

# FCN AND FCJ AUTONOMOUS CONTROLLERS IN STARDOM

**NAKAMOTO Eiji <sup>\*1</sup> OKADA Satoshi <sup>\*1</sup>**

---

*We have developed two models of autonomous controllers as the core of the STARDOM network-based manufacturing solution: the FCN (acronym for field control node) is a modulus controller with superb scalability and the FCJ (acronym for field control junction) is an all-in-one controller designed for distributed in-situ installation. Both controllers incorporate “open” and standardized technologies and feature functions to implement control logic compliant with IEC61131-3 standard of the International Electrotechnical Commission and Web-based communication functions such as autonomous e-mail transmission and communication with Web pages, all in a compact, rugged body. These features realize full-scale, practical use of the up-to-date network technologies, a highly reliable architecture for stable operation, and increased engineering efficiency with the flexibility to address system changes.*

---

## INTRODUCTION

We have developed the FCN (field control node) and FCJ (field control junction) autonomous controllers as the core of STARDOM, a solution that actualizes a network-based control system (NCS) Yokogawa proposes for increased production efficiency. In addition to conventional combined functions such as regulatory control and sequence control, the FCN and FCJ feature a highly independent architecture for linking up distributed controllers and devices via a network to integrate the control domain and autonomously transmitting information. The design oriented for distributed installation, centralized control information within each of the FCN and FCJ controllers, and highly reliable architecture allowing duplex configurations assure steady operations, and the standardized engineering methods (conforming to IEC61131-3) make engineering more efficient, from purchase to operation and on to expansion and modification.

This paper introduces the hardware and software features of the FCN and FCJ.

## SYSTEM CONFIGURATIONS

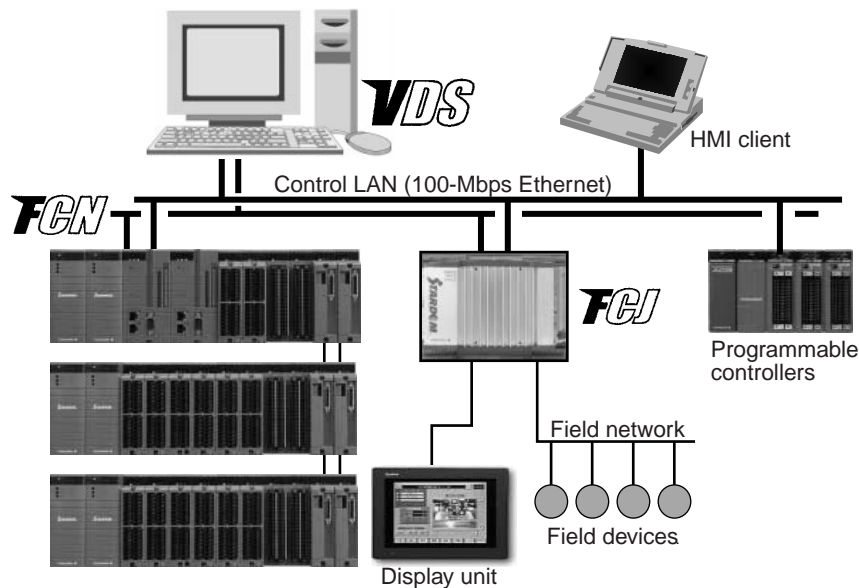
The exceptional, independent nature of the architecture allows the FCN and FCJ autonomous controllers to accommodate diverse styles of system configurations such as standalone control, interlinking control of multiple controllers connected by a control LAN, and interlinking control with data server software. Figure 1 illustrates an example of system configurations. 100-Mbps Ethernet is used as the control LAN to link various controllers and devices, and the control LAN between the autonomous controllers and VDS (Versatile Data Server Software) can be duplexed for higher reliability.

Controller hardware is offered in two models: FCN, a modulus controller with outstanding scalability, and FCJ, an all-in-one controller designed for in-situ installation. The FCN is applicable to a mission-critical system because its components can be made duplex.

In terms of software, a control (loop control and sequence control) implementation environment and Java virtual machine for running Java applications are incorporated to carry out autonomous information transmission by e-mail or to Web pages. FCN and FCJ controller applications are written in programming languages compliant with the IEC61131-3 international standard, and users can choose the most suitable language for each application or the preference of the users from five IEC61131-3-

---

<sup>\*1</sup> Industrial Automation Systems Business Div.



**Figure 1** Example of System Configurations

compliant languages. The adoption of IEC61131-3-compliant languages enable standardized engineering and reuse of the engineering results, thus simplifying the engineering. These development environment and modification and setting tools run on a generic personal computer.

## HARDWARE FEATURES

As aforementioned, autonomous controllers are offered in two models, the modulus FCN and the all-in-one FCJ; nonetheless, they have the same hardware architecture. The following describes the features and specifications of each model.

