequential function chart) graphic elements are moved to the nearest grid edge automatically by the grid function and are automatically aligned in the horizontal or vertical direction.

(2) User defined color settings

The color of all object types can be selected, and the user can improve the ease of visual check.

(3) Tool chip for online values

The tool chip is small help display that appears when the cursor is moved over a button or other object. To make online values easy to read in a worksheet that has been miniaturized to show the overall display in D300winV2, online values are displayed within the tool chip.

Figure 5 is an example of displaying an online value with a tool chip in a miniaturized graphical worksheet.

(4) Multiple line comments

A comment block, consisting of several lines of comments, can be created.

(5) Decrease of object collisions

Objects are automatically moved to empty spaces, decreasing the possibility of collision.

(6) Clipboard for SFC elements

Copying and pasting by means of the clipboard can be performed even with SFC elements, improving the productivity of the SFC program.

(7) Drag and drop for all objects

Similar to a circuit in the worksheet, all objects such as FB, contacts or variables are easily moved and copied by means of drag and drop.

2.5 Powerful online functions

(1) Powerful cross-reference function

The cross-reference function of the D300winV2 can display an abundance of information (use instructions, comments, R/W, etc.) as shown in Fig. 6, and has functions such as sort, jump-to-code, and filter, which

Fig.6 Powerful cross referencing, and monitor of array/ structure for easy debugging

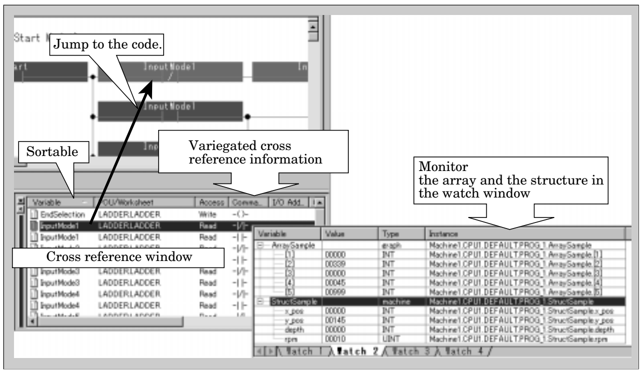
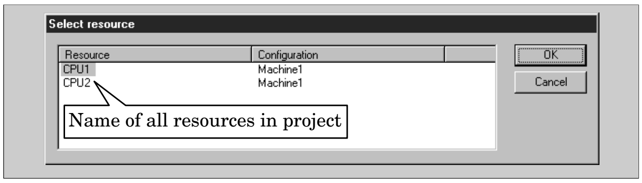


Fig.7 Reduced complexity of multi-CPU systems by resource selection



are very useful for debugging.

The cross-reference function is available both online and offline.

(2) Watch window also useful for arrays and structures

In D300win, the window where the value of the variable is displayed for debugging is called the watch window.

Debugging a user defined data type such as an array or a structure has become easier with V2 than with V1. Figure 6 shows an example display of an array and a structure in the watch window.

(3) Improved handling of multiple CPUs

In a multi-CPU system, the user should select a resource (CPU) for online communication. With V1, the user had to select the resource from the project tree. With V2, because the user can select the resource via a dialog at the beginning of online operation, the complexity of multi-CPU handling is reduced. Figure 7 shows the resource selection dialog display after the online mode has been selected.

2.6 Multi-lingual projects

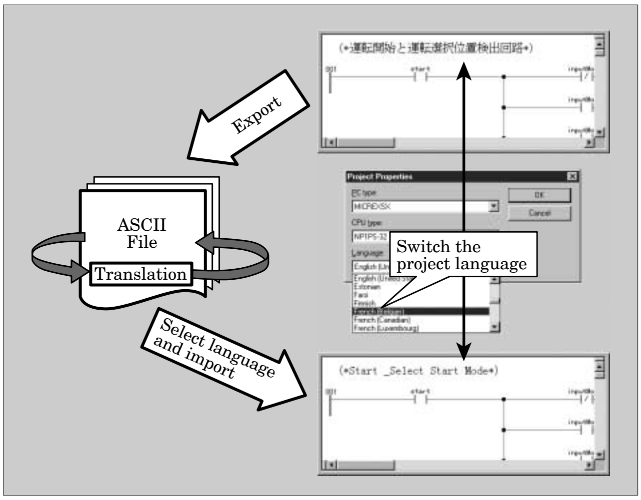
It is necessary to make localization (modification of software to be suitable for a specific region) functions more powerful to meet the trend of globalization. In D300winV2, therefore, the following functions were developed.

(1) Complete support of the Japanese (double-byte character) language

Double-byte characters were supported with all user interfaces including the tree part. As a result, all Asian languages can be supported, including Japanese.

(2) Project language

Fig.8 Switching of project language



Translated description worksheets and comments prepared for localization can be displayed in various languages. Concretely, description worksheets and comments are exported from the project to ASCII file first, then translated, and then imported to the project again. Finally, they are displayed in the language of the destination country. Figure 8 shows the procedure for switching the project language of the D300win by exporting and importing the description worksheet and comments, and also shows an example of the switched comments.

2.7 Faster compiling

Performance characteristics of the display such as the drawing speed improve due to the benefit of 32 bits compared with the previous D300winV1 (16-bit) system. The time for compilation is less that one half that of the previous system, due to the benefit of 32 bits and an improved algorithm.

Therefore, instances where the compilation speed is an issue will decrease under the present general environment.

Table 1 shows the system requirements of D300winV2.

3. Integrated Development Environment Based on D300win

Following the release of V1, application development software (support tool for extended function block for position control, PID control, failure diagnosis, POD editor, etc.) to simplify the application development using MICREX-SX has been offered. This software is based on the functions (such as the variable name service function) provided with D300win that enable variable information in the project to be easily used. Of course, these basic functions are also supported in V2.

In addition, D300winV2 provides the following

Table 1 D300winV2 system requirements

Item		Specification
Hardware		IBM-AT compatible machine or NEC PC98 series
CPU		Intel pentium 133MHz or higher is recommended.
Hard disk drive		Minimum one drive, minimum 100MB free disk space
Floppy disk drive		Minimum one drive, media size 3.5 inches
	2HD	(1.25MB/1.44MB)
Necessary memory		32MB or more
Resolution of display		VGA 640×400 dots or higher XGA 1,024×768 dots (recommended)
Communications interface	RS-232C Maximum rate	9,600bps to 38.4kbps(set default according to the resource type)
	Ethernet*1	Yes
	ISDN	Yes
Operating system(OS)		Windows 95/98/NT4.0*2
Portability		According to the commercial portable personal computer
Environmental tolerance		According to environmental condition of the commercial personal computer
Languages		Japanese and English

∗1 Ethernet : A trademark of Xerox Corporation
∗2 Windows 95/98/NT4.0 : A trademark of Microsoft Corporation

functions that enable commercial software (VB∗2 applications, Excel∗3, etc.) to use variable information in the project.

(1) Import/export function for variable files

Variable information from the project can be converted into comma separated value (CSV) format and export to a text file. Moreover, the comma separated value text file can be converted, and import to the project.

As a result, commercial software can process and use data easily.

(2) More powerful variable name service

The range of variable name service function was expanded and made more powerful. That is, derived data types that are more complex such as arrays and structures, and multi-variable worksheets, were added to the range of service. As a result, application development is even further simplified.

The integrated development environment that centers on D300win by offering the above-mentioned variable name service has been improved.

4. Conclusion

New functions and the improved performance of D300winV2 were introduced. The popularization in Japan of the programming environment in compliance with IEC61131-3 has progressed steadily. Based on the situation in Japan, Fuji Electric will continue to provide systems that contribute to the improved application development efficiency of PLC that centers on the programming environment in compliance with IEC61131-3.

∗2 Visual Basic: A registered trademark of US Microsoft Corp., USA

∗3 Excel: A brand name of US Microsoft Corp., USA

Application of Integrated Controller "MICREX-SX" to a Motion Control System

Tadakatsu Aida
Takashi Ida
Yasutaka Tominaga

1. Introduction

A scalable multi-controller SPH [hardware programmable controller (PLC)] of the integrated controller "MICREX-SX" (hereafter referred to as SX) series has the features of high speed and high performance, and is suitable for the motion control of various machines. Historically, in the servo-system of a typical motion control, a high-performance position control module (for example, the electronic-cam module of MICREX-F) was required. However, in an SX system, SPH can perform the functional calculation due to the realization of a high-speed calculation feature. That is, in an SX system, the system is configured such that only actuator interface functions such as a D-A converter, pulse distributor, etc., are on the position-control module side, and the position-control calculation is executed on the SPH side with an extended function block (FB). With this configuration, it is easy to customize position control processing for integrating user "know how" with the combination of extended FB for position control and user FB (various compensation calculations), and to support special machines. Particularly, in machines performing synchronous operation, there is demand for tuning of the machine control such as for predicting the main axis position and compensating the position gap of the control axis. This SX is the most suitable controller for these machines.

In this paper, we will present an overview of synchronous operation processing and the control characteristics during synchronous operation as an example application of the motion control system of the SX system. Further, example application to a special-purpose cutting machine that combines a rotary axis and a linear axis using the floating point calculation function of the SX will be introduced.

2. Position Control Module

2.1 Basic specifications and system configuration

Table 1 lists the basic specifications of the position-control module for the SX, and Fig. 1 shows a system configuration for position-control with the SX. The following three types of signal systems exist for a servo-amplifier in an SX system.

(1) Pulse reference signal system (NP1F-MP2, NP1F-HP2, etc.)

A pulse reference is output on the position-control module side.

This system is combined with the servo-amplifier and stepping motor of the pulse reference signal system.

(2) Analog velocity reference signal system (NP1F-MA2)

An analog voltage is output on the position-control module side.

This system is combined with the servo-amplifier of the analog velocity reference signal system.

(3) Servo-amplifier directly coupled to SX bus (L, R and V type of FALDIC-α)

This system directly signals the amplifier via the SX bus.

When combining with a FALDIC-α directly connected to the SX bus, the position-control module is unnecessary.

2.2 Function block of position-control module

Figure 2 shows a control block diagram for the position-control module.

In the function calculation unit, acceleration, deceleration and interpolation calculations are performed, and a position signal is output. In the position-control module of the pulse reference output, the position signal is the number of actual pulses. Further, in the position-control module of the analog velocity reference output, position regulator calculation (error counter, gain and feedforward control calculation) is performed. In the error counter, the difference between command position and feedback position from the servomotor is counted. The feedforward compensates for the response lag of the position signal, and is an important function in synchronous operation. The sum of the calculated gain value and the feedforward output value is added to the output value of the error counter to become the velocity reference value for the servo-amplifier.

In the conventional MICREX-F system, the function calculation unit and position-controller were en-

Table 1 Basic specifications of the position-control module

Item \ Name / Type	Analog velocity reference compound module	Pulse reference compound module	Pulse reference output module	FALDIC-α directly coupled to SX bus
	NP1F-MA2	NP1F-MP2	NP1F-HP2	RYSxxxS3-xSS
Occupied slot	1 slot	1 slot	1 slot	——
Number of occupied words	22 words (Input 14 words, output 8 words)	22 words (Input 14 words, output 8 words)	16 words (Input 8 words, output 8 words)	16 words L, R: Input 8 words, output 8 words V: Input 10 words, output 6 words
Number of control axes	2 axes/module	2 axes/module	2 axes/module	1 axis/unit
Control system	Closed loop control	Open loop control	Open loop control	Closed loop control
Reference signal	◦ Analog velocity reference ◦ 0 to ±10.24 V	◦ Pulse reference (open collector) ◦ CCW pulse+CW pulse ◦ Max. 250 kHz	◦ Pulse reference (open collector) ◦ CCW pulse+CW pulse ◦ Max. 250 kHz	——
Feedback pulse	◦ Line driver/open collector ◦ 90°phase difference signal (ϕA, ϕB signal) ◦ Max. 500 kHz (×1)	◦ Line driver/open collector ◦ 90°phase difference signal (ϕA, ϕB signal) ◦ Max. 500 kHz (×1)	——	◦ 16-bit serial encoder (integrated in motor, compatible with ABS)
Manual pulse generator/ Main axis pulse for synchronous operation	◦ Operation signal/open collector ◦ 90°phase difference signal (ϕA, ϕB signal) or CCW pulse+CW pulse ◦ Max. 500 kHz (×1)	◦ Operation signal/open collector ◦ 90°phase difference signal (ϕA, ϕB signal) or CCW pulse+CW pulse ◦ Max. 500 kHz (×1)	——	◦ Expansion counter for manual pulse generator 1 channel (V type)
Outside input/output signal	◦ Dedicated input 5 points (EMG, ±0T,beginning point LS, external interrupt) ◦ General-purpose output 2 points	◦ Dedicated input 5 points (EMG, ±0T,beginning point LS, external interrupt) ◦ General-purpose output 2 points	◦ Dedicated input 5 points (EMG, ±0T,beginning point LS, external interrupt) ◦ General-purpose output 2 points	◦ Dedicated input 5 points (Control 1 to 5) ◦ Dedicated output 2 points (Output 1 and 2)
Internal function	◦ Linear-curve accel./decel. ◦ Continuous change of frequency ◦ Reading the data for position-control in advance ◦ Feedforward calculation ◦ 2-axis simple linear interpolation	◦ Linear-curve accel./decel. ◦ Continuous change of frequency ◦ Reading the data for position-control in advance ◦ Pulse number control function ◦ 2-axis simple linear interpolation	◦ Linear-curve accel./decel. ◦ Continuous change of frequency ◦ Pulse number control function	◦ Linear position-control function (L type) ◦ Rotation calculation function (R type) ◦ Position reference system (V type)
Actuator	◦ Servo-amplifier of analog velocity reference	◦ Servo-amplifier of pulse reference ◦ Driver for stepping motor	◦ Servo-amplifier of pulse reference ◦ Driver for stepping motor	——

Fig.1 System configuration for position control

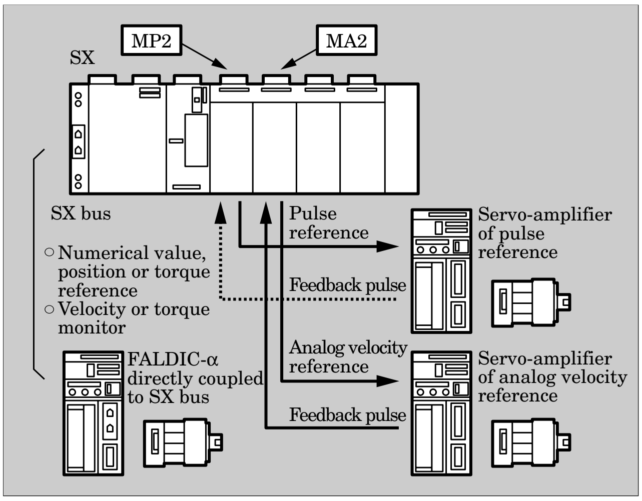
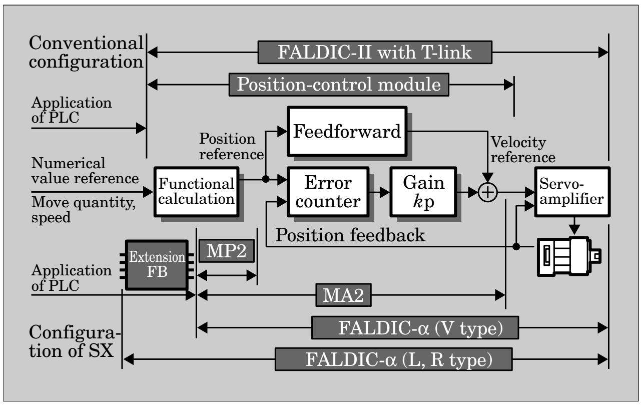


Fig.2 Control block diagram for position-control module



tirely integrated into a high-performance position-control module or servo-amplifier on the FALDIC-II side. That is, the functional calculation was performed by the combination of a high-speed microcomputer and an LSI circuit for pulse distribution mounted on the position-control module.