### FCN

The modulus FCN consists of a power supply, CPU, various I/O, and base modules. All key parts can be duplexed, and the excellent scalability means it is adaptable to small- to medium-scale systems. As shown in Figure 1, up to two I/O extension units can be connected to each FCN to install up to 25 I/O modules with the standard configuration, or up to 20 I/O modules with the fully duplexed configuration.

The CPU module is based on a PC/AT-compatible architecture as a result of placing a premium on openness to attain autonomy. This facilitated incorporation of “open” software components including a general-purpose realtime operating system, control implementation environment, and Java virtual machine.

To increase reliability, ECC (error checking and correcting) memory is used for both the main memory and storage memory, and error detection functions such as the self-diagnostics, watchdog timer, and bus parity check as well as reliability, availability, and serviceability (RAS) are enriched. The module

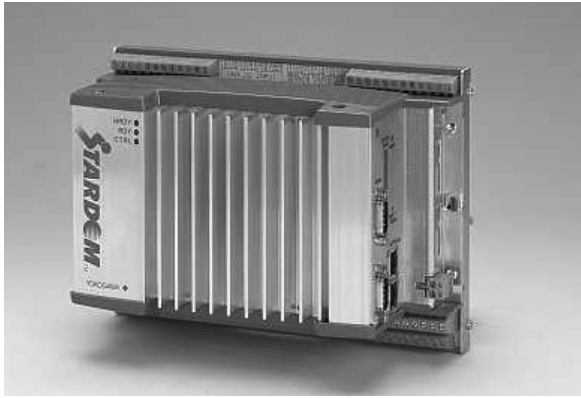
itself is devised to run on low power, and a fan-less, natural air-cooling design is employed, allowing high maintainability.

The power supply and CPU modules as well as the control LAN and internal bus connecting I/O expansion units can be duplexed individually, and modules are hot-swappable. When duplexed, a pair of CPUs run synchronously, so the continuity of control is maintained even in the event a switchover of the control right and thus the user need not care about special programming for a duplication when developing controller applications.

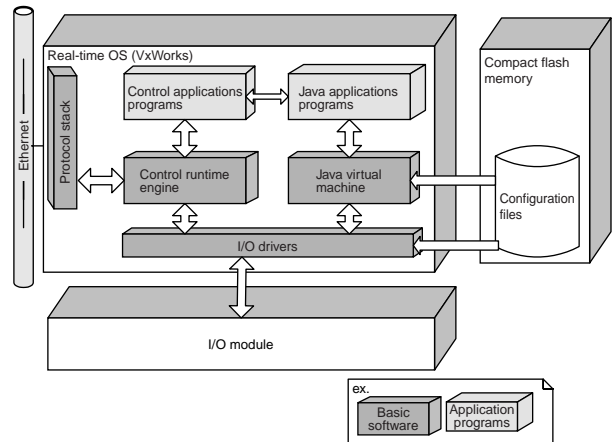
### FCJ

The FCJ is an all-in-one controller designed for distributed installation on site and has a limited number of analog and digital inputs and outputs because its main objective is forming links to field networks (see Figure 2). To be suitable for installation inside a local panel, the FCJ is made compact and slim and is equipped with a terminal block for signal connections. It is driven by a single 24 V DC power supply. There is no difference in the internal architecture from the FCN, namely, the FCJ has ECC memory, a temperature monitor and all the other aforesaid features. However, about redundancy, the control LAN can be duplexed but the power supply and CPU modules cannot.

As the I/O, the FCJ has six 1 to 5 V DC analog inputs, two 4 to 20 mA DC analog outputs, sixteen 24 V DC digital inputs, sixteen 24 V DC digital outputs, and two RS-232C serial interface ports, and will soon have two H1 FOUNDATION Fieldbus ports (under development). Using the RS-232C ports, display units can be connected to provide graphic user interfaces on site. The front and side panels are made of aluminum for efficient heat dissipation to realize natural air-cooling. Also, the terminal block is removable, so the main hardware can be replaced without disconnecting the field wiring in the case of a failure.



**Figure 2** Appearances of FCJ



**Figure 3** Software configuration

## SOFTWARE FEATURES

### Software Configuration

Figure 3 shows the software configuration. One of the key features of the controller software is that off-the-shelf or existing components are used without or with only minimal modification. The following highlights the features of each component.

#### (1) Real-time operating system

The operating system employs VxWorks® of WindRiver, which is the most widely adopted as a real-time operating system for the embedded system and is flexible, scalable, reliable, and available on all popular CPU platforms. Since VxWorks does not support CPU redundancy, it required modification. Although CPU dual-redundancy depends on the operating system, we left the kernel, the basic component of the operating system, intact as far as possible and incorporated an add-on to realize the redundancy, and thus allow the architecture to be migrated into a different operating system with ease afterwards.

#### (2) Control application programs

As the programming languages for control algorithms of control applications, languages compliant with the IEC61131-3 international standard for programmable controllers were adopted. This helps improve engineering efficiency for users in the following manners:

- The users can choose the programming language most suitable to each application or the preference of the user from five IEC61131-3-compliant languages including Function Block Diagram (FBD), Ladder Diagram (LD), and Sequential Function Chart (SFC), allowing for flexible coding.
- Thanks to object-oriented languages, applications can be structured hierarchically with modularized parts and those modules can be registered easily as reusable parts.
- Use of programming languages compliant with an international standard means that users do not need to learn manufacturer-specific programming languages and can start engineering immediately.

In addition to the inherent features of the EC61131-3-compliant languages, FCN- and FCJ-specific specifications and functions are added and they will be described later.

#### (3) Control runtime engine

As the engine for running control applications, ProConOS® of KW Software is used. ProConOS is a runtime engine for programmable logic controller (PLC) applications and can run multiple 10-ms interval control tasks concurrently. The reasons for employing ProConOS include: it faithfully incorporates the IEC61131-3 standard; and real-time operating system-dependent and CPU-dependent portions are modularized, which eases future migration and facilitates future function enhancement.

#### (4) Java virtual machine

Both the FCN and FCJ implement a Java virtual machine to run Java applications independently of the control application for the purposes of processing various data held by the control application and autonomously transmitting the processed data using, for example, Web server, e-mail, and FTP protocol network functions.

The Java virtual machine employs the JEROS system, which forms the application framework of the Yokogawa DUONUS compact field server. In the FCN and FCJ, a class for accessing control application's data from a Java application is added to JEROS so that, with the help of a Java application component, control application's data can be displayed in a Web browser and e-mail can be sent upon occurrence of an alarm. Also, use of the JEROS system means that migration of Java applications from a DUONUS is possible.

#### (5) I/O drivers

An I/O driver is a component that mediates the control codes (logical expressions) and I/O module types and locations (physical expressions). The STARDOM architecture is designed so that logical and physical expressions can be engineered independently. This makes it easy to support various types of hardware including different field networks,

which not merely reduces development costs but will also unify the engineering methods, and this is of value to users. This will be detailed later.

## Control Applications

The following describes the key features of control applications for the FCN and FCJ.

### (1) Process control

Since IEC61131-3 is a standard prepared for programmable logic controllers that are good at machine control, it requires an enormous amount of effort from users to develop a process control application which mainly uses analog expressions. For example, although function blocks containing various concepts such as mode statuses, alarms, and ranges are bases in control logic development for a distributed control system (DCS), IEC61131-3 does not support equivalent functions.

In the STARDOM, functions specifically needed for process control are realized at the IEC61131-3's application level, and are provided as libraries of software parts named Application Portfolios (APPF). For instance, the basic functions of the CENTUM (Yokogawa DCS), such as a PID controller, can be defined in a Function Block Diagram (FBD) of IEC61131-3 and provided as parts in an application portfolio. These parts are referred to as process automation system program organization units, PAS POU's in short. PAS POU's offer:

- Nearly the same functionality and interfaces as those of CENTUM's function blocks
- Support for block mode and status management and smooth transfer actions for every detail
- Data access from VDS

### (2) I/O engineering

Access to inputs/outputs in coding of a control algorithm for an FCN or FCJ is defined by specifying the respective inputs' and outputs' logical names referred to as device labels. The correspondences to the actual hardware are resolved at runtime. This method offers the following merits:

- A control application can be coded without being aware of physical expressions such as actual I/O module mounting locations, so the reusability of control applications is increased.

- Even when I/O module mounting locations are changed, the control applications need not be modified.

### (3) Independence from platform

IEC61131-3-based control applications can run without modification even if the hardware platform or software platform is changed. The aforesaid device labels are also given under the same platform-independence concept. Services dependent on the operating system are gathered in a library written in C, so they can be migrated simply by replacing the system-provided library. Even for the PAS POU's that implement process control functions, their basic logic is written in IEC61131-3-compliant languages, so they can run other manufacturers' programmable logic controllers that adopt the same control runtime engine.

## CONCLUSION

We completed, as controllers in an NCS, platforms having modulus, highly independent software and hardware architectures into which open technologies and standardized engineering methods are aggressively adopted. We hope that a variety of new solutions that make the most of the advantages of these controllers will be configured. We, as developers, are committed to further improving the FCN and FCJ autonomous controllers through upgrades as well as by adopting emerging state-of-the-art technologies while leveraging users' indispensable resources. ◆

## REFERENCE

- (1) Noguchi Akira, Ibaragi Masahiro, Ono Takeshi, Iwamura Tashin, "DUONUS Industrial Network Computer," Yokogawa Technical Report, vol. 42, no. 4, 1998, pp. 173-179, (Japanese)

\* STARDOM and CENTUM are registered trademarks of Yokogawa Electric Corporation. Application Portfolio is under application for a registered trademark. Other product and company names appearing in this paper are trademarks or registered trademarks of their respective holders.