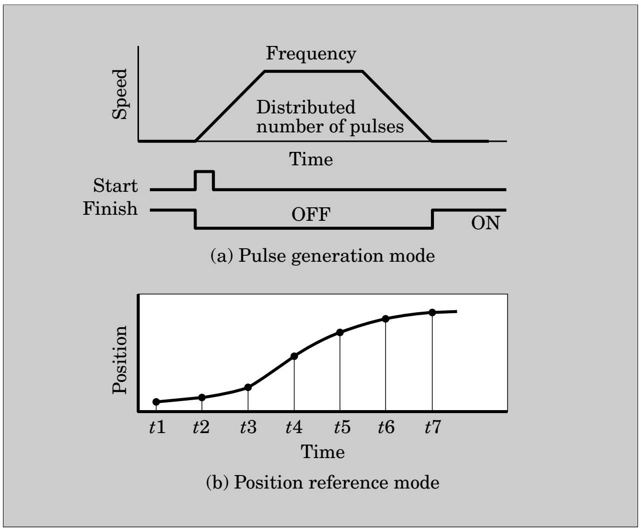
In the SX configuration, among the functional

Table 2 Combination of extended FB for position-control and position-control module

Operating system	Extended FB library name	Function	Objective position-control module			FALDIC-α	Operating CPU	
			NP1F-MA2	NP1F-MP2	NP1F-HP2	RYSxxx S3-VSS	SPH 300	SPH 200
1-axis PTP operation	Compact 1-axis PTP	◦ 1-axis PTP position-control ◦ Linear-curve accl./decel.	○	○	×	×	○	○
	1-axis PTP	◦ 1-axis PTP position-control ◦ 1-axis automatic operation of motion program ◦ Linear-curve accl./decel.	○	○	○	×	○	×
	Multi-function 1-axis PTP	◦ 1-axis PTP position-control ◦ Linear-curve accl./decel./S-form accl./decel ◦ Operation with manual pulse generator	○	○	×	△	○	×
Special synchronous operation	Special synchronous FB + Multi-function 1-axis PTP	◦ Rotary shear ◦ Flying shear ◦ Flying cutter linear operation, rotating operation ◦ Proportional synchronous operation	○	○	×	△	○	×
2- to 4-axis interpolation	Compact 1-axis PTP	◦ 2-axis simple linear interpolation in a module	○	○	×	×	○	○
	1-axis PTP	◦ 2- to 4-axis simple linear interpolation	○	○	○	×	○	×
	4-axis interpolation + Multi-function 1-axis PTP	◦ 4-axis linear interpolation ◦ 2-axis circular interpolation ◦ 4-axis automatic operation of motion program	○	○	×	△	○	×

△: An interface FB is required to match the I/O signal with the position-control compound module.
L-type (LSS) and R-type (RSS) FALDIC-α are integrated into an amplifier whose function corresponds to that of an extended FB.

Fig.3 Operation mode for position-control module



(a) Pulse generation mode

(b) Position reference mode

calculation components, functions which were conventionally processed on the microcomputer of the position-control module side, are executed on the SPH side as an extended position-control module. (In FALDIC-α, in addition to the V type used in combination with the extended FB, all functional calculation features including the L type and R type are provided, similar to the FALDIC-Ⅱ.)

Because the functional calculation feature is made into an extended FB, the user FB can be customized.

The position-control extended FBs for various operations are provided in the SX system as shown in Table 2, and the position-control modules can be directly signaled from the user FBs. Figure 3 shows two types of operating modes of the position-control modules.

(1) Pulse generation mode

After setting the distribution pulse number and frequency, a start signal is turned ON. At the position-control module side, the position-control-completed signal is turned ON after completing the pulse distribution. (Parameter values such as acceleration/deceleration time and high-speed limiter have been set in advance.)

(2) Position reference mode

Pulse position data are signaled every tact cycle using a constant cycle task. In this mode, special operation is realized by transforming the coordinate system with a user FB. In the position-control extended FB, a multi-function 1-axis PTP FB performs synchronous operation and cam operation movement by referencing this mode. (In a FALDIC-α directly coupled to an SX bus, the V type corresponds to this mode.)

3. Application to Synchronous Operation Machine

A synchronous operation function is contained in various machines such as a running crane and a running cutting machine. In this chapter, we will

Fig.4 Rotary-shear system for running cutting machine

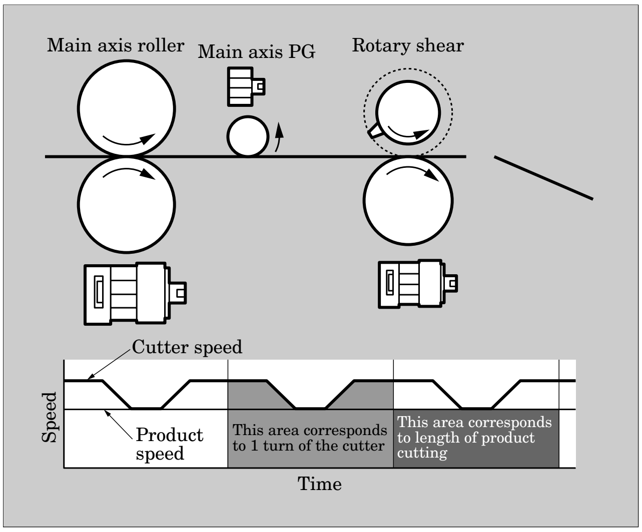
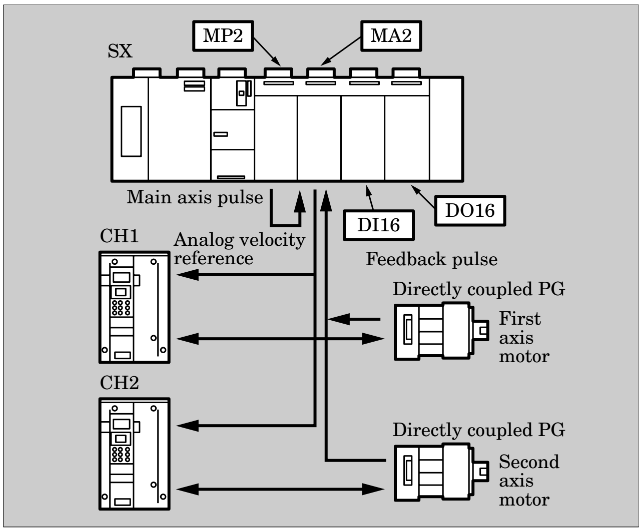


Fig.5 Test system for rotary shear



explain the running cutting machine, assuming that it a rotary shear type.

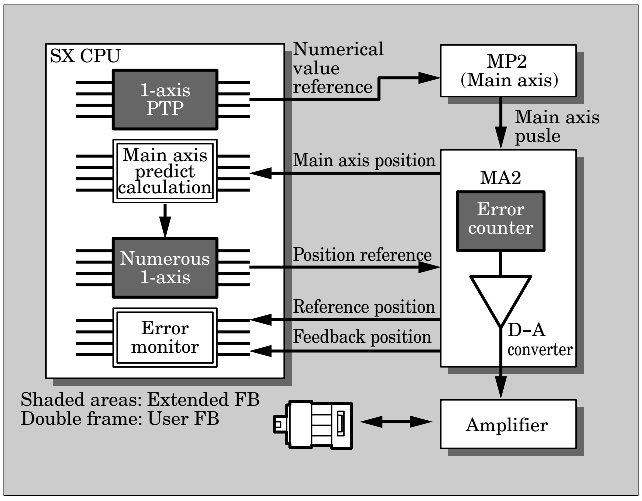
The rotary shear is equipment that cuts with a rotating blade a product which is fed continuously as shown in Fig. 4. The rotary shear is utilized in packing machines and printing machines. During operation of the rotary shear, the peripheral speed of the cutter is synchronized with the speed of the product. Further, since the cutter must rotate one turn when the product is fed by a cutting length (in the case of a single blade), the cutter shaft operates synchronously while adjusting the speed. If the motor that drives the main axis of product feeding is independent of the controller for the rotary shear control, the operation is externally synchronized, and the controller for the cutter unit performs the cutting operation while calculating the feed length of the product and feeding speed from pulses of the main axis pulse generator (PG). The following three types of errors influence cutting accuracy in the running cutting machine.

(1) Control error of the controller
(2) Control error of the actuator
(3) Control error on the machine side

We evaluated the synchronous characteristics of the SX and actuator on the assumption of rotary shear operation, and will introduce the results of that evaluation. The configuration of the evaluation system is shown in Fig. 5, and a function block diagram is shown in Fig. 6.

In this system, externally synchronized operation is assumed, and the cutter is synchronously operated, utilizing the output pulses of the position-control module of the pulse reference output [NP1F-MP2 (hereafter called MP2)] as pulses from the main axis PG. As shown in Fig. 6, processing on the SPH side is commanded from 1-axis PTP extended FB to MP2, but this is independent of the synchronous calculation. Synchronized processing is performed with a combination of the “Prediction calculation FB of main axis position” (user FB) created for this evaluation system and the “Multi-function 1-axis PTP FB” (extended PTP FB). This time, error pulses of the position-control module of the analog velocity reference [NP1F-MA2 (hereafter called MA2)] are sampled every tact cycle, and unevenness is evaluated. Further, this characteristic evaluation is performed with a combination of SX and a vector inverter (unloaded motor).

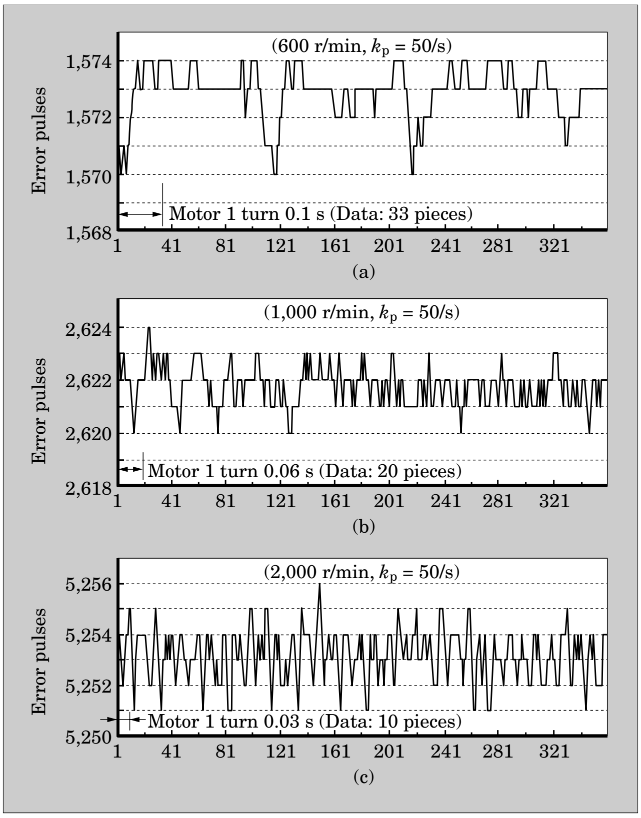
Fig.6 Rotary-shear control block diagram for test system



3.1 Evaluation system

(1) Motor: 5.5 kW synchronous motor (trial sample)
(2) Amplifier: Vector inverter FRENIC5000VG5
(3) Cutter axis PG: 2,000 P/R × 4
(4) Tact cycle: 3 ms
(5) Position regulator (MA2) calculation cycle: 0.8 ms
(6) Position regulator gain: k_p = 50/s
(7) Number of sampling points and cycle: 360 and 3 ms interval
(8) Number of control axis: 2

Fig.7 Unevenness for error pulse (evaluation result of test system)



3.2 Evaluation results

(1) Unevenness is within 5 pulses (refer to Fig. 7)

The ordinate in Fig. 7 shows error pulses, and the calculated error Ep is calculated with the following formulae:

Ep = (Rotating speed/min) × (2,000 pulses × 4) ÷ (60s × k_p) .... (1)

○ Fig. 7(a): 600 r/min

Ep = (600 × 2,000 × 4) ÷ (60 × 50) = 1,600 pulses

○ Fig. 7(b): 1,000 r/min

Ep = (1,000 × 2,000 × 4) ÷ (60 × 50) = 2,666 pulses

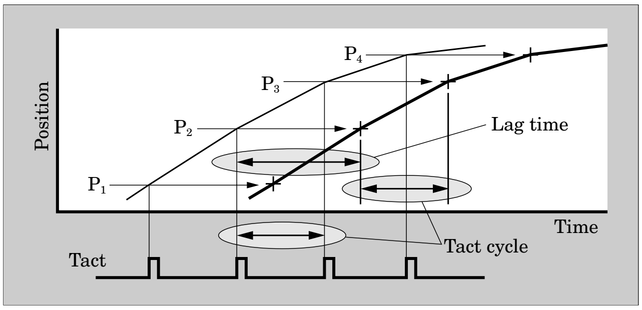
○ Fig. 7(c): 2,000 r/min

Ep = (2,000 × 2,000 × 4) ÷ (60 × 50) = 5,333 pulses

There are differences between the calculated error pulses and the measured values in Fig. 7, but this is unrelated to stability of the synchronous speed because gain adjustments of the analog velocity reference signals between MA2 and the amplifier are somewhat out of alignment.

In the synchronous operation function of the SX, by means of the position reference signal mode (refer to paragraph 2.2), target position reference signaling is performed from the “Multi-function 1-axis FB” to the MA2 side every tact cycle. As shown in Fig. 8, the reference position data P_n calculated with SPH are sent via the SX bus to the MA2 every tact cycle. On the MA2 side, after receiving the data, a distributed calculation of the reference pulses is performed so as to reach the target position after the setting time (lag time). Further, the data-receiving intervals (tact cycles) are automatically measured, and a frequency calculation is performed so as to distribute the pulses at the receiving intervals. By means of this system, even if the calculations of MA2 are performed with a 0.8 ms cycle compared to the 3 ms tact cycle, the velocity reference signal has no ripple and the motor can rotate with a stable velocity. Since the signal to MA2 is the position data, the compensation of data detected with pulses such as the length of phase compensation for synchronous operation, lag compensation, etc., is simply realized by incorporating addition and subtraction of the position data into the user FB.

Fig.8 Operation timing for position control mode



4. Application to Special-Purpose Cutting Machine

Figure 9 shows an example of a 2-axis cutting system combining a rotary axis and linear axis. A product is turned by motor M1 and a torch moves linearly with a motor M2 and ball screw. As shown in Fig. 10, many machines have 2-axis orthogonal configurations and cut with linear interpolation and arc interpolation functions of position-control modules, but machines whose configuration combines a rotary axis and linear axis can be made smaller. Conventionally, the control of a machine configured as in Fig. 9 required a dedicated controller. Otherwise, the linear axis was made to synchronize with the rotary axis using a cam pattern registered to an electronic cam module (NC1F-EC1 of MICREX-F).

The SX performs a coordinate calculation with a high-speed floating-point calculation instruction, and operates with position reference every tact cycle. Operation with the electronic cam module causes the response of the cam axis side to lag the main axis operation at the time of acceleration and deceleration of the main axis because of the following action of the cam axis, even if a prediction calculation treatment of the main axis position is added. In contrast, in an SX system, there is no response lag at acceleration and deceleration due to the calculation of commanded position because positions are calculated for both axes.

The contents of the calculation will be explained

Fig.9 Combination of rotary axis and linear axis

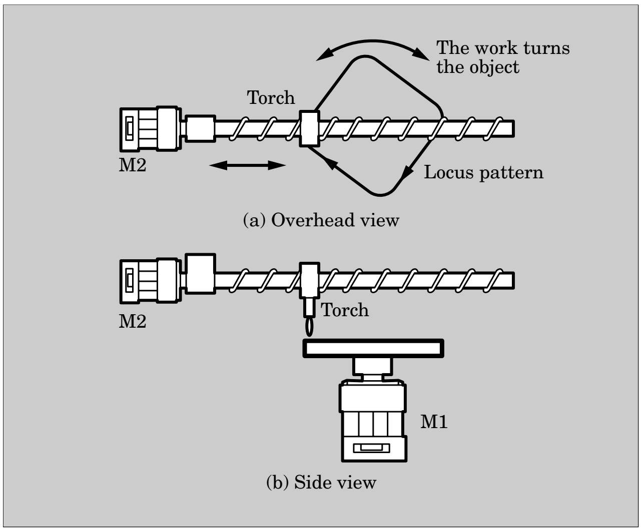


Fig.10 System configuration for 2 orthogonal axes

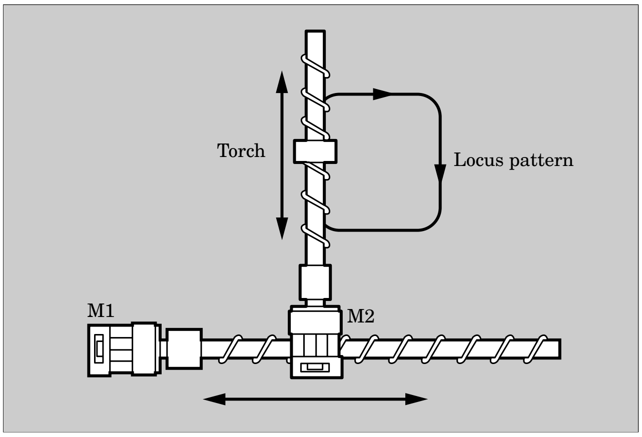


Fig.11 Operation pattern

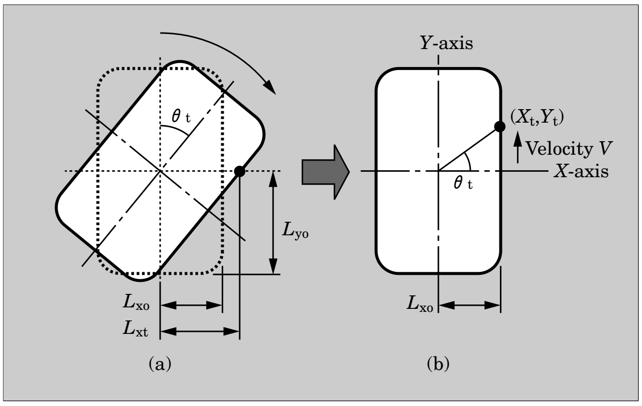
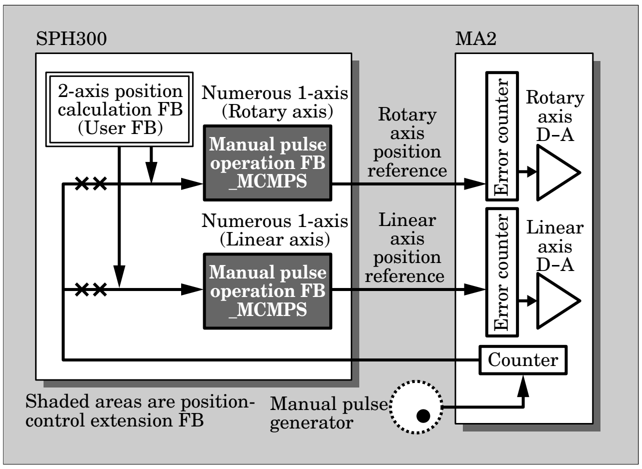


Fig.12 Control block diagram for special-purpose cutting machine



with Fig. 11. Figure 11(a) shows the fed quantity of the linear axis when the product turns by θ_t. L_{x0} is the length from the reference point at the start of operation (center of a rectangular) to the cutting start point. If the position of the torch L_{xt} when the product turns by θ_t, the length moved ΔL_t is:

$$\Delta L_t = L_{xt} - L_{x0} \quad (2)$$

Actually, the coordinate calculation is performed from the reference point as the product is fixed as shown in Fig. 11(b).

If the cutting velocity setting value is V, the X-axis and Y-axis coordinate positions (X_t and Y_t) are:

$$X_t = L_{x0} \quad (3)$$

$$Y_t = V \times t \quad (4)$$

[Calculation formula while moving toward vertical direction in the example of Fig. 11(b)]

The turning angle θ_t from the X-axis and Y-axis coordinates (X_t and Y_t) is:

$$\theta_t = \text{atan}\,(Y_t / X_t) \quad (5)$$

The length from the center to (X_t and Y_t) is:

$$L_t = L_{x0} / \cos\,(\theta_t) \quad (6)$$

or

$$L_t = \sqrt{(X_t^2 + Y_t^2)} \quad (7)$$

In Fig. 11(b), the calculation formulae for X-axis and Y-axis coordinates at corners and feeding along the horizontal direction are different from formulae (3) and (4). However, the coordinate positions are calculated first, next, the rotary axis angle and linear axis position are calculated and then position control is performed.

The FB configuration that was used in this cutting machine is shown in Fig. 12. The coordinate calculations and position calculations of formulae (3) to (7) were performed with a user FB. The calculated 2-axis position results become an input of the manual pulse operation multi-function 1-axis PTP FB. In manual pulse operation, normally, a manual pulse generator is connected to MA2, and position control is performed according to the number of the input pulses counted with a counter in MA2. The application example in this chapter utilizes high-speed floating-point calculation, and replaces the pulse counter of the manual pulse generator with an SPH calculator (user FB) to execute the special operation.

5. Conclusion

This paper introduced the unevenness in the accuracy of an SX system when applied to a synchronous operation machine. The synchronous accuracy required for running a cutting machine such as a flying shear differs depending on the product, but the accuracy proved in this system could be applied to many machines. However, there are machines such as printers for multi-color poster printing that require higher accuracy and higher speed response for position detection resolution, stability of error unevenness, etc. In the future, we intend to respond to the requirements of high accuracy and high speed by enriching the compensation processing FBs that are matched to various machines in an SX position-control system.

Further, in application to special cutting machine, we described an overview of the position-control operation which was conventionally difficult in PLC position-control module. The SX system has a function that performs position-control by means of position reference to MA2 and MP2. When integrating operation pattern calculation with the user FB on the SPH side (similar to drawing a picture of the operation pattern on the screen of a personal computer), special operations can be realized easily.

The authors will be glad if this paper is useful for applying these functions to actual machines that require synchronous operation and rotary axis operation, such as packing machines and various manufacturing machines in addition to printing machine.

Block Engineering to Efficiently Improve Control Software Development

Mitsunori Fukuzumi
Yoshiaki Shimada

1. Introduction

Innovations in the design of the D300win program support tool has greatly improved the efficiency with which one may create control programs in the MICR EX-SX series. In particular, the implementation of function blocks (FBs), which allow existing control programs to be reused, may greatly enhance control program productivity. Historically, similar programs for individual controls had to be made from scratch each time. Through the use of FBs, however, one can apply existing FBs to each new control element to construct the desired control program. Like a jigsaw puzzle, the existing FBs need only to be pieced in order to form the desired control program. This paper first introduces the concept of block engineering and then describes in detail the efficacy of FBs in the construction of control programs.

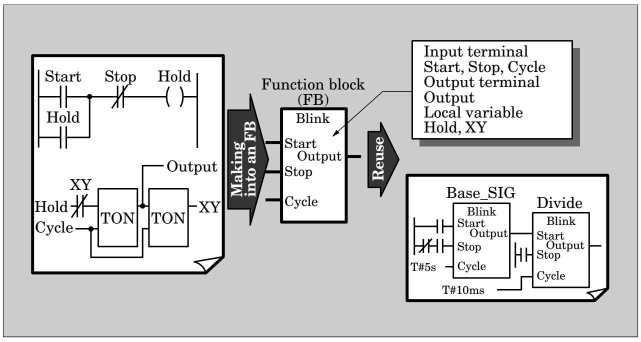
2. What is a Function Block (FB)?

A function block (FB) as prescribed in IEC61131-3 involves modularizing a program into component parts. Figure 1 shows a simple example of an FB. To the left of Fig. 1 a simple ON/OFF circuit is described with a ladder diagram. When a “Start” signal turns the circuit ON, an “Output” signal repeats the ON/OFF with a cycle twice that of the specified cycle.

In the center of Fig. 1, the circuit is made into a function block (FB). The D300win easily converts an arbitrary circuit into an FB (modularization). It is only necessary to specify signal names of an input terminal and an output terminal. This approach is analogous to hardware construction, in that a combination of discrete parts is coordinated into an integrated circuit (IC). In this scenario, signal names are only required for designation of input and output terminals. So, the input-output interface can be clearly expressed.

The right side of Fig. 1 shows an example of how FBs may be used in a ladder circuit. The same FB may be reused as another circuit by simply changing the instance name (corresponding to the part number of the hardware component).

Fig.1 General concept of the function block (FB)



3. Structured Programming in MICREX-SX

Structured programming is realized in MICREX-SX series. In this method, a whole application program is divided into blocks, each corresponding to a specific control object (elevator machine, conveyance machine, etc.) and a specific control purpose (carrying out action, coiling action, etc.). This block-based structure creates an efficient environment in which programming, debugging and administrating services may be performed. Furthermore, while still adhering to the hierarchical framework, each program block can be further modularized and functions simplified. Figure 2 shows this general idea.

In this way, creating hierarchical layers for block structures results in programs that are easy to read and maintain. In addition, the entire program or a component part can easily be divided and made into FBs. This approach allows for the program to be reused and increases the standardization of the control functions.

4. Block Engineering of MICREX-SX

If control functions are prepared in advance as well-proven FBs, most application programs will select the necessary FBs, and compile them together. This technique greatly improves programming efficiency. This is the main objective of the block engineering of MICREX-SX.

Fig.2 Structured programming of the MICREX-SX

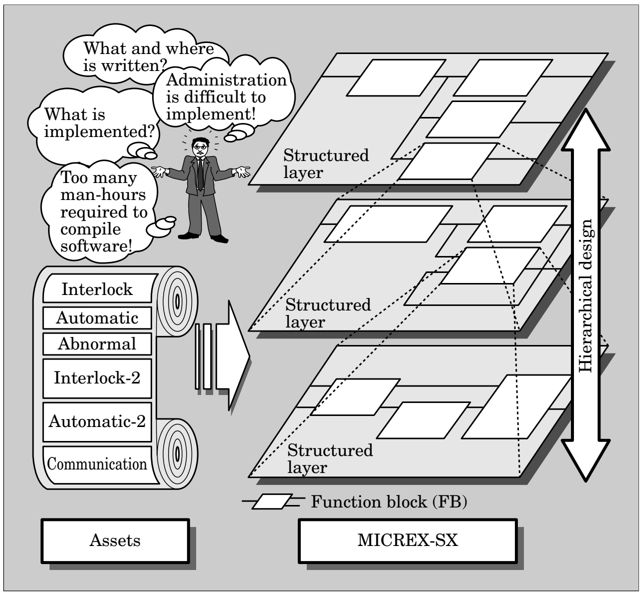
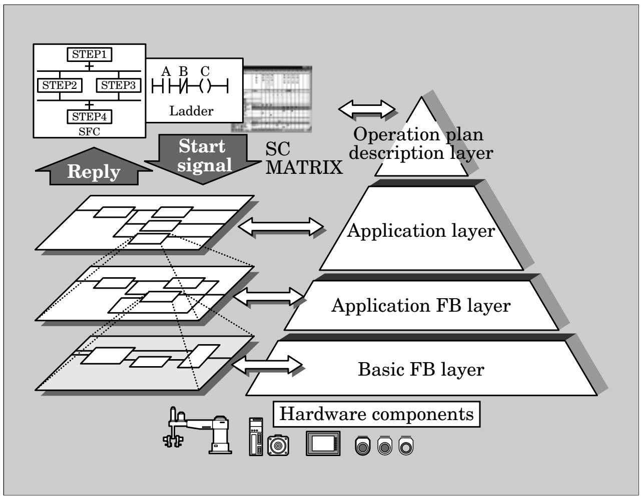


Fig.3 Hierarchical structure of block engineering



In this respect, it is important to know in what type of units should FBs be classified and provided. The block engineering of MICREX-SX builds an application program with hierarchical structure, and then classifies and provides FBs to match each hierarchy as shown in Fig. 3.

4.1 Basic FB layer

When modularizing a real control system into blocks (structured) with the intent to match up hardware and control purpose, the technique ultimately decomposes the control system into function blocks (FBs) for a small control object and its control function. Independent of the overall control purpose of the entire system, the FB functions that directly control real hardware components (sensor, actuator, etc.), that is, the functions to issue command pulses to a servo-amplifier or to receive data from a bar-code reader, depend only on the object component in the whole

Table 1 Control functions classification (partial)

Motion control	Positioning control	Point-to-point		
		Interpolation		
		Calculation of rotations		
		Synchronization	Synchronous operation	
			Traveling cutting	Rotary shear
				Flying cutter
	Speed control	Ratio control		
		Dancer control		
		Loop control		
	Torque control	Tension control		
		Torque balancing		
		Torque limiting		
	Coiling control	Dancer control		
		Tension control (tension detection type)		
		Tension control (coil diameter operation type)		
	Control operations			
Measurement control				
POD interface				
Information control	Web connection			
etc.				

Table 2 Machinery control functions for each type of business

Products suppliers	Rubber and chemical manufacturing industry Car manufacturing industry Others
Machinery system suppliers	Transportation machinery Print machinery Metalworking machinery Paper, film or plastic manufacturing machinery Semiconductor fabricating machinery Food processing machinery Wrapping machinery Others

hardware. In light of this, a set of FBs administrating the interface with hardware components is referred to as the basic FB layer.

4.2 Application FB layer

Table 1 shows a few of the classifications for machinery control. These classifications are for control algorithms, and do not theoretically depend on the control object, which is a type of sensor or actuator. As an example of the positioning control by a servo, consider the scenario in which a servo-amplifier is either related to the pulse train interface, remote I/O combination or the D-I/O combination. In this instance, the positioning control operation algorithm would be common. In this way, a set of FBs that are independent of the hardware components is referred to as an application FB layer.

4.3 Application layer

To implement the machinery control shown in

Table 2, the basic FBs and the application FBs are combined to create the desired control for each customer and for each machinery. The layer that forms this control is called an application layer. Generally, the application layer puts application FBs and basic FBs together and generates a program. The entire program or a component part can be made into an FB and registered as an application FB.

4.4 Operation plan description layer

In block engineering, a program that specifically describes the operation plan is extracted and separated from the application layer. This program is called an operation plan description layer. Between the operation plan description layer and application layer, event signals (bit signals) are exchanged.

The operation plan description layer is an administrative component that is designed to control step or state transitions in the system. In other words, it describes a set of steps in the control procedure. For example, the operation plan description layer records when (work arrives), what it is (a work axis) and how (to go forward) in order to issue event signals ("forward command to work axis") to the application layer.

The application layer and the other underlying layers receive an event signal, perform the corresponding action and then report the status ("action completed", etc.) of the action back to the operation plan layer. Thus, they play a role in the action execution component.

By separating the event administrating component from the action execution component, the functions of the latter become extremely simplified. Therefore, it becomes easier to implement structuring (modularization into blocks) and increases the standardization for programming.

Fig.4 Application of block engineering (system configuration)

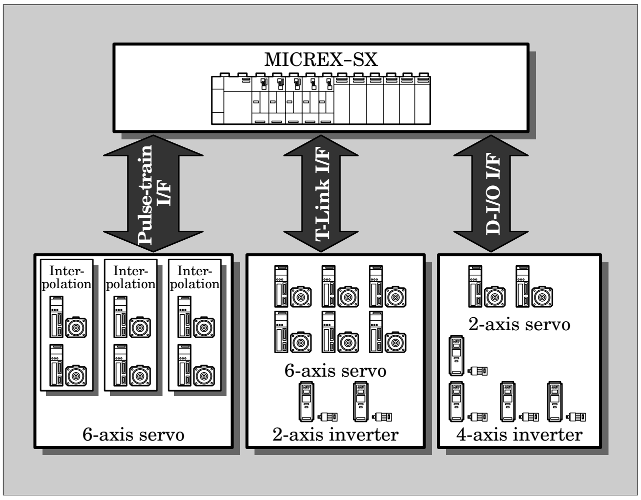
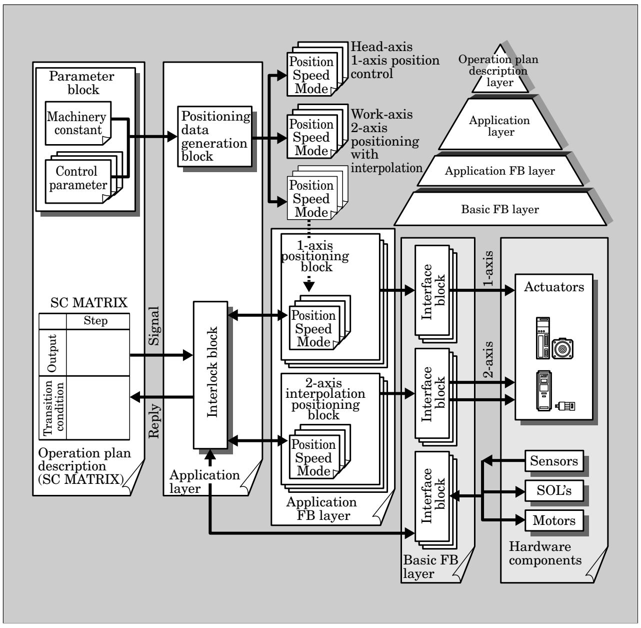


Fig.5 Application of block engineering (hardware configuration)

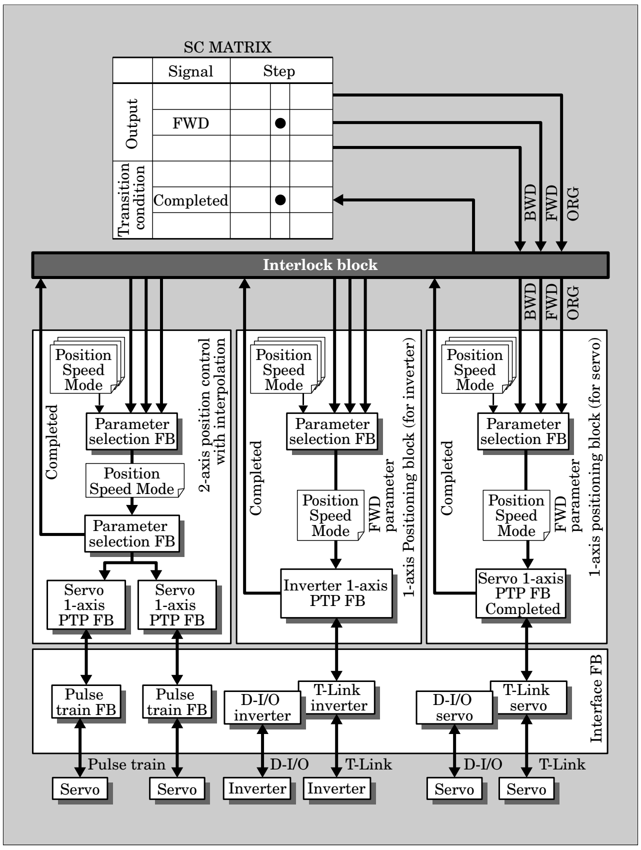


5. Application of Block Engineering

Figure 4 to Fig. 6 illustrate the application of block engineering to positioning control. It should be remembered that this example extracts only a general idea for explanation.

Figure 4 shows the system configuration. This example is of a positioning control with a total of 20 axes of servos and inverters. Of the 20 total axes, 14 axes control 1-axis PTP (point-to-point) positioning,

Fig.6 Details of function blocks (FBs)



while the remaining 6 axes control 2-axis interpolation of PTP positioning. Figure 5 shows the software configuration by block engineering. Each block of Fig. 5 is explained below.

5.1 Operation plan description layer

The operation plan description layer includes an operational plan (state transition description) describing the positioning, the control parameters and the constants of the machinery for each product to be manufactured. This example depicts the case in which the SC MATRIX is used. Any user, even those without programming knowledge can compile and revise contents of the operation plan at will by writing the operation plan description layer with the SC MATRIX. Detailed information about the SC MATRIX is discussed in another paper.

5.2 Application layer

The application layer consists of the interlock blocks and the positioning control parameter generation block program for each axis. The interlock block is a program designed to take a necessary interlock, based on command signals (head-axis DOWN, work-axis FORWARD, etc.) from the SC MATRIX, and transfer it to the application FB layer and the other underlying FB layers. The positioning control parameter generation block is a program that analyzes data trains (machinery constants and control parameters) downloaded from the SC MATRIX and then generates positioning control parameters (position, speed and mode) for each axis.

5.3 Application FB layer

The application FB layer consists of 1-axis positioning blocks and 2-axis interpolation positioning blocks. The 1-axis positioning block consists of parameter selection FBs and 1-axis positioning control FBs (Fig. 6). The parameter selection FBs select positioning control parameters (position, speed and mode) which correspond to the bit command (forward, backward and return to origin) from the SC MATRIX, and then pass it to the 1-axis positioning control FB. The 1-axis positioning control FB executes positioning control operation using given positioning control parameters, and completes by returning the transition condition of the SC MATRIX.

The 2-axis interpolation positioning block adds 2-axis interpolation control FBs to the above blocks (Fig. 6). In other words, the positioning control parameters given by the 2-axis interpolation control FB are resolved into those corresponding to each of two axes, and shared with each 1-axis positioning control FB. The processing is the same as that of the 1-axis positioning block.

5.4 Basic FB layer

The basic FB layer consists of interface control FBs for hardware components. Corresponding to the hardware configuration of Fig. 4, the basic FB layer is configured from the following five types of FBs (Fig. 6).

(1) Inverter control FB with T-Link interface (I/F)
(2) Servo control FB with T-Link I/F
(3) Inverter control FB with D-I/O I/F
(4) Servo control FB with D-I/O I/F
(5) Servo control FB with pulse train I/F

Connecting the appropriate basic FBs listed above to the 1-axis positioning FB makes the control system compatible with all types of actuators (inverters or servos) and interfaces (T-Link, D-I/O or pulse train).

6. Effects of Block Engineering

Construction of a control system using the configuration listed above produces the following favorable effects.

(1) Because FBs are divided into basic and application FB layers, independent of the interface, common use of the control component (application component) is possible. In Fig. 6, the interface is varied among T-Link, D-I/O, pulse train, etc., however, the positioning control can only commonly use the "1-axis positioning FB".
(2) Combining application FBs together can build various functions. In Fig. 6, combining two of the 1-axis positioning FBs together with one of the 2-axis positioning FBs can create 2-axis interpolation positioning.
(3) An application can be administrated and standardized on an FB unit basis. In addition, software created in this manner is durable and can be used in the future because FBs are described in the international standard language IEC61131-3.
(4) By separating the action component from the operation plan component (event component), the functionality of the action component becomes more simplified, making it easier to implement structuring (modularization into blocks) and achieve standardization. Furthermore, automatic adjustment of operation can be easily made by simply revising the event component.

7. Conclusion

The authors propose that the block engineering method, in the form of a series of integrated function blocks (FBs) based on IEC61131-3, result in an efficient operative method. As a consequence of this approach, FBs are likely to play an indispensable role on the PLC programs of the future, as the block method allows for the reuse and standardization of FBs. The authors hope that this paper will help apply the block engineering to practical applications.

Automatic Program Generation Software Package “SC MATRIX”

Mitsunori Fukuzumi
Kenji Hirukawa

1. Introduction

Functionality and performance of the programmable controller (PLC) has been enhanced and its price reduced. On the other hand, its control program has become larger in scale and more complex. Therefore it is not unusual for the programming cost to exceed the hardware cost, and there is strong demand for the more efficient generation of control-software.

In the integrated programming support tool D300win of the MICREX-SX series, an innovative environment is provided for the more efficient creation of control programs by adopting such measures as the international standard language IEC61131-3, introducing structured programming, and improving reusability by modularizing the program. In this respect, the authors have recently developed an automatic program generation software package, the “SC MA TRIX”.

2. Overview of SC MATRIX

Figure 1 shows the procedure for creating general-purpose control programs. First, a control action specification (an operation plan) is made. Next, this specification is translated into a PLC program, and is executed by the PLC. When changing the operation plan, the same processing is always performed in this cycle. If a PLC program is automatically generated directly from the operation plan, a dramatic improvement in PLC program efficiency can be expected.

As automatic program generation software, the SC MATRIX is a product that is used in combination with the D300win. More specifically, the operation plan related to step transitions of the control procedure is described in a predetermined format in a table form in easy-to-use Microsoft Excel 97 worksheet. Based on this table, the SC MATRIX automatically generates a program, i.e. a program organization unit (POU) of the STL (structured text language) prescribed in IEC61131-3.

Even with no knowledge of ladder diagram programming and such, with approximately 1 hour of study, anyone can describe an operation plan in an easy to visualize manner by using Excel 97. Furthermore, once the worksheets are described, a change of the operation plan specification during monitoring or while on-line can be made with the worksheet format. These features are quite advantageous. Figure 2 shows an overview of the SC MATRIX.

Fig.1 Conventional procedure for creating a control program

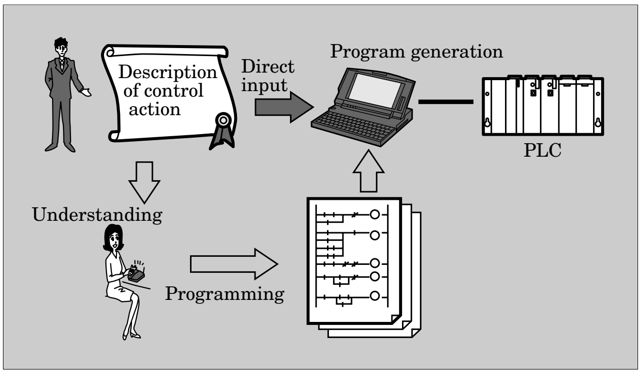
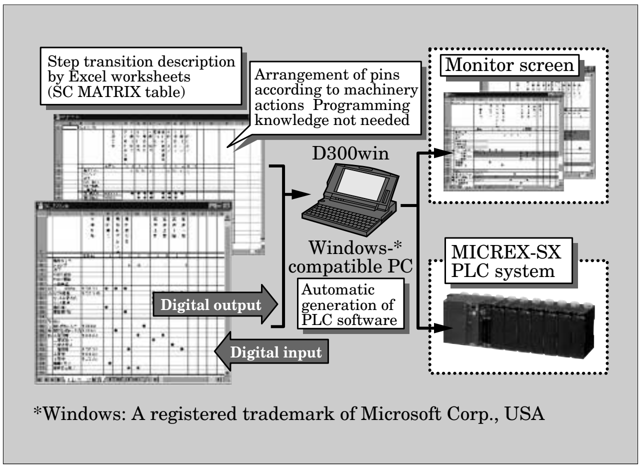


Fig.2 Overview of SC MATRIX



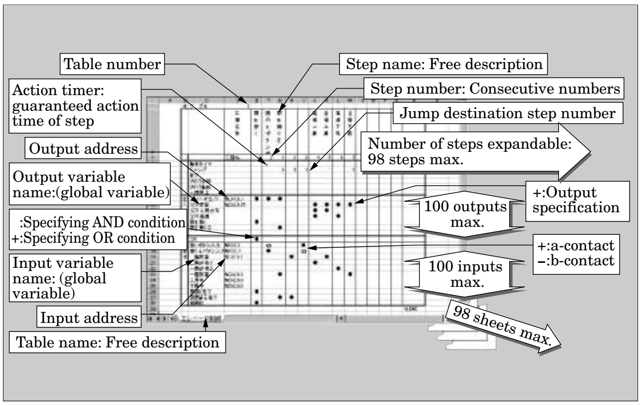
● mark means + symbol.
○ mark means – symbol.

3. Functions of SC MATRIX

3.1 Function of the operation plan description

The details of operation plan description in the SC

Fig.3 Details of operation plan description in SC MATRIX



● mark means + symbol.
○ mark means − symbol.

MATRIX are shown in Fig. 3. The serial steps in the control procedure are compiled in one Excel 97 worksheet. This is called a matrix table and a maximum of 98 sheets is available. According to the matrix tables, step actions can be executed between the matrix tables sequentially or in parallel.

A matrix table lists output signal names ("output" cells) and transition conditions to the next step ("condition" cells) in the vertical direction, and the steps in the horizontal direction.

For each step, the signal to be output at that step is specified in the "output" cell with + (a-contact). At the same time, the transition condition to the next step is specified with a + or − (b-contact) in the "condition" cell. These + and − symbols are called "pins". A "condition" pin basically indicates an AND condition, but it also indicates an OR condition if a + pin is specified in the "OR" cell.

Figure 4 shows an example of the main functions of the SC MATRIX. This table describes the operation plan of an elevator going up to the second floor and down to the first floor, and vice versa.

(1) Step transition function

The step transition function is a basic function of the SC MATRIX and sets the transition condition to the next step and the output signal for each step.

The step number is described with consecutive numbers beginning with 1. If a transition condition is established, the step shifts to the next number.

As shown in Fig. 4, step 1 switches ON output signals "braking output" and "opening door." If the transition condition signal "opening door completed" switches ON, step 1 shifts to step 2. Simultaneously, the output signal is replaced only by the signal "braking output" as specified by step 2.

(2) Jump function

A step can jump to another step if a jump destination is described in the "jump" row of the matrix table. (If nothing is described, the step switches to the next step.)

Based on an input condition, one of several steps can be selected and shifted to in the case where more than one jump destination has been described for the same step. Jumping from one matrix table to another matrix table is also possible.

Fig.4 Example of operation plan description in SC MATRIX

MS table 1

		Step name	opening door	waiting for close button	closing door			High-speed going-up	Low-speed going-up	High-speed going-down	Low-speed going-down	
		Step No.	1	2	3	3	3	4	5	6	7	
	Action timer		1 s									
	Jump				4	6	1		1		1	
	End											
	WDT detection		2 s									
	WDT release											
	Temporary halt							+				
Output	Braking output	%QX1.0.7	+	+	+							
Output	Lift start	%QX2.0.15						+	+	+	+	
Output	Lift going-up direction							+	+			
Output	Lift high-speed							+		+		
Output	Opening door		+									
Output	Closing door				+							
Transition condition	OR			+								
Transition condition	Button input to open	%IX3.0.0		−			+					
Transition condition	Button input to close	%IX3.0.1		+			−					
Transition condition	Arrival at second floor	%QX2.0.1				+			+			
Transition condition	Access up to second floor							+				
Transition condition	Access down to first floor									+		
Transition condition	Arrival at first floor	%QX2.0.9			+						+	
Transition condition	Going-up	%QX2.0.0										
Transition condition	Going-down	%QX2.0.8										
Transition condition	Opening door completed		+									
Transition condition	Closing door completed				+	+						

M_END

Step 3 of Fig. 4 shows an example of a conditional jump with three branches having three corresponding transition conditions.

In the first condition, the step shifts to step 4 if the signals of both "Arrival at the first floor" and "Closing door completed" are switched ON. In the second condition, the step shifts to step 6 if the signals of both "Arrival at the second floor" and "Closing door completed" are switched ON. In the third condition, the step shifts to step 1 if the signal "button input to open" is switched ON and the signal "button input to close" is switched OFF.

(3) Step time assurance function (an action timer)

The minimum action time of the step is assured by setting a value as an action timer in the "action timer" row of the matrix table. In other words, when switched to a new step, that step will remain active for the duration of the time set with the action timer, even if a transition condition becomes established in the meantime. The action timer can select a setting or no setting at each step.

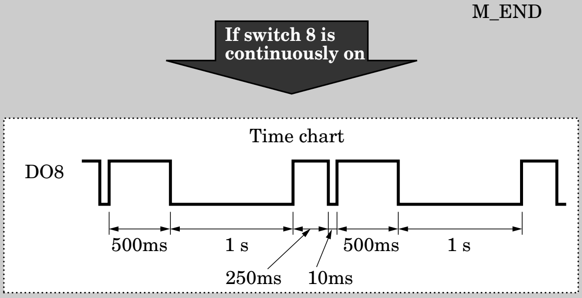
In the example of Fig. 4, when the state is switched from step 3, 5 or 7 to step 1, it will not switch to step 2

Fig.5 Example of time chart description in SC MATRIX

M: table 3

		Step name				
		Step No.	1	2	3	4
	Action timer		500ms	1 s	250ms	10ms
	Jump					1
	End					
	WDT detection					
	WDT release					
	Temporary halt					
Output	DO8	%QX1.0.8	+		+	
Transition condition	OR					
	Switch 8	%IX3.0.8	+	+	+	+
	Switch 9	%IX3.0.9				

M_END



until one second elapses, even if the input signal of “opening door completed” is already given.

In addition, if an action timer is applied, a time chart can be drawn as shown in Fig. 5.

(4) Retardation monitoring function (watch dog timer (WDT) function)

Setting a timer value in the “WDT detection” row of the matrix table makes it possible to monitor the time remaining in a given step. The signal “WDT detection n” (where n is the matrix table number) switches ON when the set watch time has elapsed and stops the step transition control of the relevant matrix table.

The signal “WDT detection n” can be used as a transition condition for other matrix tables and the a-contact signal in ladder programs.

The signal “WDT detection n” can be released by switching ON the signal “WDT release n” from another matrix table or ladder program.

In the example of Fig. 4, unless the signal “opening door completed” switches ON within 2 seconds after the transition to step 1, the signal “WDT detection 1” will switch ON and the step transition control will be stopped.

(5) Temporary halt function

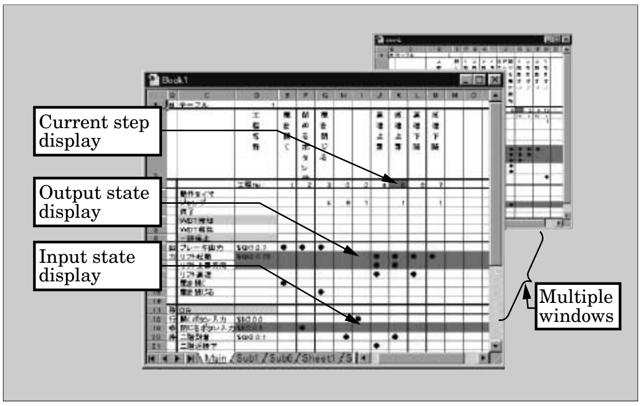
When the + or – pin is set to the “temporary halt” row of the matrix table, the step transition control can be stopped temporarily or restarted. A temporary halt pin acts on judgment of the transition condition with an AND condition. In other words, when a temporary halt is set (the pin is set in the temporary halt cell), the next step will not be transitioned to unless a condition in the halt cell is temporarily established, even if the transition condition is established.

At step 4 of Fig. 4, even if the transition condition signal “access up to the second floor” switches ON, there will be no transition to the step 5 if the pin “temporary halt” is ON.

(6) Data setting function

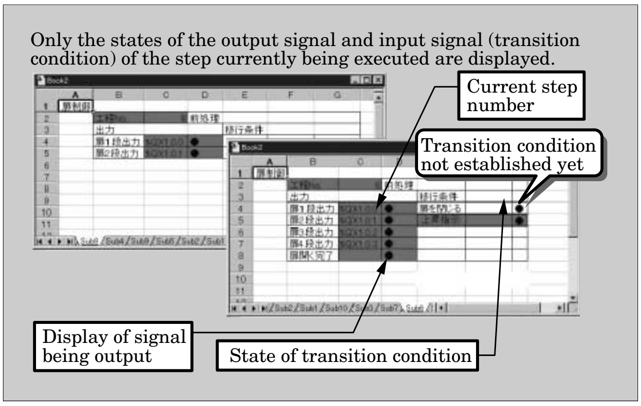
The data setting function can renew specific data in the PLC memory at an arbitrary step of the matrix table. In addition, it can enter a series of data strings into the PLC by manual operation. This function is also used to change control parameters locally or globally.

Fig.6 Example of overall monitoring in SC MATRIX



● mark means + symbol.
○ mark means – symbol.

Fig.7 Example of partial monitoring in SC MATRIX



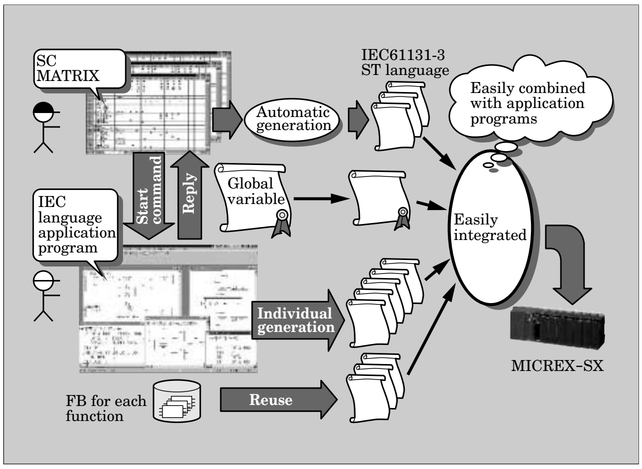
● mark means + symbol.
○ mark means – symbol.

3.2 Monitoring functions

(1) Overall monitoring

This function monitors the entire matrix table with the format that has been created. The current state of the step in execution, its input signal and output

Fig.8 Example of combination with PLC program



● mark means + symbol.
○ mark means – symbol.

signal are displayed. In addition, this function can monitor more than one sheet simultaneously as shown in Fig. 6.

(2) Partial monitoring

This function is provided to automatically gather and display only the current state of the step in execution, its output and transition-condition signals. This function is called partial monitoring. Only the signals specified by the current step are displayed. When the number of steps and signals is large, use of partial monitoring makes it possible to identify at a glance which signal is being output currently and which transition condition signal is being waited for. Figure 7 shows one such example.

The use of overall or partial monitoring can be switched at any time.

3.3 On-line revision function

When setting up and adjusting a control object, it is important that the operation plan can be revised repeatedly within a short time. In the SC MATRIX, a data file is compiled which contains information regarding the existence of pins and the timer setting values. As a result, regeneration of the PLC programs becomes unnecessary based on the following revision operations. Data revision can be performed as if it was an online operation in an extremely short period of time (approximately ten seconds).

(1) Addition or removal of pins (– or +) in the transition condition cell
(2) Addition or removal of the pin (+) in the output signal cell
(3) Time revision of action timer
(4) Time revision of watchdog timer
(5) Revision of jump number

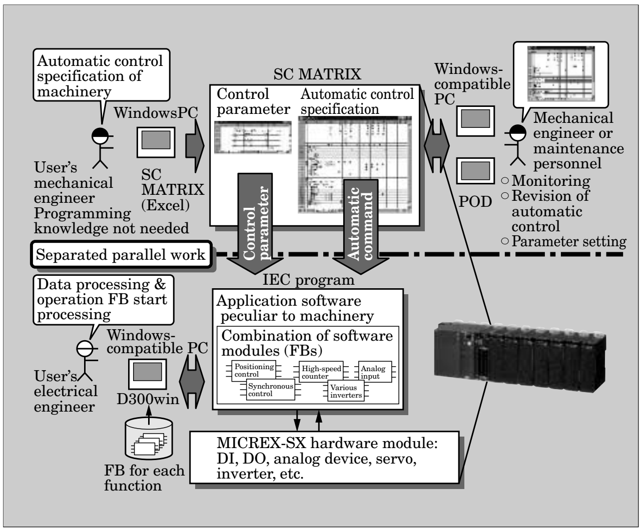
4. Combination with a PLC Program

The SC MATRIX can easily be combined with an application program described by a ladder diagram and the like. The matrix table of the SC MATRIX is compiled and converted into STL programs (POU) of IEC61131-3 for each matrix table unit and are registered on the project tree of the D300win. These programs are equivalent to the user-made programs created with a ladder diagram using the D300win, and can easily be combined with each other on the project tree.

In addition, all the signal names defined in each matrix table are automatically converted to global variables. An application program to be generated with the D300win receives command signals from the matrix table through these global variables and sends back reply signals in the transition conditions of the SC MATRIX (Fig. 8).

As a result of further applications, for example, it becomes possible to interlock the output signal of the SC MATRIX with a ladder program and output that signal, or to generate with an application program a positioning function which is issued a start command from a step of the SC MATRIX and inputs the completion of positioning as one step transition condition of the SC MATRIX.

Fig.9 Example of efficient use of SC MATRIX



● mark means + symbol.
○ mark means – symbol.

5. Efficient Use of the SC MATRIX

Figure 9 shows an example of efficient operations with the SC MATRIX. In this figure, the upper section shows the domain of an engineer who fully understands the mechanical operations (mechanical engineer), and the lower section shows the domain of an engineer who fully understands PLC programming for sensor and actuator controls (electrical engineer).

In the upper section, an operation plan using the SC MATRIX is described, for example, as “when (a work sensor switches ON)”, “what” (drives the loader)” and "how (to raise up)". At the operation plan level, the engineer generally does not refer to the classification of sensors or actuators (general-purpose motors, servos, inverters, etc.) nor the input and output interlocks.

In the lower section, machine-specific application software is provided beforehand as the actual control program and the interlock circuits which are compatible with the types of sensors and actuators.

The following effects are obtained by distributing the programming work.

(1) Parallel work between the mechanical and electrical engineers can shorten the work term.

(2) A mechanical engineer at the work site can easily describe automatic controls without knowledge of the PLC programs.

(3) According to the action behavior of the machinery, a mechanical engineer can revise the operation plan in an extremely short time.

6. Conclusion

This paper has described the advantageous features of the "SC MATRIX" automatic program generation software package of the MICREX-SX, its functions and methods for efficient use. Fuji Electric will continue to advance the study of automatic programming in the future, and will contribute to improving the efficiency of application program development for control systems which are becoming increasingly complicated.

Integrated Controller "MICREX-SX" System and FA Application Software

Chitose Nakamura
Yoshitaka Takeda

1. Introduction

Today, the environment surrounding manufacturing industries is undergoing a significant shift. It has been several years since the trends toward openness, multi-vendor systems, networked systems and multimedia as symbolized by PC (personal computer) systems have advanced from information processing systems to FA (factory automation) systems. In the meantime, personal computers equipped with a standard OS (operating system) such as Windows*1 have been increasingly introduced in factory lines for the purpose of production management, production control, monitoring and operation. The realization of custom manufacturing to meet a variety of consumer needs and achieve labor savings, however, requires seamless data transfer between an information management system with PCs and an FA system, allowing quick and satisfactory real-time response to changes at each phase based on ever-changing data. Nowadays, the environment surrounding manufacturing industries is changing remarkably due to the latest trend toward IT (information technology).

This paper introduces the MICREX-SX (hereinafter referred to as SX) system's response to theses changes by focusing on the data transfer between FA application software and a PLC (programmable controller) system.

2. Technical Trends in Manufacturing Industries

The trend toward IT based on open and multi-vendor systems, which has become widespread in information systems, is eventually infiltrating FA systems.

Ethernet*2 is widely used as a LAN (local area network) in information management systems and also as a LAN between controllers as represented by FL-net. Standard operating systems such as Windows NT*3 and CE*4 are extending their application range from monitoring systems to control systems. The same is surely expected of Java*5. Distributed object-oriented technology such as DCOM (distributed component object mode) and CORBA (common object request broker architecture) have great potential to alter control systems in the future. In addition, controllers with a built-in function to log onto the internet have been developed and introduced to the market.

With this being the situation, a variety of technologies born from IT have become essential for FA systems and the problem of how to combine these technologies to construct a system has become important.

3. Current Status of FA Application

When constructing FA applications in the past, there has been a tendency to customize individual functions by adopting a custom HMI (human machine interface) and SCADA (supervisory control and data acquisition). In most cases, data transfer between an FA system and PLCs was implemented by newly developed communication functionality using a PLC-specific protocol or by using a PLC access function provided by the PLC manufacturer. In addition, it was necessary for the SCADA manufacturer to develop a specific driver for each PLC or a DDE- (dynamic data exchange) interface.

4. Requirements for FA Application Development

When constructing an FA system, it is always important to realize the system in a short period and at low cost, and to facilitate its possible modification or enhancement in the future. Moreover, to survive in a fiercely competitive market it is necessary to utilize the data in a system to realize strategic management.

The technology level in manufacturing industries

*1 Windows: A registered trademark of Microsoft Corp., USA
*2 Ethernet: A registered trademark of Xerox Corp., USA
*3 Windows NT: A registered trademark of Microsoft Corp., USA
*4 Windows CE: A registered trademark of Microsoft Corp., USA
*5 Java: A trademark of Sun Microsystems, Inc., USA

is now sufficiently high to meet those requirements. Utilization of the technologies described above and components based on these technologies allows access via a network to the shop floor PLCs and PLC data seamlessly in a short time and at low cost. In other words, it must be possible to display PLC data on the screen by merely specifying definitions and creating a simple program. Now is the time manufacturers should offer solutions to the challenges described above.

5. Countermeasures in an SX System

Under the above circumstances, the following interfaces have been developed so that data in an SX system can be accessed by variable name to reduce the man-hours required and to facilitate system development.

(1) OPC [OLE (object linking and embedding) for process control] interface
(2) C interface

These interfaces permit the sharing and exchanging of PLC data with machines other than PLCs and the transferring of data to the application. That is to say, a seamless connection between the shop floor and office can be realized via a network.

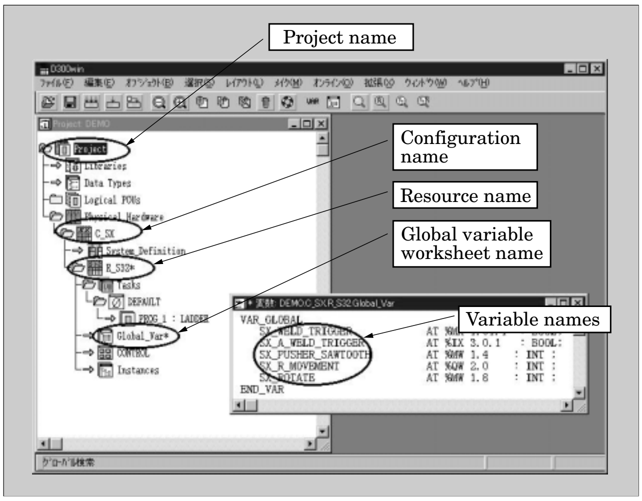
Specifically, a conventional FA application can be realized with an office application such as Excel*6 equipped with a VBA (Visual Basic*7 for application). In Excel, a GUI (graphical user interface) screen can easily be created, allowing those without expert knowledge to easily develop a simple monitoring HMI.

5.1 Variable names

I/O and memory in an SX system can be accessed by variable names. Variable names are defined with an integrated programming tool (hereafter referred to as the D300win) in compliance with the international standard language (IEC61131-3). As shown in Fig. 1, variable names are controlled at the layers of project name, configuration name, resource name and global variable worksheet name. Users can access PLC memory and I/O with variable names defined according to the D300win. Each interface shares the properties of variable names defined with the D300win, facilitating control of the variable names.

*6 Excel: A registered trademark of Microsoft Corp., USA
*7 Visual Basic: A registered trademark of Microsoft Corp., USA

Fig.1 Definition of variable names with D300win



5.2 Data access through an OPC interface

OPC is a data exchange specification for FA/PA (process automation), based on OLE/COM (component object model), and provides a client-sever type object interface. Several specifications for OPC have been formulated. Among these, the OPC data access (OPC DA) specification was made public in October 1998 by the OPC foundation, a nonprofit organization, and products made to this specification have been put on the market by several manufacturers. A report issued by the Japan OPC Council shows that as of October 30, 1999, there are approximately 500 such products worldwide. As shown in Fig. 2, in conventional systems, vendors had to support a different data exchange interface for each PLC manufacturer and client. As shown in Fig. 3, the introduction of an OPC DA interface, a software interface common among PLC manufacturers, however, has facilitated the seamless data exchange of PLC data. Figure 4 shows the mounting of the OPC DA interface. The OPC DA interface features access to PLC data with the variable name of "ItemID."

Products made to the OPC DA specification are expected to bring the merits shown in Fig. 5 to users, system integrators and manufacturers.

With the proliferation of products in compliance with the OPC specification, system expansion, component selection and data access have been facilitated. This has resulted in a high degree of data integration such as the direct transfer of process control application data to data base software and to spreadsheet

Fig.2 Conventional system

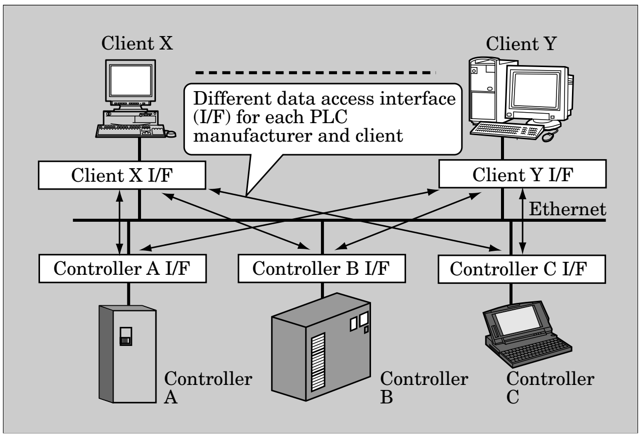


Fig.3 OPC-mounted system (conceptual diagram)

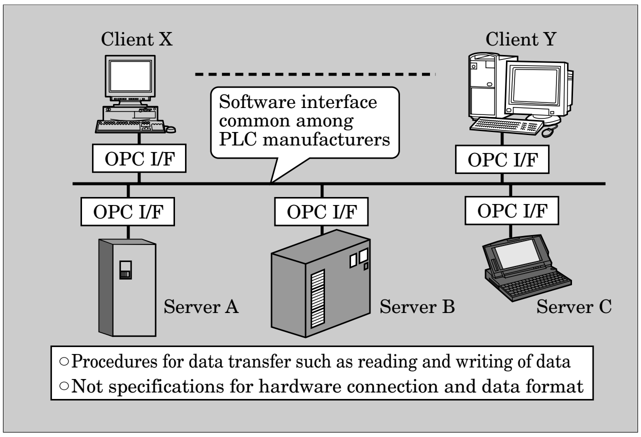


Fig.4 Mounting of the OPC interface

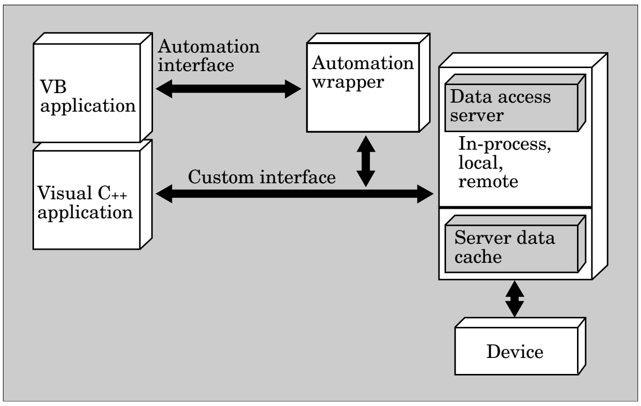


Fig.5 Merits of OPC

□End user
- Increased flexibility in selecting software and hardware
- Organic integration of FA/PA systems and information system

□System integrator
- Facilitated integration, test, verification and maintenance of instrumentation, monitoring and control systems
- Increased range of solutions to be provided
- Improved productivity and reduced amount of time required for system development

□Control systems and equipment manufacturers
- Facilitated development of multi-vendor-capable products
- Reduced amount of time required for system development

software, and automatic editing of the data.

Based on the trend of standardization by OPC, Fuji Electric has developed OPC DA servers intended for an SX system in compliance with specifications of OPC DA version 1.0A and 2.0.

Servers for SX support data access as shown in Table 1. The servers are connected to PLCs in the manner shown in Fig. 6.

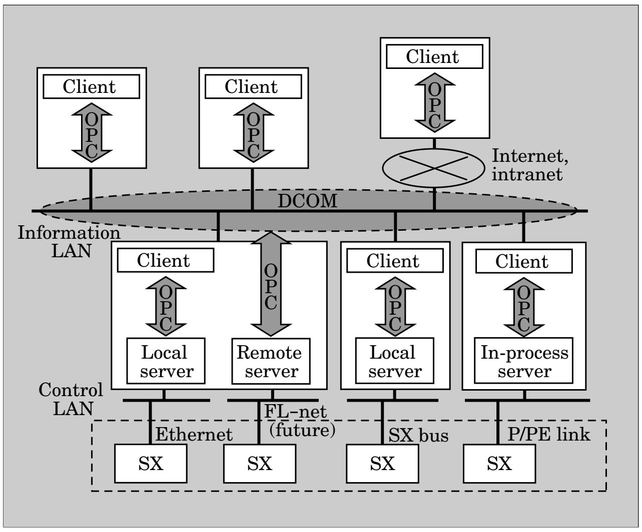
5.3 Data access through a C interface

A C interface permits data access without requiring awareness of the PLC-specific protocol and allows a developer to access an SX system by programming in

Table 1 Support functions for OPC DA in compliance with SX

Function	Method	Description
Process data read	Synchronous read	Read process data corresponding to an item. Wait until the read is completed.
	Asynchronous read	Read process data corresponding to an item. Return to original state immediately when required. When the read is completed, call a method on the data access client side.
	Refresh	Read process data from all active items.
	Notification of data change (subscription)	Check data in a certain cycle. If there is a data change beyond certain limits, notify the data access client.
Process data write	Synchronous write	Output data to an item. Wait until the write is completed.
	Asynchronous write	Output data to an item. Return to original state immediately when required. When the write is completed, call a method on the data access client side.

Fig.6 System configuration using SXs



the familiar C language. The C interface is particularly useful for adding SX to a system that uses conventional machines such as MICREX-F and FLEX-PC, and is also useful when replacing existing machines with new ones.

To be specific, the C interface supports an interface for reading and writing multi-point data into PLCs. As in the case of OPC, the C interface is connected to PLCs in various networks.

6. Example Applications

Various combinations of the above mentioned interfaces allow flexible and seamless data transfer between PLCs and an information system.

At this time, FA applications can be constructed with the following methods.

① Visual C++*8
② VB (Visual Basic) and VBA
③ SCADA
④ Java
⑤ Web

We introduce here ③, ④ and ⑤ as examples of application.

(1) Application construction using SCADA

The connection to a SCADA system is made through an OPC DA interface.

SCADA has previously existed as a system that performs data acquisition and device control by combining various input/output boards around a personal computer. However, in recent years it has chiefly been customized around packaged software.

Since the introduction of an OPC DA interface, SCADA products designed for OPC DA clients are being introduced to the market one after another. Before the introduction of OPC, a DDE interface created by Microsoft was used to facilitate the connection to PLCs, but in some cases it was difficult to process a large amount of data at once and FA applications did not achieve their expected performance. OPC DA solves that problem. By installing an OPC DA sever and defining its connection to the OPC DA server and defining variable names, access of PLC data is possible.

(2) Application construction using Java

Connection to a Java application can be made through either an OPC or C interface.

Access from a Java application to a C application is commonly made through a JNI (Java native interface). A library with high reliability developed using the C language can be reused through a JNI.

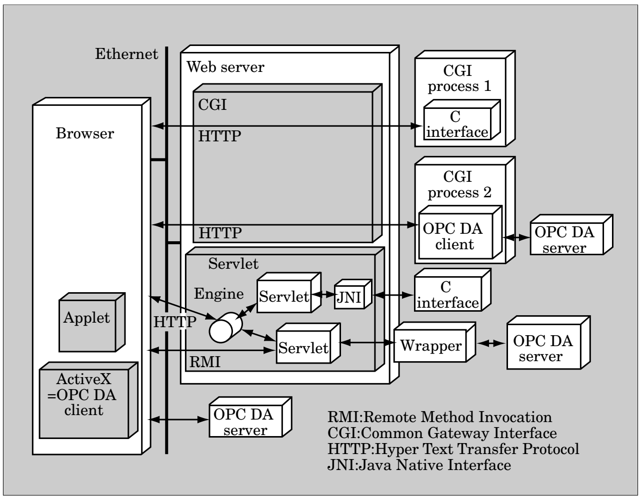
When an OPC DA interface is used in a Java application, PLC data can be accessed through the OPC DA server by creating a wrapper with Visual J++*9.

(3) Application construction using Web

In most cases, a Web system is constructed with ActiveX*10 control, Java and a CGI (common gateway interface). A Web system can be constructed with ActiveX control created by a VB, facilitating the construction of a client server system. If the server-side Web system configuration described in Fig. 7 is used, a Web system can be constructed with Java in the procedures described above in item (2).

*8 Visual C++: A registered trademark of Microsoft Corp., USA
*9 Visual J++: A registered trademark of Microsoft Corp., USA
*10 ActiveX: A registered trademark of Microsoft Corp., USA

Fig.7 System configuration using Web



7. Conclusion

This paper presented newly developed interfaces applied to FA applications in an SX system. With regard to OPC, the standardization of specifications for a DA and other interfaces is being advanced and the OPC Foundation is assuming a leading role. Standardization activities are promoted for the purpose of applying Java to FA.

In the future, as IT advances, it is believed that FA systems will merge with higher level systems, and super distributed systems will be promoted for the purpose of integration with the objects to be controlled. Fuji Electric is determined to play a significant role in application development in order to continue to be a reliable vendor of control systems.

References

(1) OPC Foundation: OPC Data Access Custom Interface Specification, Version 2.0, p.4 (1998)

Real-Time Expansion of Software Programmable Logic Controllers

Hiroshi Matsuda
Taiji Mori

1. Introduction

Fuji Electric is providing "SPS" as a software PLC (programmable controller) for the "MICREX-SX" series of integrated controllers. The SPS is a PLC operated on Windows NT*1, a general-purpose OS (operating system) for the PC (personal computer), and is well received because it shares the same programming language and development environment as the hardware PLC for the MICREX-SX. Realization of the PLC function on personal computers allows the development of applications utilizing the rich resources of personal computers and the merging of information technology (internet, intranet) with control technology, a recent trend.

OSs widely used in personal computers are primarily intended for office use. The SPS, whose processing speed is sometimes inadequate for direct machine control, is mainly used in a management system one layer above.

This paper introduces trends of software technology aimed at achieving an adequate processing speed for machine control and utilizing many of the rich and useful resources in personal computers and general-purpose OSs.

2. Real-Time Processing

Personal computer hardware is constructed as a system around a microprocessor and can be applied to every field including high-speed control.

On the other hand, OSs such as the Windows*2 series contain functions to perform applications such as word processing and spreadsheets, where importance is placed on the human-computer interface, allowing comfortable operation. In other words, an OS places special emphasis on the batch processing of a large amount of information and on the response of human-computer interface processing. As a result, these OSs are not necessarily most suitable for the processing of events such as machine control within a specified period of time.

The control performance required of machine control is the ability to monitor the conditions of machinery, and in response to their changes, to compute and output data within a given period of time, generally 1 to 10ms. In addition, response to a given change must be consistent from the viewpoint of accurate repetition of control.

OS functions that respond to and process external events are classified into soft real-time and hard real-time processing functions. In soft real-time processing, the output of results after the receipt of changes (interruptions) of an external event is not necessary performed within a certain period of time, but is used for clerical work. Soft real-time processing is used with an OS such as Windows. In hard real-time processing, it must be guaranteed that the output of results after the receipt of changes (interruptions) of an external event be performed within a certain period of time. In most case machine control requires hard real-time processing.

Software PLC with additional hard real-time performance is described below.

3. Software PLC by Real-Time Expansion of Windows NT

3.1 Real-time expansion of Windows NT

Since being released in 1994, Windows NT has steadily come into widespread use as an enterprise key system with its increased functions due to upgrades and improved stability. However, because Windows NT is based on a TSS (time-sharing system), it has been said that it is inferior to conventional real time OSs with regard to real-time characteristics and reliability when applied to an embedded system. To solve this problem, it has been proposed to develop a system that gives Windows NT the same real-time characteristics as that of a specific-OS and the reliability to continue real-time operation in the event of blue screen crash by expanding Windows NT with third parties. RTX of VenturCom Co. and INtime of Radisys

---

*1 Windows NT: A registered trademark of Microsoft Corp., USA

*2 Windows: A registered trademark of Microsoft Corp., USA

Fig.1 Software construction (software PLC on Windows NT real-time expansion)

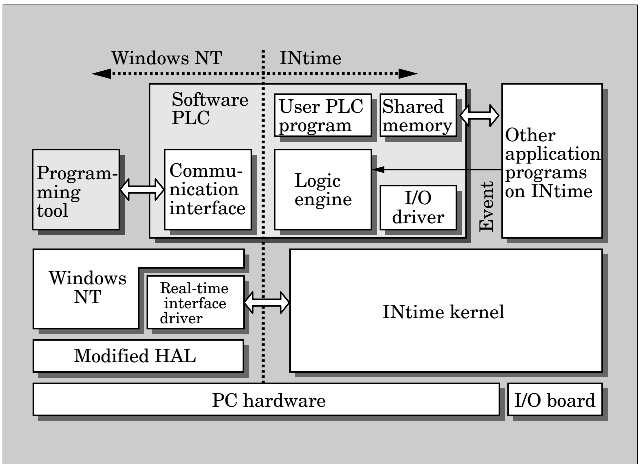


Table 1 Main specifications (software PLC on WindowsNT/ real-time expansion)

Item	Specification
Operating environment	OS: Windows NT4.0 SP3 INtime2.01 CPU: Pentium<Note> 75 MHz or more Main memory: 48MB or more Hard disk: 1GB or more Extension bus: PCI
Cyclic task	1 (minimum period: 2ms)
Periodic task Event task	16 priorities, 32 tasks
I/O control method	Task synchronization
Programming language	IL, ST, LD, FBD, SFC (conforming to IEC61131-3)
Program capacity/POU	ca. 5k steps/POU

<Note> Pentium: A trademark of Intel Corp., USA

Co. are well known as such systems. Here, Fuji Electric implemented a software PLC on INtime2.0, and the construction and main features are described below.

3.2 Software PLC construction on INtime

INtime incorporates the iRMX kernel, an OS developed by Intel, into Windows NT4.0 to realize real-time characteristics. Specifically, Windows NT is positioned as the lowest priority task and is controlled by the INtime kernel, where all Windows NT tasks run. Such construction guarantees real-time characteristics of INtime real-time programs and continuous operation in the event of a Windows NT crash.

Figure 1 shows the software construction of the software PLC that was installed. The PC hardware is a CPU board for PC/AT[*3] compatible personal computers or embedded PC boards. A modified HAL (hardware abstraction layer) expands the time-base and interruption management in the HAL that depends on the PC hardware, and upgrades real-time characteristics of INtime applications. A real-time interface driver supports communications between Windows NT and INtime applications. Software PLC runs on this Windows NT real-time expansion. Software PLC consists of two modules and runs on Windows NT and INtime. Running on the INtime side are a control engine to perform sequence control and a field bus I/O driver; user programs are also executed. On the Windows NT side there exists a communication interface for programming tools, which initializes programs on the Intime side. As a result, support is available from programming tools installed on the same PC or other PCs.

3.3 Linking with external application programs

To establish a link with applications on either INtime or Windows NT, software PLC is provided with the following.

(1) Event task start

PLC event tasks can be started by applications on another INtime, permitting the synchronous execution of external applications and PLC tasks.

(2) Shared memory

Software PLC can share the same area of memory with other applications. The shared memory can be directly accessed by a PLC user program and used for data exchange with other applications.

(3) Function block for file operation

Software PLC is provided with a function block to allow access of file systems on Windows NT from PLC user programs and to permit the setting and storing of machine control parameters from a file.

(4) Function block for message data exchange

Software PLC is provided with a message function block to perform message data exchange with other applications and to permit alarm and status notification from user programs.

3.4 Main features and specifications

The use of a real-time expansion OS on Windows NT allows the software PLC to achieve excellent real-time characteristics, robustness and linkage with the rich software assets of Windows NT. Table 1 shows the main specifications of the system.

4. Software PLC Using a Real-Time OS

In systems embedding in machines, cost-effective development within a short time period is increasingly required. Based on the requirements of embedded systems, Fuji Electric has developed a software PLC using a real-time OS. This chapter introduces the software PLC based on the platform of a general-purpose real-time OS.

4.1 Features of a real-time OS

The Windows OS generally requires a large auxil-

*3 PC/AT: A trademark of International business Machines, Corp., USA

Fig.2 Software construction (software PLC on a real-time OS)

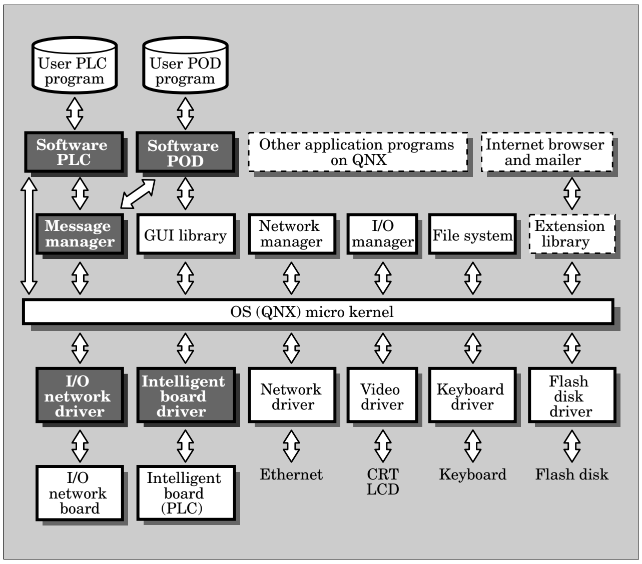
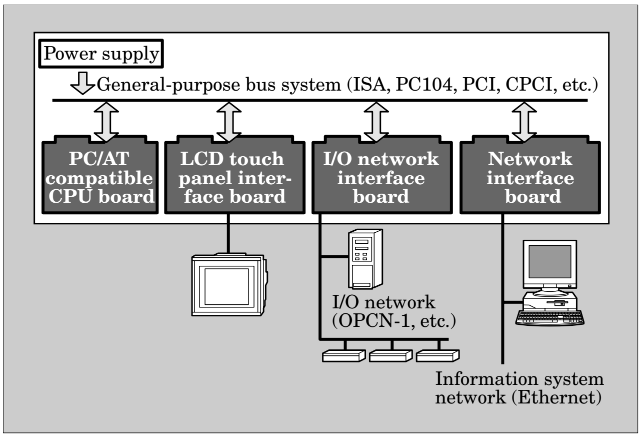


Fig.3 Example hardware configuration (software PLC on a real-time OS)



iary storage device and prescribed procedures for turning power on and off to the system.

In most cases, controllers for machines are installed in environments that are severer with regard to vibration and temperature, than those of office use. Power is turned on and off at random, different from office use. They also require easy maintenance. In other words, a system is required that can be executed with a silicon disk device of small storage capacity having no moving elements, and for which boot-up and shutdown processes are not necessary. In addition, to facilitate maintenance, moving elements of limited serviceable life, such as a fan, must be reduced.

To meet these requirements, a software PLC was installed on a general-purpose real-time OS.

4.2 Software construction

Figure 2 shows the software construction. QNX*4 was utilized as the real-time OS. Centered around the OS is a PLC system and an interface for performing I/O operations with the outside. A HMI (human-machine-interface) is connected to a POD (programmable operation display) through a GUI (graphical user interface) library, permitting the construction of the HMI in the same environment as that of Fuji Electric's hardware POD.

Interfacing with the outside is performed by I/O networks and intelligent boards. Connected to the I/O network are various Fuji Electric networks for I/O devices such as an SX bus and T-link, and an interface board for open networks such as OPCN-1 and DE VICENET. Data exchange between the boards and the interior is performed by device drivers. The software PLC and POD are connected to outside devices through a message manager, and perform upload/download operations of user programs using a general-purpose network (such as an Ethernet*5) as well as read/write operations of external information.

The OS performs task management of these programs. Various other general-purpose libraries can be added to the OS, and an internet- and intranet-capable browser and mailer can be installed on the OS, establishing a system with high expandability.

Figure 3 shows an example of the hardware configuration, where various boards are connected by a general-purpose bus.

HMI operation is performed through LCD keypad panel interface boards and the HMI exchanges information with external actuators and sensors through I/O network interface boards. In addition, network interface boards are connected to an information management system through general-purpose interface boards such as an Ethernet.

4.3 Main features and specifications

Using a low-end OS allows the system as a whole to be made compact in size. Personal computer software for clerical work cannot be used for machine control because the OSs are different. Various open networks can be used for embedded applications, enabling cost-effective development within a short time period and man-hour reduction, compared with the development of a dedicated controller.

Control performance of the newly developed controller is equivalent to that of a dedicated hardware microcomputer, and in most cases realizes satisfactory performance without using a state-of-the-art CPU. Consequently, a cooling fan necessary for a high-speed CPU can be eliminated, leading to cost reduction and easy maintenance.

Table 2 lists the main system specifications.

*4 QNX: A registered trademark of QNX Software Systems Ltd., Canada

*5 Ethernet: A registered trademark of Xerox Corp., USA

Table 2 Main specifications (software PLC for real-time OS)

Item	Specification	Remarks
Operating environment	OS : QNX CPU : Intel386 or later version Main memory : 8MB or more File memory : 8MB or more Extension bus:ISA, PC104, PCI, CPCI	CPU and main memory disk differ according to the system size. Extension buses require their corresponding device drivers.
Cyclic task	1 (default task)	Min.: 1ms
Periodic task	4	*N* times of 1ms
I/O control method	Task synchronization	
Programming language	IL, ST, LD, FBD, SFC	Conforming to IEC61131-3
Processing speed (Sequence-/data-instruction)	200 ns/200 ns (Pentium 75MHz)	Data instruction : ADD
Program capacity /POU	Ca. 5k steps/POU	
Memory capacity	Max.: 256k words	
I/O points	Max.: 8,192	
Number of programs	Max.: 128	
Amount of FB registration	Max.: 512	

<Note> The system is intended for embedded use. The system configuration must meet the performance requirements.

5. Future Trends

With the lower prices and higher performance that result from the proliferation of personal computers, controllers for industrial use which are based on the personal computer platform are expected to increase significantly in the future. Where open architecture is demanded, it has become important to realize machine control using a general-purpose OS. Merits of using a general-purpose OS are as follows.

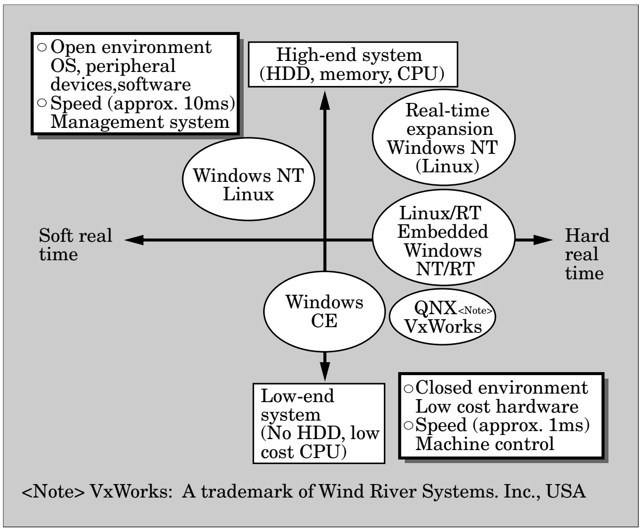
(1) Can use various network protocols installed previously without conducting new development
(2) General-purpose OSs have been installed in various machines and most have few bugs and high reliability
(3) Complete GUI tools and library are available for constructing HMI
(4) Can run third party software

General-purpose business-use OSs are produced primarily to satisfy requirements for the overwhelming number of household appliances and business machines. Those requirements are not necessarily appropriate for industrial use. To realize the merit mentioned in (4) above, it is useful to employ a business-use OS in the manner introduced in chapter 3, and to realize the merits mentioned in the above (1) to (3) in the manner introduced in chapter 4. Figure 4 shows various OSs and their compatibility.

○ Windows NT

Windows NT has a predominantly large market share and an abundance of application programs. Many OS manufacturers offer real-time expansion.

Fig.4 Various OSs and their compatibility



○ Linux*6

Linux is a UNIX*7 OS which recently has received much attention. Its source code is public and is beginning to be used for embedded applications.

○ Windows CE*8

Windows CE is used for handheld personal computers. Reportedly, its real-time characteristics will be improved in the next and subsequent versions. Application to FA fields is expected.

○ Real-time OSs such as QNX

These real-time OSs are specific to embedded applications and superior to other OSs with regard to size and processing speed. OS manufacturers deliver network-capable open OSs.

For the time being, it is believed that an OS appropriate for each device will be used and it takes a long time before OSs for industrial use are unified.

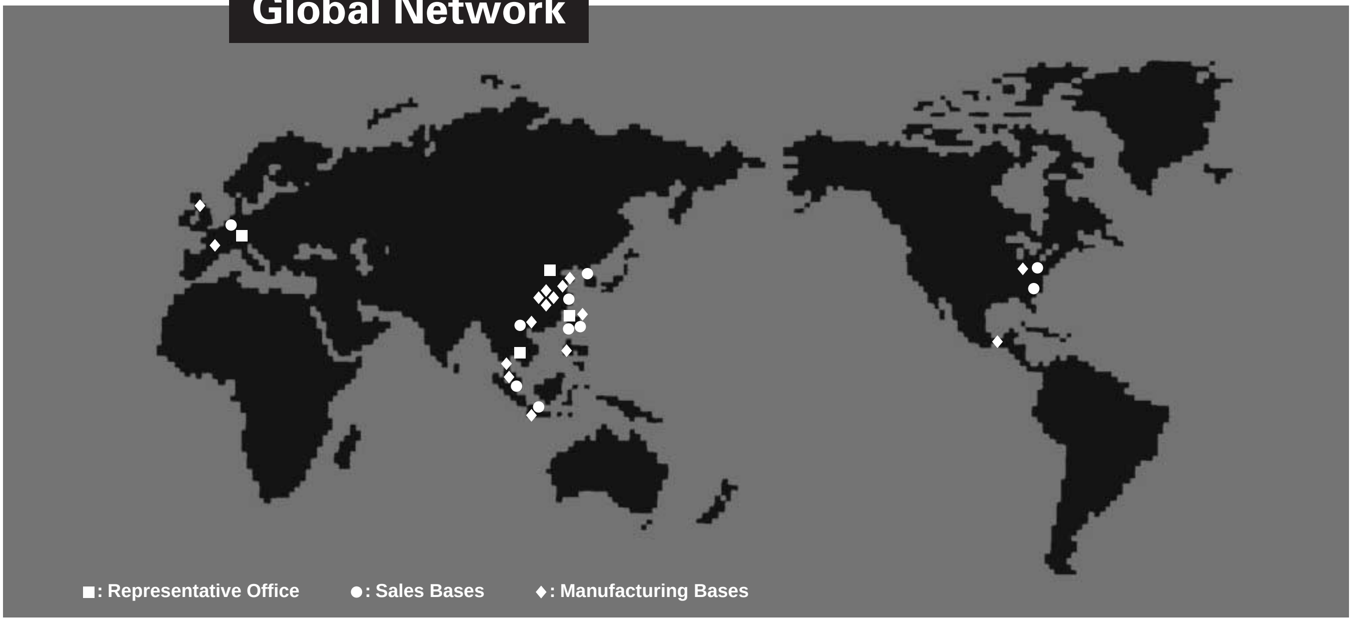
In the future, a software PLC compatible with various platforms will be required. PLC based on real-time Java*9 will be realized in the future.

6. Conclusion

This paper introduced real-time expansion of the software PLC for the purpose of application to machine control. Keeping abreast of the rapid advances of PC-based controller systems, Fuji Electric will continue to develop and deliver optimum control systems to meet the needs of the time.

*6 Linux: PC UNIX created by Linus Torvalds
*7 UNIX: A registered trademark in USA and other countries licensed by X/Open Co. Ltd.
*8 Windows CE: A registered trademark of Microsoft Corp., USA
*9 Java: A trademark of Sun Microsystems, Inc., USA

Global Network



AMERICA

● **FUJI ELECTRIC CORP. OF AMERICA**
USA
Tel : +1-201-712-0555 Fax : +1-201-368-8258

◆ **U.S. FUJI ELECTRIC INC.**
USA
Tel : +1-732-560-9410 Fax : +1-732-457-0042

● **GE FUJI DRIVES USA, INC.**
USA
Tel : +1-540-387-5925 Fax : +1-540-387-8580

◆ **GE FUJI DRIVES AMERICA S.A. DE C.V.**
MEXICO
Tel : +52-8-154-7000 Fax : +52-8-154-7007

EU

■ **FUJI ELECTRIC CO., LTD.**
Erlangen Representative Office
F.R. GERMANY
Tel : +49-9131-729613 Fax : +49-9131-28831

● **FUJI ELECTRIC GmbH**
F.R. GERMANY
Tel : +49-69-6690290 Fax : +49-69-6661020

◆ **FUJI ELECTRIC (SCOTLAND) LTD.**
U.K.
Tel : +44-1355-234111 Fax : +44-1355-238810

◆ **FUJI ELECTRIC FRANCE S.A.**
FRANCE
Tel : +33-4-73-98-26-98 Fax : +33-4-73-98-26-99

ASIA

East Asia

■ **FUJI ELECTRIC CO., LTD.**
Beijing Representative Office
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-10-6505-1263 Fax : +86-10-6505-1851

● **FUJI ELECTRIC (SHANGHAI) CO., LTD.**
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-21-6466-2810 Fax : +86-21-6473-3292

◆ **FUJI ELECTRIC DALIAN CO., LTD.**
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-411-762-2000 Fax : +86-411-762-2030

◆ **SHANDONG LUNENG FUJI ELECTRIC CO., LTD.**
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-531-723-1138 Fax : +86-531-726-1737

◆ **SHANGHAI FUJI ELECTRIC SWITCHGEAR CO., LTD.**
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-21-5718-5741 Fax : +86-21-5718-1448

◆ **SHANGHAI FUJI ELECTRIC TRANSFORMER CO., LTD.**
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-21-5718-5746 Fax : +86-21-5718-5745

◆ **SHANGHAI GENERAL FUJI REFRIGERATION EQUIPMENT CO., LTD.**
Temporary Office
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-21-6921-1088 Fax : +86-21-6921-1066

◆ **FUJI GE DRIVES (WUXI) CO., LTD.**
THE PEOPLE'S REPUBLIC OF CHINA
Tel : +86-510-521-7932 Fax : +86-510-521-5052

■ **FUJI ELECTRIC CO., LTD.**
Taipei Representative Office
TAIWAN
Tel : +886-2-2561-1255 Fax : +886-2-2561-0528

● **FUJI ELECTRIC TAIWAN CO., LTD.**
TAIWAN
Tel : +886-2-2515-1850 Fax : +886-2-2515-1860

● **FUJI/GE TAIWAN CO., LTD.**
TAIWAN
Tel : +886-2-2556-0716 Fax : +886-2-2556-0717

◆ **ATAI FUJI ELECTRIC CO., LTD.**
TAIWAN
Tel : +886-3-358-5000 Fax : +886-3-356-4359

◆ **HONG KONG FUJIDENKI CO., LTD.**
HONG KONG
Tel : +852-2664-8699 Fax : +852-2664-8040

● **FUJI ELECTRIC (ASIA) CO., LTD.**
HONG KONG
Tel : +852-2311-8282 Fax : +852-2312-0566

● **FUJI ELECTRIC KOREA CO., LTD.**
KOREA
Tel : +82-2-780-5011 Fax : +82-2-783-1707

Southeast Asia

■ **FUJI ELECTRIC CO., LTD.**
Bangkok Representative Office
THAILAND
Tel : +66-2-653-2020, 2021 Fax : +66-2-653-2022

◆ **FUJI ELECTRIC (MALAYSIA) SDN. BHD.**
MALAYSIA
Tel : +60-4-403-1111 Fax : +60-4-403-1496

◆ **FUJI ELECTRIC PHILIPPINES, INC.**
PHILIPPINES
Tel : +63-2-844-6183 Fax : +63-2-844-6196

● **P. T. BUKAKA FUJI ELECTRIC**
INDONESIA
Tel : +62-21-5266717 Fax : +62-21-5266718

◆ **P. T. FUJI DHARMA ELECTRIC**
INDONESIA
Tel : +62-21-4606247 Fax : +62-21-4610338

● **FUJI ELECTRIC SINGAPORE PRIVATE LTD.**
SINGAPORE
Tel : +65-479-5531 Fax : +65-479-5210

◆ **FUJI/GE PRIVATE LTD.**
SINGAPORE
Tel : +65-533-0010 Fax : +65-533-0021

Fuji Electric, the Pioneer in Energy and Electronics

FUJI ELECTRIC

富士電機

Printed on recycled paper